# A Study on DNA based Computation and Memory Devices

Harmeet Singh
Department of Computer Science
Punjab College of Technical Education (PCTE)
Baddowal, Punjab, India

Sangeeta
Punjab Technical University, Jalandhar

Harpreet Kaur
Department of Electronics and Communication Engineering
Sant Longowal Institute of Engineering and Technology (SLIET) Longowal,
Punjab, India

**Abstract**: The present study delineates Deoxyribonucleic Acid (DNA) based computing and storage devices which have good future in the vast era of information technology. The traditional devices mostly used are made up of silicon. The devices are costly and have physical limitations to cause leakage of electrons and circuit to shorten. So, there is a need of materials which are capable of doing fast processing and have vast memory storage. DNA which is a bio-molecule has all these characteristics capable of providing ample storage. In classical computing devices, electronic logic gates are elements which allow storing and transforming of information. Designing of an appropriate sequence or a net of “store” and “transform” operations (in a sense of building a device or writing a program) is equivalent to preparing some computations. In DNA based computation, the situation is analogous. The main difference is the type of computing devices since in this new method of computing instead of electronic gates, DNA molecules have been deployed for the processing of dossier. Moreover, the inherent massive parallelism of DNA computing may lead to methods solving some intractable computational problems. The aim of this research study is to analyze the logical features and memory formation using DNA bio molecules in order to achieve proliferated speed, accuracy and vast storage.

**Keywords**: DNA; information technology; nanotechnology; bio- molecules; DNA computing

## 1. INTRODUCTION

Computing is ordinarily defined as the use of computer hardware and software to ameliorate the speed and accuracy of mathematical calculations and manipulations. It is the computer-specific part of information technology. Moore's law portrays a long-term trend in the history of computing hardware, in which the number of transistors that can be placed inexpensively on an integrated circuit has doubled approximately every two years. The silicon chip which has supplied several decades’ worth of remarkable features proliferates in computing power and speed. As silicon computer circuitry gets even smaller in the quest to pack more components into smaller areas on a chip, ultimately the miniaturized electronic devices are undermined by fundamental physical limits. They start to become leaky, making them incapable of holding onto digital information. Many researchers are working to overcome the dilemma of silicon chip technology [1]. In order to overcome the flaws of the current scenario, there is a need of better performance material to process and store the information. DNA bio molecules which are genetic materials have the capability to store and process large amount of data. The concept of computing using DNA was initialized by Leonard Alderman who solved the Hamiltonian path problem using DNA molecules [2]. Nowadays, the research in the area of DNA computing has been continued in designing algorithms, designing new basic operations, developing new ways of encoding information in DNA bio molecules and reduction of errors in computations based upon DNA [3]. This paper affianced is to study and discuss the concept of DNA, logical operations based upon DNA and data storage, its main advantages delineating the crucial role of DNA computing in the field of information technology.

## 2. FOUNDATION OF DNA COMPUTING

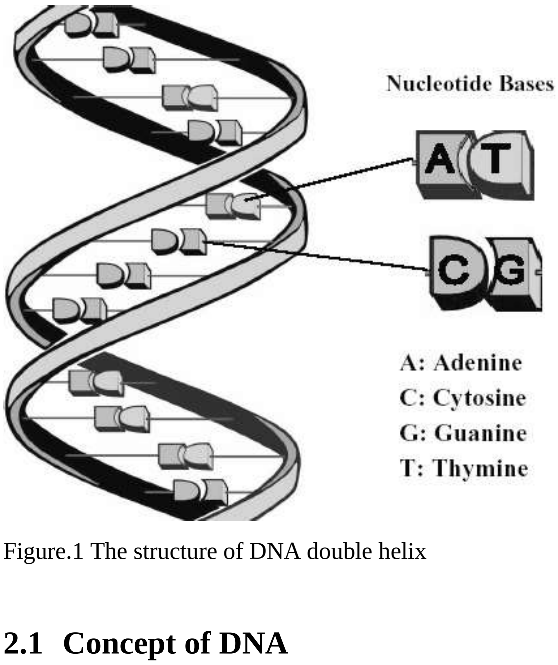


Figure.1 The structure of DNA double helix

### 2.1 Concept of DNA

Deoxyribonucleic acid (DNA) is a nucleic acid that incorporates the genetic instructions used in the development and functioning of all known living organisms and some viruses. The fundamental role of DNA molecules is the long-term storage of information. DNA is oftentimes compared to a set of blueprints or a recipe, or a code, since it contains the instructions required to construct other components of cells, such as proteins and RNA molecules. The DNA segments that carry this genetic information are called genes, but other DNA sequences have structural purposes, or are involved in regulating the use of this genetic information.

Figure 1 shows DNA double helix structure which is a double stranded sequence of four nucleotides; the four nucleotides that compose a strand of DNA are as follows: adenine (A), guanine (G), cytosine (C), and thymine (T); they are intermittently called as bases. The chemical structure of DNA (the famous double- helix) was discovered by James Watson and Francis Crick in 1953. It consists of a particular bond of two linear sequences of bases. This bond follows a property of complementarity: adenine bonds with thymine (A-T) and vice versa (T-A), cytosine bonds with guanine (CG) and vice versa (G-C). This is known as Watson-Crick complementarity.

## 2.2 General Working of DNA Computation

DNA is the dominant information storage molecule in living cells. On behalf of using electrical impulses to represent bits of information, the DNA computer adopts the chemical properties of these molecules by examining the patterns of combination or growth of the molecules or strings. DNA can do this through the manufacture of enzymes, which are biological catalysts that could be called the 'software', used to accomplish the desired calculation. DNA computers use deoxyribonucleic acids A (adenine), C (cytosine), G (guanine) and T (thymine) as the memory units and recombinant DNA techniques already in existence carry out the fundamental operations. From computer science point of view, a DNA strand is a word over alphabet ΣDNA = {A, C, G, T}. In a DNA computer, computation takes place in test tubes or on a glass slide coated in 24K gold. The input and output are both strands of DNA, whose genetic sequences encode certain information. A program on a DNA computer is executed as a series of biochemical operations, which have the effect of synthesizing, extracting, modifying and cloning the DNA strands.

## 2.3 Basic Operations

Hence, the basic operations of DNA algorithms are usually constructed for selecting sequences which satisfy some particular conditions. On the other hand, there may be different sets of such basic operations. In fact, any biochemical procedure which may be interpreted as a transformation (or storing) information encoded in DNA molecules may be treated as a basic operation of DNA based algorithms. One of the possible set of such operations is the following [4]:

MERGE: given two test tubes N1 and N2 create a new tube N containing all strands from N1 and N2.

AMPLIFY: given tube N create a copy of them.

DETECT: given tube N return true if N contains at least one DNA strand, otherwise return false.

SEPARATE: given tube N and word w over alphabet ΣDNA create two tubes +(N, w) and !(N, w), where +(N, w) consists of all strands from N containing w as a substring and: (N, w) consists of the remaining strands.

LENGTH-SEPARATE: given tube N and positive integer n create tube (N, ≤ n) containing all strands from N which are of length n or less.

POSITION-SEPARATE: given tube N and word w over alphabet ΣDNA create tube B(N, w) containing all strands from N which have w as a prefix and tube E(N, w) containing all strands from N which have w as a suffix. Each of the above operations is a result of some standard biochemical procedure.

## 3. MOLECULAR INFORMATION STORAGE

As in magnetic information storage, where magnetic states of ferromagnetic compounds are used, electrochemical information storage is also being studied in biological systems. In this case, distinct oxidation states of certain complex chemicals are being used for multiple bits of information storage at the molecular level. In principle, the amount of information stored is directly related to the number of oxidation states obtainable. By converting the redox state of the molecules into electrical signals, the information can be easily read. This is accomplished by allowing the chemical complexes to self-assemble into monolayers on gold electrodes. This technology is an example of how, as used in Dimensional Design, different kinds of input and output stimuli (chemical input and electrical energy as the output) can be used to store and read information.

A variety of complex "triple-decker" complexes of ferrocenes and porphyrins have been recently synthesized which have four stable and distinct redox states [5]. This is another example of how to build systems which can store enormous amounts of information.

The genetic code is a prime example of how biological systems store an enormous amount of information at the molecular level. Biological information storage in DNA is based on a genetic alphabet. The information content is believed to be encoded in two base pairs (A and G or C and G) which hold the two strands together. In fact there is a special type of bond (the hydrogen bond) between the two base pairs which holds the DNA molecule together. Genetic information is encoded in the specific sequences, along the DNA, of the A-C base pair and the C-G base pair. The number of bits of information is directly related to the sequences of base pairs. The genetic information is then "read" by a complex series of interactions between DNA and special enzymes and proteins. The output of the system is the generation of a replicate strand of DNA (as in cell replication) or the generation of specific amino acids, which are then assembled into specific proteins. Deciphering of the information stored in the base sequences of the genetic code allows biological systems to survive. Although DNA uses a four letter alphabet, DNA is fundamentally a binary storage media for imprinting and retrieving chemical information.

Although DNA has a binary storage capacity, the information content in DNA is so vast we can only conclude that another mechanism, in addition to the known and previously described information storage mechanisms must be at work here. One example of the enormity of the information in DNA is the fact that DNA contains output information not only about specific amino acids and bases, but also about how these building blocks are organized into complex three dimensional (3D) bio-molecular structures. In order to understand and utilize the enormous information in DNA the new field of bio-informatics has developed which uses complex mathematical modeling, algorithm-based computer technology, DNA chips and cDNA microarrays (DNA coated microcircuits) [6]. A high-throughput sequencing ability is characteristic of this new sophisticated genomic computational technology.

## 4. DNA BASED LOGIC GATES

To build a computational system, it is firstly necessary to have the development of DNA based logic devices. A logic gate performs a logical operation on one or more logic inputs and produces a single logic output. The logic normally performed is Boolean logic and is most commonly found in digital circuits. Figure 2 shows the block diagram of a simple logic device with two inputs and one output.

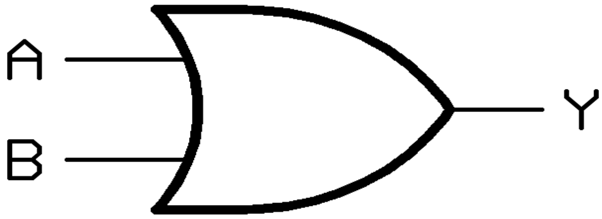


Figure.2 Schematic of Logic gate with input and outputs

In vitro studies have been used to design combinations of molecules that have emergent properties related to information processing--molecular computing devices. Both the inputs and outputs consist of molecular species, with the output being a biologically active molecule. The extent to which these devices will be used with the cellular context is unclear--however, they are bound to inspire new directions for research in synthetic biology, and have potential applications in biochemical sensing, pathway engineering, and medical diagnosis and treatment.

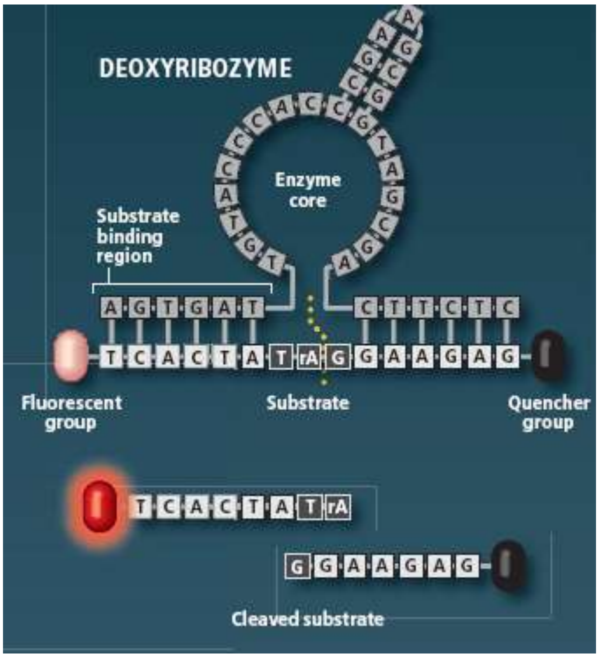


Figure.3 Schematic of DNA enzymes combination

Combining DNA enzymes with stem-loop controllers yields a variety of fundamental logic gates that use short strands of DNA as both inputs and outputs. The cleaving action of the enzyme produces the strands that serve as the gate's output of 1 as shown in Figure 3.

### 4.1 AND Gate

A logical AND gate has two inputs and produces an output of 1 only if both inputs are 1. A deoxyribozyme with a stem-loop on each of its arms acts as an AND gate. Figure 4 shows the working of AND gate made with DNA [7] .The closed stems disable the enzyme (left), and only when both loops' matching input strands are added can the enzyme cleave substrates (middle). Truth table (right) summarizes the gate's function.

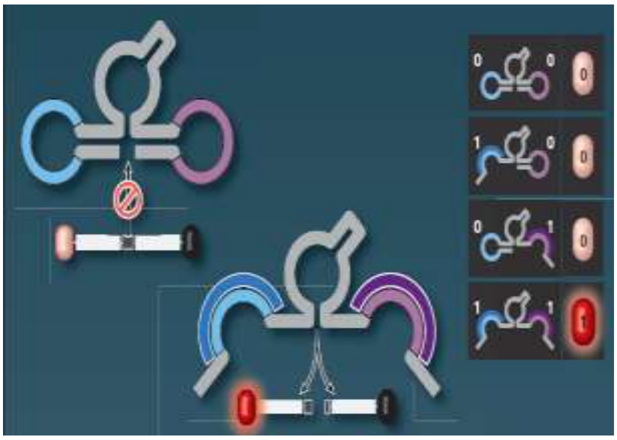


Figure.4 Working of DNA based AND gate

### 4.2 AND-AND-NOT Gate

A stem-loop controller on the "back" of a deoxyribozyme acts as a NOT input that inhibits the enzyme when the matching input strand is present. If the stem-loop's input strand is not present (0), the stem remains closed and the enzyme cleaves substrates to produce output strands, provided that the enzyme's arms are free (left). Figure 5 shows working of AND-AND-NOT Gate. When the input strand binds to the controller, the stem opens, deforming the enzyme core and rendering it inactive (middle). A deoxyribozyme with controllers on both arms and its back thus behaves as an AND-AND-NOT gate. The enzyme is active, cleaving substrates and thus producing the 1 output, only if inputs X (blue) AND Y (purple) AND NOT Z (yellow) are present.

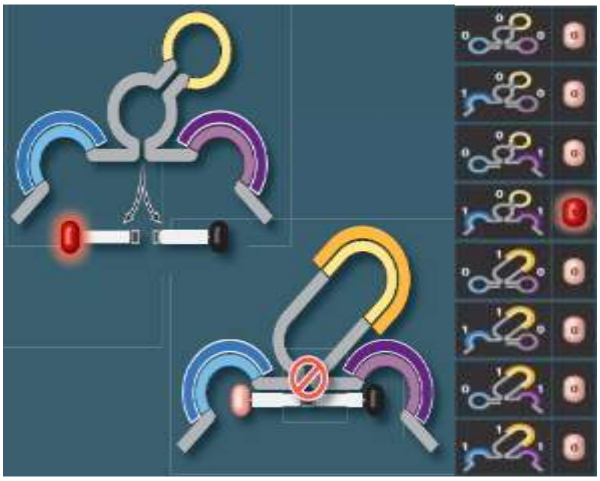


Figure.5 Working of DNA based AND-AND-NOT gate

## 5. DNA'S INFORMATION STORAGE AND PROCESSING CAPABILITIES

Nucleic Acids are used because of density, efficiency and speed. DNA molecules can store far more information than any existing computer memory chip. It has been estimated that a gram of DNA can hold as much information as a trillion CDs.

Most electronic computers operate linearly and they manipulate one block of data after another, biochemical reactions are highly in parallel: a single step of biochemical operations can be set up so that it affects trillions of DNA strands. While a DNA computer takes much longer than a normal computer to perform each individual calculation, it performs an enormous number of operations at a time and requires less energy and space than normal computers.

## 6. CONCLUSIONS

The field of DNA computing remains alive and promising, even as new challenges emerge. Most important among these are the uncertainty, because of the DNA chemistry, in the computational results, and the exponential increase in number of DNA molecules necessary to solve problems of interesting size. So, with the commencement of this field, lots more are expected in the betterment of Information technology. In addition, new paradigms based on molecular evolution have emerged from molecular biology to inspire new directions in DNA computing. As has been the case in the recent development of new fields, only further work will allow the determination of the proper scope and niche of DNA based computation and memory storage.

## 7. FUTURE WORK

Some centers of research in this area are at the University of Southern California at Los Angeles, with Dr. Adleman, Princeton, with Dr. Richard Lipton and his graduate students Dan Boneh and Eric Baum, and the NEC Research Institute in Princeton, NJ. With others elsewhere, they are developing new branches in this young field. Advancements are being made in cryptography. Researchers are working on decreasing error and damage to the DNA during the computations/reactions. The Princeton contingent has published papers on models for universal DNA computers, while others have described methods for doing addition and matrix multiplication with these computers.

Currently, molecular computing is a field with a great deal of potential, but few results of practical value. In the wake of Adleman's solution of the Hamiltonian path problem, there came a host of other articles on computation with DNA; however, most of them were purely theoretical. Currently, a functional DNA "computer" of the type most people are familiar with lies many years in the future. But work continues: in his article Speeding Up Computation via Molecular Biology Lipton shows how DNA can be used to construct a Turing machine, a universal computer capable of performing any calculation. While it currently exists only in theory, it's possible that in the years to come computers based on the work of Adleman, Lipton, and others will come to replace traditional silicon-based machines.

The field of DNA computing is truly exciting for the revolution it implies will occur within the next few years. It also demonstrates the current trend of merging and lack of distinction between the sciences, where a computer scientist can mess around with biology equipment and come up with something new and valuable.

## 8. REFERENCES

[1] R.W Keyes, "Silicon technology-in the chips for the future", Circuits and Devices Magazine, IEEE, 11,32-36 (1995)

[2] L. Adleman, "Molecular computations of solutions to combinatorial problems", Science 266, 1021-1024 (1994).

[3] S. Roweis and E. Winfree, "On the reduction of errors in DNA computation", Journal of Computational Biology, 6, 65-75 (1999).

[4] G. Păun, G. Rozenberg and A. Salomaa, "DNA Computing.New Computing Paradigms", Springer-Verlag, Berlin (1998)

[5] Gryko DT, Zhao, F, Yasseri AA, et al "Synthesis of thiol-derivatized ferrocene-porphyrins for studies of multibit information storage", J. Org. Chem, 65, 7356-7362 (2000)

[6] Roweis S, Winfree E, Burgoyne R, et al. "A sticker based model for DNA computation", J. Computational Bio, 5, 615-629 (1998)

[7] Macdonald Joanne, Stefanovic Darko and Stojanovic Milan N. "DNA computers for work and play", Scientific American, inc (2008)

International Journal of Computer Applications Technology and Research
Volume 4– Issue 3, 179 - 183, 2015, ISSN:- 2319–8656

# Implementation of 2D Optimal Barcode (QR Code) for Images

Awadhesh Kumar
AIET Jaipur,
India

Ajeet Kumar Nigam
Dr.KNMEC,
Modinagar, India

**Abstract:** Quick Response (QR) Code is very useful for encoding the data in an efficient manner. Here data capacity in 2D barcode is limited according to the various types of data formats used for encoding. The data in image format uses more space. The data capacity can be increased by compressing the data using any of the data compression techniques before encoding. In this paper, we suggest a technique for data compression which in turn helps to increase the data capacity of QR Codes generated for image. Finally, results are compared with the normal QR Codes to find the efficiency of the new technique of encoding followed by compression for generating optimal QR code.

**Keywords:** 2D barcodes, Data Capacity, Data Compression, Lossless Compression, QR Code

## 1. INTRODUCTION

Bar codes have become widely popular because of their reading speed, accuracy, and superior functionality characteristics. Barcodes can be divided as 1D, 2D and 3D. 1D barcodes can express information in horizontal direction only. Also, the data capacity is limited. 2D barcodes can hold data both in horizontal and vertical direction. As a result, the data capacity is 100 times more than the 1D barcode [1]. 3D barcode is usually engraved on a product or applied on a product so that the barcode has depth and thickness.

As bar codes became popular and their convenience universally recognized, the market began to call for codes capable of storing more information, more character types, and that could be printed in a smaller space. However, these improvements also caused problems such as enlarging the bar code area, complicating reading operations, and increasing printing cost. 2D Code emerged in response to these needs and problems [2].

QR Code is a kind of 2-D (two-dimensional) symbology developed by Denso Wave and released in 1994 with the primary aim of being a symbol that is interpreted by scanning equipment [3]. 2D bar codes can act like identifier (like in 1D) but takes less space. 2-D barcode minimizes the use of database; alternatively, it functions as database itself.

QR Code holds a considerably greater volume of information than a 1D bar code. These can be numeric, alphanumeric or binary data – of which up to 2953 bytes can be stored. Only a part of each QR bar code contains actual data, including error correction information. A large area of the QR code is used for defining the data format and version as well as for positioning, alignment and timing purposes. The smallest square dot or pixel element of a QR code is called a module. QR Codes have an empty area around the graphic. This quiet area is ideally 4 modules wide. Examination certificates can also use the QR Encoding techniques [4].

This paper proposes a method in which data capacity can be increased by first compressing the data and then encoding it. Actual requirement for compression arises when we need to encode image data into QR code. A lossless compression technique is proposed to increase the data capacity. For decoding the data, two steps will be followed: (i) de-compressing the data using the techniques which are just the reverse of compression technique used here and (ii) decoding the decompressed data. For this, the reverse technique used for encoding the data can be used.

## 2. LITERATURE SURVEY

QR Codes have already overtaken the conventional 1-D bar codes because of the capacity of data that can be stored by a 2-D barcode(QR Code) is much greater than that of conventional 1-D bar code. QR Code contains data both in horizontal and vertical directions. This stems in many cases from the fact that a typical 1-D barcode can only hold a maximum of 20 characters, whereas as QR Code can hold up to 7,089 characters [3]. QR Codes are capable of encoding the same amount of data in approximately one tenth the space of a traditional 1-D bar code. A great feature of QR Codes is that they do not need to be scanned from one particular angle, as QR Codes can be read regardless of their positioning. The data can be read successfully even if QR code is tampered while 1-D barcode can’t. QR Codes can be easily decoded with a smart phone with appropriate barcode reader software (for example:, Kaywa Reader, QRafter and I-Nigma etc.) [5]. Secure communication can also be established using QR Encoding techniques [6].

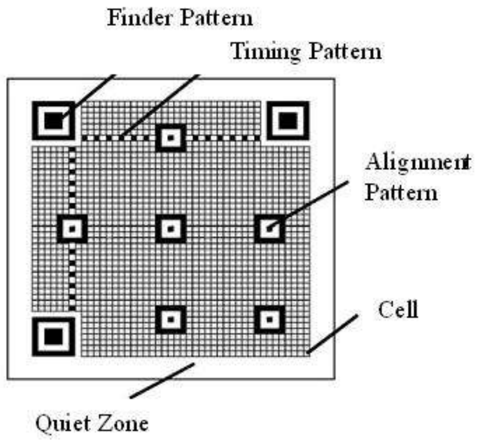


**Fig.1: Structure of QR Code**

## 2.1 Structure of QR Codes

QR Codes are actually black modules in square patterns on white background but many researchers have been working for colored QR code. It consists of the following areas having specific significance.

- Finder Pattern
- Alignment Pattern
- Timing Pattern
- Quiet Zone
- Data Area

Fig.1 shows the structure of QR Code. The significance of each area is as described as follows:
Each QR Code symbol consists of mainly two regions: an encoding region and function patterns. Function patterns consist of finder, timing and alignment patterns which does not encode any data. The symbol is surrounded on all the four sides by a quiet zone border [7]. A QR Code can be read even if it is tilted or distorted. The size of a QR Code can vary from 21 x 21 cells to 177 x 177 cells by four cell increments in both horizontal and vertical direction.

### *2.1.1 Finder Pattern*

This pattern can be used for detecting the position, size and angle of the QR Code. These can be determined with the help of the three position detection patterns (Finder Patterns) which are arranged at the upper left, upper right and lower left corners of the symbol as shown in Fig. 1.

### *2.1.2 Alignment Pattern*

The alignment pattern consists of dark 5x5 modules, light 3x3 modules and a single central dark module. This pattern is actually used for correcting the distortion of the symbol [8]. The central coordinate of the alignment pattern will be identified to correct the distortion of the symbol.

### *2.1.3 Timing Pattern*

The timing patterns are arranged both in horizontal and vertical directions. These are actually having size similar to one module of the QR Code symbol. This pattern is actually used for identifying the central co-ordinate of each cell with black and white patterns arranged alternately.

### *2.1.4 Quiet Zone*

This region is actually free of all the markings. The margin space is necessary for reading the bar code accurately. This zone is mainly meant for keeping the QR Code symbol separated from the external area [9]. This area is usually 4 modules wide.

### *2.1.5 Data Area*

It consists of both data and error correction code words. According to the encoding rule, the data will be converted into 0’s and 1’s. Then these binary numbers will be converted into black and white cells and will be arranged accordingly. Reed-Solomon error correction is also used here [10].

## 2.2 Data Capacity

The data storage capacity of QR Code is very large as compared to 1-D barcode. The number of characters that can be encoded as QR Code varies according to the type of information that is to be encoded. The various information types and the volume that the QR Code can hold are explained in Table 1.

**Table 1. Information Types and Volume of Data**

| Information Type | Volume of Data |
|---|---|
| Alphabets and Symbols | 4296 |
| Numeric Characters | 7089 |
| Binary Data (8 bit) | 2953 |
| Kanji Characters | 1817 |

## 2.3 Data Compression

In the history of computer science, data compression, source coding [1] or bit-rate reduction includes encoding information using fewer bits than the original representation. There are two kinds of data compression: lossy and lossless. Lossy compression reduces bits by identifying marginally important information and removing it. Lossless compression reduces bits by identifying and eliminating statistical redundancy. No information is lost in lossless compression.

Data Compression is very useful due to reducing the consumption of resources such as data space or transmission capacity. Because compressed data must be decompressed to be used, this extra processing imposes computational or other costs through decompression. The design of data compression schemes involve trade-offs among various factors, including the degree of compression, the amount of distortion introduced and the computational resources required to compress and uncompress the data [11].

Lossless data compression algorithms usually exploit statistical redundancy to represent data more concisely without losing information. Lossless compression is possible because most real-world data has statistical redundancy. The Lempel–Ziv (LZ) compression methods are among the most popular algorithms for lossless compression. DEFLATE is a variation on LZ which is optimized for decompression speed and compression ratio, but compression can be slow.

## 3. PROPOSED SCHEME

The efficiency of QR Codes is increased by applying compression before encoding. This paper focuses on the high capacity QR Codes for encoding image within barcode symbol. Our approach consists of mainly four steps for encoding:

1) Convert data in image format to base64 character format,
2) Compresses the data obtained in step 1.
3) Encodes the compressed data into a QR Code.

The whole process of converting image into QR Code is represented by flowchart in Fig 2.

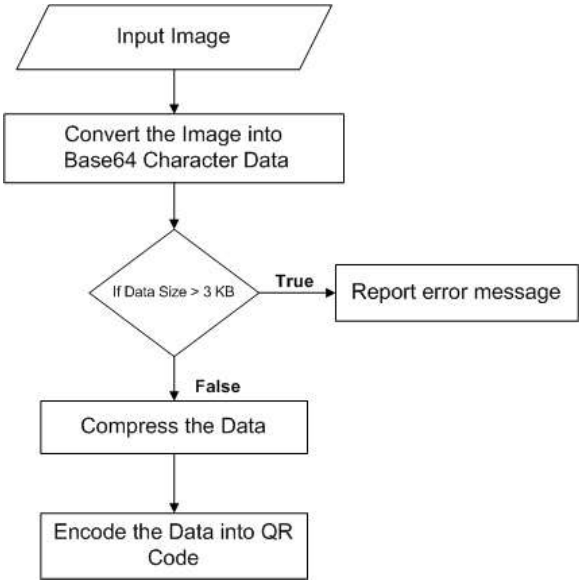


**Fig. 2 Flowchart for generation of QR Code for Image**

### 3.1 Compression Technique

This is a lossless approach to short the string such that after expansion we get the same string without any loss of character. In QR Code, we are using alphabets (A-Z), numbers(0-9) and some special characters ($,space,%,_,+,□,_,:,= ). Thus, a total of 45 characters we needed for QR Code. Normally, each character in the string is represented by 8-bits. So, if there are 100 characters in a string then we need 800 bits to represent that string. Our approach gives each 45 characters a particular code. The code is a fixed length code. We need 6 bits fixed length for each character to distinguish from one another, since $45<2^6$.

### 3.2 QR Encoding

The normal encoding of data is done through various steps such as [1]:

*1)* Analyse the data to be encoded. Convert the data to symbol characters. Find out the error correction and detection level.
*2)* Encode the data.
*3)* Error Correction Coding
*4)* Add reminder bits and data masking patterns.
*5)* Generate the format information and version information [12].

The entire process can be made clear with the help of the simple flowchart given below.

### 3.3 QR Decoding

Normally, QR decoding is done with the help of camera equipped mobile phones. Decoding process is just the reverse of the encoding procedure applied. We need to identify the quiet zone in order to decode the correct data. Alignment patterns help the decoding procedure by correcting the distortion of the symbol.

Kaywa reader is the most commonly used software to decode the original text. Image processing with J2ME is found to be more powerful. J2ME is designed to work on low-end devices in terms of processing power and memory capabilities. The various pre processing steps include gray scaling of the captured colour image, histogram stretching of the image, local adaptive thresholding, noise filtering, cropping, rotation correction and tilt correction. After pre processing, the finder patterns are detected and then the original data is decoded [5]. Another technique for decoding involves "edge to similar edge" estimation method is employed to check whether the detected bar-space pattern is correct [12].

### 3.4 Compression and Decompression of String

**Algorithm 1: Compression**

```
//String1 is the input and output.
1.  for each character in String1. do
       Extract a character from the String1.
       Convert this character into respective 6-bit length code.
       Append this code to String2.//Initially String2 is empty.
    end for
2.  Remainder = (Length of the String2) mod 8.
3.  Convert (Remainder+1) into 8 bits and add in the
    starting of String2.
4.  Add 0's equal to the Remainder at the end of String2.
5.  for each 8 characters in String2. do
        Extract 8 character from the String2.
        Convert this value into equivalent ASCII character.
        Store this character in String1.
    end for
6. Return String1
```

**Algorithm 2: Decompression**

```
//String1 is the input and output.
1. for each character in String1. do
        Extract a character from the String1.
        Convert this character into respective 8-bit length
        ASCII code.
        Append this code to String2.//Initially String2 is
        empty.
    end for
2.  Extract starting 8-bits from String2.
3.  Convert this into number.
4.  Remove _rst 8-bits from String2.
5.  Remove 0's equal to (number-1) from the end of
    String2.
6.  for each 6 characters in String2. do
        Extract 6 character from the String2.
        Convert it into equivalent code.
        Store this character in String1.
    end for
7. Return String1.
```

## 4. RESULTS

Using the approach discussed above, we are compressing the data before generating QR Code, and hence efficiency can be improved. This technique suggests simple ways to accomplish this task. The whole concept is implemented by designing a small application using C#.Net on Visual Studio 2008. Fig-3 shows the conversion of image in png format to Base64 character format.

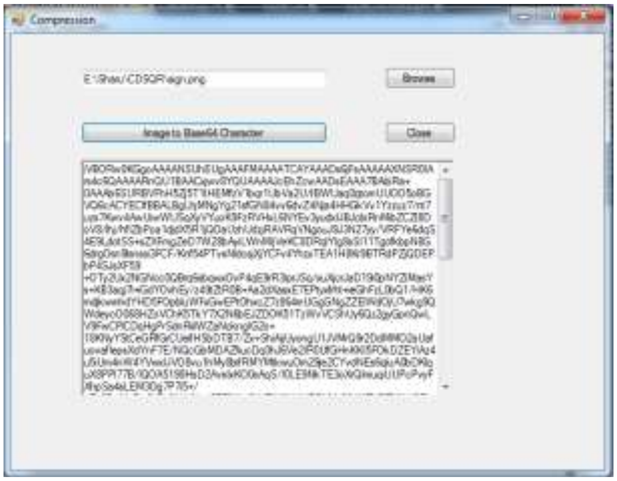


**Fig-3: Image to Base64 Character Format**

Fig-4 depicts the generation of QR code from image. The difference between uncompressed and compressed QR code can be seen in the Fig-4 below. The size of QR code without compression is greater (i.e. 26KB for sample signature image) than the QR code (i.e. 19KB for same sample image) of compressed data.

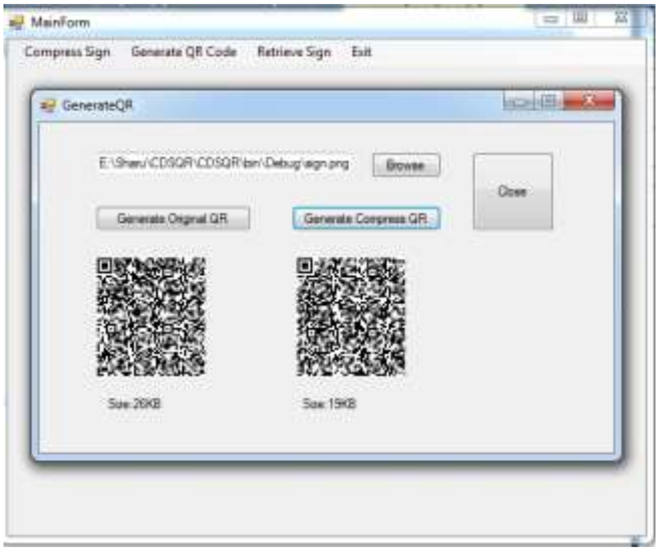
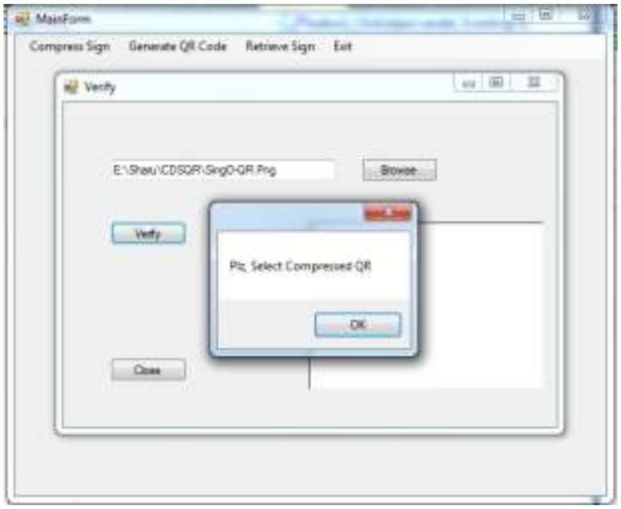


**Fig-4: Generation of QR Code for Image**

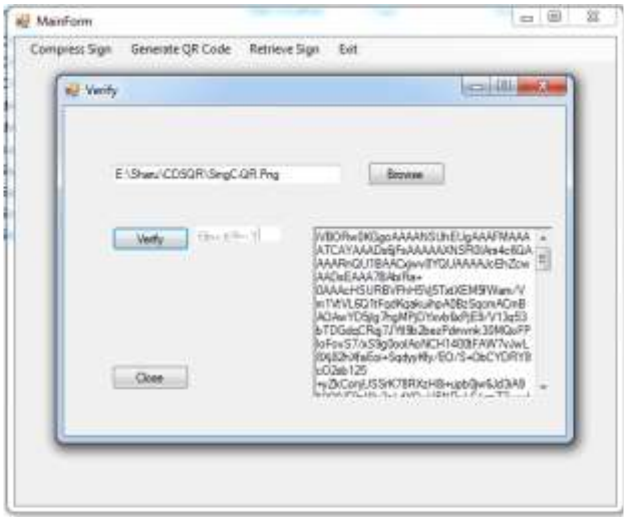
As our objective focuses on compression of data (after converting image to character format) to before generating QR code since QR code has limited data storage capacity. Fig-5(a) shows the error message when uncompressed image is browsed.



Plz, Select Compressed QR

**Fig-5(a): Verification of QR Code**

Fig-5(b) shows the successful verification of signature image encoded in QR code.



E:\Sharu\CDSQR\SingC.QR.Png

```
iVBORw0KGgoAAAANSUhEUgAAAFMAAA
ATCAYAAADw6jFsAAAAAXNSR0IArs4c6QA
AAARnQU1BAACxjwv8YQUAAAAJcEhZcw
AACxEAAA7BAbfFa+
DAAAcHSURBVFhH5V5TxtXEM9Wam/V
m1VtVL6Q1tFqdKgakuhpA0BzSqomACmB
AOAwYD6jg7hgMPjOYevb8xPjE8/V13q53
bTDGdsCRq7J/W8b2bezPdmvnk35MQoFP
loFoxS7/xS9g0oolAoNCH1400FAW7vJwL
l0lj82h/4faEoi+Sqdyy4fly/EO/S+ObCYDRY8
cO2ab125
+yZkConjUSSnK78RXzH8+upbQw6Jd3A8
```

Fig-5(b): Verification of QR Code

## 5. CONCLUSIONS

Normal QR Codes can compress only up to 4 KB of data. Using the techniques followed here, the data capacity can be increased drastically. As compared to the normal QR Codes, the data capacity of the QR Code after following technique was found to be more than 4 KB. Efficient data compression techniques can be used to store more than 4 KB of data inside a QR Code. A variety of data compression techniques can be used to obtain more data storage capacity. Comparing with the existing technologies used to generate bar codes, QR Codes were found to be of great advantage to the manufacturer because of its great data storage capacity, reading speed and accuracy. The data capacity was further improved by combining the most distinguishing features of compression and bar code generation. Using this novel technique of data compression followed by data encoding, the data storage capacity of QR Codes were increased drastically.

Currently only Smartphone's are technically equipped to do this. Many users that have mobile phones that have cameras are unable to get QR reading software for their phones. Future enhancements focus on QR Encoding of images which is more than 4 KB of size. Secure QR Coding can also be implemented using encryption techniques. Also, more advanced data compression techniques can be used to add more to the data capacity of the normal QR Codes.

## 6. REFERENCES

[1] Xiaofei Feng, Herong Zheng, "Design and Realization of 2D Color Barcode with High Compression Ratio" 2010 International Conference On Computer Design And Appliations (ICCDA 2010), 978-1-4244-7164-51, 2010 IEEE, 978-1-4244-7164-51, 2010 IEEE, Volume 1

[2] Nancy Victor, "Enhancing the Data Capacity of QR Codes by Compressing the Data before Generation", International Journal of Computer Applications (0975- 8887), Volume 60 - No.2, December 2012.

[3] Peter Kieseberg, Manuel Leithner, Martin Mulazzani, Lindsay Munroe, Sebastian Schrittwieser, Mayank Sinha, Edgar WeipplT. J., "QR Code Security"

[4] Chun-lei XIA, "Examination Certificate Based on Two-Dimensional Bar Code Technology", 2008 International Symposium on Computer Science and Computational Technology, 978-0-7695-3498-5/08/2008 IEEE DOI 10.1109/ISCSCT.2008.102

[5] Tasos Falas, Hossein Kashani, "Two-Dimensional Bar-code Decoding with Camera-Equipped Mobile Phones", Proceedings of the Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops(PerComW'07) 0-7695-2788-4/07/2007

[6] William Claycomb, Dongwan Shin, "Using A Two Dimensional Colorized Barcode Solution for Authentication in Pervasive Computing", 1-4244-0237-9/06/2006 IEEE

[7] Guenther Starnberger, Lorenz Froihofer and Karl M. Goeschka, "QR-TAN: Secure Mobile Transaction Authentication", 2009 International Conference on Availability, Reliability and Security, 978-0-7695-3564-7/09 IEEE DOI 10.1109/ARES.2009.96

[8] ISO/IEC 18004:2000 Information Technology - Automatic Identification and Data Capture Techniques – Barcode Symbology- QR Code (MOD), June 2000.

[9] Sarah Lyons and Frank R. Kschischang, "Two-Dimensional Barcodes for Mobile Phones", 25th Biennial Symposium on Communications, 978-1-4244-5711-3/10/2010

[10] R. Bose and D. Ray-Chaudhuri. On a class of errorcorrecting binary group codes*. Information and control, 3(1):68{79, 1960.

[11] David L. Donoho, Martin Vetterli, Fellow, IEEE, R. A. DeVore, and Ingrid Daubechies, Senior Member, IEEE," Data Compression and Harmonic Analysis", IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 44, NO. 6, OCTOBER 1998, 0018–9448/98$10.00 ã 1998 IEEE

[12] Hee I1 Hahn and Joung Koo Joung, "Implementation of Algorithm to Decode Two-Dimensional Barcode PDF-417", ICSP'O2 Proceedings, 0-7803-7488-6/02/$17.00 Q 2002 IEEE.

International Journal of Computer Applications Technology and Research
Volume 4– Issue 3, 184 - 187, 2015, ISSN:- 2319–8656

# Identifying Disease-Treatment Relation Using ML and NLP Approach

Dhamne Kalpesh
SIEM, Nashik, University of Pune, India

Mistari Sachin
SIEM, Nashik, University of Pune ,India

Dahite Sushil
SIEM, Nashik, University of Pune, India

Dalvi Suraj
SIEM, Nashik, University of Pune, India

R. S. Shirsath
SIEM, Nashik, University of Pune, India

**Abstract**: This paper presents the efficient machine learning algorithm and techniques used in extracting disease and treatment related sentences from short text published in medical papers. **.** In this paper better machine learning algorithms and techniques are used for extracting disease treatment relations from various medical related articles. The proposed system gives the user exactly the Disease and Treatment related sentences by avoiding unnecessary information, advertisements from the medical web page namely Medline. For making better medical decisions we can make use of this proposed technique.

**Keywords**: Machine Learning, Disease Treatment, Medline, Stemming Algorithm & Natural Language Processing Multinomial Naive Bayes algorithm

## 1. INTRODUCTION

Now a day's people are more aware about their health and healthcare. In spite of their busy schedules they want information regarding to their health for each and everything in a suitable way. People want Fast access to reliable information and in a manner that is suitable to their habits and work-flow. Medical field has grown in a wider to such an extent that information about latest discoveries are published day by day. The proposed system gives more reliable information and classification performances regarding medline database. Our proposed technique provides the doctors in making better medical decisions.

Medline is the database which contains the latest medical articles with disease and treatment information. Medical related article are very large. Now days to read complete articles published in these databases is not possible. It is a tedious work. So to avoid such problems we extract informative sentences related to disease, treatment, and three semantic relations between them like cure, prevent and side effect [3].

Our proposed system is to work with NLP and ML technique which has the task of identifying and disseminating information. The work that we present in this paper is focused on two tasks:

Task1: It automatically identifying sentences published in medical abstracts (Medline) containing information about diseases and treatments, and identifying semantic relations that exist between diseases and treatments. Task1 is done by using the "stemming algorithm".

Task2: It is focused on three semantic relations: Cure, Prevent, and Side Effect. This project will be more useful for common users who find difficulty in reading medline. It will be done by using "Multi nominal Naive Bayes algorithm" [1].

## 2. LITERATURE SURVEY

The work presents various Machine Learning (ML) and information for classification of short texts and finds the relation between diseases and treatments. As per the ML technique related information are shown in short texts when identifying relations between two entities such as diseases and treatment. It is better to identify and remove the sentence that does not contain information relevant to disease or treatments [4]. The remaining sentences can be classified according to the relation. It will be very difficult to identify the exact solution if everything is done in one step by classifying sentences based on interest and also including the sentences that do not provide relevant information. The data set used in this work is UMLS. The data set contains information from medline with all relevant information including diseases, treatments and eight relations between diseases and treatment. For mapping the words into semantic categories they used medical subject headings. In this work they compare different graphical models and generative models. For extracting semantic relations the Naive Bayes algorithm is used [3].

Although this system does not provide accurate results these systems are successful in this biomedical field. New rules have to be followed each time because the semantic rule based system has a disadvantage that lexicon changes from domain to domain. To get the good results semantic and syntactic based systems are combined so that they provide flexibility of syntactic information and good precision of semantic rule. Statistical approaches are used to solve different tasks. So that rules will extract automatically [2]. This method is used to solve different NLP tasks. So this approach works well even with fewer amounts of data. Considering relation extraction the rule checks whether the text information contains any relation or not. The statistical approach uses bag-of-words technique for the relation extraction. Some researchers combined this technique with POS which provides two sources of information such as relation between their specific contexts and entities. Since it is proved that simple technique can produce accurate results [4].

The traditional healthcare system is also becoming one that hug the Internet and the electronic world. In the healthcare domain, Electronic Health Records (EHR) is becoming the standard. Studies and researches show that the potential benefits of having an EHR system are:

Health information recording and clinical data repositories immediate access to patient diagnoses, allergies, and lab test results that enable better and time-efficient medical decisions; The EHR is web base application require server client communication it is very critical to maintain the connection for long term use [3]. These disadvantages overcome in propose system.

In this system individual sentences are considered as instances that are to be processed by the naive bayes classifier. Here each instance is considered as positive training set. Alternative relation extractions are made through relational learning. Relational learning involves parsing sentences and from the parsed sentences, parse tree is constructed. From the parsed tree grouping of the relevant sentence made. The extracted results are in proper form. The task of relation extraction was previously tackled in medical literature for gene-disorder association. It involves automatic extraction of relation between medical concepts. UMLS is used for finding the medical concept in sentence classification. Using semantic parser the sentences are automatically parsed. After applying semantic extraction a set of extraction, alteration, validation rules are applied to distinguish the actual semantic relation to be extracted [2].

## 3. PROPOSED SYSTEM

In this proposed system for easily identifying and collecting the healthcare information's published in various medical related midlines. The difficult problem here is that to know about a particular disease and its treatment people have to read the entire article. So in order to avoid such tedious work we provide them with an easy method of extracting only related or informative sentences from the medical articles. So here people get the information regarding a particular disease in the form of three semantic relations cure, prevent and side effects. We also find the symptoms focused in the articles related to disease. For removing the unwanted information from the articles we use many methods [1]. We drop out the stop words form the articles and then by using the stemming algorithm we remove the repetition of words and after that with the help of Multinominal Naive Bayes algorithm and semantic probability calculations extract the informative words. The application used is designed using dot net. The command named relation finder finds the relation between diseases and treatments and also provides us other information's. Whenever the button is pressed the user or doctor obtains the relevant information regarding that particular disease. In order to improve the quality of the result the process are performed in a sequential manner. To avoid uninformative sentences we first perform the stop word removal. We remove stop words such as a, an, is, any, about, of, if, in etc. from the text file. There are about 174 English stop words and we remove the entire stop words from the text file so that we can improve the quality of the result. By stop word removal content is reduced but quality is improved to a greater extend [4].

Next step is removal of repeated words from midline. We know that after the stop word removal process the remaining text file contains repeated words such as expressing and expressed etc. The stream of such words for example express is same for two words we combine both of them to one word so that the repetition can be avoided. all the repeated words are removed. This removal of repeated words will increase the quality of result to a much upper level. For the removing the repeated word we use the suffix stemming algorithm. There are many different stemming algorithms that we are known. From this different stemming algorithm here we use the suffix stripping algorithm [1].

We have to find the disease treatment relations from the remaining text document. In the form of three semantic relations cure, prevent and side effect. We also find the symptoms associated with the disease. For finding the semantic relations here the Multinominal Naive Bayes algorithm is used. The algorithms will easily finds the relation and we can easily display it to the end user. Naive Bayes algorithms drawbacks are overcome in Multinominal Naive Bayes.

In text classification we make use of this Multinominal Naive Bayes algorithm due to its computational advantage and simplicity. The algorithm is a specialized version of Naive Bayes [3]. The Naive Bayes algorithm is not used here because it suffers from some drawbacks. The major difference is that it assumes that the attributes of a given class are not dependent on each other. In some cases the attributes are related to each other. For example consider the classifier for in the case of assessing the risk of issuing a check book. For a worthy customer it will not be true to assume that there is no dependency or relation between that customers age, worth and education status. We prefer Multinominal Naive Bayes algorithm to avoid this problem. In naïve Bayes algorithm we calculate the semantic probability, which helps in easily recognizing the disease treatment relation.

The above described method of finding disease treatment relation can be used in various other applications in future work. The result quality can be found out with the help of f-measure values, recall and precision [1]. This will helps in saving the time of various users especially doctors by easily extracting the informative sentences from the medical related midline. There are various important modules used to perform these task and they are described as follows

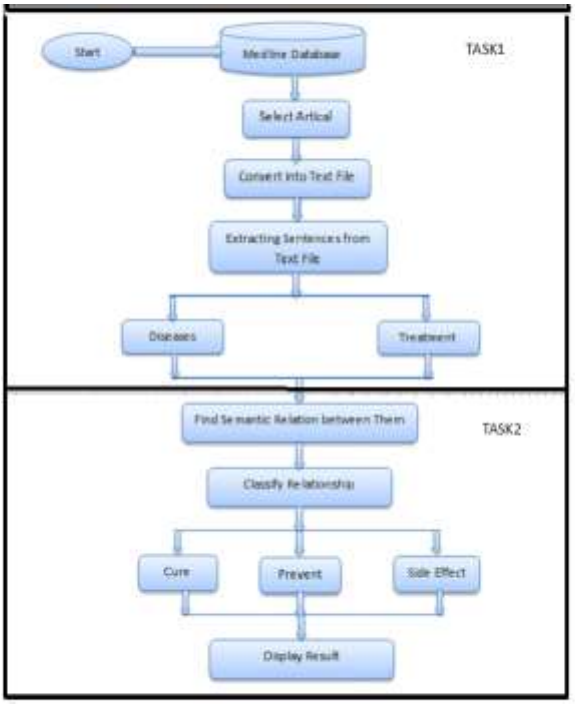


**Fig 1: System Architecture**

## 3.1 Html to text conversion:

The saved .html document is converted into a text file and is stored with .txt extension. This involves removing all the HTML tags, cascading style sheets and it retrieves, stores only the text content in the html file as text file with .txt extension. The obtained text file may be stored in location mentioned by the user [1].

## 3.2 Extraction of informative data:

Bag-Of-Word (BOW) representation is used for text classification where each of the word is used as feature for training the classifier in training dataset. BOW represents a document as a frequency of word occurrences. This classification and representation is unable to maintain any sequential information. In our proposed system, Weighted Bag-Of-Word representation is used to overcome the drawbacks of above mentioned problem of BOW [1].

### 3.2.1 *Stop Word Removal Process:*

As the first process we remove the stop words associated with each sentence. After the stop word removing the content size is reduces & document quality is improves. There are about 174 English stop words and all these when present in the document are successfully removed. Ex. a, an, is, for, the etc.

### 3.2.2 *Repeated words Removing:*

After the removal of stop words the remaining document contains repeated words and phrases and these words have to be removed from the contents extracted from above to improve the quality of the contents . To remove the repeated words and phrases we use the stemming algorithm. But the stemming algorithms has different types. Out of this algorithms here we make use of the suffix stripping algorithm. This may be done by removal of the various suffixes like -ED, -S, -ING, -ION, -IONS. For Ex.-

GENERALIZATIONS
GENERALIZATION
GENERALIZE
GENERAL

### 3.2.4 *Sentence Identification And Relationship Extraction:*

From the extracted contents, related with a disease and its treatment the three semantic relations such as cure, prevent and side effects are find out. To resolve the above problem and to result in efficient sentence identification Multi-nominal Naive Bayes classification algorithm is used in the proposed system. This algorithm is mostly used for the text classification. This algorithm finds the relations between disease and treatment and we can easily display it to the user by using related data set. Multi-nominal Naive Bayes classification (MNB) algorithm adopts parameter learning method [4].

### 3.2.4 *Output Performance Evaluation:*

This proposed system output is evaluated for various medline abstracts. The results we obtained shows informative sentences relevant to disease, treatments and the three disease treatment relations and symptoms related to the disease. The different data sets are used to extracting information associated to the three semantic relations that are cure, prevent and side effects. The predictable model is created to show the information regarding above mentioned semantic relations [3].

## 4. EVALUATION AND RESULT

The performance measurement is the efficiency of solution to given problem. It considers the performance of the trained models which yields the best predictive and classified results from the test dataset. Various standard measures gives the better score in relation extrication which is relevant to our problem domain. Ex. Accuracy which is measured by, Accuracy = total corrected corrections /total predictive [3].

From the recovered sentences, choose a testing dataset and a training dataset. ML setting worked on the training dataset and computed against the testing dataset. It gone very simple for selecting randomly in separation of data (Ex. 63% in training dataset, 37% in testing dataset) or may contains more complex sampling or extrication methods. But while processing on both datasets, they should be represent the solution for the problem.

## 4.1 Evaluation & performance Measures:

The important evaluation measures in ML algorithms are: accuracy, recall, precision and F-measure. As per the predictive concept of a model: confusion matrix (figure out the accuracy, cost of classification, F-measure). We can calculate ROC curve, and roles of every classifier is shown as a point on ROC curve. Whenever changes in the threshold value in the algorithms, cost matrix of classification, the point locations on ROC curve will alter respectively [1].

All above mentioned measures are evaluated to form a confusion matrix which includes information of the true classes, the actual classes and the classes prophecies by classifiers. The test dataset on which the predictive models are calculated include the true classes and the performance tries to recognize how many of true classes were forecasted by the model classifier. In the ML algorithms, focus needs towards the evaluation or performance measures that are used [4].

## 4.2 Efficiency of Identifying Informative Sentences:

This gives the evaluation for the first task, i.e. sentences are positive or negative (informative or non-informative). The ML algorithms are predicted for classification and represented as described above. Results of a classifier give the majority for improvement of datasets [7].

## 4.3 Efficiency of Identifying Semantic Relations:

Second task recognises sentences which contain information about 3 semantic relations like Cure, Prevent, and Side Effects. While performing operations on imbalanced data, F-measure is reported [3].

## 4.4 Performance of overall system:

In second task solution, to find the semantic relation we compare the results in 4 classes: 3 semantic relations and set of non-informative sentences. Performance of overall system can be computed as an evaluation measures of first task (results of classifiers) and second task (reporting F-measure results for imbalanced data).

## 4.5 Future Work:

These predictive models have stability and reliability for various tasks brings off on short texts in the medical field. The classification techniques gives more impact on ML algorithm results, but more informative classified results are the ones that regularly gives the best results to the users. The first task fulfilled in this paper is a task that has applications in recovery of important information, extricate the recovered information and text categorization. When more information is available for extrication or classification, there is an improvement in forecast results.

BOW method yields the best results in the text summarization or information extrication, can be more relevant when attaching more information from different kinds of things. The second task that performed can be seen as a task that could give profit by performing the first task first. To perform a handling or sorting of the sentences to get results for a relation classification. We adjoined the information from relation extraction that includes any of the three relations like disease, treatment and preventions, and excluded the sentences which did not contain above three semantic relations. This search is very useful in over out the positive and negative sentences before classification or extrication of those sentences.

## 5. CONCLUSION

It provides reliable & efficient medical information in short-text. The proposed work provides us only informative sentences and removes uninformative sentences from the medical related articles in a pipelined manner. This system helps users especially doctors in saving their time and they can know easily about a disease its treatment and symptoms and can analyses more about a various treatments associated with a particular disease. This system will be more useful to common users who want to know more about a disease in simpler manner.

## 6. REFERANCES

[1] Oana Frunza, Diana Inkpen, and Thomas Tran, Member "A Machine Learning Approach for Identifying Disease-Treatment Relations in Short Texts" june 2011.

[2] Ancy Sudhakar and Merin Meleet "A System for Extraction of Semantic Biomedical Relations Using Multinominal Naive Bayes Algorithm", March 2014.

[3] Janani.R.M.S and Ramesh V.," Efficient Extraction of Medical Relations using Machine Learning Approach", March 2013.

[4] Mouratis, S.Kotsiantis, "Increasing The Accuracy Of Discriminative Of Multinominal Bayesian Classifier In Text Classification**",** ICCIT'09 Proceedings Of The 2009 Fourth International Conference On Computer Science And Convergence Information Technology.

[5] R. Kohavi and F. Provost, "Glossary of Terms," Machine Learning, Editorial for the Special Issue on Applications of Machine Learning and the Knowledge Discovery Process, vol. 30, pp. 271-274, 1998.

[6] J. Li, Z. Zhang, X. Li, and H. Chen, "Kernel-Based Learning for Biomedical Relation Extraction," J. Am. Soc. Information Science and Technology, vol. 59, no. 5, pp. 756-769, 2008.

[7] T.K. Jenssen, A. Laegreid, J. Komorowski, and E. Hovig, "A Literature Network of Human Genes for High-Throughput Analysis of Gene Expression," Nature Genetics, vol. 28, no. 1, pp. 21-28, 2001

# Neural Network Based Context Sensitive Sentiment Analysis

S.Suruthi, M.Pradeeba, A.Sumaiya
Pondicherry Engineering College
Pudhucherry, India

J.I Sheeba
Pondicherry Engineering College
Pudhucherry, India

**Abstract**: Social media communication is evolving more in these days. Social networking site is being rapidly increased in recent years, which provides platform to connect people all over the world and share their interests. The conversation and the posts available in social media are unstructured in nature. So sentiment analysis will be a challenging work in this platform. These analyses are mostly performed in machine learning techniques which are less accurate than neural network methodologies. This paper is based on sentiment classification using Competitive layer neural networks and classifies the polarity of a given text whether the expressed opinion in the text is positive or negative or neutral. It determines the overall topic of the given text. Context independent sentences and implicit meaning in the text are also considered in polarity classification.

**Keywords**: Sentiment analysis, neural network, data mining, implicit meaning, soft computing.

## 1. INTRODUCTION

Sentiment analysis also known as emotion mining refers to the identification of the emotion in the given text. Sentiment analysis determines the attitude of a speaker or a writer. Sentiment analysis of short texts such as online political debate post is challenging because of the limited contextual information which normally contain. Measuring public opinions emerge as a challenging task [1]. This type of posts are increasingly used to determine consumer sentiment towards a brand. The existing literature on sentiment analysis uses various methods that are used in many text classification problems. Sentiment classification has received considerable attention in the natural language processing research community due to its many useful applications such as online product review classification and opinion summarization[2][3].

Sentiment analysis has become a mainstream research field since the early 2000s. Its impact can be seen in many practical applications, ranging from analyzing product reviews to predicting sales and stock markets using social media monitoring [4][5]. The users' opinions are mostly extracted either on a certain polarity scale, or binary (positive, negative); various levels of granularity are also taken into account, e.g., document-level, sentence-level, or aspect-based sentiment [6].

Techniques for sentiment analysis can be broadly categorized into two classes of approaches. The first class involves the application of a sentiment lexicon of opinion-related positive or negative terms to evaluate text in an unsupervised fashion [7]. The second class of approaches utilize a textual feature representation coupled with machine learning algorithms to derive the relationship between features of the text segment and the opinions expressed in the writing in a supervised fashion [8]. Models based upon such supervised techniques require a large training set of instances complete with class labels, to calibrate the models and tune the relevant parameters. Labels can be assigned to training instances manually through human evaluation of the text, or resources with explicitly defined ratings (such as the number of stars assigned in movie and product reviews) can be leveraged.

## 2. RELATED WORKS

Sentiment topic models for social emotion mining presents two sentiment topic models called Multi-label Supervised Topic Model (MSTM) and Sentiment Latent Topic Model (SLTM).Both MSTM and SLTM can be applied to the tasks of social emotion classification and generating social emotion lexicons. Both MSTM and SLTM allow to distinguish between different affective senses of the same word, and to discover meaningful topics evoking social emotions [9].

In Sentiment analysis in Czech social media using supervised machine learning describes in-depth research on machine learning methods for sentiment analysis of Czech social media. It evaluates state-of-the-art supervised machine learning methods for sentiment analysis. It also explores different pre-processing techniques and employ various features and classifiers. It significantly outperformed the baseline in three-class classification and achieved an F-measure of 0.69 using a combination of features and preprocessing techniques [10].

In Bi-view semi-supervised active learning for cross-lingual sentiment classification proposed a new model based on bi-view classification by combining active learning and semi-supervised co-training approaches in order to reduce the human labeling effort in cross-lingual sentiment classification. Both labeled and unlabeled data are represented in the source and target languages using a machine translation service to create two different views of data [11].

Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network makes several contributions to twitter sentiment analysis, demonstrated through application on a corpus of tweets related to the Justin Bieber brand. Using this approach, they state that ‘‘more than 80%’’ of tweets contain no sentiment. Their process resulted in a more accurate estimation of sentiment in experimentation on the Justin Bieber Twitter corpus [12]. A neural network based approach for sentiment classification in the blogosphere proposed a neural network based approach which combines the advantages of the Machine language techniques and the Information Retrieval techniques. The back-propagation neural network has been selected as the basic learner. This method uses the results of the Semantic Orientation indexes as the inputs for the BPN. The experimental results indicate that method can efficiently increase the performance of sentiment classification [13].

Document-level sentiment classification aims to automate the task of classifying a textual review, which is given on a single topic whether it is positive or negative. Except for some unbalanced data contexts, their experiments indicated that ANN produce superior or at least comparable results to SVM’s. ANN outperformed SVM by a statistically significant difference, even on the context of unbalanced data [14].

To overcome all the problems of the existing techniques the proposed framework has been introduced here.

## 3. THE PROPOSED FRAMEWORK

Figure 1 shows the overall architecture of the proposed work. The input data is taken from social media. In the first step, the input data is preprocessed that means stop words are removed. After preprocessing, feature extraction is done. In the feature Extraction, features like Noun, Adjective and Verbs will be extracted using QTag tool. All the words are tagged based on the POS Tagger. The occurrence of words are calculated in the frequency calculation step. The output of feature extraction has been given as input to the competitive layer neural network. Finally the positive, negative and neutral words are classified using competitive layer neural network. The topic of the given input is also identified in the proposed framework.

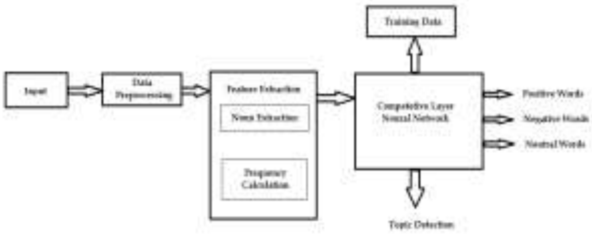


Figure. 1 Architecture of proposed framework

In the proposed framework in order to analyze the polarity of the given input the following steps are carried out.

- Input
- Data preprocessing
- Feature Extraction
- Competitive layer neural network

## 3.1 Input

The input data can be a conversation or comments or other types of unstructured documents for which the polarity is to be determined. This distribution contains the data used in the publication “Recognizing Stances in Ideological On-line Debates.” There are 6 directories, one for each debate domain. There are totally 84 debates and 3921 posts are available in this dataset [16].

## 3.2 Data preprocessing

Data preprocessing is an important step in the data mining process. First, the reduced feature set decreases the computing demands for the classifier, and, second, removing irrelevant features can lead to better classification accuracy [10]. Quality decisions must be based on quality data. Quality data is obtained through preprocessing. First step is to remove the stop words from the dataset. The stop words add little semantic value to a sentence. For example “to”, “I”, “has”, “the”, “be”, “or”, etc. are the stop words. Stop words bloat memory space and processing time without providing any extra value.

## 3.3 Feature Extraction

### *3.3.1 Noun Adjective and Verb Extraction*

Feature extraction is done to further reduce the data for achieving maximum efficiency in the output. All the sentiment words will be mostly bundled in the noun, adjective and verbal part of a sentence. So, the noun, verbs and the adjectives are extracted. Q tag tool is used to tag the sentence. This tagged words are parsed and only Noun Verbs and Adjectives are extracted.

### *3.3.2 Frequency Extraction*

The frequency extraction involves extracting the occurrence count of the words in the data. This is used to find the overall topic of the given data. Only those words whose counts are above the threshold set will be considered.

Those words that are noun, adjective and verb and those words that are above the given threshold will be sent as input to the neural network. Feature extraction is done in order to reduce the processing time in the neural net.

### 3.4 Competitive layer Neural Network

It is a type of unsupervised neural network. Unsupervised networks are trained by letting the network continually adjust itself to new updates [15]. Competitive layer neural network recognizes and group similar input vectors. Some data whose individual meaning changes when seen as a whole sentence. Even in such context independent situation sentiment analysis are made to find a positive or negative output. Neural networks are organized in layers which is made up of a number of interconnected nodes. These nodes are also called as neurons. Input layer communicates with one or more hidden layers where actual processing is done in hidden layer and it is connected to the output.

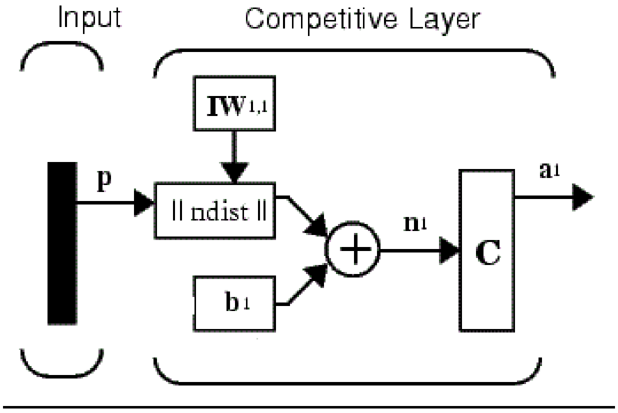


Figure. 2 Competitive layer neural network architecture.

The ‖ dist ‖ box in the figure 2 accepts the input vector p and the input weight matrix IW and gives a vector having elements S1. They are the negative of the distances between the input vector and vectors $iIW_{1,1}$ formed from the rows of the input weight matrix. Then n1 is the net input which is computed by adding the negative distance and the bias value. The competitive transfer function accepts a net input vector and returns 1 for all neurons that win the competition and 0 for those neurons that does not win the competition [15].

#### *3.4.1Kohonen learning rule:*

The weights of the winning neuron are adjusted with the help of Kohonen learning rule. If the $i^{th}$ neuron wins, all the elements of the $i^{th}$ row of the input weight matrix are adjusted as shown below.

$$iIW1,1(q)=iIW1,1(q-1)+\alpha(p(q)-iIW1,1(q-1))$$

The Kohonen rule will make the weights of a neuron to learn an input vector [15].

Thus, the neuron whose weight is closest to the input vector is changed such that it is even closer. Thus the winning neuron is more likely to win the competition next time and less likely to win when a very different input vector is presented. As more inputs are given, each neuron that is closest to a group of input vectors adjusts its weight vector toward the respective input vectors. Thus, the competitive network learns to categorize the input vectors it sees.

#### *3.4.2Bias leaning rule:*

One of the limitations of competitive networks is that some neurons will not be allocated. Some neuron weight may start very far away from input vectors and never win the competition, no matter how long the training is made. These neurons, is called as dead neurons, does not perform any function [15].

In order to stop this, biasing is given to the neurons. A positive bias, added to the negative distance, makes a distant neuron to win. To do this an average of neuron outputs is taken. It is equivalent to the percentages of times each output is 1[15]. This average is used to update, so that frequently active neurons bias value become smaller, and infrequently active neurons bias value become larger.

Infrequently active neurons bias value increase and so the input space to those neurons respond increases. As the input space increases, the infrequently active neuron responds and moves toward input vectors. Thus the neuron responds to the same number of vectors as other neurons.

## 4. CONCLUSION

The proposed framework mainly used for finding the polarity of the text using competitive layer neural network whether it is positive or negative or neural. It also considers implicit meaning and context independent data in the polarity classification and the overall topic of the given data is identified. This proposed framework will give better accuracy when compared to the other techniques.

## 5. ACKNOWLEDGMENTS

I would like to thank my guide Ms. J.I Sheeba, Assistant Professor, Department of Computer Science and Engineering, Pondicherry Engineering College for her valuable comments and suggestions in helping us to write this paper.

## 6. REFERENCES

[1] E. Cambria, B. Schuller, Y. Xia, & C. Havasi, (2013) Knowledge-based approaches to concept-level sentiment analysis: new avenues in opinion mining and sentiment analysis, IEEE Intel. Syst. Vol.28, pp. 15–21.

[2] Kang, H., Yoo, S. J., & Han, D. (2012). Senti-lexicon and improved Naïve Bayes algorithms for sentiment analysis of restaurant reviews. Expert Systems with Applications, Vol. 39(5), pp. 6000-6010.

[3] Ku, L. W., Liang, Y. T., & Chen, H. H. (2006, March). Opinion Extraction, Summarization and Tracking in News and Blog Corpora. In AAAI spring symposium: Computational approaches to analyzing weblogs Vol. 100107.

[4] Stepanov, E. A., & Riccardi, G. (2011, December). Detecting general opinions from customer surveys., 2011 IEEE 11th International Conference on Data Mining Workshops (ICDMW) pp. 115-122

[5] Yu, L. C., Wu, J. L., Chang, P. C., & Chu, H. S. (2013). Using a contextual entropy model to expand emotion words

and their intensity for the sentiment classification of stock market news. Knowledge-Based Systems, Vol. 41, pp. 89-97.

[6] Sadegh, M., Ibrahim, R., & Othman, Z. A. (2012). Opinion mining and sentiment analysis: A survey. International Journal of Computers & Technology, Vol. 2(3), pp. 171-178.

[7] Turney, P. D. (2002, July). Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In Proceedings of the 40th annual meeting on association for computational linguistics pp. 417-424.

[8] Pang, B., Lee, L., & Vaithyanathan, S. (2002, July). Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing, Vol. 10, pp. 79-86.

[9] Rao, Y., Li, Q., Mao, X., &Wenyin, L. (2014). Sentiment topic models for social emotion mining. Information Sciences, Vol. 266, pp. 90-100.

[10]Habernal, I., Ptácek, T., & Steinberger, J. (2013, June). Sentiment analysis in Czech social media using supervised machine learning. In Proceedings of the 4th workshop on computational approaches to subjectivity, sentiment and social media analysis pp. 65-74.

[11]Hajmohammadi, M. S., Ibrahim, R., &Selamat, A. (2014). Bi-view semi-supervised active learning for cross-lingual sentiment classification. Information Processing & Management, Vol. 50(5), pp.718-732.

[12]Ghiassi, M., Skinner, J., &Zimbra, D. (2013). Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network.Expert Systems with applications,Vol. 40(16), pp. 6266-6282.

[13]Chen, L. S., Liu, C. H., & Chiu, H. J. (2011). A neural network based approach for sentiment classification in the blogosphere. Journal of Informetrics, Vol.5(2), pp.313-322.

[14]Moraes, R., Valiati, J. F., &Neto, W. P. G. (2013). Document-level sentiment classification: An empirical comparisonbetween SVM and ANN. Expert Systems with Applications, Vol.40 (2), pp.621-633.

[15] http://in.mathworks.com/help/nnet/competitive-layer.html

[16] http://www.aclweb.org/anthology/W10-0214.

International Journal of Computer Applications Technology and Research
Volume 4– Issue 3, 192 - 196, 2015, ISSN:- 2319–8656

# Health Monitoring System of Elderly using Wireless Sensor Network

Kothuru Anudeep
Dept. of ECE
Vardhaman College of Engineering
Hyderabad, India

S.Srinivas
Dept. of ECE
Vardhaman College of Engineering
Hyderabad,India

**Abstract**: Wireless-sensor-network-based home monitoring system for elderly activity behaviour involves functional assessment of daily activities. In this paper, we report a mechanism for estimation of elderly well-being condition based on usage of house-hold appliances connected through various sensing units. We define a two new wellness functions to determine the status of the elderly on performing essential daily activities. The modernized system for monitoring and evaluating the essential daily activities was tested at homes for four different elderly persons living alone and the results are encouraging in determining wellness of the elderly.

**Keywords**: Activities of daily living, elder care, home monitoring, smart home, wellness, wireless sensor network .

## 1. INTRODUCTION

WSN based health monitoring system for patient activities like body temperature, heartbeat ,blood pressure etc . By the using sensing units we can get the updates from the patient .An intelligent home monitoring system based on ZigBee wireless sensors network has been designed and developed to monitor and evaluate the patient details ./Health conditions of an elderly  people can be unsafe situations during regular works. Here is an software to get a health monitoring system to determine patient health care system.[4]Also, the system interprets all the essential elderly activities such as regular activities. Basically, the system function based on the usage data of electrical and non-electrical appliances within a home. At the hardware level, wireless sensor network with ZigBee [1] components are connected in the form of mesh topology, and a central coordinator of the sensing units collect data from the sensors connected to various appliances. In this system ,a required number of sensors for monitoring the daily activities of the elderly have been used. A smart sensor coordinator collects data from the sensing units and forward to the computer system for data processing. Collected sensor data are of low level information containing only status of the sensor as active or inactive and identity of the sensor. To sense the activity behavior of elderly in real time, the next level software module will analyze the collected data by following an intelligent mechanism at various level of data abstraction based on time and sequence behavior of sensor usage.

## 1.1 OVER VIEW OF THE SYSTEM

### *1.1.1 Block diagram of Patient Section*

Above block diagram representing the patient section, in that we are continuously monitoring the patient information by using wireless sensor networks, i.e., Temperature and heart beat of patient. And that information we can forward to the control room section by using Zigbee technology[1].

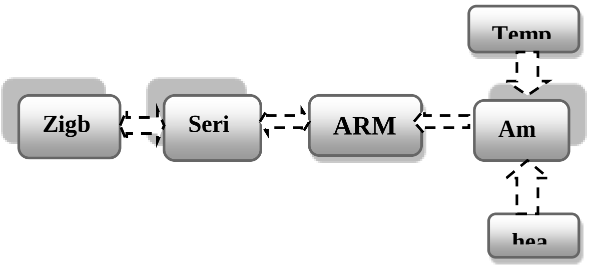


Figure 1. Patient Section

### *1.1.2 Control Section*

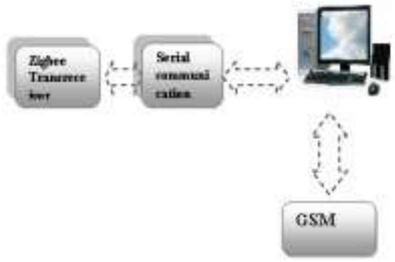
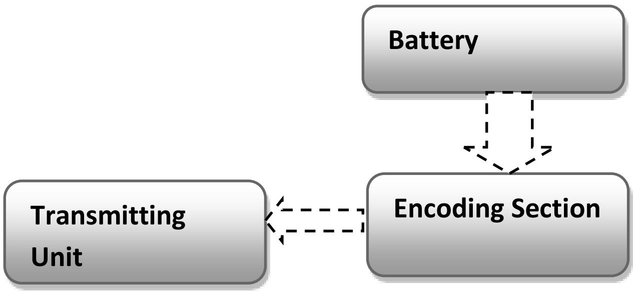


Figure 2. Control system

From patient section we transferring the information that is received by control room section through Zigbee and in pc it will check and if any abnormal condition occur it will send SMS to user by using GSM technology.

### *1.1.3 Medical section*



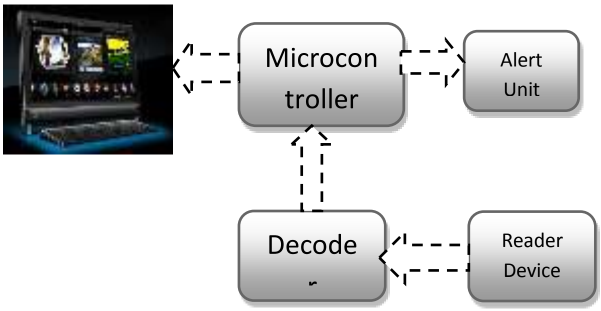


Figure 3. Medical Section

From the medical alert we are providing the patient information by using RF technology means, Person section consist RF transmitter followed Encoder. It transmitting the signal and the Medical section receiving the signal using RF receiver and if the person unauthorized means it will give the alert.

## 1.2 Micro Controller (ARM7) Family

The ARM7 family includes the ARM7TDMI-S, ARM7TDMI, ARM720T and ARM7EJ-S processors. The ARM7TDMI core is the most widely used 32-bit embedded RISC microprocessor solution. Optimized for power-sensitive applications and cost , the ARM7TDMI solution provides the low power consumption, high performance and small size needed in portable, embedded applications. The synthesizable version of the ARM7TDMI core, available in both VHDL and VERILOG, ready for compilation into processes supported by in-house or commercially available synthesis libraries. The ARM720T hard macro cell contains the ARM7TDMI core, Memory Management Unit (MMU), and a 8kb unified cache that allows the use of protected execution spaces and virtual memory. This macro cell is compatible with leading operating systems including Windows CE, Linux, SYMBIAN OS, and palm OS.

The ARM7EJ-S processor comprises of ARM's latest DSP extensions and enabling acceleration of java-based applications. Compatible with the ARM9E™, ARM9™, and ARM10™ families, and Strong-Arm® architecture software written for the ARM7TDMI processor is 100% binary-compatible with other members of the ARM7 family and forwards-compatible with the ARM9E, ARM9 and ARM10 families, as well as products in Intel's Strong ARM and xscale architectures. This gives designers a software-compatible processor with strong price-performance points. Supporting the ARM architecture today includes:

- Operating systems such as Linux, Windows CE, SYMBIAN OS and palm OS.
- More than 50 real-time operating systems including qnx and wind river's vx works

## 1.3 LPC2148 Microcontroller

LPC2148 Microcontroller Architecture. which offers very low power consumption and high performance . The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles and the instruction set and related decode mechanism are much simpler than those of micro programmed Complex Instruction Set Computers (CISC). This simplicity results in a high instruction throughput and impressive real-time interrupt response from a small and cost-efficient processor core.

Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory. The ARM7TDMI-S processor also employs a unique architectural strategy known as Thumb, which makes it ideally suited to high-volume applications with memory restrictions, or applications where code density is an issue.

The key idea behind Thumb is that of a super-reduced instruction set. Essentially, the ARM7TDMI-S processor has two instruction sets:

- The standard 32-bit ARM set.
- A 16-bit Thumb set.

The Thumb set's 16-bit instruction length allows it to approach twice the density of standard ARM code while retaining most of the ARM's performance advantage over a traditional 16-bit processor using 16-bit registers. This is possible because Thumb code operates on the same 32-bit register set as ARM code. Thumb code is able to provide up to 65 % of the code size of ARM, and 160 % of the performance of an equivalent ARM processor connected to a 16-bit memory system.

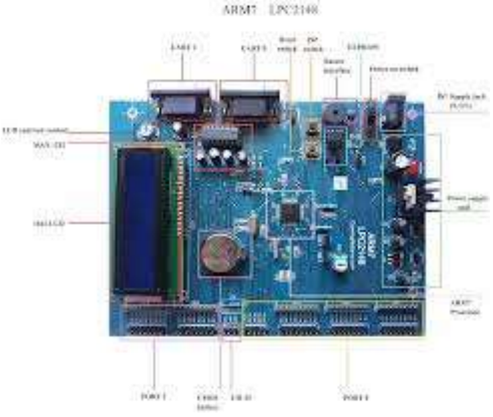


Figure 4. GSM Module

## 1.4 Heart Beat Sensors

The sensor consists of a light source and photo detector; light is shone through the tissues and variation in blood volume alters the amount of light falling on the detector. The source and detector can be mounted side by side to look at changes in reflected light or on either side of a finger or earlobe to detect changes in transmitted light. The particular arrangement here uses a wooden clothes peg to hold an infra red light emitting diode and a matched phototransistor. The infra red filter of the phototransistor reduces interference from fluorescent lights, which have a large AC component in their output

The skin may be illuminated with visible (red) or infrared LEDs using transmitted or reflected light for detection. The very small changes in reflectivity or in transmittance caused by the varying blood content of human tissue are almost invisible. Various noise sources may produce disturbance signals with amplitudes equal or even higher than the amplitude of [5] the pulse signal. Valid pulse measurement therefore requires extensive preprocessing of the raw signal. The setup described here uses a red LED for transmitted light illumination and a pin Photodiode as detector. With only slight changes in the preamplifier circuit the same hard- and software could be used with other illumination and detection concepts. The detectors photo current (AC Part) is converted to voltage and amplified by an inexpensive operational amplifier (LM358). A PIC16F877 microcontroller converts the analog signal with 10 bits resolution to a digital signal. An average is calculated from 250 readings taken over a 20 milliseconds period (This equals one period of the European power line frequency of 50 Hz).

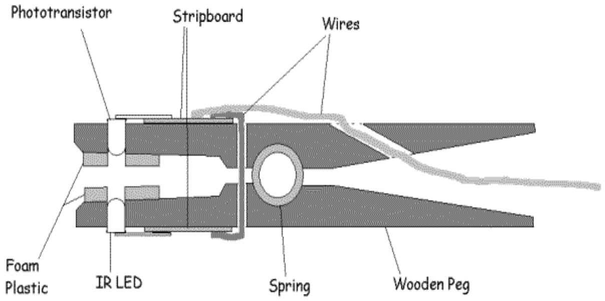


Figure 5. Heart Beat Sesnor

## 1.5 Temperature Sensor

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of ±1⁄4°C at room temperature and ±3⁄4°C over a full −55 to +150°C temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only 60 µA from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a −55° to +150°C temperature range, while the LM35C is rated for a −40° to +110°C range (−10° with improved accuracy). The LM35 series is available packaged in hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-220 package.

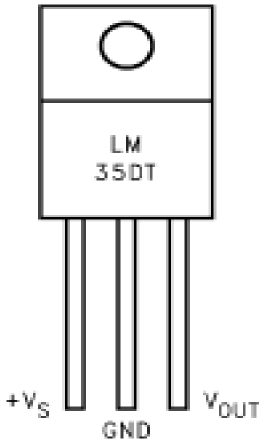


Figure 6. Temperature Sensor

# 2. WIRELESS COMMUNICATION

## 2.1 GSM Technology

To achieve important information of cars, one GSM Module is added into the car security system. Siemens TC35I GSM modem can quickly send SMS messages to appointed mobile phone or SMS server [3]. So the owner and the police can be informed at the first time. If another GPRS module is added in, the image data could also sent to information.

A GSM modem can be an external device or a PC Card or a PCMCIA Card. Typically, an external GSM modem is connected to a computer through a USB cable or a serial cable. A GSM modem in the form of a PC Card or a PCMCIA Card is designed for a laptop computer, which should be inserted into one of the PC Card or a PCMCIA Card slots of a laptop computer. A GSM modem needs a SIM card in order to operate. As mentioned in the earlier sections of the SMS tutorial, computers use a common set of standard AT commands to control the both the GSM and dial-up modems. GSM modem can be used just like a dial-up modem.

Figure 7. GSM Module

#### *2.1.1 SMS commands*

- **AT+CIMI**
  Note: Read the IMSI

- **AT+CMGS=”+33146290800”**
  Note: Send a message in text mode

- **AT+CMGR=3**
  Note: Read it

- **AT+CMGD=3**
  Note: Delete it Note: Message

## 2.2 Zigbee Module

The Xbee/Xbee-PRO RF Modules [1] are designed to operate within the ZigBee protocol and support the unique needs of low-cost, low-power wireless sensor networks. The modules require minimal power and provide reliable delivery of data between remote devices. The modules operate within the ISM 2.4 GHz frequency band and are compatible with the following:

- XBee RS-232 Adapter
- XBee RS-232 PH (Power Harvester) Adapter
- XBee RS-485 Adapter
- XBee Analog I/O Adapter
- XBee Digital I/O Adapter
- XBee Sensor Adapter
- XBee USB Adapter
- XStick
- Connect Port X Gateways
- 00 XBee Wall Router.

The XBee/XBee-PRO ZB firmware release can be installed on XBee modules. This firmware is compatible with the ZigBee 2007 specification, while the ZNet 2.5 firmware is based on Ember's proprietary "designed for ZigBee" mesh stack (EmberZNet 2.5). ZB and ZNet 2.5 firmware are similar in nature,(1) but not over-the-air compatible. Devices running ZNet 2.5 firmware cannot talk to devices running the ZB firmware.

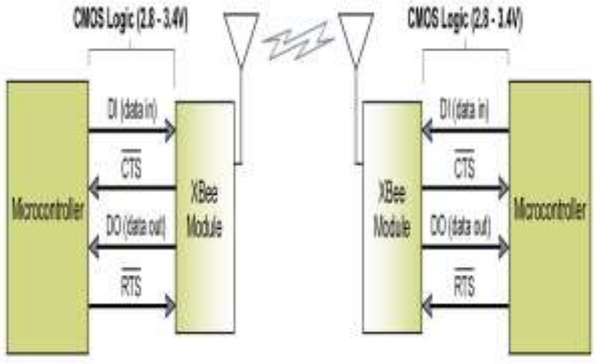


Figure 8. Zigbee Section

# 3. MEDICAL SECTION

## 3.1 RF Transmitter and Receiver

Radio Frequency, any frequency within the electromagnetic spectrum associated with radio wave propagation. When an RF current is supplied to an antenna, an electromagnetic field is created that then is able to propagate through space. Many wireless technologies are based on RF field propagation Radio Frequency: The 10 kHz to 300 GHz frequency range that can be used for wireless communication. Also used generally to refer to the radio signal generated by the system transmitter, or to energy present from other sources that may be picked up by a wireless receiver(6).

## 3.2 Transmitter

The TWS-434 extremely small, and are excellent for applications requiring short-range RF remote controls(2). The transmitter module is only 1/3 the size of a standard postage stamp, and can easily be placed inside a small plastic enclosure.

TWS-434: The transmitter output is up to 8mW at 433.92MHz with a range of approximately 400 foot (open area) outdoors. Indoors, the range is approximately 200 foot, and will go through most walls.

#### *3.2.1.1 (c)Receiver*

***3.2.1.2 RWS-434: The receiver also operates at 433.92MHz, and has a sensitivity of 3uV. The WS-434 receiver operates from 4.5 to 5.5 volts-DC, and has both linear and digital outputs.***

#### *3.2.1.3 (d)Transmitting and receiving*

Full duplex or simultaneous two-way operation is not possible with these modules. If transmit and receive module are in close proximity and data is sent to a remote receive module(2) while attempting to simultaneously receive data from a remote transmit module, the receiver will be overloaded by its close proximity transmitter. This will happen even if encoders and decoders are used with different address settings for each transmitter and receiver pair. If two way communications is required, only half duplex operation is allowed.(6)

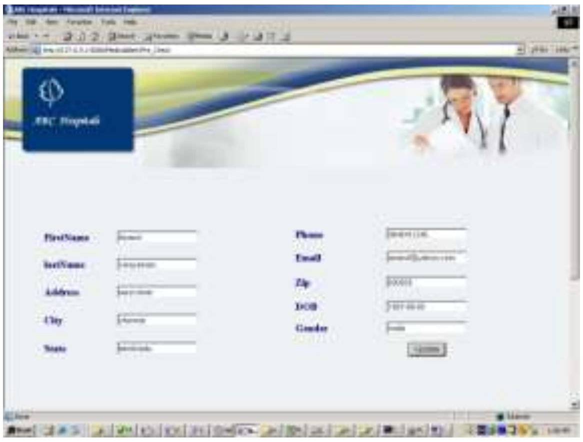


Figure 9. Screens for Medical Data Storing

## 4. CONCLUSION

In this time, model biotelemetry system is being implemented into working solution. Nevertheless, there is space for improvements in both concept and implementation details of this system. Model biotelemetry system is currently designed for indoor use by one patient only. More nearby instances of inner part of model biotelemetry system managed by single outer part of system are possible, but there exists one to one mapping between patient and ZigBee network [1]. Future improvements may include support for outdoor operation with communication implemented using 3G mobile technology [3] and patient's tracking by GPS system. With advancements in low-power high-density FPGA solutions, FPGA programmable system on chip technology seems to be promising for purpose of this biotelemetry system.

## 5. REFERENCES

[1] Safaric S., Malaric K., "ZigBee wireless standard",Multimedia Signal Processing and Communications, 48th International Symposium ELMAR-2006, Zadar, Croatia,June 2006.

[2] Ze Zhao and Li Cui, "EasiMed: A remote health care solution", Proceeding of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai,China, September 2005.

[3] Krejcar, O., Janckulik, D., Motalova, L., Kufel, J., "Mobile Monitoring Stations and Web Visualization of Biotelemetric System - Guardian II". In EuropeComm 2009. LNICST vol. 16, pp. 284-291. R. Mehmood, et al. (Eds). Springer, Heidelberg (2009).

[4] Krejcar, O., Janckulik, D., Motalova, L., "Complex Biomedical System with Mobile Clients". In The World Congress on Medical Physics and Biomedical Engineering 2009, WC 2009, September 07-12, 2009 Munich, Germany. IFMBE Proceedings, Vol. 25/5. O.Dössel, W. C. Schlegel, (Eds.). Springer, Heidelberg. (2009).

[5] Krejcar, O., Janckulik, D., Motalova, L., Frischer, R., "Architecture of Mobile and Desktop Stations for Noninvasive Continuous Blood Pressure Measurement". In The World Congress on Medical Physics and Biomedical Engineering 2009, WC 2009, September 07-12, 2009 Munich, Germany. IFMBE Proceedings, Vol. 25/5. O. Dössel, W. C. Schlegel, (Eds.). Springer, Heidelberg. (2009).

[6] Idzkowski A., Walendziuk W.: Evaluation of the static posturograph platform accuracy, Journal of Vibroengineering, Volume 11, Issue 3, 2009, pp.511-516, ISSN 1392 - 8716M. Penhaker , M. Cerny, L. Martinak, et al. HomeCare "Smart embedded biotelemetry system" Book Series IFMBE proceedings World Congress on Medical Physics and Biomedical Engineering, AUG 27-SEP 01, 2006 Seoul, SOUTH KOREA, Volume: 14 Pages: 711-714, 2007, ISSN: 1680- 0737, ISBN: 978-3-540-36839-7.

International Journal of Computer Applications Technology and Research
Volume 4– Issue 3, 197 - 201, 2015, ISSN:- 2319–8656

# A Comparison between FPPSO and B&B Algorithm for Solving Integer Programming Problems

Mahmoud Ismail
Department of Operations Research
Faculty of Computers and Informatics
Zagazig University
El-Zera Square, Zagazig, Sharqiyah
Egypt

Ibrahim El-Henawy
Department of Computer science
Faculty of Computers and Informatics,
Zagazig University
El-Zera Square, Zagazig, Sharqiyah
Egypt

**Abstract**: Branch and Bound technique (B&B) is commonly used for intelligent search in finding a set of integer solutions within a space of interest. The corresponding binary tree structure provides a natural parallelism allowing concurrent evaluation of sub-problems using parallel computing technology. Flower pollination Algorithm is a recently-developed method in the field of computational intelligence. In this paper is presented an improved version of Flower pollination Meta-heuristic Algorithm, (FPPSO), for solving integer programming problems. The proposed algorithm combines the standard flower pollination algorithm (FP) with the particle swarm optimization (PSO) algorithm to improve the searching accuracy. Numerical results show that the FPPSO is able to obtain the optimal results in comparison to traditional methods (branch and bound) and other harmony search algorithms. However, the benefits of this proposed algorithm is in its ability to obtain the optimal solution within less computation, which save time in comparison with the branch and bound algorithm.

**Keywords**: Branch and bound, flower pollination Algorithm; meta-heuristics; optimization; the particle swarm optimization; integer programming.

## 1. INTRODUCTION

We ask that authors follow some simple guidelines. This document is a template. An electronic copy can be downloaded from the journal website. For questions on paper guidelines, please contact the conference publications committee as indicated on the conference website. Information about final paper submission is available from the conference website

The real world optimization problems are often very challenging to solve, and many applications have to deal with NP-hard problems [1]. To solve such problems, optimization tools have to be used even though there is no guarantee that the optimal solution can be obtained. In fact, for NP problems, there are no efficient algorithms at all. As a result of this, many problems have to be solved by trial and errors using various optimization techniques [2]. In addition, new algorithms have been developed to see if they can cope with these challenging optimization problems. Among these new algorithms, many algorithms such as particle swarm optimization, cuckoo search and firefly algorithm, have gained popularity due to their high efficiency. In this paper we have used IBACH algorithm for solving integer programming problems. Integer programming is NP-hard problems [3-10]. The name linear integer programming is referred to the class of combinatorial constrained optimization problems with integer variables, where the objective function is a linear function and the constraints are linear inequalities. The Linear Integer Programming (also known as LIP) optimization problem can be stated in the following general form:

$$\text{Max } cx \qquad (1)$$
$$\text{s.t. } Ax \leq b, \qquad (2)$$
$$x \in Zn \qquad (3)$$

where the solution x ∈ Zn is a vector of n integer variables: x = (x1, x2 , …, xn)T and the data are rational and are given by the m×n matrix A, the 1×n matrix c, and the m×1 matrix b. This formulation includes also equality constraints, because each equality constraint can be represented by means of two inequality constraints like those included in eq. (2).

Integer programming addresses the problem raised by non-integer solutions in situations where integer values are required. Indeed, some applications do allow a continuous solution. For instance, if the objective is to find the amount of money to be invested or the length of cables to be used, other problems preclude it: the solution must be discrete [3]. Another example, if we are considering the production of jet aircraft and x1 = 8.2 jet airliners, rounding off could affect the profit or the cost by millions of dollars. In this case we need to solve the problem so that an optimal integer solution is guaranteed.

The possibility to obtain integer values is offered by integer programming: as a pure integer linear programming, in which all the variables must assume an integer value, or as a mixed-integer linear programming which allows some variables to be continuous, or a 0-1 integer model, all the decision variables have integer values of zero or one[10].

A wide variety of real life problems in logistics, economics, social sciences and politics can be formulated as linear integer optimization problems. The combinatorial problems, like the knapsack-capital budgeting problem, warehouse location problem, travelling salesman problem, decreasing costs and machinery selection problem, network and graph problems, such as maximum flow problems, set covering problems, matching problems, weighted matching problems, spanning trees problems and many scheduling problems can also be solved as linear integer optimization problems [11-14].

## 2. BRANCH AND BOUND

The branch and bound is the divide and conquer method. We divide a large problem into a few smaller ones. (This is the “branch” part). The conquering part is done by estimate how good a solution we can get for each smaller problems (to do this, we may have to divide the problem further, until we get a problem that we can handle), that is the “bound” part. The branch and bound algorithm is able to be parallelized by distributing computation of subproblems on multiple computing nodes. Parallel branch and bound algorithms with the master-worker algorithm, where a single master process dispatches tasks to multiple worker processes, have been proposed in many literatures [3]. In master-worker algorithm, a single master process dispatches subproblems, which correspond to leaf nodes on the search tree, to multiple worker processes and receives the computed results from the worker processes. The computed results contain the best upper bound of the objective function, and subproblems that have generated by branching and have not been pruned on a worker process. Also, the parallel algorithm with the hierarchical master-worker paradigm is proposed to improve performance on large-scale computing environment.

Exact integer programming techniques such as cutting plane techniques [15-17]. The branch and the bound both have high computational cost, in large-scale problems [18-19]. The branch and the bound algorithms have many advantages over the algorithms that only use cutting planes. One example of these advantages is that the algorithms can be removed early as long as at least one integral solution has been found and an attainable solution can be returned although it is not necessarily optimal. Moreover, the solutions of the LP relaxations can be used to provide a worst-case estimate of how far from optimality the returned solution is. Finally, the branch method and the bound method can be used to return multiple optimal solutions.

Since integer linear programming is NP-complete, for that reason many problems are intractable. So instead of the integer linear programming, the heuristic methods must be used. For example, Swarm intelligence metaheuristics, amongst which an ant colony optimization, artificial bee colony optimization particle swarm optimization [20-24].Also Evolutionary algorithms, differential evolution and tabu search were successfully applied into solving integer programming problems [25-27]. Heuristics typically have polynomial computational complexity, but they do not guarantee that the optimal solution will be captured. In order to solve integer programming problems, most of the heuristics truncate or round the real valued solutions to the nearest integer values. In this paper, an improved version of flower pollination algorithm is applied to integer programming problems and the performance was compared with other harmony search algorithms.

This paper is organized as follows: after introduction, the original branch and bound algorithm is introduced in section 2. The flower pollination algorithm is briefly introduced in section 3. Section 4 introduces the particle swarm optimization algorithm. Section 5 introduces the meaning of chaos. While the results are discussed in section 6. Finally, conclusions are presented in section 7.

## 3. FLOWER POLLINATION ALGORITHM

Flower Pollination Algorithm (FPA) was founded by Yang in the year 2012. Inspired by the flow pollination process of flowering plants are the following rules [28]:

**Rule 1**: Biotic and cross-pollination can be considered as a process of global pollination process, and pollen-carrying pollinators move in a way that obeys Le'vy flights.
**Rule 2**: For local pollination, a biotic and self-pollination are used.
**Rule 3**: Pollinators such as insects can develop flower constancy, which is equivalent to a reproduction probability that is proportional to the similarity of two flowers involved.
**Rule 4**: The interaction or switching of local pollination and global pollination can be controlled by a switch probability p∈[0,1], with a slight bias toward local pollination.

In order to formulate updating formulas, we have to convert the aforementioned rules into updating equations. For example, in the global pollination step, flower pollen gametes are carried by pollinators such as insects, and pollen can travel over a long distance because insects can often fly and move in a much longer range [56].Therefore, Rule 1 and flower constancy can be represented mathematically as:

$$x_i^{t+1} = x_i^t + \gamma L(\lambda)(x_i^t - B) \quad (1)$$

Where $x_i^t$ is the pollen i or solution vector xi at iteration t, and B is the current best solution found among all solutions at the current generation/iteration. Here γ is a scaling factor to control the step size. In addition, L(λ) is the parameter that corresponds to the strength of the pollination, which essentially is also the step size. Since insects may move over a long distance with various distance steps, we can use a Le'vy flight to imitate this characteristic efficiently. That is, we draw L > 0 from a Levy distribution:

$$L \sim \frac{\lambda\Gamma(\lambda)\sin(\pi\lambda/2)}{\pi}\frac{1}{S^{1+\lambda}}, (S >> S_0 > 0) \quad (2)$$

Here, Γ(λ) is the standard gamma function, and this distribution is valid for large steps s > 0.
Then, to model the local pollination, both Rule 2 and Rule 3 can be represented as

$$x_i^{t+1} = x_i^t + U(x_j^t - x_k^t) \quad (3)$$

Where $x_j^t$ and $x_k^t$ are pollen from different flowers of the same plant species. This essentially imitates the flower constancy in a limited neighborhood. Mathematically, if $x_j^t$ and $x_k^t$ comes from the same species or selected from the same population, this equivalently becomes a local random walk if we draw U from a uniform distribution in [0, 1].Though Flower pollination activities can occur at all scales, both local and global, adjacent flower patches or flowers in the not-so-far-away neighborhood are more likely to be pollinated by local flower pollen than those faraway. In order to imitate this, we can effectively use the switch probability like in Rule 4 or the proximity probability p to switch between common global pollination to intensive local pollination. To begin with, we can use a naive value of p = 0.5as an initially value. A preliminary parametric showed that p = 0.8 might work better for most applications [28].

The basic steps of FP can be summarized as the pseudo-code shown in Figure 1.

```
Flower pollination algorithm

Define Objective function f (x), x = (x_1, x_2, ..., x_d)
Initialize a population of n flowers/pollen gametes with random solutions
Find the best solution Bin the initial population
Define a switch probability p ∈ [0, 1]
Define a stopping criterion (either a fixed number of generations/iterations or accuracy)
while (t <MaxGeneration)
    for i = 1 : n (all n flowers in the population)
        if rand <p,
            Draw a (d-dimensional) step vector L which obeys a L´evy distribution
            Global pollination via $x_i^{t+1} = x_i^t + L(B - x_i^t)$
        else
            Draw U from a uniform distribution in [0,1]
            Do local pollination via $x_i^{t+1} = x_i^t + U(x_j^t - x_k^t)$
        end if
        Evaluate new solutions
        If new solutions are better, update them in the population
    end for
    Find the current best solution B
end while
Output the best solution found
```

Fig. 1 Pseudo code of the Flower pollination algorithm

## 4. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) was developed by Kennedy and Eberhartin 1995 based on the swarm behavior such as fish and bird schooling in nature [29]. Since then, PSO has generated much wider interests and forms an exciting, ever expanding research subject called swarm intelligence. This algorithm searches the space of an objective function by adjusting the trajectories of individual agents, called particles, as the piecewise paths formed by positional vectors in a quasi stochastic manner. The movement of a swarming particle consists of two major components: a stochastic component and a deterministic component. Each particle is attracted toward the position of the current global best g and its own best location xi* in history, while at the same time it has a tendency to move randomly. Let xi and vi be the position vector and velocity of particle i, respectively. The new velocity vector is determined by the following formula:

$$v_i^{t+1} = v_v^t + c_1 r_1 (g - x_i^t) + c_2 r_2 (x_x^* - x_i^t) \qquad (4)$$

Wherer1 and r2are two random vectors and each entry takes the values between 0and 1. The parameters c1and c2are the learning parameters or acceleration constants, which can typically be taken as, say, c1≈c2≈ 2.The initial locations of all particles should be distributed relatively uniformly so that they can sample over most regions, which is especially important for multimodal problems. The initial velocity of a particle can be taken as zero, i.e. vit=0 =0. The new positions can then be updated by:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \qquad (5)$$

Although vi can be any value, it is usually bounded in some range [0,vmax].

## 5. CHAOS

Generating random sequences with longer periods and good consistency is very important for easily simulating complex phenomena, sampling, numerical analysis, decision making and especially in heuristic optimization [30]. Its quality determines the reduction of storage and computation time to achieve a desired accuracy [31]. Chaos is a deterministic, random-like process found in a nonlinear, dynamical system, which is non-period, non-converging and non-bounded. Moreover, it depends on its initial condition and parameters [32-34]. Applications of chaos has several disciplines including operations research, physics, engineering, economics, biology, philosophy and computer science [35-37].

Recently chaos has been extended to various optimization areas because it can more easily escape from local minima and improve global convergence in comparison with other stochastic optimization algorithms [34-38]. Using chaotic sequences in flower pollination Algorithm can be helpful to improve the reliability of the global optimality, and they also enhance the quality of the results.

### 5.1 Chaotic Maps

At random-based optimization algorithms, the methods using chaotic variables instead of random variables are called chaotic optimization algorithms (COA) [34]. In these algorithms, due to the non-repetition and ergodicity of chaos, it can carry out overall searches at higher speeds than stochastic searches that depend on probabilities [43-48]. To resolve this issue, herein one-dimensional and non-invertible maps are utilized to generate chaotic sets. We will illustrate some of well-known one-dimensional maps as:

*5.1.1 The Logistic map*

The Logistic map is defined by:

$$Y_{n+1} = \mu Y_n (1 - Y_n)\ Y \in (0,1)\ 0 < \mu \leq 4 \quad (6)$$

*5.1.2 The Sine map*

The Sine map is written as the following equation:

$$Y_{n+1} = \frac{\mu}{4} \sin(\pi Y_n)\ Y \epsilon\ (0,1)\ 0 < \mu \leq 4 \qquad (7)$$

*5.1.3 The iterative chaotic map*

The iterative chaotic map with infinite collapses is described as:

$$Y_{n+1} = \sin\left(\frac{\mu\pi}{Y_n}\right)\ \mu \in (0,1) \qquad (8)$$

*5.1.4 The Circle map*

The Circle map is expressed as:

$$Y_{n+1} = Y_n + \alpha - (\frac{\beta}{2\pi}) \sin(2\pi Y_n)\ mod\ 1 \quad (9)$$

*5.1.5 The Chebyshev map*

The family of Chebyshev map is written as the following equation:

$$Y_{n+1} = \cos(k cos^{-1}(Y_n))\ \ Y \in (-1,1) \qquad (10)$$

### *5.1.6 The Sinusoidal map*

Sinusoidal map can be represented by

$$Y_{n+1} = \mu Y_k^2 \sin(\pi Y_n) \qquad (11)$$

### *5.1.7 The Gauss map*

The Gauss map is represented by:

$$Y_{n+1} = \begin{cases} 0 & Y_n = 0 \\ \frac{\mu}{Y_n} \ mod\ 1 & Y_n \neq 0 \end{cases} \qquad (12)$$

### *5.1.8 The Sinus map*

Sinus map is formulated as follows:

$$Y_{n+1} = 2.3(Y_n)^{2\sin(\pi Y_n)} \qquad (13)$$

### *5.1.9 The Dyadic map*

Dyadic map Also known as the dyadic map bit shift map, 2x mod 1 map, Bernoulli map, doubling map or saw tooth map. Dyadic map can be formulated by a mod function:

$$Y_{n+1} = 2Y_n \ mod\ 1 \qquad (14)$$

### *5.1.10 The Singer map*

Singer map can be written as:

$$Y_{n+1} = \mu(7.86Y_n - 23.31Y_n^2 + 28.75Y_n^3 - 13.3Y_n^4) \qquad (15)$$

μ between 0.9 and 1.08

### *5.1.11 The Tent map*

Tent map can be defined by the following equation:

$$Y_{n+1} = \begin{cases} \mu Y_n & Y_n < 0.5 \\ \mu(1 - Y_n) & Y_n \geq 0.5 \end{cases} \qquad (16)$$

## 6. NUMERICAL RESULTS

Several examples have been done to verify the weight of the proposed algorithm. The initial parameters setting of the algorithms is as follows: HMS=50 and itermax=1000, HMCR = 0.9; PARmax = 1; PARmin =0.1; bwmax = 1; bwmin = 0.01. The results of FPPSO algorithm are conducted from 50 independent runs for each problem and measured according to the best values in these runs.The selected chaotic map for all examples is the Sinusoidal map, whose equation is shown below: $Y_{n+1} = \mu Y_k^2 \sin(\pi Y_n)$ (16)

Where n is the iteration number.

**Table 1.** Optimal solution of selected problems

| **Exact Method** | | **The Best Solution** | | |
|---|---|---|---|---|
| No.of Variables | Optimal Solution | Optimal values | BB | FPPSO |
| 2 | 55 | $X_i$=(4,3) | 55 | **55** |
| 3 | 26 | $X_i$=(2,1,6) | 22 | **26** |
| 5 | 9 | $X_i$=(1,1,0,0,0) | 7 | **9** |
| 10 | 9 | $X_i$=(0,2,0,2,3,1,0, 0,2,3) | 7 | **9** |
| 20 | 16 | $X_i$=(0,0,0,0,0,0,0, 0,0,1,4,0,4,3,0,2,4 ,0,3,0) | 12 | **16** |
| 30 | 446 | $X_i$=(0,0,0,0,0,0,0, 0,0,16,20,4,4,0,3, 0,0,0,24,3,0,0,0,0, 0,4,0,1,0,8) | 401 | **446** |

Table 1 shows the results of FPPSO algorithm are privileged compared with the results of Branch and bound (B&B). In comparison with exact values we find that the results of FPPSO algorithm are very close to the exact values of selected problems under the study. If a large number of variables are to be found, then it is hard to go past the classical methods. More usually, though, users will choose to use the proposed algorithm, to save their own time and to gain reliability. for example when we solved test problem number 6 by proposed algorithm it took time 7 seconds ,but when we solved it by branch and bound(exact method) it took time 396 seconds .

The reason for getting better results than the other algorithm considered is that the search power of FP algorithm. Adding to this, using PSO algorithm improves the performance of the algorithm.

## 7. CONCLUSIONS

This paper has introduced an improved flower pollination Algorithm by blending with practical swarm optimization algorithm for solving integer programming problems. Several examples have been used to prove the effectiveness of FPPSO. FPPSO algorithm managed to solve a large scale of problems that traditional method could not solve due to exponential growth in time and space complexities. The solution procedure will not face the same time waste in going through non-converging iterations as traditional methods do. FPPSO algorithm is superior to B&B in terms of both efficiency and success rate. This implies that FPPSO is potentially more powerful in solving NP-hard problems.

who have contributed towards development of the template.

## 8. REFERENCES

[1] L. A. Wolsey, Integer programming, IIE Transactions, vol. 32, pp. 2-58, 2000.

[2] G. B. Dantzig, Linear programming and extensions: Princeton university press, 1998.

[3] G. L. Nemhauser and L. A. Wolsey, Integer and combinatorial optimization vol. 18: Wiley New York, 1988.

[4] E. Beale, "Integer programming," in *Computational Mathematical Programming*, ed: Springer, 1985, pp. 1-24.

[5] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*: Courier Dover Publications, 1998.

[6] H. Williams, "Logic and Integer Programming, International Series in Operations Research & Management Science," ed: Springer, 2009.

[7] A. Schrijver, Theory of linear and integer programming: Wiley. com, 1998.

[8] D. Bertsimas and R. Weismantel, Optimization over integers vol. 13: Dynamic Ideas Belmont, 2005.

[9] J. K. Karlof, *Integer programming: theory and practice*: CRC Press, 2005.

[10] M. Jünger, T. Liebling, D. Naddef, G. Nemhauser, W. Pulleyblank, G. Reinelt, *et al.*, *50 Years of Integer Programming 1958–2008*: Springer, Berlin, 2010.

[11] D.-S. Chen, R. G. Batson, and Y. Dang, *Applied integer programming: modeling and solution*: Wiley. com, 2011.

[12] K. L. Hoffman and M. Padberg, "Solving airline crew scheduling problems by branch-and-cut," *Management Science*, vol. 39, pp. 657-682, 1993.

[13] J. D. Little, K. G. Murty, D. W. Sweeney, and C. Karel, "An algorithm for the traveling salesman problem," *Operations research*, vol. 11, pp. 972-989, 1963.

[14] M. Grotschel and L. Lovász, "Combinatorial optimization," *Handbook of combinatorics*, vol. 2, pp. 1541-1597, 1995.

[15] R. E. Gomory, "Outline of an algorithm for integer solutions to linear programs," *Bulletin of the American Mathematical Society*, vol. 64, pp. 275-278, 1958.

[16] R. E. Gomory, "An algorithm for integer solutions to linear programs," *Recent advances in mathematical programming*, vol. 64, pp. 260-302, 1963.

[17] R. E. Gomory, "Early integer programming," *Operations Research*, pp. 78-81, 2002.

[18] J. Tomlin, "Branch and bound methods for integer and non-convex programming," *Integer and Nonlinear Programming, American Elsevier Publishing Company, New York*, pp. 437-450, 1970.

[19] S. Rouillon, G. Desaulniers, and F. Soumis, "An extended branch-and-bound method for locomotive assignment," *Transportation Research Part B: Methodological*, vol. 40, pp. 404-423, 2006.

[20] M. Tuba, "Swarm intelligence algorithms parameter tuning," in *Proceedings of the 6th WSEAS international conference on Computer Engineering and Applications, and Proceedings of the 2012 American conference on Applied Mathematics*, 2012, pp. 389-394.

[21] R. Jovanovic and M. Tuba, "An ant colony optimization algorithm with improved pheromone correction strategy for the minimum weight vertex cover problem," *Applied Soft Computing*, vol. 11, pp. 5360-5366, 2011.

[22] R. Jovanovic and M. Tuba, "Ant colony optimization algorithm with pheromone correction strategy for the minimum connected dominating set problem," *Computer Science and Information Systems*, vol. 10, pp. 133-149, 2013.

[23] B. Akay and D. Karaboga, "Solving integer programming problems by using artificial bee colony algorithm," in *AI* IA 2009: Emergent Perspectives in Artificial Intelligence*, ed: Springer, 2009, pp. 355-364.

[24] M. G. Omran, A. Engelbrecht, and A. Salman, "Barebones particle swarm for integer programming problems," in *Swarm Intelligence Symposium, 2007. SIS 2007. IEEE*, 2007, pp. 170-175.

[25] G. Rudolph, "An evolutionary algorithm for integer programming," in *Parallel Problem Solving from Nature—PPSN III*, ed: Springer, 1994, pp. 139-148.

[26] M. G. Omran and A. P. Engelbrecht, "Differential evolution for integer programming problems," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, 2007, pp. 2237-2242.

[27] F. Glover, "Tabu search—part II," *ORSA Journal on computing*, vol. 2, pp. 4-32, 1990.

[28] X-S. Yang, Flower pollination algorithm for global optimization, Unconventional Computation, lecture Notes in Computer Science, Vol. 7445, pp. 240-249,2012.

[29] J. Kennedy and R. Eberhart, Particle swarm optimization, in Proceedings of IEEE International Conference on Neural Network, pp. 1942–1948, December 1995.

[30] O. Abdel-Raouf, , M.Abdel-Baset, and I. El-Henawy. "An Improved Chaotic Bat Algorithm for Solving Integer Programming Problems." *International Journal of Modern Education and Computer Science (IJMECS)* 6.8 (2014): 18.

[31] O. Abdel-Raouf, , M. Abdel-Baset, and I. El-henawy. "A New Hybrid Flower Pollination Algorithm for Solving Constrained Global Optimization Problems." *International Journal of Applied* 4.2 (2014): 1-13.

[32] O. Raouf, I. El-henawy, and M. Abdel-Baset. "A novel hybrid flower pollination algorithm with chaotic harmony search for solving sudoku puzzles." *International Journal of Modern Education and Computer Science* 3 (2014): 38-44.

[33] O. Abdel-Raouf, I. El-henawy and M. Abdel-Baset "chaotic Harmony Search Algorithm with Different Chaotic Maps for Solving Assignment Problems "International Journal of Computational Engineering & Management, Vol. 17, pp. 10-15 ,2014.

[34] O. Abdel-Raouf, I. El-henawy and M. Abdel-Baset. "Chaotic Firefly Algorithm for Solving Definite Integral", IJITCS, vol.6, no.6, pp.19-24, 2014.

[35] O. Abdel-Raouf, I. El-henawy and M. Abdel-Baset "Improved Harmony Search with Chaos for Solving Linear Assignment Problems", IJISA, vol.6, no.5, pp.55 61, 2014.

[36] O. Abdel-Raouf, , M. Abdel-Baset, and I. El-henawy. "An Improved Flower Pollination Algorithm with Chaos." 2014.

[37] I. El-henawy, , O. Abdel-Raouf, and M. Abdelbaset. "Improved harmony search algorithm with chaos for solving definite integral." *International Journal of Operational Research* 21.2 (2014): 252-261.

International Journal of Computer Applications Technology and Research
Volume 4– Issue 3, 202 - 208, 2015, ISSN:- 2319–8656

# Efficient & Lock-Free Modified Skip List in Concurrent Environment

Ranjeet Kaur
Department of Computer Science and Application
Kurukshetra University, Kurukshetra
Kurukshetra, Haryana

Pushpa Rani Suri
Department of Computer Science and Application
Kurukshetra University, Kurukshetra
Kurukshetra, Haryana

**Abstract**: In this era the trend of increasing software demands continues consistently, the traditional approach of faster processes comes to an end, forcing major processor manufactures to turn to multi-threading and multi-core architectures, in what is called the concurrency revolution. At the heart of many concurrent applications lie concurrent data structures. Concurrent data structures coordinate access to shared resources; implementing them is hard. The main goal of this paper is to provide an efficient and practical lock-free implementation of modified skip list data structure. That is suitable for both fully concurrent (large multi-processor) systems as well as pre-emptive (multi-process) systems. The algorithms for concurrent MSL based on mutual exclusion, Causes blocking which has several drawbacks and degrades the system's overall performance. Non-blocking algorithms avoid blocking, and are either lock-free or wait-free.

**Keywords**: skip-list, CAS, Modified Skip List, concurrency, lock-free

## 1. INTRODUCTION

Modern applications require concurrent data structures for their computations. Concurrent data structures can be accessed simultaneously by multiple threads running on several cores. Designing concurrent data structures and ensuring their correctness is a difficult task, significantly more challenging than doing so for their sequential counterparts. The difficult of concurrency is aggravated by the fact that threads are asynchronous since they are subject to page faults, interrupts, and so on. To manage the difficulty of concurrent programming, multithreaded applications need synchronization to ensure thread-safety by coordinating the concurrent accesses of the threads. At the same time, it is crucial to allow many operations to make progress concurrently and complete without interference in order to utilize the parallel processing capabilities of contemporary architectures. The traditional approach that helps maintaining data integrity among threads is to use lock primitives. Mutexes, semaphores, and critical sections are used to ensure that certain sections of code are executed in exclusion[1]

To address these problems, researchers have proposed non-blocking algorithms for shared data objects. Nonblocking methods do not rely on mutual exclusion, thereby avoiding some of these inherent problems. Most non-blocking implementations guarantee that in any infinite execution, some pending operation completes within a finite number of steps. Nonblocking algorithms have been shown to be of big practical importance in practical applications [2][3]

In the previous work we presented the concurrent access of Modified skip list with locking techniques[12] , as we have discussed the limitation due to locking method ,we present the lock free access of modified skip list data structure. This one is the initial efforts in this direction.

## 2. SKIP LIST AND MODIFIED SKIP LIST

Skip-lists [4] are an increasingly important data structure for storing and retrieving ordered in-memory data. SkipLists have received little attention in the parallel computing world, in spite of their highly decentralized nature. This structure uses randomization and has a probabilistic time complexity of O(logN) where N is the maximum number of elements in the list.

The data structure is basically an ordered list with randomly distributed short-cuts in order to improve search times, see Figure 1. In this paper, we propose a new lock-free concurrent modified skip-list pseudo code that appears to perform as well as the best existing concurrent skip-list implementation under most common usage conditions. The principal advantage of our implementation is that it is much simpler, and much easier to reason about. The original lock-based concurrent SkipList implementation by [13] is rather complex due to its use of pointer-reversal,

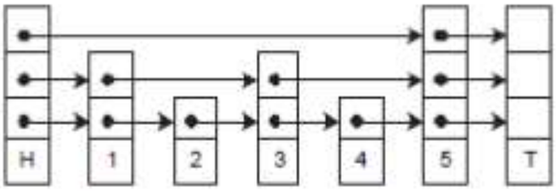


Figure:1Skiplist data structure.

While the search, insert, and delete algorithms for skip lists are simple and have probabilistic complexity of O (log n) when the level 1 chain has n elements. with these observations in mind [5] introduced modified skip list(MSL) structure in which each node has one data field and three pointer fields :left, right, and down. Each level l chain worked separate doubly linked list. The down field of level l node x points to the leftmost node in the level l-1 chain that has key value larger than the key in x. H and T respectively , point to the head and tail of the level lcurrent chain. Below Figure 2 shows the MSL.

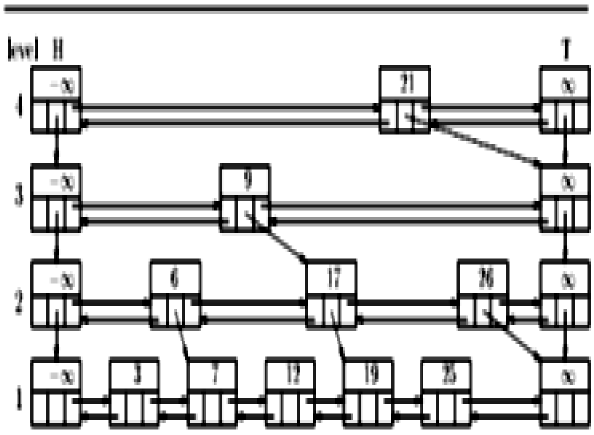


Figure: 2 modified skip list

## 3. CONCURRENT OPERATIONS ON MSL

This paper describes the simple concurrent algorithms for access and update of MSL. Our algorithm based on [11] . In this paper we present a lock-free algorithm of a concurrent modified skip list that is designed for efficient use in both pre-emptive as well as in fully concurrent environments. The algorithm is implemented using common synchronization primitives that are available in modern systems. Double link list is used as a basic structure of modified skip list.

A shared memory multiprocessor system configuration is given in Figure 3.Each node of the system contains a processor together with its local memory. All nodes are connected to the shared memory via an interconnection network. A set of co- operating tasks is running on the system performing their respective operations. Each task is sequentially executed on one of the processors, while each processor can run many tasks at a time. The co-operating tasks, possibly running on different processors, use shared data objects built in the shared memory to co-ordinate and communicate. Tasks synchronize their operations on the shared data objects through sub-operations on top of a cache-coherent shared memory. The shared memory may not though be uniformly accessible for all nodes in the system; processors can have different access times on different parts of the memory [6].

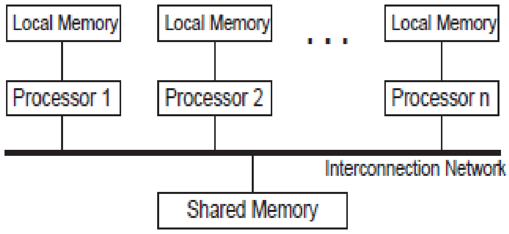


Figure: 3 Shared Memory Access

## 4. OUR ALGORITHM

We present a modified skip list algorithm in the context of an implementation of n set objects supporting three methods, search_node, insert_node, del_node:search_node (key) search for a node with key k equal to key, and return true if key found otherwise return false. Insert_node ( d) inserts adds d to the set and returns true iff d was not already in the set; del_node (v) removes v from the set and returns true iff v was in the set, the below Figure 4 & 5 shows the field of a node. Using the strategy of [11]. To insert or delete a node from the modified skip list we have to change the respective set of prev and next pointers. These pointers have to be changed consistently, but it is not necessary to change them at once. According to Sundell & Tsigas [11] we can consider the doubly linked list as being a singly linked list with auxiliary information in the left pointers, with the right pointers being updated before the left pointers. Thus, the right pointers always form a consistent singly linked list and thus define the nodes positional relations in the logical abstraction of the doubly linked list, but the left pointers only give hints as to where to find the previous node. The down pointer of modified skip list is according its criteria.

```
Union link::word
<p, d> :< pointer to  node , Boolean>
Structure Node
Value: pointer to word
left: union link
right: union  link
down :union link
//local variables
Node, prev,prev2,next,next2:pointer to node
Last,link1:union link
```

Figure :4 local and global variables

The concurrent traversal of nodes makes a continuously allocation and reclamation of nodes, in such kind of scenario several aspects of memory management need to be considered, like No node should be reclaimed and then later re-allocated while some other process is traversing this node. This can be done with the help of reference counting. We have selected to use the lock-free memory management scheme invented by Valois [7] and corrected by Michael and Scott [8], which makes use of the FAA,TAS and CAS atomic synchronization primitives. The operation done by these primitives given below figure 5 & 6.

```
procedure FAA (address: pointer to word, number: integer)
atomic do
*address := *address + number;
```

Figure:5 FAA Atomic primitive

```
function CAS (address: pointer to word, oldvalue: word,
new value: word):boolean
atomic do
if *address = old value then
*address: = new value;
return true;
else
return  false;
```

Figure:6 CAS Atomic primitive

One problem, that arises with non-blocking implementations of MSL that are based on the linked-list structure, is when inserting a new node into the list. Because of the linked-list structure one has to make sure that the previous node is not about to be deleted. If we are changing the next pointer of this previous node atomically with CAS, to point to the new node, and then immediately afterwards the previous node is deleted then the new node will be deleted as well, as illustrated in Figure 7. This problem can be resolved with the latest method introduced by Harris [4] is to use one bit of the pointer values as a deletion mark. On most modern 32-bit systems, 32-bit values can only be located at addresses that are evenly dividable by 4, thereof ore bits 0 and 1 of the address are always set to zero. Any concurrent Insert operation will then be notified about the deletion, when its CAS operation will fail.

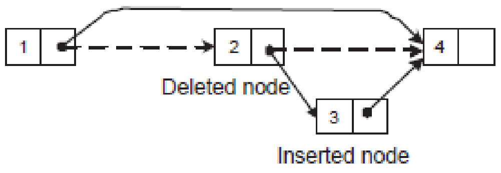


Figure: 7 Concurrent insert and delete operation can delete both nodes

One memory management issue is how to de-reference pointers safely. If we simply de-reference the pointer, it might be that the corresponding node has been reclaimed before we could access it. It can also be that bit 0 of the pointer was set, thus marking that the node is deleted, and therefore the pointer is not valid. The following functions are defined for safe handling of the memory management: shown in figure 8, 9 & 10.

```
function READ_NODE (node  **address):

/* De-reference the pointer and increase the reference
counter for the corresponding node. In case the pointer is
marked, NULL is returned */
```

Figure: 8 Memory management function

```
procedure RELEASE_NODE(node: pointer to Node)

/* Decrement the reference counter on the corresponding
given node. If the reference count reaches zero, then call
RELEASE_NODE on the nodes that this node has owned
pointers to, then reclaim the node */
```

Figure: 9 Memory management function

```
function COPY_NODE(node: pointer to Node):pointer

to Node /* Increase the reference counter for the
corresponding given node */
```

Figure: 10 Memory management function

## 4.1 search_node

Searching in MSL is accomplished by taking a value v and search exactly like a searching in sequential linked list, starting at the highest level and proceeding to the next down level, each time it encounters a node whose key is greater than or equal to v. the search process also save the predecessor and successor of a searched node v for further reference. In concurrent environment, while searching for a node in MSL a processes will eventually reach nodes that are marked to be deleted. It might be due to forcefully preemption of deletion process that invoked the corresponding operation. The searching operation helps the delete process to finish the pending work before continuing the search operation. However, it is only necessary to help the part of the delete operation on the current level in order to be able to traverse to the next node. The search operation traverses in several steps through the next pointers (starting from node1) at the current level until it finds a node that has the same or higher key value than the given key. See Figure 11.

```
pointer to node function search _node ( int   v)
{
node *t,*t_right, *save [maxlevel], *found_node
int i
t=COPY_NODE (head)
t_right=head→right
while (t! =NULL)
{
while( t→value<v)
{
If ( IS_MARKED (t→right))
t=help_del(t)
t_right=READ_NODE(t→right)
save[i]=t
}
if (t→value==v)
break;
else
{
t=t→left→down
i=i-1
}
}
found_node=t
return found_node
}
```

Figure:11

## 4.2 insert_node operation

The implementation of insert operation is shown in figure:12 With a search_node operation to find the node after which the new node should be inserted. This search phase starts from the head node at the highest level and traverses down to the lowest level until the correct node is found .if there exist already  a node with key same as new node with value v, then insertion algorithm exit otherwise it searched for node with key value more than the new node value v. after inserting a new node, it is possible that it is deleted by a concurrent delete_node operation before it has been inserted completely at the particular level. The main steps for insertion algorithm are (I) after setting the new node's left and right pointers, atomically update the right pointer of the to-be-previous node, (II) atomically update the left pointer of the to-be-right node. Iii) atomically update the down pointer of nodes according to the value generated by random no. generator function. if it is more than current level, a  new level is created for new node ,if it is less than current level  new node is to be inserted in between of existing levels ,say it inserted at level l<current level. the down pointer of node at (l+1) level is to be changed accordingly ,as well as the  down  pointer of new node is to be updated with address of a node at level l-1 with value greater than the value at new node respectively.

```
function insert_node(key  int , value: pointer to word)
node *p,*t,*save[max],*t_right,*up,*found_node
k=randomlevel ()
temp,i =current_level
t=COPY_NODE (head)
while (t! =NULL)
{
while( t→key<key)
{
If ( IS_MARKED (t→right→value))
t=Help_Del(t)
save[i]=t
t_right=READ_NODE(t→right)
}
if (t→key==key)
break;
else
{
t=t→left→down
i=i-1
save[i]=t
}
}
found_node=t
new_node=create_node ( value)
node1=COPY_NODE(head)
If(k>temp)
{
//create new head and tail
h1=createnode(∞)
COPY_NODE(h1)
h1→left=null
h1→right=x
h1→down=h
RELEASE_NODE(H1)
t1=CreateNode(∞)
COPY_NODE (t1)
t1→left=x
t1→right=NULL
t1→down=t
RELEASE_NODE (t1)
new_node→left=h1
new_node→right=t1
if((save[k-1]→right→down)==NULL              OR(save[k-
1]→right→value>new_node→value))
new_node→down=save[k-1]→right
RELEA SE_NODE(new_node)
RELEASE_NODE(t1)
RELEASE_NODE(h1)
}
If(k<temp)//insert the new node after save[k]
{
prev=COPY_NODE(save[k])
next=READ_NODE(& prev→right)
While T do
{
If ( prev→right != next)
RELEASE_NODE(next)
next=READ_NODE(&prev→right)
continue
new_node→left=prev
new_node→tight=next
If   CAS(&prev→right,next,new_node)//update   the   next
pointer of Prev of to be inserted node
COPY_NODE(new_node)
break
back-off
}
While  T do //update the left pointer of next of to be inserted
node
{
Link1= next→left
If (IS_MARKED(link1) || new_node→right!=next)
Break
if CAS(&next→left,link1,<new_node,F>)
COPY_NODE(new_node)
RELEASE_NODE(link1)
if(IS_MARKED(new_node)
prev2=COPY_NODE(new_node)
prev2=update_prev(prev2,next)
RELEASE_NODE(prev2)
break
back-off
}

RELEASE_NODE(next)
RELEASE_NODE(new_node)
//update the down pointer
if (k>1)
{
if((save[k-1]→right→down)==NULL OR(save[k-1]→right→ value >new_node→ value)) then
new_node→down= save [k-1]→right

if ((save[k+1]→right→down)==NULL OR (save[k+1]→right→ value <new_node→ value )) then
save [k+1]→right→down =new_node

RELEASE_NODE(new_node)
return true
}
 if (k==1)
{
if(save[k+1]→right→down)==NULL &&
(save[k+1]→right→value<x→value)) then

save[k+1]→right→down= new_node

new_node→down=NULL

}
RELEA SE_NODE(new_node)
return true
}  }
```

figure:12

## 4.3 Delete_node operation

The delete operation uses search operation to locate the node with key k, and then uses two stage process to perform the deletion. Firstly the node is logically deleted by marking the reference contained in it (delete_node→right→value). , secondly the node is physically deleted. The main steps of the algorithm for deleting a node at an arbitrary position are the following: (I) Set the deletion mark on the right pointer of the to-be-deleted node, (II) Set the deletion mark on the left pointer of the to-be-deleted node, (III) Set the deletion mark on the down pointer of the to-be-deleted node, IV) Atomically update the right pointer of the previous node of the to-be-deleted node,(V) Atomically update the left pointer of the right node of the to-be-deleted node. (VI) atomically update the down pointer for a node which was pointed by down pointer of to be deleted node.

```
Function delete_node (int key):boolean
{
node *delete_node,*prev,*succ,*up,*s[max]
temp, I =current_level
t=COPY_NODE (head)
while (t! =NULL)
{
while( t→key<key)
{
If ( IS_MARKED (t→right→value))
t=help_del(t)
save[i]=t
t_right=READ_NODE(t→right)
}
if (t→key==key)
break
else
{
t=t→left→down
i=i-1
save[i]=t
}
}
del_node=t
While T do
if  (del_node==NULL) then
RELEASE_NODE(del_node)
return NULL
linkk1=del_node→right
IF  IS_MARKED(link1) then
help_del(del_node)
RELEASE_NODE(node)
continue
if CAS( &del_node→right,link1<link1.p,T>) then
help_del(del_node)
next=READ_NODE(&del_node→right)
prev=update_prev(del_node,next)
RELEASE_NODE(prev)
release_node(next)
break
RELEASE_NODE(del_node)
```

```
Procedure mark_prev(pointer to node node)
while T do
 link1=node→left
 if IS_MARKED(link1) or
CAS(&node→left,link1,<link1.p,T>)then break
```

Figure:14

```
Pointer to node  function Help_Del(node: pointer to
Node)
 Mark_Prev(node);
last=NULL;
prev= READ_NODE(&node→left)
next= READ_NODE (&node→right)
 while T do
 if prev == next  then
break
 if IS_MARKED(next→right) then
mark_prev(next)
 next2:= READ_NODE (&next→right)
 RELEASE_NODE(next)
next:=next2
 continue
 prev2= READ_NODE (&prev→right)
 if prev2 = NULL then
 if last != NULL then
mark_prev(prev)
next2= READ_NODE (&prev→right)
if CAS(&last→right,<prev,F>),<next2,F>) then
 RELEASE_NODE(prev)
 else
 RELEASE_NODE(next2)
 RELEASE_NODE(prev)
prev=last
 last=NULL
 else
prev2=READ_NODE(&prev→left)
 RELEASE_NODE(prev)
prev=prev2
 continue
 if prev2 != node then
 if last !=NULL   then
RELEASE_NODE(last)
 last:=prev
 prev=prev2
 continue
 RELEASE_NODE(prev2)
 if CAS(&lprev→right, <node,F>,<next,F>) then
 COPY_NODE(next)
 RELEASE_NODE(node)
 break
 Back-Off
 if last != NULL then RELease_node(last)
 RELEASE_NODE(left)
RELEASE_NODE (next)
```

figure: 15

```
function update_prev(prev, node: pointer to
Node): pointer to Node
 last=null
 while T do
prev2:=READ_NODE(&prev→right)
 if prev2 = null then
 if last != null then
mark_prev(prev)
 next2:=READ_NODE(&prev→right)
 if CAS(&last→right,<prev,F>,<next2,F>) then
RELEASE_NODE (prev)
 Else
 RELEASE_NODE (next2)
 RELEASE_NODE (prev)
prev=last
 last=null
 else
 prev2=READ_NODE(&prev→left)
 RELEASE_NODE (prev)
prev=prev2
 continue
 link1=node→left
 if IS_MARKED(link1) then
RELEASE_NODE (prev2)
 break;
 if prev2!= node then
 if last!= null then
 RELEASE_NODE (last)
 last=prev
 prev:=prev2
 continue
 RELEASE_NODE (prev2)
 if link1→p = prev then
break
 if prev→right = node and CAS(
&node→left,link1,<prev,F>) then
COPY(prev)
 RELEASE_NODE (link1→p)
 if IS_MARKED(prev→left) then break
back-off
if last != NULL then
RELEASE_NODE (last)
 return prev
```

Figure:16

## 5. Correctness

In this section we describe the correctness of presented algorithm .here we outline a proof of linearizability [10] and then we prove that algorithm is lock-free. Few definitions are required before giving proof of correctness.

**Definition 1** We denote with $M_t$ the abstract internal state of a modified skip list at the time t. Mt is viewed as a list of values ( $v_1$ ,----, $v_n$) .The operations that can be performed on the modified skip list are Insert (I) and Delete(D). The time t1 is defined as the time just before the atomic execution of the operation that we are looking at, and the time t2 is defined as the time just after the atomic execution of the same operation. The return value of $true_2$ is returned by an Insert operation that has succeeded to update an existing node, the return value of true is returned by an Insert operation that succeeds to insert a new node. In the following expressions that defines the sequential semantics of our operations, the syntax is S1 : O1; S2, where S1 is the conditional state before the operation O1, and S2 is the resulting state after performing the corresponding operation:

$$M_{t1}: \ I(v_1), \quad M_{t2} = M_{t1} + [v_1] \qquad (1)$$

$$M_{t1} = \theta : D\,() = NULL \ , M_{t2} = \quad \theta \qquad (2)$$

$$M_{t1} [ \ v_1 \ ] + M_1 : D\,() = v_1 \,, \quad M_{t2}= M_1 \qquad (3)$$

**Definition 2** In order for an implementation of a shared concurrent data object to be linearizable [10], for every concurrent execution there should exist an equal (in the sense of the effect) and valid (i.e. it should respect the semantics of the shared data object) sequential execution that respects the partial order of the operations in the concurrent execution.

**Definition 3** The value v is present (∃ i.M[i]=v) in the abstract internal state M of implementation , when there is a connected chain of next pointers (i.e. prev→link→right) from a present node in the doubly linked list that connects to a node that contains the value v, and this node is not marked as deleted (i.e. is_marked(node)=false) ).

**Definition 4** The decision point of an operation is defined as the atomic statement where the result of the operation is finitely decided, i.e. independent of the result of any suboperations after the decision point, the operation will have the same result. We also define the state-change point as the atomic statement where the operation changes the abstract internal state of the priority queue after it has passed the corresponding decision point.
We will now use these definitions to show the execution history of point where the concurrent operation occurred atomically.

**Lemma 1 :** *A insert_node operation (I(v)) , takes effect atomically at one statement.*
Proof: The decision, state-read and state-change point for an insert operation which succeeds (I(v)), is when the CAS sub-operation **CAS(&prev→right,next,new_node)** of insert operation succeeds. The state of the modified skip list was (Mt1 = M1) directly before the passing of the decision point. The state of the modified skip list after passing the decision point will be MT2 = [v] + M1 as the next pointer of the save[k] node was changed to point to the new node which contains the value v. Consequently, the linearizability point will be the CAS sub-operation in that line.

**Lemma 2 :** *A delete_node operation which fails (D*() =NULL*),takes effect atomically at one statement*
Proof: The decision point for a delete operation which tails (D() =NULL) is the check in line **if (del_node==NULL) then** . Passing of the decision point gives that the value v we are searching for deletion is not exist in modified skip list i.e ($M_{t1}$ = NULL) .

**Lemma 3** : *A delete_node operation which succeeds (D*() *=v), takes effect atomically at one statement.*
Proof: The decision point for a delete operation which succeeds (D() = v) is when the CAS sub-operation inline [**next=read_node(&del_node→right)]** succeeds. Passing of the decision point together with the verification in line [ **if is_marked(link1) then ].** Directly after passing the CAS sub-operation **in [if CAS( &del_node→right,link1<link1.p,T>) then]** (i.e. the state-change point) the to-be-deleted node will be marked as deleted and therefore not present in the Modified skip list (¬∃i.$M_{t2}$ [i] = v). Unfortunately this does not match the semantic definition of the operation.

## 6. Conclusion

We introduced a concurrent modified Skiplist using a remarkably simple algorithm in a lock free environment. Our implementation is raw, various optimization to our algorithm are possible like we can extend the correctness proof. Empirical study of our new algorithm on two different multiprocessor platforms is a pending work. The presented algorithm is first step to lock free algorithmic implementation of modified skip list; it uses a fully described lock free memory management scheme. The atomic primitives used in our algorithm are available in modern computer system.

## 7. REFERENCES

[1] Eshcar Hillel. Concurrent Data Structures: Methodologies and Inherent, Limitations, PhD thesis, Israel Institute of Technology, 2011]

[2] P. TSIGAS, Y. ZHANG. Evaluating the performance of non-blocking synchronization on shared-memory multiprocessors. Proceedings of the international conference on Measurement and modeling of computer systems (SIGMETRICS 2001), pp. 320-321, ACM Press, 2001.

[3] P. TSIGAS, Y. ZHANG. Integrating Non-blocking Synchronisation in Parallel Applications: Performance Advantages and Methodologies. Proceedings of the 3rd ACM Workshop on Software and Performance (WOSP ’02), ACM Press, 2002.

[4] Pugh, W. Skip lists: A probabilistic alternative to balanced trees. Communications of the ACM 33, 6 (June 1990),

[5] S. Cho and S. Sahni. Weight-biased leftist trees and modified skip lists. ACM J. Exp. Algorithmics, 1998.

[6] H. Sundell and P. Tsigas. Fast and Lock-Free Concurrent Priority Queues for Multi-Thread Systems. In Proceedings of the 17th International Parallel and Distributed Processing Symposium, page 11. IEEE press, 2003.

[7] J. D. VALOIS. Lock-Free Data Structures. PhD. Thesis, Rensselaer Polytechnic Institute, Troy, New York, 1995.

[8] M. MICHAEL, M. SCOTT. Correction of a Memory Management Method for Lock-Free Data Structures. Computer Science Dept., University of Rochester, 1995.

[9] T. L. HARRIS. A Pragmatic Implementation of Non-Blocking Linked Lists. Proceedings of the 15th International Symposium of Distributed Computing, Oct. 2001.

[10] M. Herlihy and J. Wing, “Linearizability: a correctness condition for concurrent objects,” ACM Transactions on Programming Languages and Systems,vol. 12, no. 3, pp. 463–492, 1990.

[11] H. Sundell, P. Tsigas, Lock-free and practical doubly linked list-based deques using single-word compare-and-swap, in: Proceedings of the 8th International Conference on Principles of Distributed Systems, in: LNCS, vol. 3544, Springer Verlag, 2004, pp. 240–255.

[12] Ranjeet Kaur, Dr. Pushpa Rani Suri, Modified Skip List in Concurrent Environment, in : Proceedings of the IJSER, Aug 2014

[13] I. LOTAN, N. SHAVIT. Skiplist-Based Concurrent Priority Queues. International Parallel and Distributed Processing Symposium, 2000.

# Web Content Mining equipped Natural Language Processing for handling web data

**Karan Sukhija**
Research Scholar
Department of Computer Science and Application
Panjab University, Chandigarh
rs.karansukhija@gmail.com

**Abstract:** The growing usage of the web has unfolded the Web mining technology to a great extent. Web mining helps in extraction of useful knowledge from web data. (i.e. a range of web pages, hyperlinks among various pages, web sites usage logs and so on. This paper has threefold aspect. Firstly, it defines how web mining research area focuses on mining research and retrieval research (i.e. retrieval of data, information on web, data and text mining). Secondly, it categorizes the Web mining as content mining (i.e. retrieval of information from texts, images and other contents), structure mining (i.e. finding of facts from association of web pages) and usage mining (i.e. mining of information about usage of web sites). Web content mining mainly focuses on the structure of inner-document whereas web structure mining aim is to discover the linkage assembly of the hyperlinks at the inter-document level. Web usage mining includes three major phases i.e. preprocessing, pattern discovery and pattern analysis. Thirdly, it focuses on natural language processing as a backbone for web content mining that helps in handling of unstructured data over the web by offering various techniques. This paper concluded the web mining as trending research area for various research communities such as Databases, Artificial intelligence, Information retrieval and E-commerce.

**Keywords:** Web mining, Content mining, Structure mining, Usage mining, Opinion mining, Natural language processing.

## 1. Introduction

Web mining is promising trend of data mining that helps in finding of useful facts from web data. Web data includes the various web documents, hyperlinks among pages, web sites usage log and so on. Web mining can be defined either by the process-centric view approach or by the data-centric view approach. In process-centric view, web mining is defined as a sequence of tasks whereas in data-centric view, it is defined by means of web data that helps in the mining process [1]. Web mining research area focuses on mining research (i.e. discover hidden facts) and retrieval research (i.e. retrieves existing data or documents from a large database or document repository). Table 1 summarizes the possible categorization of retrieval and mining. This categorization is founded on twofold phases: Purpose and sources of data. The purpose of data retrieval techniques is to enhance the fetching of data from a databank and data mining techniques is to identify interesting patterns by analysis of data [2].

**Table 1: Retrieval and Mining techniques Classification [2]**

| Purpose | Sources | | |
|---|---|---|---|
| | **Data** | **Textual Data** | **Web Data** |
| Retrieving well-known facts or documents efficiently and effectively | Data Retrieval | Information Retrieval | Web Retrieval |
| Finding new patterns or knowledge previously unknown | Data Mining | Text Mining | Web Mining |

From table-1 it is concluded that web mining research is the juncture of different areas (i.e. data retrieval, information retrieval, web retrieval, data mining and text mining). This paper is organized as follows. In section 2 Web mining is classified as content mining, structure mining and usage mining. In section 3 web mining is highlighted as trending research area for various research communities such as Databases, Artificial intelligence, Information retrieval and E-commerce. In section 4, Natural language processing technology is highlighted to explain how to handle unstructured data over the web using various NLP techniques (i.e. Part-of-Speech tagging). Finally, the paper is concluded in section 5.

## 2. Classification of Web Mining

Web mining can be categorized as content mining, structure mining and usage mining. Web content mining is a technique of fetching information from texts, images and other contents. Web structure mining is a technique of extracting information from linkages of web pages. Web usage mining is a process of take out information about the usage of web sites [7].

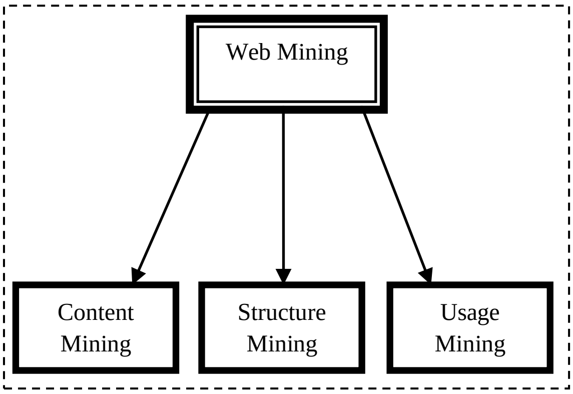


**Figure 1: Classification of Web mining**

***Web Content Mining:*** Content mining is a process of finding information from millions of sources across the World Wide Web and mining these web data contents. Web data contents can be structured (i.e. data stored in the tables or HTML pages generated from database), semi- structured (i.e. HTML documents) or unstructured (i.e. text data) [11] [14].

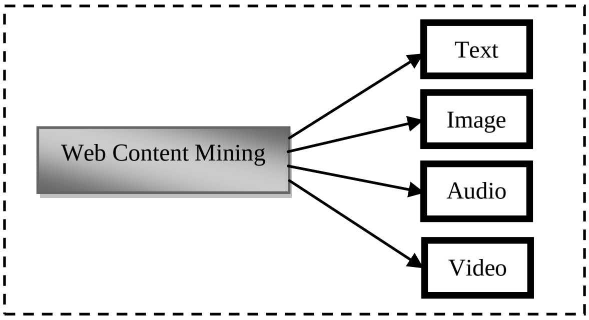


**Figure 2: Taxonomy of web content mining**

The un-structured properties of web data potency the web content mining in the direction of a further complex approach [3].

- Mining by developing a knowledge-base repository of the domain
- Interpretation of Mined Knowledge
- Iterative refinement of user queries for personalized search
- Process of Iterative Query Refinement

***Web Structure Mining:*** As web content mining chiefly emphases the construction of inside the document, whereas web structure mining aim is to determine the link structure of the hyperlinks in the inter-document level [3]. It creates the structural framework about the web site along with web page. The structural outline consists of the following information [12]:

- Measure the frequency of the local links in the web tuples in a web table.
- Measure the frequency of web tuples in a web table containing links that are interior and the links that are within the same document.
- Measure the frequency of web tuples in a web table that having links that is global and the links that span different web sites [13].
- Measure the frequency of identical web tuples that appear in the web table or among the web tables.

The configuration of a web (directed graph) consists of web pages as nodes and hyperlinks as edges i.e. connection among related pages. Web structure graph terminology is as follows as given in table 2:

**Table 2: Lexicon of web structure graph**

| | |
|---|---|
| Web-graph | A directed graph that exemplifies the web. |
| Node | Each Node represents the web page of the web-graph. |
| Link | Each hyperlink represents the directed edge of the web-graph |

Figure 3 depicts the web graph structure by means of document structure and hyperlinks.

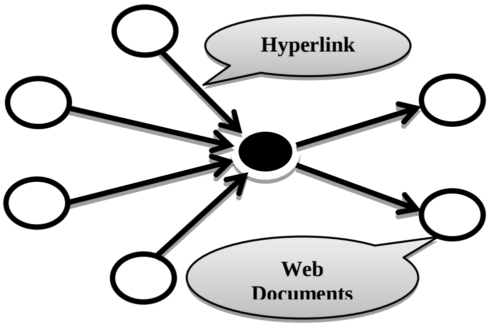


**Figure 3: Web Graph Structure**

A hyperlink connects a web page to a poles apart location within the same web page is called an intra-document hyperlink. A hyperlink that links two diverse pages is called an inter-document hyperlink [1].

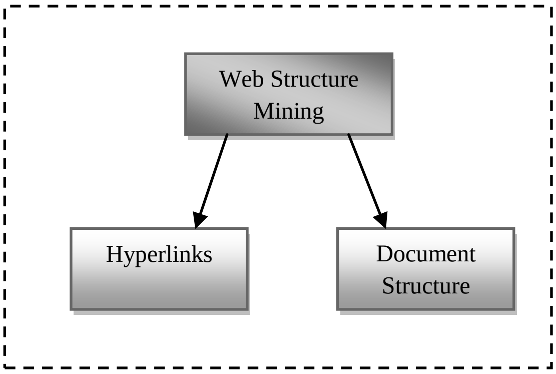


**Figure 4: Web structure mining taxonomy**

***Web Usage Mining:*** Web usage mining is an activity that automatically discovers the user access [15] patterns from different web servers. It keeps track of earlier retrieved pages by a user that helps in identifying the distinctive behavior of the user and to make forecast about preferred pages [4]. Web usage mining includes three major phases i.e. preprocessing, pattern discovery and pattern analysis as shown in figure 5.

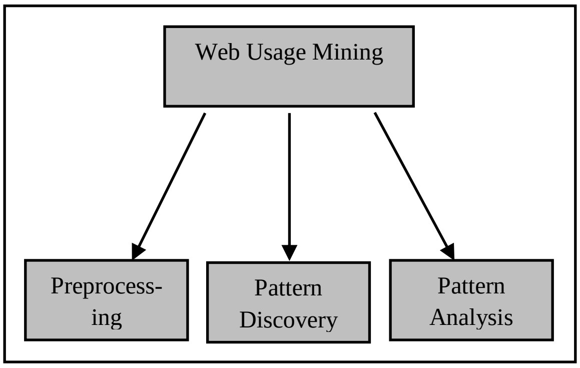


**Figure 5: Segments of web usage mining**

Web usage mining phases are defined as follows: -

- **Preprocessing:** This phase converts the raw usage data into the data abstractions. It includes usage preprocessing, content preprocessing and structure preprocessing. Preprocessing stage follow some steps such as data cleaning, efficient user identification, session identification and path completion, and transaction identification [6].

- **Pattern Discovery:** This phase includes different techniques such as association rules, clustering etc for pattern discovery. It draws upon methods and algorithms developed from several fields such as statistics, data mining, machine learning and pattern recognition [10].

- **Pattern Analysis:** The purpose of pattern analysis is to filter out uninteresting rules or patterns from the set found in the pattern discovery phase. It requires the analysis of the structure of hyperlinks and the contents of the pages.

The difficulties in web usage mining occur due to the anomalies in existing data.

## 3. Applications of Web mining

Web mining spreads analysis much further by combining other corporate information with Web traffic data. Practical applications of Web mining technology are abundant, and are by no means the limit to this technology. Web mining tools can be extended and programmed to answer almost any question. It can be applied in following areas:

- **Managerial decision making:** Web mining can provide companies managerial insight into visitor profiles that helps in taking strategic actions by top management [4].
- **Marketing effectiveness:** Companies can have some subjective measurements through web mining regarding the effectiveness of their marketing campaign or marketing research, which will help the business to improve and align their marketing strategies timely.
- **Business related decisions:** In the business world, structure mining can be quite useful in determining the connection between two or more business web sites [7].
- **Accounting and Inventory:** Web mining also allows accounting, customer profile, inventory, and demographic information to be correlated with web browsing.
- **Improvement feedback:** Companies can identify the strength and weakness of their web marketing campaign through feedback from web mining, and can make the strategic adjustments accordingly [8].
- **Searching enhancement:** Search engine such as Google provides advanced and efficient searching capabilities by the usage of web mining [5]
- **E-commerce (Infrastructure):** Generate user profiles, targeted advertizing**,** fraud and similar image retrieval [9].
- **Information retrieval (Search) on the Web:** Automated generation of topic hierarchies**,** web knowledge bases**,** extraction of schema for XML documents.
- **Network Management:** results in performance management and fault management.

## 4. Handling of Web Content using Natural Language Processing

Web data consists of various types like structured data (i.e. data retrieved from backend databases), Semi-structured data (i.e. data organized as a hierarchy of blocks) and unstructured data (i.e. natural language text) as shown in figure 6.

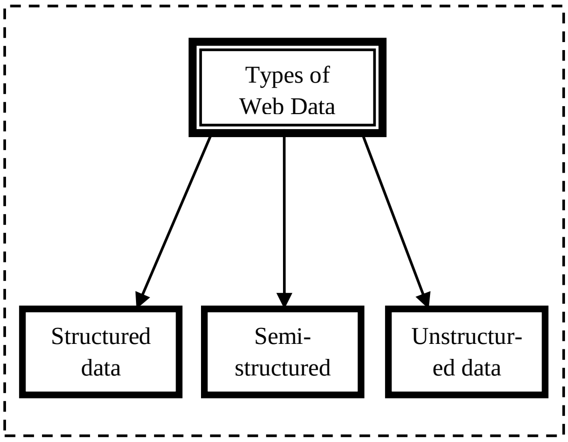


**Figure 6: Types of Web data**

The aforementioned web contents can be handled efficiently by following the below given techniques that is related to natural language processing to some extent. Natural language processing helps to understand how to manage unstructured data over the machine processing platform using various NLP techniques along with the web content mining.

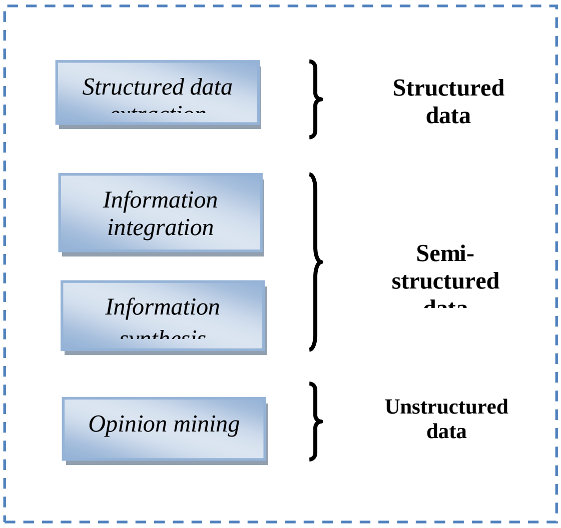


**Figure 7: NLP techniques to handle different types of web content**

***Structured Data Extraction:*** web data or information is arranged/ managed as structured data objects on regular basis. Such as retrieval of various data records from databases. Extraction of these structured objects is possible by either the Wrapper induction technique (i.e. Supervised) or Automatic extraction (i.e. unsupervised) technique [16].

- *Wrapper induction technique* is based on machine learning.
- *Automatic extraction technique* is based on POS tagging (Part-Of-Speech). POS tagging (NLP technique) is to automatically assign part-of-speech tags (i.e. noun, verb, adjective etc.) to words in context [17].

***Information Integration:*** Structured data objects extraction is followed by integration of data to design a consistent and reliable database. Integration can be in terms of schema match or data instance match.

- *Schema match:* From various data tables match the columns (e.g., Item names).
- *Data instance match:* From various data fields match the values, e.g., “Coke” = “Coca Cola”?

***Information/ knowledge synthesis:*** It works upon a web search paradigm where a request for some words is given and a ranked list of pages is returned by search engine and top-ranked pages read by the user to find required information. This technique is adequate/suitable for navigational queries (i.e. specific information) but not for informational queries (i.e. open-ended research problems).

***Opinion mining:*** Web content mining study is to extract precise sorts of information from text in Web documents e.g. Opinion (positive or negative) and factual information (i.e. Find economic data from rumors of different countries). Opinion mining or sentiment analysis purpose is to excerpt and abridge opinions.

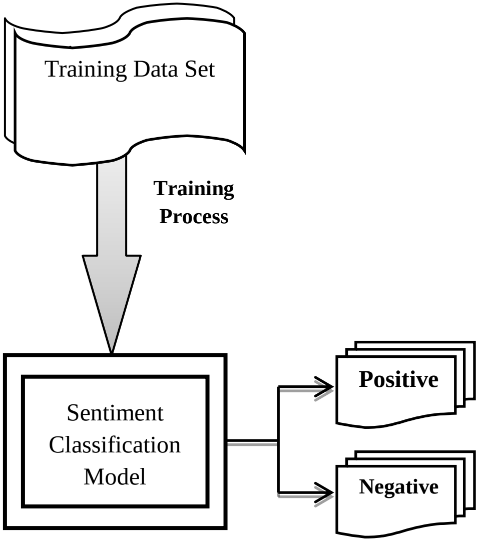


**Figure 8: Opinion Mining**

## 5. Conclusion

The increasing demand of the web has greatly evolved the web mining technology. Web mining is concerned with the mining of data from the various web documents, hyperlinks between documents and usage logs of web sites. Mining approach is to be followed can be either process-centric or data centric. This paper presents an overview of web content mining, web structure mining and web usage mining. Unstructured data of web contents can be handled by using natural language processing mechanism. An important research area in web mining is web usage mining which focuses on the sighting of interesting outlines in the glancing and steering data of web users. It helps in personalization of web content by having track of earlier retrieved pages by a user that result in identification of the distinctive behavior of the user and to make forecast about favorite pages. The outcomes formed by web usage mining can be exploited to expand the performance of web servers and web-based applications.

## 6. References

[1] Jaideep Srivastava, Prasanna Desikan, Vipin Kumar, “Web Mining: Concepts, Applications, and Research Directions”.

[2] Hsinchun Chen and Michael Chau, “Web Mining: Machine learning for Web Applications”, Annual Review of Information Science and Technology.

[3] Sarita Dalmia, “Web Mining : Survey and Research”.

[4] Ankita Kusmakar, Sadhna Mishra, “Web Usage Mining: A Survey on Pattern Extraction from Web Logs”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 9, September 2013.

[5] Monika Yadav Mr. Pradeep Mittal, “Web Mining: An Introduction” , International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 3, March 2013.

[6] DeMin Dong, "Exploration on Web Usage Mining and its Application", International Workshop on Intelligent Systems and Applications, Pp. 1.

[7] Robert Cooley, Bamshad Mobasher, Jaideep Srivastava , “Web Mining: information and Pattern Discovery on the WWW”.

[8] Mary Garvin, “Data Mining and the Web: What They Can Do Together”.

[9] R. Kosala, H. Blockeel, “Web Mining Research: A Survey”, in SIGKDD Explorations, ACM, Volume 2, Issue 1, July 2000.

[10] J. Srivastava, R. Cooley, M.Deshpande, P-N. Tan. “ Web Usage Mining: Discovery and Applications of usage patterns from Web Data”, SIGKDD Explorations, Volume 1, Issue 2, 2000

[11] Etzioni, "The World Wide Web: Quagmire or gold mine" , Communications of the ACM, Vol. 39, issue ll, 1996, pp. 65-68.

[12] Cooley, R., Mobasher, B., & Srivastava, "Web mining: information and pattern discovery on the World Wide Web" , In Proceedings of the 9th ZEEE International Conference on Tools with Artificial Intelligence, 1997, pp. 558-567.

[13] Liu Bin, "Web Data Ming Exploring Hyperlinks, Contents, and Usage Data".

[14] Kshitija Pol, Nita Patil, Shreya Patankar, Chhaya Das, " A Survey on Web Content Mining and extraction of Structured and Semistructured data".

[15] Bing Liu, "From Web Content Mining to Natural Language Processing", 2007.

[16] M. Rajman, R. Bseancon, "Text Mining: Natural Language Techniques and Text Mining Applications"

[17] Lihui Chen, Wai Lian Chue, "Using Web structure and summarisation techniques for Web content mining".