DRONACHARYA

Patron

Dr. Satish Yadav
Chairman, DCE, Gurgaon

Chairman

Prof (Dr.) BMK Prasad
Principal, DCE, Gurgaon

Co-Chairman

Prof. Onkar Singh
Dean Academics, DCE, Gurgaon

Convener

Mr. Jitender Kumar
HoD (CSE & IT), DCE, Gurgaon

Coordinator

Dr.(Mrs.) Supriya Panda
Professor(CSE & IT) DCE, Gurgaon

Proceedings

NATIONAL SEMINAR ON ARTIFICIAL INTELLIGENCE

NSAI-09

17th July, 2009

Contents

Proceedings

AICTE SPONSORED

NATIONAL SEMINAR ON ARTIFICIAL INTELLIGENCE

17th July, 2009

NSAI-09

Editor

Jitender Kumar
HoD, CSE & IT Department

Organized By

Dronacharya College of Engineering
Khentawas, Farrukh Nagar, Gurgaon-123506, Haryana, India

Acknowledgements

Sponsors

All India Council For Technical Education

Disclaimer

The opinions expressed and figures provided in this Proceedings of NSAI-09 are the sole responsibility of the authors. The organizers and the editor bear no responsibility in this regard. Any and all such liabilities are disclaimed.

Published and printed by Mr. Jitender Kumar, HoD, CSE & IT Department, Dronacharya College of Engineering, Khentawas, Farrukh Nagar, Gurgaon - 123506, Tel. 0124-2375502/03/04, Fax 0124-2275328, E-mail: nsai09@dronacharya.info

Patron
Dr. Satish Yadav
Chairman
Dronacharya Group of Institutions

Chairman
Dr. B.M.K. Prasad
Principal
Dronacharya College of Engineering, Khentawas, Gurgaon

Co- Chairman
Prof. Onkar Singh
Dean Academics
Dronacharya College of Engineering, Khentawas, Gurgaon

Convener
Mr. Jitender Kumar
HOD (CSE & IT Deptt.)
Dronacharya College of Engineering, Khentawas, Gurgaon

Coordinator
Dr. Supriya Panda
Professor (CSE & IT Deptt.)
Dronacharya College of Engineering, Khentawas, Gurgaon

MESSAGE

The maturity of a technical college is reflected in the seminars, conferences, publications and other such academic events conducted by the institution. I am happy to see the maturing of our institute's academic and research environment, with the National Seminar on Artificial Intelligence, NSAI-09, adding more rich colours to our institute's vibrant academic canvas.

I congratulate the Principal, Dean Academics, Head of CSE & IT department and their entire team for making the event a resounding success.

Dr. Satish Yadav
Chairman
Dronacharya College of Engineering

MESSAGE

I feel proud and privileged to note the grand success of the AICTE Sponsored National Seminar on Artificial Intelligence, NSAI-09 organized by Department of CSE & IT. The seminar's objective was to bring eminent academicians, scholars, scientists, researchers, industrialists and experts from different technical domains to explore new horizons of the ever innovative field of Artificial Intelligence.

It is heartening to see that the seminar not only met the above objective successfully but also proved to be a boost towards our academic commitment and the promotion of research environment. I am sure that the contributions in the form of research papers will enrich our knowledge and motivate everyone of us to take up the challenging application areas in contribution towards the development of the society, the industry and the nation as a whole.

I congratulate the seminar team for the success of the event and wish them luck for their future endeavours.

Prof. (Dr.) B.M.K. Prasad
Principal
Dronacharya College of Engineering

MESSAGE

It gives me immense pleasure and satisfaction to note that the Department of CSE& IT has successfully organized a One Day AICTE sponsored National Seminar on Artificial Intelligence, a topic of immense contemporary relevance.

A very promising sign of the growing importance of Artificial Intelligence is clearly indicated by the papers in this volume of proceedings. The wide range and importance of Artificial Intelligence applications are likely to increase further as the time goes by and we intend to reflect these developments in our future seminars and conferences.

I would like to thank the Patron, Hon'ble Dr. Satish Yadav, Chairman, Dronacharya College of Engineering, All India Council of Technical Education, and the Program Chair, Dr. (Prof.) B.M.K. Prasad, for their support in organizing the seminar. I would also like to take this opportunity to congratulate the convener, Mr. Jitender Kumar and Coordinator Dr. (Mrs.) Supriya Panda for having taken all the pains to edit and compile the proceedings, along with their dedicated team and brought out the same in such a short time. I am sure that the proceedings will be of immense use to one and all.

I shall also like to convey my deep appreciation to all the participants for their applause worthy efforts and quality papers.

I wish the seminar team success in all future endeavours

Prof. Onkar Singh
Dean Academics
Dronacharya College of Engineering

Foreword

Imagine tools, technologies, environments, data sets, that get better the more you use them, that learn about you and adapt and improve as a result of your interactions and use. Think about being able to create new knowledge based on capturing observed patterns, recognizing behaviours, gleaning and understanding the context of events and actions. This is the promise of Artificial Intelligence to the human kind.

Artificial Intelligence (AI) is the key technology in many of today's novel applications, ranging from banking systems that detect attempted credit card fraud, to telephone systems that understand speech, to software systems that notice when you're having problems and offer appropriate advice. These technologies would not exist today without the sustained support of fundamental AI research over the past three decades. AI does not produce stand-alone systems, but instead adds knowledge and reasoning to existing applications, databases, and environments, to make them friendlier, smarter, and more sensitive to user behaviour and changes in their environments. Beyond the myriad of currently deployed applications, ongoing efforts that draw upon these decades of fundamental research point towards even more impressive future capabilities including producing valuable spin-off technologies. AI researchers tend to look very far ahead, crafting powerful tools to help achieve the daunting tasks of building intelligent systems. Artificial Intelligence first conceived and demonstrated such well-known technologies as the mouse, time-sharing, high-level symbolic programming languages, computer graphics, the graphical user interface (GUI), computer games, the laser printer, object-oriented programming, the personal computer, email, hypertext, symbolic mathematics systems, and, most recently, the software agents which are now popular on the World Wide Web.

There is every reason to believe that AI will continue to produce such spin-off technologies. Surely, we must equally push forward, explore, learn, develop and advance the capabilities presenting themselves today. Just imagine the impact on human life if we had technology that could learn! The **National Seminar on Artificial Intelligence - NSAI-09** is our modest attempt in contributing towards the rich legacy of Artificial Intelligence. I wish to thank the AICTE and College Management for their support and cooperation for organizing this seminar. I am also thankful to the Principal and Dean Academics for their valuable suggestions to make this event possible.

Mr. Jitender Kumar (**Convener**, NSAI-09)
HOD, CSE & IT Deptt.
Dronacharya College of Engineering

Message from the Coordinator

Change is the unchangeable law of nature. Change is a constant factor in this rapidly raging world. Of course, the computing world has not been insulated from change. Embracing all changes from programming languages, modern high speed processors to green computing and internet, we have surged into the 5G computing paradigm, where computational intelligence is a well-established paradigm. Artificial Intelligence (AI) has made a great progress in short history of 60-years; starting from a gentle evolution to now a raging revolution, where new theories with a sound biological understanding have been evolving.

In light of the current research pursuits, the realm is heterogeneous as being dwelled on such technologies as fuzzy systems, neurocomputing, artificial life, genetic algorithms, multi-agent systems etc. This leads to a theory-matrix, which involving these factors and their corresponding impacts is menacing a formidable intelligent explosion.

A lot of change involves a lot of people making the end result worthwhile. National Seminar on Artificial Intelligence (NSAI-2009) focused on this key objective to provide the academic community a medium for presenting cutting edge research related to AI and its applications.

So, thanks for all the support that Dronacharya College of Engineering provided to host this seminar and also, thanks to all the active participants and the delegates to make this platform a two way learning domain.

Dr. (Mrs.) Supriya Panda (**Coordinator**,NSAI-09)
Professor (CSE & IT Deptt.)
Dronacharya College of Engineering

CONTENTS

Neural Network and Fuzzy Logic

Amrita Koul

Lecturer, CSE & IT Deptt, DCE, Gurgaon

Abstract

This paper deals with Artificial Neural Network, fuzzy logic and their combination as hybrid system. In section 1 of the paper introduction about ANN is given after which neurons and their work is described. After that paper contains neural net architecture and learning of neural networks.

Second section of this paper gives the introduction about Fuzzy Logic and how it is used.

Third section of this paper proceeds under the topic Fuzziness in Artificial Neural Network. In this section detail description about fusion of artificial neural network and fuzzy logic is given. This section also contains different fusion methods of ANN and fuzzy logic.

Applications of neuro fuzzy systems are described in section 4. The information about applications such as language processing, financial application, character recognition is given in this section.

After that paper concludes with some future directions of artificial neural network, fuzzy logic and their hybrid systems.

1. Introduction to Artificial Neural Networks

1.1 What is a Neural Network?

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process.

1.2 Artificial Neurons and how they work

The fundamental processing element of a neural network is a neuron. This building block of human awareness encompasses a few general capabilities. Basically, a biological neuron receives inputs from other sources, combines them in some way, performs a generally nonlinear operation on the result, and then outputs the final result. The basic unit of neural networks, the artificial neurons, simulates the four basic functions of natural neurons.

In Figure 1.2.1, various inputs to the network are represented by the mathematical symbol, x (n). Each of these inputs are multiplied by a connection weight. These weights are represented by w (n). In the simplest case, these products are simply summed, fed through a transfer function to generate a result, and then output. This process lends itself to physical implementation on a large scale in a small package. This electronic implementation is still possible with other network structures which utilize different summing functions as well as different transfer functions. Some applications require "black and white," or binary, answers. These applications include the recognition of text, the identification of speech, and the image deciphering of scenes. These applications are required to turn real-world inputs into discrete values.

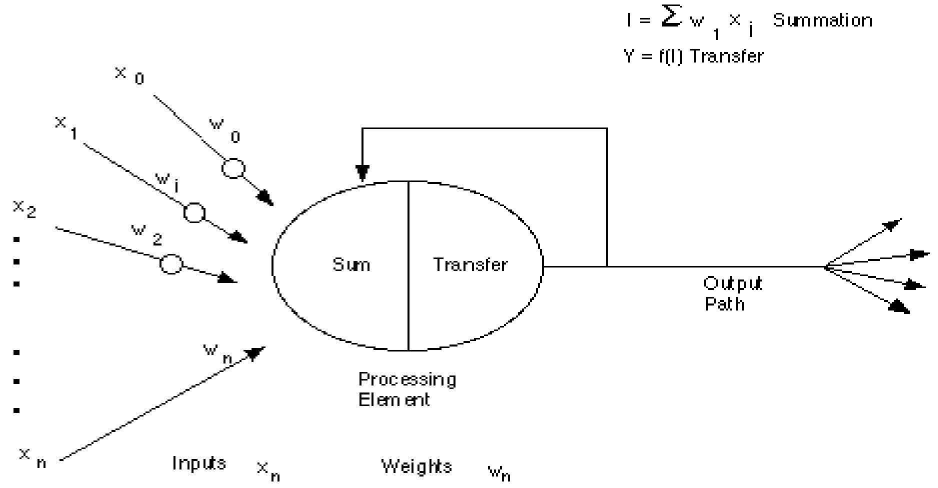


Figure 1.2.1 A Basic Artificial Neuron

These potential values are limited to some known set, like the ASCII characters or the most common 50,000 English words. Because of this limitation of output options, these applications don't always utilize networks composed of neurons that simply sum up, and thereby smooth, inputs. These networks may utilize the binary properties of ORing and ANDing of inputs.

1.3 Architecture of neural networks

Single neuron is insufficient for many practical applications therefore networks with large number of neurons are frequently used. Following are some network structures.

a. Layered network.
b. Acyclic network.
c. Feed forward network.
d. Modular neural network.

Once a network has been structured for a particular application, that network is ready to be trained.

2. Introduction to Fuzzy Logic

2.1 What is Fuzzy Logic?

FL is a problem-solving control system methodology that lends itself to implementation in systems ranging from simple, small, embedded micro-controllers to large, networked, multi-channel PC or workstation-based data acquisition and control systems. It can be implemented in hardware, software, or a combination of both. FL provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information. FL's approach to control problems mimics how a person would make decisions, only much faster.

2.2 How does FL work?

FL requires some numerical parameters in order to operate such as what is considered significant error and significant rate-of-change-of-error, but exact values of these numbers are usually not critical unless very responsive performance is required in which case empirical tuning would determine them. These values don't have to be symmetrical and can be "tweaked" once the system is operating in order to optimize performance. Generally, FL is so forgiving that the system will probably work the first time without any tweaking.

3. Artificial Neural Network and Fuzzy Logic

3.1 Fusion of ANN and Fuzzy Logic

Both NN and fuzzy theories object the humanness, and the concerns to each had been aroused spontaneously and rapidly at a same time. So the similarities and mutual compensations between these are much discussed. The study of fusion can be started with the combination of either the individual merits or the similarities between these

Difference: Fuzzy logic: Logicality

NN: Learning function

Similarity: (1) Output characteristics of NN and membership function

(2) Multiply-add operation of neuron and MAX-MIN operation

of approximate reasoning.

The first fusion pattern is a method combining individual advantages. Fuzzy logic can express logic explicitly taking a form of rule. NN is helpful when it is employed for pattern identification because of its learning function. From these advantages of view, (a) a method to endow learning function to fuzzy logic or to conduct pattern processing before fuzzy logic is applied, and (b) a method to incorporate logics in NN structure etc. could be possible for combining these two techniques.

The second fusion method is to superpose similarities. The first similarity shown above is to give a membership function to NN without causing a crisp boundary between classes formed by a pattern classification type NN. The reason is that threshold function of its neuron have sigmoid characteristics to attain continuous values of 0, 1. The second similarity is that (a) the MIN operation of input and fuzzy variables conducted at each proposition of IF parts of fuzzy inference rule corresponds to a product of input to the neuron and synaptic weights, and (b) the MAX operation to obtain a final inference value from the THEN part of these plural inference rules corresponds to the input sum within neuron. The endowment of learning function to the fuzzy logic is one of the major particular characteristics of NN only. The degree of association between keywords is expressed in a form of matrix, and the learning is conducted by using the steepest decent method. Although the matrix in this case could be interpreted as a two-layered NN in a way, it could apart from the general accepted image of NN.

3.3 Status of Fusion Technology

Auto-design of Membership Function.

One of the most significant characteristic features of fuzzy logic attained in view of the knowledge processing is the separation of its logic and fuzziness

Into a rule and a membership function respectively and another is that the quantitative expression and processing of fuzziness became possible by this.

Thus, those objects that had been considered highly difficult to deal with because of their fuzziness can now be dealt easily by using a framework of conventional knowledge processing. While a considerable time has to be spent for tuning the rules for dealing those objects having inherent fuzziness, such as talent or skillfulness of expert, by two valued logic based system wherein the fuzziness affects the logic, there exists no needs to tune the rules of logical expression when an approximate reason is employed. One of the most significant characteristic features of fuzzy logic attained in the designing methods of membership function is roughly divided into three categories shown below.
(a) Manual cut-and-try

(b) Fuzzy clustering

(c) Neural Network.

The following four advantages shown below are noticeable in these approaches.

i. Shorter designing time since it is algorithmically determined without requiring manual works.
ii. Design of nonlinear membership function is possible because of inherent nonlinearity of NN.
iii. Automatical acquisition of rule from experts using the learning function of NN. Dynamical adaptation to inference environment by the learning function of NN.

4. Applications of ANN and Fuzzy Logic

4.1 Language Processing

Language processing encompasses a wide variety of applications. These applications include text-to-speech conversion, auditory input for machines, automatic language translation, secure voice keyed locks, automatic transcription, aids for the deaf, aids for the physically disabled which respond to voice commands, and natural language processing.

Many companies and universities are researching how a computer, via ANNs, could be programmed to respond to spoken commands. If this capability could be shrunk to a chip, that chip could become part of almost any electronic device sold today. Literally hundreds of millions of these chips could be sold.

4.2 Financial

Banking, credit card companies, and lending institutions deal with decisions that are not clear-cut. They involve learning and statistical trends. The loan approval process involves filling out forms which hopefully can enable a loan officer to make a decision. The data from these forms is now being used by neural networks which have been trained on the data from past decisions. Indeed, to meet government requirements as to why applications are being denied, these packages are providing information on what input, or combination of inputs, weighed heaviest on the decision.

4.3 Character Recognition

Neural network based product that can recognize hand printed characters through a scanner. This product can take cards, like a credit card application form, and put those recognized characters into a database. This product has been out for two and a half years. It is 98% to 99% accurate for numbers, a little less for alphabetical characters. Currently, the system is built to highlight characters below a certain percent probability of being right so that a user can manually fill in what the computer could not. This product is in use by banks, financial institutions, and credit card companies.

4.4 Since neural networks are best at identifying patterns or trends in data, in fusion with fuzzy logic they are well suited for prediction or forecasting needs including:

- Sales forecasting
- Industrial process control
- Customer research
- Data validation
- Risk management
- Target marketing

4.5 Combination of ANN and fuzzy logic are also used in Robotics, Geophysical applications and prediction of stock index.

5. Conclusion

The computing world has a lot to gain from neuro-fuzzy systems. Their ability to learn by example and processing of unconventional data makes them very flexible and powerful. They are also very well suited for real time systems because of their fast response and computational times.

ANN and Fuzzy Logic together as hybrid systems can be used for many practical applications. Yet, the future holds even more promises. Neural networks need faster hardware. They need to become part of hybrid systems which also utilize fuzzy logic and expert systems. It is then that these systems will be able to hear speech, read handwriting, and formulate actions.

Perhaps the most exciting aspect of neural networks is the possibility that some day 'consious' networks might be produced. There are number of scientists arguing that consciousness is a 'mechanical' property and that 'consious' neural networks are a realistic possibility.

According to me, neuro-fuzzy systems can not produce miracles but when used in right direction they can give very amazing results. They will be able to become the intelligence behind robots that never tire nor become distracted. It is then that they will become the leading edge in an age of "intelligent" machines.

References:

[1] Neural Network and Fuzzy Logic. Dr.Valluru B. Rao, Hayagriva V. Rao

[2] Neural Network and Fuzzy Systems. Bart Kosko

[3] Artificial Neural Network. Kishan Mehrotra , Chilukuri K. Mohan , Sanjay Ranka

[4] Artificial Neural Network Technology (www.dacs.com)

[5] Neural Networks (http://www.aaai.org/AITopics/html/neural.html)

[6] Fuzzy Logic – An Introduction (www.seatlerobotics.com)

Fuzzy Logic and Neural Network

Bhanu Pratap Saini

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

Abstract

This paper gives a general overview of fuzzy logic and Neural Network. It describes the concepts of fuzzy sets and operations used in their manipulation, and the study of Neural Network modern neuroscience, artificial neurons, neuron networks and architecture, supervise learning, Statistical Frame work, Statistical learning theory, Associative Memory, Fuzzy principle, applications Application examples, Forecasting, Prediction Rain, temperature, flood, Humidity etc.and neurocomputing and how they are related to each other.developed by Lofti Zadeh in 1965. The paper gives examples of the fuzzy logic applications, with emphasis on the field of artificial intelligence

1. Introduction

Fuzzy logic is a powerful problem-solving methodology with a myriad of applications in embedded control and information processing. Fuzzy provides a remarkably simple way to draw definite conclusions from vague, ambiguous or imprecise information. In a sense, fuzzy logic resembles human decision making with its ability to work from approximate data and find precise solutions. Fuzzy Logic has been gaining increasing acceptance during the past few years. There are over two thousand commercially available products using Fuzzy Logic, ranging from washing machines to high speed trains. Nearly every application can potentially realize some of the benefits of Fuzzy Logic, such as performance, simplicity, lower cost, and productivity.

FL is a problem-solving control system methodology that lends itself to implementation in systems ranging from simple, small, embedded micro-controllers to large, networked, multi-channel PC or workstation-based data acquisition and control systems. It can be implemented in hardware, software, or a combination of both. FL provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information. FL's approach to control problems mimics how a person would make decisions, only much faster.

2. How is FL different from conventional control methods?

FL incorporates a simple, rule-based IF X AND Y THEN Z approach to a solving control problem rather than attempting to model a system mathematically. The FL model is empirically-based, relying on an operator's experience rather than their technical understanding of the system. For example, rather than dealing with temperature control in terms such as "SP =500F", "T <1000F", or "210C <TEMP <220C", terms like "IF (process is too cool) AND (process is getting colder) THEN (add heat to the process)" or

"IF (process is too hot) AND (process is heating rapidly) THEN (cool the process quickly)" are used. These terms are imprecise and yet very descriptive of what must actually happen. Consider what you do in the shower if the temperature is too cold: you will make the water comfortable very quickly with little trouble. FL is capable of mimicking this type of behavior but at very high rate.

3. How does FL work?

FL requires some numerical parameters in order to operate such as what is considered significant error and significant rate-of-change-of-error, but exact values of these numbers are usually not critical unless very responsive performance is required in which case empirical tuning would determine them. For example, a simple temperature control system could use a single temperature feedback sensor whose data is subtracted from the command signal to compute "error" and then time-differentiated to yield the error slope or rate-of-change-of-error, hereafter called "error-dot". Error might have units of degs F and a small error considered to be 2F while a large error is 5F. The "error-dot" might then have units of degs/min with a small error-dot being 5F/min and a large one being 15F/min. These values don't have to be symmetrical and can be "tweaked" once the system is operating in order to optimize performance. Generally, FL is so forgiving that the system will probably work the first time without any tweaking. **Neural Network** based on nodes and connections Analogous to a nerve cell - 1012 neurons and 1014 synaptic connections in the human brain

Biological Neuron

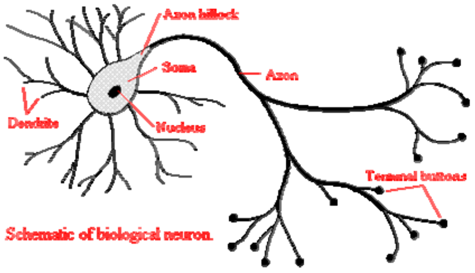


Fig-1. The Biological Neuron

The brain is a collection of about 10 billion interconnected neurons. Each neuron is a cell [right] that uses biochemical reactions to receive process and transmit information.

A neuron's dendritic tree is connected to a thousand neighbouring neurons. When one of those neurons fire, a positive or negative charge is received by one of the dendrites. The strengths of all the received charges are added together through the processes of spatial and temporal summation. Spatial summation occurs when several weak signals are converted into a single large one, while temporal summation converts a rapid series of weak pulses from one source into one large signal. The aggregate input is then passed to the soma (cell body). The soma and the enclosed nucleus don't play a significant role in the processing of incoming and outgoing data. Their primary function is to perform the continuous maintenance required to keep the neuron functional. The part of the soma that does concern itself with the signal is the axon hillock. If the aggregate input is greater than the axon hillock's threshold value, then the neuron fires, and an output signal is transmitted down the axon. The strength of the output is constant, regardless of whether the input was just above the threshold, or a hundred times as great. The output strength is unaffected by the many divisions in the axon; it reaches each terminal button with the same intensity it had at the axon hillock. This uniformity is critical in an analogue device such as a brain where small errors can snowball, and where error correction is more difficult than in a digital system.

Artificial Neuron:

Our basic computational element (model neuron) is often called a **node** or **unit**. It receives input from some other units, or perhaps from an external source. Each input has an associated **weight** w, which can be modified so as to model synaptic learning.

The unit computes some function f of the weighted sum of its inputs:

$$y_i = f(\sum_j w_{ij} y_j)$$

Its output, in turn, can serve as input to other units.

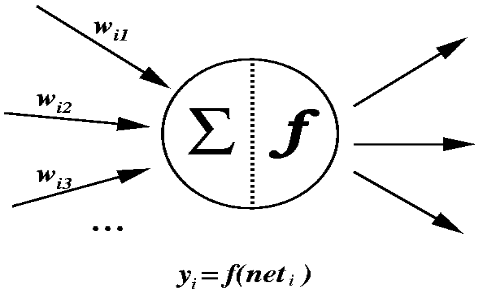


Fig-2 Artificial Neuron

- The weighted sum $\sum_j w_{ij} y_j$ is called the **net input** to unit i, often written net_i.
- Note that w_{ij} refers to the weight from unit j to unit i (not the other way around).

The function f is the unit's **activation function**. In the simplest case, f is the identity function, and the unit's output is just its net input. This is called a **linear unit**.

4. What is Neurocomputing?

Very loosely based on how the brain is thought to work.

It is an emulation of primitive neural processes in software (or hardware).

It attempts to mimic (it cannot yet copy) the workings of a biological brain.

best neural net is currently equivalent to a brain damaged worm but astonishingly this is good enough for many practical problems..

Nodes have input signals Dendrites carry an impulse to the neuron

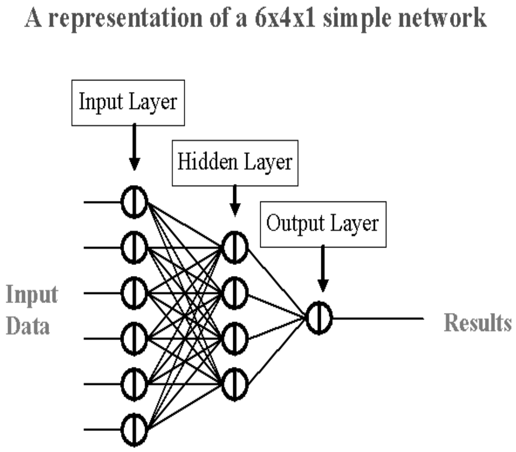


Fig.3 Representation of 6×4×1 simple network

In figure 1 Nodes have one output signal Axons carry signal out of neuron and synapses are local regions where signals are transmitted from the axon of one neuron to dendrites of another.

Input signal weights are summed at each node Nerve impulses are binary; they are “go” or “no go”. Neurons sum up the incoming signal and fire if a threshold value is reached.

5. Single-Layer Perceptron Neural Networks

The perceptron is the simplest form of a neural network used for the classifications of patterns said to be linearly separable basically it consists of a single neuron with adjustable synaptic weights and bias. The algorithm used to adjust the free parameters of this neural network first approach in learning procedure developed by Rosenblatt(1958,1962) for this perceptron brain model.

A single-layer perceptron network consists of one or more artificial neurons in parallel. The neurons may be of the same type we've seen in the Artificial Neuron Applet.

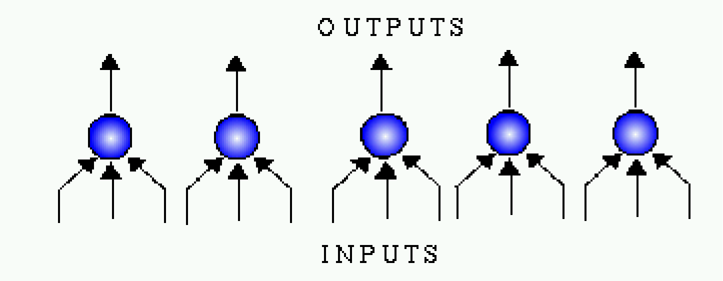


Fig4, Single Layer Perceptron

- Each neuron in the layer provides one network output, and is usually connected to all of the external (or environmental) inputs.
- The applet in this tutorial is an example of a single-neuron, single-layer perceptron network, with just two inputs.

The perceptron learning rule, which we study next, provides a simple algorithm for training a perceptron neural network. However, as we will see, single-layer perceptron networks cannot learn everything: they are not computationally complete. As mentioned in the introduction, two-input networks cannot approximate the XOR (or XNOR) functions. Of the $(2^2)^n$ or 16 possible functions, a two-input perceptron can only perform 14 functions. As the number of inputs, n, increases, the proportion of functions that can be computed decreases rapidly.

5.1 Multilayer perceptron

Multilayered Perceptron have been applied successfully to solve difficult and diverse problems by training them in a supervised manner with a highly popular algorithms known as the error back-propagation algorithm. This algorithm is based on the error-correction learning rule. As such, it may be viewed as a generalization of an equally popular adaptive filtering algorithm.

Multi-layer perceptron is the most widely used type of neural network. It is both simple and based on solid mathematical grounds. Input quantities are processed through successive layers of "neurons". There is always an input layer, with a number of neurons equal to the number of variables of the problem, and an output layer, where the perceptron response is made available, with a number of neurons equal to the desired number of quantities computed from the inputs (very often only one). The layers in between are called "hidden" layers. With no hidden layer, the perceptron can only perform linear tasks (for example a linear discriminant analysis, which is already useful). All problems which can be solved by a perceptron can be solved with only one hidden layer, but it is sometimes more efficient to use 2 hidden layers. Each neuron of a layer other than the input layer computes first a linear combination of the outputs of the neurons of the previous layer, plus a bias. The coefficients of the linear combinations plus the biases are called the weights. They are usually determined from examples to minimize, on the set of examples, the (Euclidian) norm of the desired output - net output vector. Neurons in the hidden layer then compute a non-linear function of their input. In MLPfit, the non-linear function is the sigmoid function y(x) = 1/(1+exp(-x))). The output neuron(s) has its output equal to the linear combination. Thus, a Multi-Layer Perceptron with 1 hidden layer basically performs a linear combination of sigmoid function of the inputs. A linear combination of sigmoids is useful because of 2 theorems:

- a linear function of sigmoids can approximate any continuous function of 1 or more variable(s). This is useful to obtain a continuous function fitting a finite set of points when no underlying model is available.
- trained with a desired answer = 1 for signal and 0 for background, the approximated function is the probability of signal knowing the input values. This second theorem is the basic ground for all classification applications.

6. How Neural Net Works

(I)Feedforward network

(II)Feedback network

A given node's output can be transmitted back to itself or to other previous nodes as another input all outputs only go forward

Learning Process

Learning is a process by which the free parameters of neural networks are adapted through a process of stimulation by the environment in which the network is embedded. The type of learning is determined by the manner in which the parameter change takes place

- **UNSUPERVISED LEARNING** (i.e. without a "teacher"):

It is possible to use neural networks to learn about data that contains neither target outputs nor class labels. There are many tricks for getting error signals in such **Un-supervised** settings; here we'll briefly discuss a few of the most common approaches: autoassociation, time series prediction, and reinforcement learning.

- **SUPERVISED LEARNING** (i.e. with a "teacher"):

In supervised learning Data comprises a set of discrete sample drawn from the pattern space where each sample relates as input vector

the set of samples describe the behaviour of unknown function. In supervised learning procedure we want the system to generate an output in response to an input and we say that the system has learn the underlying map if a stimulus input close to response output.

7. Applications

(I) Machine learning:

Having a computer program itself from a set of examples so you don't have to program it yourself. This will be a strong focus of this course: neural networks that learn from a set of examples.

Optimization: given a set of constraints and a cost function, how do you find an optimal solution? E.g. travelling salesman problem.

Classification: grouping patterns into classes: i.e. handwritten characters into letters. Associative memory: recalling a memory based on a partial match.

Regression: function mapping

(II)Neurobiology:

Modelling models of how the brain works. Neuron-level higher levels: vision, hearing, etc. Overlaps with cognitive folks.

(III) Mathematics:

Nonparametric statistical analysis and regression.

(IV)Philosophy:

Can human souls/behavior be explained in terms of symbols, or does it require something lower level, like a neurally based model?

(V) Finance & Banking

Firm failure prediction , Bank failure prediction, Credit card fraud prevention at Chase Manhattan Bank, American Express, and Mellon Bank examine unusual credit-charge patterns over a history of usage and compute a fraud potential rating ,Country risk rating for early warning of financial risk, Stock price prediction,Asset allocation ,Corporate merger prediction

(VI) Marketing

Customer mailing list management,Spiegel Inc. mail order catalog targets saved $1 million from reduced costs and increased sales, Airline seating allocation and passenger demand, Customer purchasing behavior and merchandising-mix strategies, Hotel room pricing - yield management

(VII) Medicine

Analysis of electrocardiogram data, Improved prosthetic devices, Pap smear ,detection of cancerous cells to drastically reduce errors,RNA & DNA sequencing in proteins, Medical image enhancement, Drug development without animal testing

(VIII) Pattern Recognition

Signature validation,OCR scanning for machine printed character recognition; also used at Post Office to sort mail,Hand printed character recognition (i.e. insurance forms) to reduce clerical data entry costs,Cursive handwriting recognition (i.e. for pen-based computing),Airport bomb detection (1989 JFK International in NY) analyzes gamma ray patterns of various objects after being struck with neutrons.

(IX) Telecommunication

Network line Fault detection

(X) Real Estate

Real Estate appraisal

(XI) Weather Forecasting

Neural network used in Weather Forecasting to predict the Temperature, Rain, Flood Humidity etc.

(XII) Monitoring

networks have been used to monitor the state of aircraft engines. By monitoring vibration levels and sound, early warning of engine problems can be given.

British Rail has also been testing a similar application monitoring diesel engines.

Software & Tools

The Neural Network Toolbox extends MATLAB with tools for designing, implementing, visualizing, and simulating neural networks. Neural networks are invaluable for applications where formal analysis would be difficult or impossible, such as pattern recognition and nonlinear system identification and control. The Neural Network Toolbox provides comprehensive support for many proven network paradigms, as well as graphical user interfaces (GUIs) that enable you to design and manage your networks. The modular, open, and extensible design of the toolbox simplifies the creation of customized functions and networks.

Conclusions

The Fuzzy logic and Neuron network technique is very useful in problem-solving methodology with a applications in embedded control and information processing. Fuzzy provides a remarkably simple way to draw definite conclusions. with the achieved results exhibiting high levels of accuracy, consistency and reliability, with acceptably low computational time. The fuzzy-neural based classifier provides better performance and combines the benefits of both neural networks and fuzzy logic.

References

[I] Fuzzy Logic Research and Life, Japanese Technology Evaluation Center, 1995, (http://itri.loyola.edu/kb/c5_s4.htm)

[II] Hochreiter, Sepp and Schmidhuber, Juergen, (1997) "Long Short-Term Memory", Neural Computation, Vol 9 (8), pp: 1735-1780

[III] Fausett L., Fundamentals of Neural Networks, Prentice-Hall, 1994. ISBN 0 13 042250 9

[IV] Neural Computing and Applications, Springer-Verlag. (address: Sweetapple Ho, Catteshall Rd., Godalming, GU7 3DJ)

[V] Staff Development Program on Neurocomputing from 8-20 Jan 2007 by Prof. J.P.Saini H.O.D, ECE Deptt. B.I.E.T. Jhansi

[VI] IEEEVolume 3, Issue , 2-6 Nov. 2003 Page(s): 2103 - 2107 Vol.3

[VII] Wikipedia The free encyclopedia

[VIII] Neural Networks A comprehensive Foundamental by Simon Haykin

Proposed Approach for Access of Data

Dharmendra Pal

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

Abstract

This paper is a proposed work of using internet services for application of data sources at the internet traffic disorders. I am presenting architecture for data access from various data sources based on internet Services to assist with access of data lying at different locations in various data sources and document generation. This may assist for the automatic production of file documentation lying there for application uses.

The autonomous nature of the actively participating data source is preserved and evolution of the system is allowed by addition and removal of data sources. With these efforts, the proposed architecture supports the technical and organizational problems generally posed by the wide spread use of different technologies and implementation standards. It may improve the faced problems of imposing a single technology throughout the organization. The reason behind this all is the characteristics of the system of hiding the information itself. The autonomous nature of participation to data sources binds it with them.

1. Introduction

Accessing data and applications at a remote end is provided by the development of distributed computing and distributed networking. The changed environment has become more useful due to the development of different systems. Having a large number of applications interoperating with each other is the main problem to be solved. The applications define different data formats, having their own communication protocols and, are developed on different platforms because of not been built to be integrated. The main challenge is to introduce Interoperability of distributed systems.

This type of problem is found in the Internet traffic report which monitors the flow of data around the world. It displays a value between zero and 100. Internet traffic leads the efforts to manage inefficiencies of the access or /and manipulation. It helps internet to modernize and expand efforts in the field of interest areas such as: knowledge based mining, research field, personality development and finding new domains of development by providing assistance in all respect. To collect, analyze and disseminate information is the function of it to assist internet traffic to find knowledge based information and achieve security of resources available within the working area. An idea is proposed to establish the information management and access and/or manipulation improvement. The data represented in various distinct formats and in various languages is stored in different types of data sources. Through this paper I am presenting an idea to log up interoperability of the different data sources in the Internet traffic. The approach is based on the use of an Interpretability of Data Sources - providing information for allowing exchange of data between various information sources implemented by using different technologies.

2. Problems

Working at the Internet traffic report of the world, I got in close contact of approximately 200 systems supplying information for access on the World Wide Web, deployed using mainly two different technologies: Microsoft ASP [2] and Java JSP/ servlets [3]. The available data sources share and exchange data between each other in an easy mode. Using these technologies was already widespread, so it was a difficult task to apply a single technology. The existing information infrastructure has database systems containing different types of data including, but not limited to, different types of documents. Different people generate documents in different formats, which are inserted in the databases using web interfaces. The HTML format may best fited for the access. Organization may access /manipulate data lying in various sources of it. Data should be shared between systems quickly and easily without any tight copulation. Main problem is the widespread use of Microsoft ASP and Java JSP/ Servlets within the organization. Some of the requirements of integrating disparate distributed systems in the existing architecture were having limitations involving budgetary or technical challenges, inflexibility, and difficulty of scalability. I came to the conclusion that there must be a technology that is inexpensive, easy to implement, easy to maintain and based on open standards, to allow influence of knowledge and existing resources. The technology needs to support interoperability of existing data sources and management. It is necessary to convert and add database structures for each different language. At the same time the database models are not easily extensible when new data or language variants are introduced.

3. The objective

The main idea is to allow interoperability between existing data sources in the organization, in a way to be implemented on multiple vendor platforms, with minimal effort. The objective of the approach is to create an environment where new web-based information systems can be developed quickly and easily, using any technology platform, by accessing information from any of the existing information systems. The objective is to show that regardless of the technology platform used, Internet Services could be used to integrate rapidly data extracted from any of the RDC systems.

3.1 The Architecture

It is proposed that the development of two (functionally) identical applications for the organization must sustain, but using two different Web Services technologies - Microsoft .NET [9] and JAVA (from the Apache Software Foundation [1]).

- Diagram 'A' presents an overall overview of the architecture with all its components. The architecture consists of an Interpretability of Data Sources with XML structures. The different data sources are associated with Internet services interfaces.
- There is no need to re-engineer existing systems to new XML standards. For maintaining XML standards in the Internet Services interfaces the parameters for operations involving language codes using the 2-character ISO 639 code.

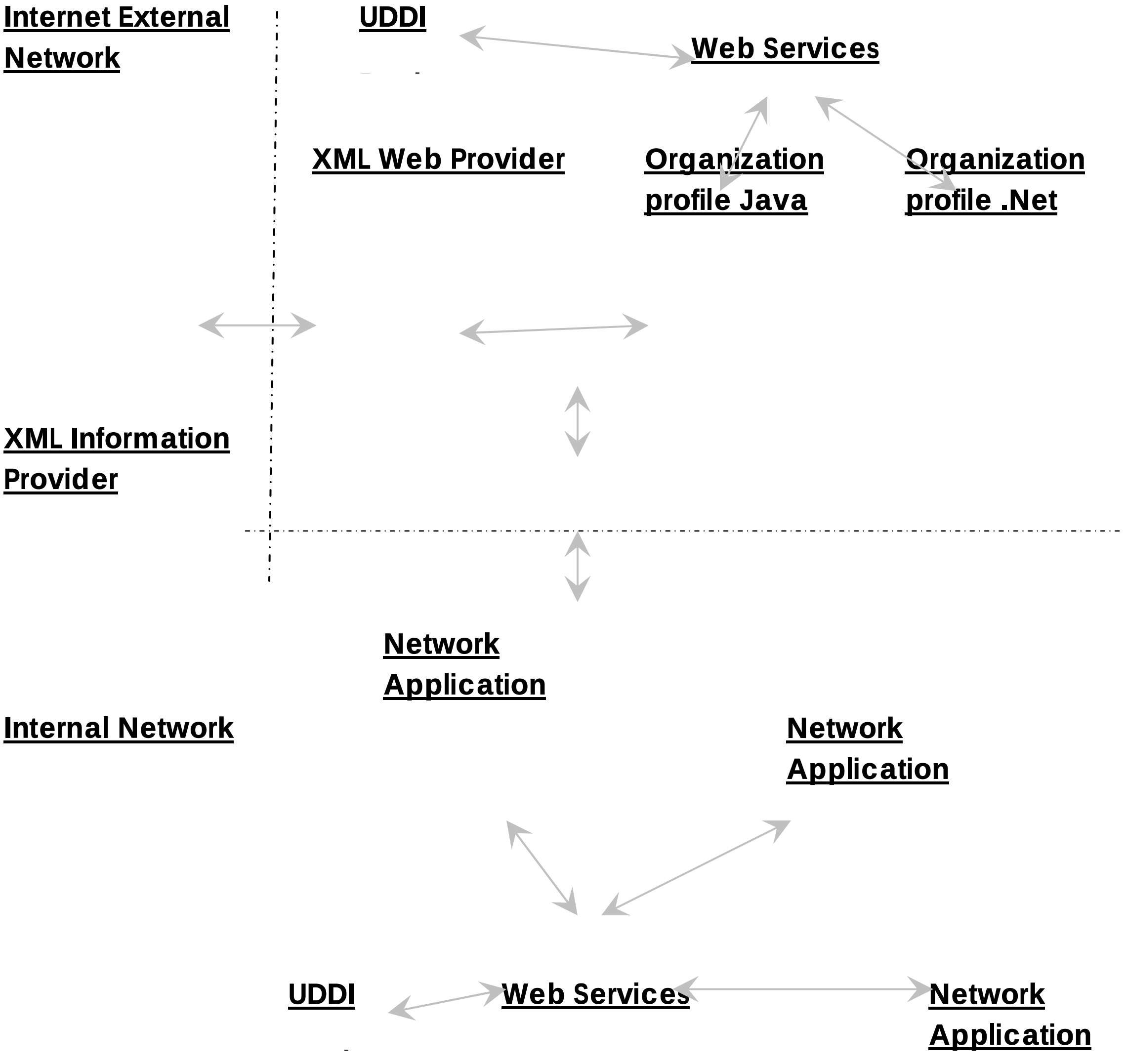


Diagram A: Proposed Architecture for the Interpretability of Data Sources

3.2 Procedural Development

The concept behind the proposed architecture of the Interpretability of Data Sources is that all data passes through it in standard XML formats. These formats must be applied in a regulated fashion by publishing the XML schemas and, the same XML syntax is used for input /output parameters on the Internet Services. For existing applications, to which the development team had access to the source code, one great advantage of the .NET framework was the ease with which Internet Services Packets could be created. These Internet Services did not integrate with the JAVA platform without some problems.

There were problems integrating .NET Services with JAVA Services, this is because .NET uses Document-style web services by default, whereas the JAVA implementation (Apache Axis) uses RPC-style invocation. To solve this problem in .NET, I used the SoapRpcService() property to indicate that the .NET web service was RPC-style. The reason behind the problems was- Axis did not yet implement support for multi-dimensional arrays for generating complex type definitions which are created automatically by .NET. For allowing developers to create Internet Services quickly and easily from existing MS applications, a second tier of Internet Services was created that automatically made the transformation from the data types generated by .NET to XML arrays that could be used by both Services .NET and JAVA.

3.3 Related development work

The challenge of interoperating distributed systems, in particular database systems, has existed for a long time and researched much. So many ideas have been proposed to allow integration and interoperability of distributed systems developed in some free way. These ideas have come as outcomes of research work in most prominent fields. The construction of a global schema does not guarantee the autonomy of the participating database systems, and doesn’t allow easy evolution of the system in terms of adding /removing of participating databases. For solving the problem of constructing a global integrated schema the ideas have been proposed. Examples of these ideas include the federated architecture [10] and multidatabase architecture . Within the approaches , some of them proposed the use of mediators and packets. In these ideas data sources are encapsulated to form it usable in a much convenient way by hiding / exposing the internal interface of the data sources, reformat data, and translating the queries.

There is no standard way of formatting or describing the values in the files as many systems use ASCII-based text files to represent their data.. The different systems exchange data in ASCII format must have custom-built loading software to handle different file formats. Other systems exchange data via a specified file format, which does not scale well (e.g. Microsoft Excel). Data transmission has also become difficult to implement. The use of the File Transfer Protocol (FTP) facilitates file transfer. Anyhow this is not a tight, object-oriented approach to exchange data. Electronic Data Interchange (EDI) [8] has also been used for exchanging data for the same purpose..

Conclusion

An architecture is proposed with some idea to allow interoperability of different data sources with distinct types of information of various systems, developed on different platforms. My scrutiny results are very accelerating as it is very easy to develop the packets around the data sources. The GUI of MS .NET is very intuitive and facilitates the development of Web Services from existing Microsoft applications.

It was not possible to directly integrate .NET Internet Services with JAVA Services, as there was difference in handling complex data types and as well as inconsistencies in the use of WSDL. It was possible to integrate Web Services from the different platforms by writing simple and generic services. The concept offers permission of sharing, exchanging, and merging of data in that way which was not possible earlier anyway in any condition. However based on the working experience with this Interpretability of Data Sources I expect to be able to use the approach in other situations where it is necessary to interoperating distributed systems.

References

[1] Apache. Apache Project. http://www.apache.org

[2] ASP. Microsoft Active Server Pages. http://www.microsoft.com/asp

[3] Java. Java Java Server Pages/Servlets. http://java.sun.com

[4] D. Brickley and R.V. Guha. Resource Description Framework (RDF) Schema Specification 1.0, March 2002. http://www.w3.org/TR/rdf-schema

[5] D. Box, D. Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H. Nielsen, S. Thatte, and D. Winer. Simple Object Access Protocol (SOAP) 1.1. http://www.w3.org/TR/SOAP.

[6] C. Chung. DATAPLEX: An Access to Heterogeneous Distributed Databases. Communications of the ACM, 33(1):70-80, January 1990.

[7] CORBA/IIOP. Common Object Request Broker Architecture. http://www.omg.org/technology/documents/formal/corba-iiop.htm.

[8] EDI - Electronic Data Interchange (See http://www.diffuse.org/edi.html)

[9] .Net. Microsoft .Net. http://www.microsoft.com/net

[10] D. Heimbigner and D. McLeod. A Federated Architecture for Info. Management. ACM Transaction on Office Information Systems, 3(3):253-278, July 1985.

Wired Equivalent Privacy and Fuzzy Logic

Kavita Choudhary

Lecturer, KIIT College of Engineering, Gurgaon

kavitapunia@gmail.com

ABSTRACT

Wired **E**quivalent **P**rivacy, a security protocol for wireless local area networks (WLANs) defined in the 802.11b standard. WEP is designed to provide the same level of security as that of a wired LAN. LANs are inherently more secure than WLANs because LANs are somewhat protected by the physicalities of their structure, having some or all part of the network inside a building that can be protected from unauthorized access. WLANs, which are over radio waves, do not have the same physical structure and therefore are more vulnerable to tampering. WEP aims to provide security by encrypting data over radio waves so that it is protected as it is transmitted from one end point to another. However, it has been found that WEP is not as secure as once believed. WEP is used at the two lowest layers of the OSI model - the data link and physical layers; it therefore does not offer end-to-end security.WEP is often the first security choice presented to users by router configuration tools even though it provides a level of security that deters only unintentional use, leaving the network vulnerable to deliberate compromise.

WEP (Wired Equivalent Privacy)—WEP is an 802.11 standard encryption algorithm originally designed to provide your wireless LAN with the same level of privacy available on a wired LAN.However, the basic WEP construction is flawed, and an attacker can compromise the privacy with reasonable effort. LANs are inherently more secure than WLANs because LANs are somewhat protected by the physicalities of their structure, having some or all part of the network inside a building that can be protected from unauthorized access. WLANs, which are over radio waves, do not have the same physical structure and therefore are more vulnerable to tampering. WEP aims to provide security by encrypting data over radio waves so that it is protected as it is transmitted from one end point to another.

1. Why Fuzzy Logic is required in WEP?

- WEP is a security protocol defined in IEEE 802.11 standard. Fuzzy Logic is needed to overcome the drawback of WEP.

1.1 802.11 Wireless Networks

802.11 wireless networks operate in one of two modes- ad-hoc or infrastructure mode. The IEEE standard defines the ad-hoc mode as Independent Basic Service Set (IBSS), and the infrastructure mode as Basic Service Set (BSS). In the remainder of this section, we explain the differences between the two modes and how they operate. In ad hoc mode,each client communicates directly with the other clients within the network. Ad-hoc mode is designed such that only the clients within transmission range (within the same cell) of each other can communicate. f a client in an ad-hoc network wishes to communicate outside of the cell, a member of the cell MUST operate as a gateway and perform routing. In infrastructure mode, each client sends all of it's communications to a central station, or access point (AP). The access point acts as an Ethernet bridge and forwards the communications onto the appropriate network–either the wired network, or the wireless network. Prior to communicating data, wireless clients and access points must establish a relationship, or an association. Only after an association is established can the two wireless stations exchange data. In infrastructure mode, the clients associate with an access point. The association process is a two step process involving three states:

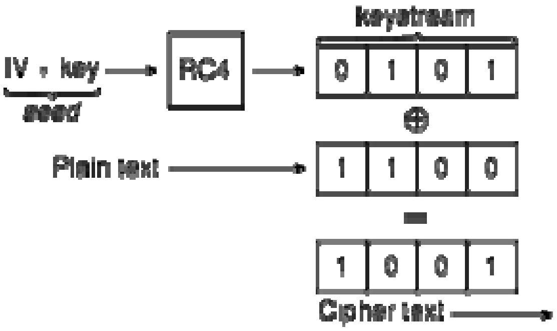
1. Unauthenticated and unassociated,

2. Authenticated and unassociated, and

3. Authenticated and associated.

To transition between the states, the communicating parties exchange messages called management frames. The Wireless Ethernet Compatibility Alliance (WECA) claims that WEP - which is included in many networking products - was never intended to be the sole security mechanism for a WLAN, and that, in conjunction with traditional security practices, it is very effective. WEP was included as the privacy of the original IEEE 802.11 standard ratified in September 1999. WEP uses the stream cipher RC4 for confidentiality, and the CRC-32 checksum for integrity. It was deprecated as a wireless privacy mechanism in 2004, but for legacy purposes is still documented in the current standard.

Encryption details



1.2 Basic WEP encryption: RC4 keystream XORed with plaintext

Standard 64-bit WEP uses a 40 bit key (also known as WEP-40), which is concatenated with a 24-bit initialization vector (IV) to form the RC4 traffic key. At the time that the original WEP standard was being drafted, U.S. Government export restrictions on cryptographic technology limited the key size. Once the restrictions were lifted, all of the major manufacturers eventually implemented an extended 128-bit WEP protocol using a 104-bit key size (WEP-104). A 128-bit WEP key is almost always entered by users as a string of 26 hexadecimal (base 16) characters (0-9 and A-F). Each character represents four bits of the key. 26 digits of four bits each gives 104 bits; adding the 24-bit IV produces the final 128-bit WEP key. A 256-bit WEP system is available from some vendors, and as with the 128-bit key system, 24 bits of that is for the IV, leaving 232 actual bits for protection. These 232 bits are typically entered as 58 hexadecimal characters. (58 × 4 = 232 bits) + 24 IV bits = 256-bit WEP key.

Key size is not the only major security limitation in WEP. Cracking a longer key requires interception of more packets, but there are active attacks that simulate the necessary traffic. There are other weaknesses in WEP, including the possibility of IV collisions and altered packets that are not helped at all by a longer key.

2. Authentication

Two methods of authentication can be used with WEP: Open System authentication and Shared Key authentication. For the sake of clarity, we discuss WEP authentication in the Infrastructure mode (i.e., between a WLAN client and an Access Point), but the discussion applies to the Ad-Hoc mode as well.

In Open System authentication, the WLAN client need not provide its credentials to the Access Point during authentication. Thus, any client, regardless of its WEP keys, can authenticate itself with the Access Point and then attempt to associate. In effect, no authentication (in the true sense of the term) occurs. After the authentication and association, WEP can be used for encrypting the data frames. At this point, the client needs to have the right keys.

In Shared Key authentication, WEP is used for authentication. A four-way challenge-response handshake is used:

a) The client station sends an authentication request to the Access Point.
b) The Access Point sends back a clear-text challenge.
c) The client has to encrypt the challenge text using the configured WEP key, and send it back in another authentication request.
d) The Access Point decrypts the material, and compares it with the clear-text it had sent. Depending on the success of this comparison, the Access Point sends back a positive or negative response.

After the authentication and association, WEP can be used for encrypting the data frames. At first glance, it might seem as though Shared Key authentication is more secure than Open System authentication, since the latter offers no real authentication. However, it is quite the reverse. It is possible to derive the keystream used for the handshake by capturing the challenge frames in Shared Key authentication. Hence, it is advisable to use Open System authentication for WEP authentication, rather than Shared Key authentication. (Note that both authentication mechanisms are weak).

Because RC4 is a stream cipher, the same traffic key must never be used twice. The purpose of an IV, which is transmitted as plain text, is to prevent any repetition, but a 24-bit IV is not long enough to ensure this on a busy network. The way the IV was used also opened WEP to a related key attack. For a 24-bit IV, there is a 50% probability the same IV will repeat after 5000 packets. Many WEP systems require a key in hexadecimal format. Some users choose keys that spell words in the limited 0-9, A-F hex character set, for example C0DE C0DE C0DE C0DE. Such keys are often easily guessed.

3. Weakness

Because RC4 is a stream cipher, the same traffic key must never be used twice. The purpose of an IV, which is transmitted as plain text, is to prevent any repetition, but a 24-bit IV is not long enough to ensure this on a busy network. The way the IV was used also opened WEP to a related key attack. For a 24-bit IV, there is a 50% probability the same IV will repeat after 5000 packets.

- Key management is not specified in the WEP standard and because of this problem , keys will tend to be long-lived and of poor quality.

- Brute Force attack.

- Authentication messages can be easily forged.

- Fluhrer, Mantin, and Shamir Attack- The attack exploits the method in which the standard describes using IVs for the RC4 stream cipher.

4. Patches and Progress

- WEPplus remedies the initialization vector problem.
- Temporal Key Integrity Protocol (TKIP) and Advanced Encryption Standard (AES) is designed to remedy WEP's key problem by changing the temporal key after every 10,000 packets.

5. Future Scope

The biggest WEP issue today is the weakness that remains even as the technology evolves. For a 24-bit IV, there is a 50% probability the same IV will repeat after 5000 packets. We can apply artificial intelligence technique after every 4000-5000 packet to restrict the value repetition. For this problem Fuzzy logic is required. We can implement WEP through Fuzzy logic to solve this problem.

References

http://delivery.acm.org/10.1145/1370000/1366341/p159-bernaschi.pdf?key1=1366341&key2=7129557421&coll=ACM&dl=ACM&CFID=45420903&CFTOKEN=18303323

http://delivery.acm.org/10.1145/1520&CFID=45417991&CFTOKEN=93598617

[3] www.**wep**india.com

[4] www.searchsecurity.techtarget.com/.../0,,sid14_gci549087,00.htm

Proceedings of the National Seminar on Artificial Intelligence - NSAI-09
July 17, 2009

Dronacharya College of Engineering, Khentawas, Farrukh Nagar, Gurgaon-123506

An Introduction to Data mining and its Applications

Kumar Rahul

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

rahulmese@gmail.com

Abstract

Data mining is a combination of database and artificial intelligence technologies. Although the AI field has taken a major dive in the last decade; this new emerging field has shown that AI can add major contributions to existing fields in computer science. In fact, many experts believe that data mining is the third hottest field in the industry behind the Internet, and data warehousing.

Data mining is really just the next step in the process of analyzing data. Instead of getting queries on standard or user-specified relationships, data mining goes a step farther by finding meaningful relationships in data. Relationships that were thought to have not existed or ones that give a more insightful view of the data. For example, a computer-generated graph may not give the user any insight; however data mining can find trends in the same data that shows the user more precisely what is going on. Using trends that the end-user would have never thought to query the computer about. Without adding any more data, data mining gives a huge increase in the value added by the database. It allows both technical and non-technical users get better answers, allowing them to make a much more informed decision, Saving their companies millions of dollars.

1. Introduction

"Data mining is the process of discovering meaningful new correlations, patterns, and trends by sifting through large amounts of data stored in repositories, using pattern recognition technologies as well as statistical and mathematical techniques" (SPSS). However, really data mining turns databases into knowledge bases which are one of the fundamental components of expert systems. Instead of the computer just blindly pulling data from a database, the computer is able to take all the data and interpret it, which is a huge step to make. If it was not for existing AI technologies this field could not have emerged as quickly; if at all.

Data mining allows companies to focus on the more important information in their data warehouses. Data mining can be broken down into two major categories. Automated prediction of trends and behaviors, and automated discovery of previously unknown patterns. In the first category, data mining automates the process of finding predictive information in large databases. Questions that traditionally required exhaustive hands-on analysis can now be quickly answered directly from data. In the second category, data mining tools sweep through databases and identify previously hidden patterns in one step. This category is where the major focus of research has been on.

2. Discussion

"Data mining is a rather new term for a challenge that has been growing for many years: how to scan very large databases to retrieve the high level conceptual information of the greatest interest" (Lindsay). With the advances in data acquisition and storage technologies, the problem of how to turn measured raw data into useful information becomes a important one. Having reached sizes that defy even partial examination by humans, the data volumes are literally swamping users. For example, large US retail chains now mine their data bases with sophisticated data mining programs to look for general trends and geographic clustering in purchases that are not easily visible in the huge multitude of products and sales.

Data mining has come from an evolution of searching through data trying to find useful business information. There are four major steps: Data Collection, Data Access, Data Warehousing & Decision Support, and finally Data Mining (Pilot).

Data Collection started in the 1960s. This is a static data delivery system that came from pulling information from computers, tapes, and disks. For example, what is the total revenue in the last five years. Data Access is the next step and it started in the 1980s. This allowed dynamic data delivery at the record level. Data access mainly uses relational databases using SQL. A Sample Query would be: What were unit sales in Florida last October. Then in the 1990s came Data warehousing and decision support. This allowed dynamic data delivery at multiple levels. This technology came about, because of multidimensional databases and on-line analytic processing (OLAP). This will let the query above go as detailed as city to city in Florida. Finally came data mining, which allowed proactive information delivery. Data mining uses Advanced AI algorithms, multiprocessor computers, and massive databases. With data mining a person ask questions like what is likely to happen to Florida unit sales next month and why (Pilot).

Fundamentally, data mining does two things with data: It finds relationships and makes forecasts. Within these two categories, data mining is good at producing the following six information types (Newquist): Classes, Clusters, Associations, Sequences, Forecasts, and Similar Sequences. Classes are the most common form of data mining, and consist of shared characteristics, such as how many or what percentage of people over the age of 40 have checking and saving accounts but no investments in mutual funds. A data mining tool uses pattern recognition to create classes. Clusters are a subset of classes that consist of patterns and relationships that have not been predefined or were not previously have known to exist. Data mining finds these relationships even though the user was not specifically looking for them.

Associations deal with events. That is, an association exists when the completion of one occurrence implies the existence of another. For example, when people buy beer, 60 percent of the time they buy some form of snack. Sequences deal with events also. However they are linked over time instead. For example, credit card holders that ask for an increase in limit usually buy a large item within the next two weeks.

Forecasts involve predicting the future on current data. Forecasts are applicable to almost any corporate situation. Extracting all relevant data and applying them with relevant fluctuations makes forecasts. Similar sequences extend the concept of sequences by combining them conceptually with classes. For example, after discovering a sequence in a particular time, a user might want to find other sequences occurring at the same time or search for similar sequences over time (Newquist).

The most common techniques in data mining are artificial neural networks, decision trees, genetic algorithms, nearest neighbor method, and rule induction (Pilot).

Artificial neural networks are non-linear predictive models that learn through training, and closely resemble biological networks. Decision trees are tree-shaped data structures that represent sets of decisions. These decisions generate rules for the classification of a dataset. Genetic Algorithms use optimization techniques that use concepts of evolution such as combination, mutation, and natural selection. Nearest neighbor method is a technique that classifies each record in a dataset based on a combination of classes of records most similar to them in a historical dataset. Rule induction is the extraction of useful if-then rules from databases on statistical significance (Pilot).

The main reason for the necessity of automated computer systems for intelligent data analysis is the enormous volume of existing and newly appearing data that requires processing. The amount of data accumulated each day by various businesses, scientific, and governmental organizations around the world is appalling. According to GTE research, scientific organizations store about 1 terabyte of new information each day (Mega computer). It is impossible for human analysts to cope with such overwhelming amounts of data.

Two problems that surface when human analysts process data are the inadequacy of the human brain when searching for complex dependencies in data, and the lack of objectiveness in their analysis. Therefore, one of the benefits of using automated data mining systems is that this process has a much lower cost than hiring an army of highly trained and paid professional statisticians. Although data mining does not completely eradicate the need for humans, it allows an analyst who has no programming and statistics skill to extract knowledge from databases (Mega computer).

These applications of data mining:

- Ad revenue forecasting
- Churn (turnover) management
- Claims processing
- Credit risk analysis

- Cross-marketing
- Customer profiling
- Customer retention
- Electronic commerce
- Exception reports
- Food-service menu analysis
- Fraud detection
- Government policy setting
- Hiring profiles
- Market basket analysis
- Medical management
- Member enrollment
- New product development
- Pharmaceutical research
- Process control
- Quality control
- Shelf management / store management

Conclusion

Data Mining is the extraction of hidden predictive information from large databases. This is a new powerful new technology with great potential to help companies focus on the most important information in data warehousing. Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. "The automated, prospective analyses offered by data mining move beyond the analyzes of past events provided by retrospective tools typical of decision support systems".

Data mining is important to large systems because it finds things in large data repositories that you did not know existed. "A simple metaphor would be finding two needles in a haystack that match. The haystack is the database, the individual lengths of the hay represent your data fields, and the needles represent data fields with a relationship worth more to you than all the hay put together".

References

[1] Mega Computer. "Reasons for the growing popularity of data mining." Online. Internet. 3 Oct. 2008.

[2]Lindsay, Clark. "Data Mining." Online. Internet. 2007 Available: http://msia02.msi.se/~lindsay/datamine.html

[3] Usama Fayyad, "Mining Databases: Towards Algorithms for Knowledge Discovery", Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, vol. 21, no. 1, March2008.

http://www.research.microsoft.com/research/db/debull/98mar/issue.html

[4] Christopher Matheus, Philip Chan, and Gregory Piatetsky-Shapiro, "Systems for Knowledge Discovery in Databases", IEEE Transactions on Knowledge and Data Engineering, 5(6):903-913, November 2006.

[5] Rakesh Agrawal and Tomasz Imielinski, "Database Mining: A Performance Perspective", IEEE Transactions on Knowledge and Data Engineering, 5(6):914-925, December 2007.

Database Security from Malicious Attack: Multi Agent System Approach

Narendra Kumar Tyagi

Asstt.Professor, CSE & IT Deptt, DCE, Gurgaon

Abhilasha Vyas

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

Abstract

The paper is the analysis modeling and simulation of malicious attacks against database security in the networks. The attack, damage and prevention to network security are the main research work in this paper. The authors present the real experience in applying multi agent technology for attack simulation. The approach suggested attack simulation is realized through self generated replication of computer network attacks particularly in distributed environment. The paper describes Agent-Based Attack Simulator with tools.

Keywords

Agent-based modeling and simulation, active vulnerability assessment, computer network attacks, penetration testing

1. Introduction

The appearance of incidents of attacking the computers and networks has given a way for the necessity of attack modeling and simulation. Using the knowledge obtained from the generalization and formalization of computer systems and cases of malicious attacks could improve the efficiency of protection mechanisms in the system. The research of the essence and the particular features of malicious attacks against computer networks are strongly favored by this approach. The author realized the focus of this research not only for the generalization of the experience; but it also is based on formal models of attacks and attack simulation tools. The security systems which are capable of operating with identifying of malicious attacks can take the advantages for their designing in their model and tools. These capabilities enable the system to dominate the development of malicious attack on-line before the irreparable consequences may occur. The role of the attack simulation tool is in the validation of security systems and setting policies, thus become the testing equipment cutting the costs of and decreasing the time for security policy validation. This research paper contains a general analysis approach of development of research. Mathematical models and software tools proposed for agent-based modeling and simulation of attacks and active analysis of computer network.

2. History Modeling and Simulation against Malicious Attacks

The researches till now reveal the attack modeling and simulation in various groups. Main works are depicted in Table 1.

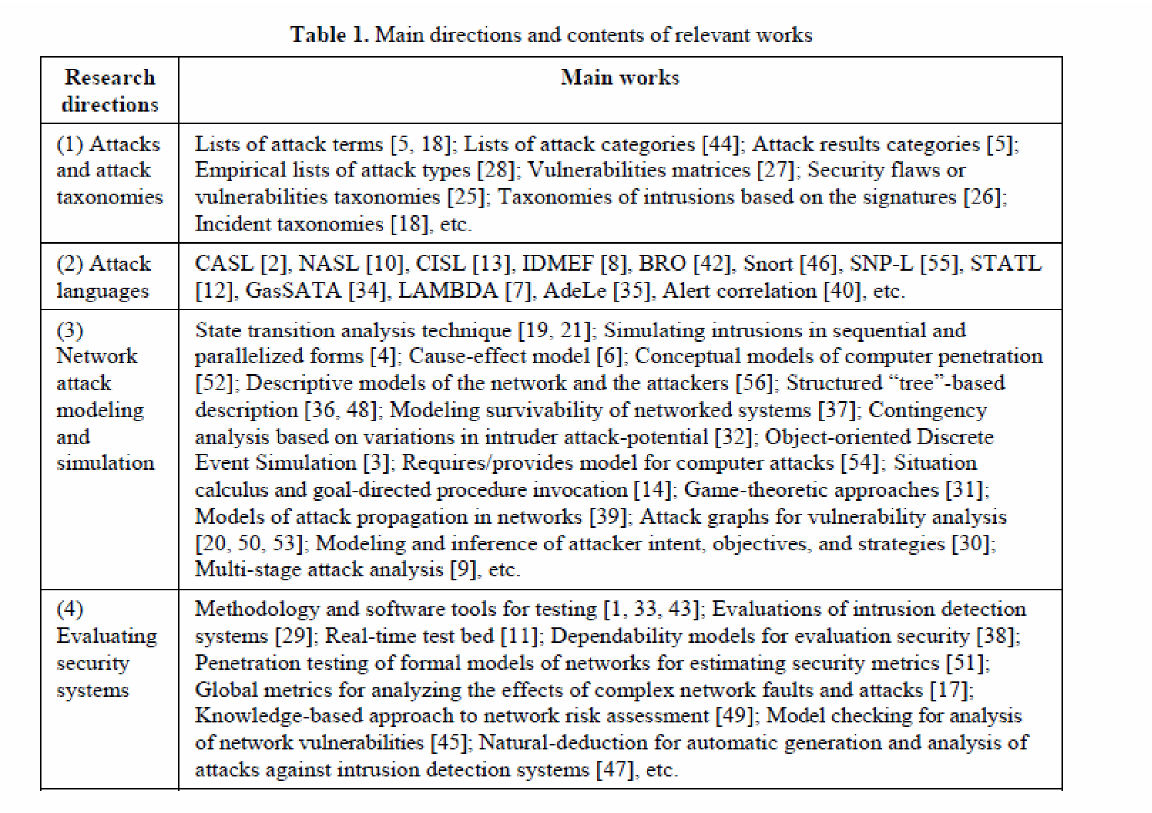


Table 1. Main directions and contents of relevant works

Research directions	Main works
(1) Attacks and attack taxonomies	Lists of attack terms [5, 18]; Lists of attack categories [44]; Attack results categories [5]; Empirical lists of attack types [28]; Vulnerabilities matrices [27]; Security flaws or vulnerabilities taxonomies [25]; Taxonomies of intrusions based on the signatures [26]; Incident taxonomies [18], etc.
(2) Attack languages	CASL [2], NASL [10], CISL [13], IDMEF [8], BRO [42], Snort [46], SNP-L [55], STATL [12], GasSATA [34], LAMBDA [7], AdeLe [35], Alert correlation [40], etc.
(3) Network attack modeling and simulation	State transition analysis technique [19, 21]; Simulating intrusions in sequential and parallelized forms [4]; Cause-effect model [6]; Conceptual models of computer penetration [52]; Descriptive models of the network and the attackers [56]; Structured "tree"-based description [36, 48]; Modeling survivability of networked systems [37]; Contingency analysis based on variations in intruder attack-potential [32]; Object-oriented Discrete Event Simulation [3]; Requires/provides model for computer attacks [54]; Situation calculus and goal-directed procedure invocation [14]; Game-theoretic approaches [31]; Models of attack propagation in networks [39]; Attack graphs for vulnerability analysis [20, 50, 53]; Modeling and inference of attacker intent, objectives, and strategies [30]; Multi-stage attack analysis [9], etc.
(4) Evaluating security systems	Methodology and software tools for testing [1, 33, 43]; Evaluations of intrusion detection systems [29]; Real-time test bed [11]; Dependability models for evaluation security [38]; Penetration testing of formal models of networks for estimating security metrics [51]; Global metrics for analyzing the effects of complex network faults and attacks [17]; Knowledge-based approach to network risk assessment [49]; Model checking for analysis of network vulnerabilities [45]; Natural-deduction for automatic generation and analysis of attacks against intrusion detection systems [47], etc.

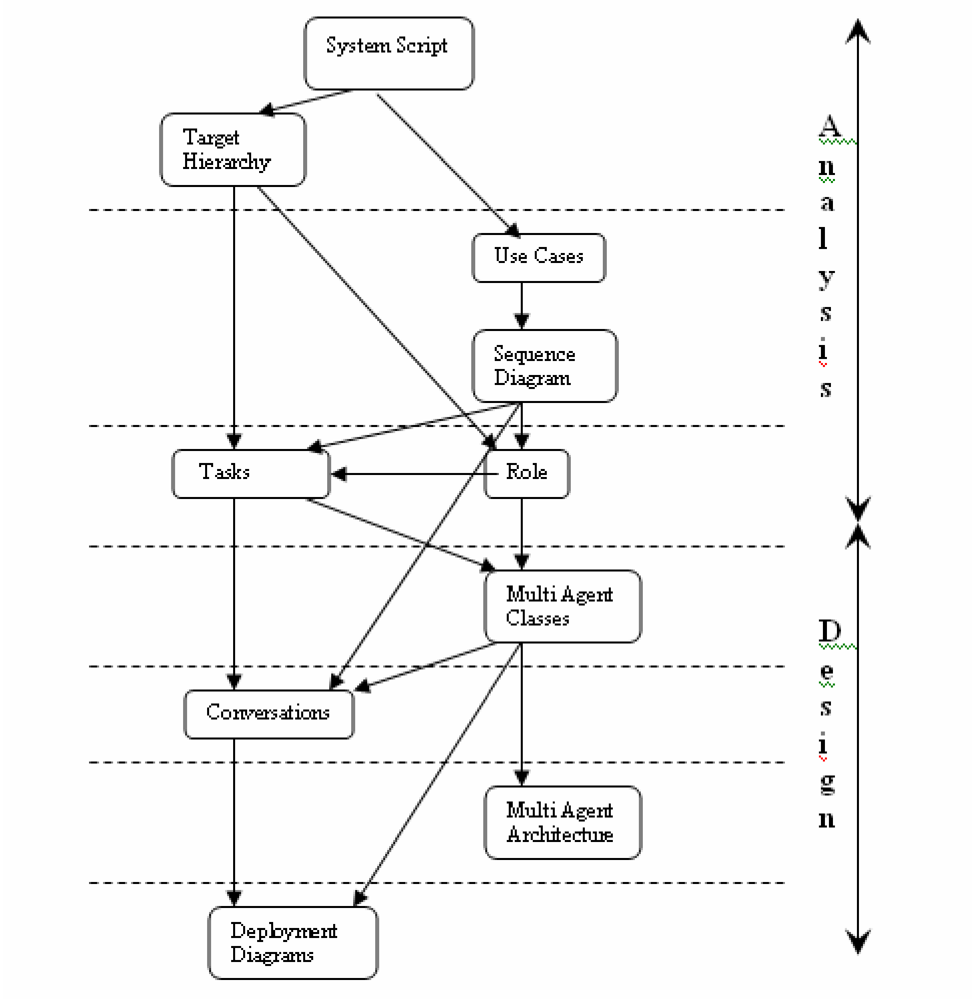
We proposed an approach for the investigation of remote distributed malicious attacks on computer networks which has not been analyzed in depth is agent-based modeling and simulation. Our approach has applied the results of reviewed relevant works, but is evolving own theoretical and practical ideas about using formal models and multi-agent technology. We extend the idea that particular components of attack simulation system must be represented as a distributed system of self generated adaptive software entities interacting through message exchange and making decisions in a cooperative way [22]. Thus a computer network malicious attack can be considered as a sequence of coordinated actions of the spatially distributed factors. Each offender is routed like an intelligent agent of the similar architecture having the same functionality and interoperability. The group interpretation of the offenders' activity performing distributed attacks on the basis of combination of the shared plans theories [22, 24]. The attack simulation system is based on mechanism of self generating construction and replaying of distributed attacks scripts. It is done by combining known attacks fragments, taking into account the offenders' malicious intentions and knowledge of computer network attacked. The attack model specifies the functioning of the attack simulation system. This is defined as hierarchical structure consisting of several levels. We defined main saying of attack generation that are formalized in the problem domain ontology "Computer network attacks" [16]. The basis of the domain matches with offenders' intentions. Everything is structured according to his intentions. That's why the developed approach is also called as "intention-centric approach". The levels of the attack model correspond to a particular script and script stages. This level is the realization of series of scenarios e implemented by a group of offenders. The single script rank defines one offender's intention. The group of script ranks contain: investigation, initial access to a host, threat realization and covering tracks. The lowest rank points the offender's low level actions executing different exploits. The attack simulation system operates with the model of analyzed computer network. This approach can be used at different stages of computer network life cycle, including design and exploitation stages.

3. Agent-Based Attack Simulator

The Agent-Based Attack Simulator (ABAS) [16] is built as a multi-agent system .It uses two classes of agents: the "Network Agent" which simulates an attacked computer network and the "Hacker Agent" which a hacker performing attacks against the computer network. The basis of the technology implementation agents are supported by Multi-Agent System Development Kit (MASDK) [15]. Different parts of the application ontology (designed by use of the MASDK editor) are used by the agents. The communication component supports the interaction between agents. During simulating an attack Hacker Agent sends a certain message to the Network Agent, which analyzes the received message and forms a responsive message. This responsive message is formed based on the Network Agent's knowledge base. It represents the network configuration. It also represents the information about possible attacks which exists and reaction of the network on them. The state-machine models explain the behaviors of the agents. These behaviors of agents are interpretations of behavior specified formally by use of formal grammar framework. Hacker Agent acts on the basis of nested state machines. The state machine model of Network Agent is represented by a single state machine. Looking at the behaviour discussed above, it is clear that we can model agents as "active objects", or, an object with targets. We will base our approach on existing object-oriented analysis and design techniques such as OMT and UML and will add additional features such as goals, sensors, and effectors.

The primary focus of Multi Agent System Approach is to help a designer take an initial set of requirements and analyze, design, and implement a working multiagent system. This methodology is the foundation for the agent Tool development system serving as a validation platform a proof of concept. The agent Tool system is a graphically-based, fully interactive software engineering tool for the Multi Agent System methodology. Agent Tool supports the analysis and design in each of the seven Multi Agent System steps as well as automatic verification of inter-agent communications and code generation for multiple multi agent system frameworks.

The Analysis phase consists- a) Capturing Targets, b) Applying Use Cases, c) Refining Roles. The Design phase consist a) Creating Multi Agent Classes, b) Constructing Conversations, c) Assembling Multi Agent Classes, and d) System Design. The rectangular shaped models are used to capture the output, while the arrows show the effects of the models on each other. The objective is that the analyst/ designer should be allowed to move between the above said phases and phases freely such that with each successive pass, additional detail is added and, eventually, a complete and consistent system design is produced.



The chief objective of the experiments with the prototype was to evaluate the tool's efficiency for simulation of various attacks. We have thoroughly inspected and observed the prototypes possibilities for realization of few tasks. These tasks are mainly - checking a security policy at stages of design of network security system which is solved by simulation of attacks at a macro-level and research of responding a network model being designed. The other task is- checking security policy including vulnerabilities recognition of a real-life computer network, which is fulfilled by means of simulation of attacks at a micro level. This is done by generating a network traffic corresponding to real activity of offenders. These experiments were carried out for various parameters of the attack task specification and an attacked computer network configuration.

3.1 Offender's model realization module

It determines the offender's skill level, which is a mode of action and attack target. The data and knowledge repository contains data and knowledge. These are as a principle used by offender while planning and experiencing attacks. This contains a knowledge base (KB) about analyzed system, a KB of operation rules, and a database (DB) of attack tools. The knowledge base about analyzed system includes data about the architecture and particular parameters of computer network .These are needed for scripts generation and attack execution.

The offender obtains this data by using actions and methods of social engineering. The "IF-THEN" type determining AVAS operation on different levels of detail consists in meta and low level rules of the database of operation. IF-part of rule contains (meta-) action goal and (or) condition parts. The goal is chosen in accordance with a scenario type, an attack intention .The condition is compared with the data from database about analyzed system. THEN-part contains the name of attack action which can be applied and the link on exploit. The DB of attack tools consist exploits and parameters of the execution, where a choice of a parameter is determined by the data in KB about analyzed system. The module of database and knowledge repository update downloads the open vulnerability databases. This module then translates them into KB of operation rules of low level. The AVAS prototype was implemented and the experiments were held based on the case-study developed.

Conclusion

This paper described basic ideas of the agent-based modeling and simulation of network attacks. We developed the approach to be used for conducting experiments to analyze the efficiency and effectiveness of security policy against different network attacks. Software prototypes were developed. They allow imitating a wide spectrum of real life attacks. Experiments with the prototypes ware conducted. These include the investigation of attack scenarios against networks with a variety of different security policies. Further development of our modeling and simulation framework and software tools will consist of improving capabilities of the attack agents by expansion of the attack classes, implementing more sophisticated attack scenarios. These will provide comprehensive experimental assessment of offered approach. Our future theoretical work is directed on development of formal basis for agent-based modeling and simulation of counteraction between attack and defense teams in the Internet.

Acknowledgement

We are very thankful to our family members who paved way in our research with their valuable co-operation and moral support.

References

[1] D.Alessandri, C.Cachin, M.Dacier, etc., Towards a taxonomy of intrusion detection systems and attacks, MAFTIA deliverable D3, Version 1.01, Project IST-1999-11583. (2001).

[2] Custom attack simulation language (CASL), Secure Networks. (1998).

[3] S.-D.Chi, J.S.Park, K.-C.Jung, J.-S.Lee, Network security modeling and cyber attack simulation methodology, ACISP 2001, Lecture Notes in Computer Science, Vol.2119. (2001).

[4] M.Chung, B. Mukherjee, R.A.Olsson, N.Puketza, Simulating concurrent intrusions for testing intrusion detection systems: parallelizing intrusions. Proceedings of the 18th NISSC. (1995).

[5] F.B.Cohen, Information system attacks: a preliminary classification scheme, Computers and Security, Vol. 16, No. 1. (1997).

[6] F.Cohen. Simulating cyber attacks, defenses, and consequences, IEEE Symposium on Security and Privacy, Berkeley, CA. (1999).

[7] F.Cuppens and R.Ortalo, Lambda: A language to model a database for detection of attacks, Proceedings of RAID'2000. (2000).

[8] D.Curry, Intrusion detection message exchange format, extensible markup language (xml) document type definition. draft-ietf-idwg-idmef-xml-02.txt. (2000).

[9] J.Dawkins, J. Hale, A Systematic approach to multi-stage network attack analysis, Proceedings of the Second IEEE International Information Assurance Workshop (IWIA'04). (2004).

[10] R.Deraison, The nessus attack scripting language reference guide, http://www.nessus.org. (1999).

[11] R.Durst, T.Champion, B.Witten, E.Miller, L.Spanguolo, Testing and evaluating computer intrusion detection systems, Communications of ACM, 42(7). (1999).

[12] S.T.Eckmann, G.Vigna, R.A.Kemmerer, STATL: An attack language for state-based intrusion detection, Proceedings of the ACM Workshop on Intrusion Detection. (2000).

[13] R.Feiertag, C.Kahn, P.Porras, D.Schnackenberg, S.Staniford-Chen, B.Tung, A common intrusion specification language (cisl), Specification draft, http://www.gidos.org. (1999).

[14] R.P.Goldman, A Stochastic model for intrusions, Recent Advances in Intrusion Detection. Fifth International Symposium, RAID 2002, Lecture Notes in Computer Science, V.2516. (2002).

[15] V.Gorodetski, O.Karsayev, I.Kotenko, A.Khabalov, Software development kit for multi-agent systems design and implementation, Lecture Notes in Artificial Intelligence, Vol.2296. (2002).

[16] V.Gorodetski, I.Kotenko, Attacks against computer network: formal grammar-based framework and simulation tool, Recent Advances in Intrusion Detection. Fifth International Symposium. RAID 2002. Lecture Notes in Computer Science, Vol.2516. (2002).

[17] S.Hariri, G.Qu, T.Dharmagadda, M.Ramkishore, C. S.Raghavendra, Impact analysis of faults and attacks in large-scale networks, IEEE Security & Privacy, September /October. (2003).

[18] J.D.Howard, T.A.Longstaff, A common language for computer security incidents, SANDIA Report, SAND98-8667. (1998).

[19] K.Iglun, R.A.Kemmerer, P.A.Porras, State transition analysis: a rule-based intrusion detection system, IEEE Transactions on Software Engineering, 21(3). (1995).

[20] S.Jha, O.Sheyner, J.Wing, Minimization and reliability analysis of attack graphs, Technical Report CMU-CS-02-109, School of Computer Science, Carnegie Mellon University. (2002).

[21] R.A.Kemmerer, G.Vigna, NetSTAT: a network-based intrusion detection approach, Proceedings of the 14th Annual Computer Security Applications Conference, Scottsdale, Arizona. (1998).

[22] I.Kotenko, Agent-based modeling and simulation of cyber-warfare between malefactors and security agents in Internet, 19th European Simulation Multiconference. ESM'05. (2005).

[23] I.Kotenko, M.Stepashkin, Analyzing vulnerabilities and measuring security level at design and exploitation stages of computer network life cycle, MMM-ACNS-05, Lecture Notes in Computer Science, Springer Verlag, Vol.3685. (2005).

[24] I.Kotenko, A.Ulanov, Multiagent modeling and simulation of agents' competition for network resources availability, Second International Workshop on Safety and Security in Multiagent Systems (SASEMAS '05). Utrecht, The Netherlands. (2005).

[25] I.V.Krsul, Software vulnerability analysis, Ph.D. Dissertation, Computer Sciences Department, Purdue University, Lafayette, IN. (1998).

[26] S.Kumar, E.H.Spafford, A software architecture to support misuse intrusion detection. Technical Report CSD-TR-95-009. Purdue University. (1995).

I. Kotenko et al. / Agent-Based Modeling and Simulation of Malefactors' Attacks 145

[27] C.E.Landwehr, A.R.Bull, J.P.McDermott, W.S.Choi, A taxonomy of computer security flaws, ACM Computing Surveys, Vol. 26, No. 3. (1994).

[28] U.Lindqvist, E.Jonsson, How to systematically classify computer security intrusions. Proceedings of the 1997 IEEE Symposium on Security and Privacy, Los Alamitos, CA. (1997).

[29] R.Lippmann, J.W.Haines, D.J.Fried, J.Korba, K.Das. The 1999 DARPA off-line intrusion detection evaluation, RAID'2000, Lecture Notes in Computer Science, Vol.1907. (2000).

[30] P.Liu, W.Zang, Incentive-based modeling and inference of attacker intent, objectives, and strategies. ACM Transactions on Information and System Security, Vol. 8, No. 1. (2005).

[31] K.Lye, J.Wing, Game strategies in network security, International Journal of Information Security, February. (2005).

[32] J.McDermott, Attack-potential-based survivability modeling for high-consequence systems, Third IEEE International Workshop on Information Assurance, College Park, MD, USA. (2005).

[33] J.McHugh, The 1998 Lincoln Laboratory IDS evaluation: a critique, RAID'2000, Lecture Notes in Computer Science, Vol. 1907. (2000).

[34] L.Me.Gassata, A genetic algorithm as an alternative tool for security audit trails analysis, Proceedings of the first international workshop on the Recent Advances in Intrusion Detection (RAID'98). (1998).

[35] C.Michel, L.Me, ADeLe: an attack description language for knowledge-based intrusion detection, Proceedings of the 16th International Conference on Information Security. Kluwer. (2001).

[36] A.P.Moore, R.J.Ellison, R.C.Linger, Attack modeling for information security and survivability, Technical Note CMU/SEI-2001-TN-001. Survivable Systems. (2001).

[37] S.D.Moitra, S.L.Konda, A simulation model for managing survivability of networked information systems, Technical Report CMU/SEI-2000-TR-020. (2000).

[38] D.M.Nicol, W.H.Sanders, K.S.Trivedi, Model-based evaluation: from dependability to security, IEEE Transactions on Dependable and Secure Computing. Vol.1, N.1. (2004).

[39] S.Nikoletseas, G.Prasinos, P.Spirakis, C.Zaroliagis, Attack propagation in networks, Theory of Computing Systems, 36. (2003).

[40] P.Ning, D.Xu, C.G.Healey, R.A.St.Amant, Building attack scenarios through integration of complementary alert correlation methods, Proceedings of the 11th Annual Network and Distributed System Security Symposium (NDSS '04). (2004).

[41] OMNeT++ homepage. http://www.omnetpp.org

[42] V.Paxson, Bro: A system for detecting network intruders in real-time. Proceedings of the 7th Usenix Security Symposium. (1998).

[43] N.Puketza, M.Chung, R.A.Olsson, B.Mukherjee, A software platform for testing intrusion detection systems, IEEE Software, Vol.14, No 5. (1997).

[44] M.Ranum, A Taxonomy of Internet Attacks, Web Security Sourcebook. John Wiley & Sons. (1997).

[45] R.W.Ritchey, P.Ammann, Using model checking to analyze network vulnerabilities, Proceedings SOOO IEEE Computer Society Symposium on Security and Privacy. (2000).

[46] M.Roesch, Snort - lightweight intrusion detection for networks, Proceedings of the USENIX LISA'99 conference. (1999).

[47] S.Rubin, S.Jha, B.P.Miller, Automatic generation and analysis of NIDS attacks, 20th Annual Computer Security Applications Conference (ACSAC), Tuscon, Arizona. (2004).

[48] B.Schneier, Attack trees: modeling security threats, Dr. Dobb's Journal, December. (1999).

[49] B.Shepard, C.Matuszek, C.B.Fraser, etc., A Knowledge-based approach to network security: applying Cyc in the domain of network risk assessment, The Seventeenth Innovative Applications of Artificial Intelligence Conference on Artificial Intelligence (IAAI-05), Pittsburgh, Pennsylvania. (2005).

[50] O.Sheyner, J.Haines, S.Jha, R.Lippmann, J.M.Wing, Automated generation and analysis of attack graphs, Proceedings of the IEEE Symposium on Security and Privacy. (2002).

[51] S.Singh, J.Lyons, D.M.Nicol, Fast model-based penetration testing, Proceedings of the 2004 Winter Simulation Conference. (2004).

[52] A.J.Stewart, Distributed metastasis: a computer network penetration methodology, Phrack Magazine, Vol 9, Issue 55. (1999).

[53] L.Swiler, C.Phillips, D.Ellis, S.Chakerian, Computer-attack graph generation tool, Proceedings DISCEX '01: DARPA Information Survivability Conference & Exposition II. (2001).

[54] S.J.Templeton, K.Levitt, A requires/provides model for computer attacks, Proceedings of the New Security Paradigms Workshop. (2000).

[55] E.Turner, R.Zachary, Securenet pro software's snp-l scripting system, White paper.

http://www.intrusion.com, July. (2000).

[56] J.Yuill, F.Wu, J.Settle, F.Gong, R.Forno, M.Huang, J.Asbery, Intrusion-detection for incident-response, using a military battlefield-intelligence process, Computer Networks, No.34. (2000).

Artificial Neural Networks and Fuzzy Logic

Manisha Saini

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

Abstract

Artificial neural networks have emerged as fast computation tools with learning and adaptively capabilities whereas fuzzy logic has emerged as a mathematical tool to deal with the uncertainties in human perception and reasoning. They also provide a framework for an inference mechanism that allows approximate human reasoning capabilities to be applied to knowledge-based systems. Recently, these two fields have been integrated into a new emerging technology called fuzzy neural networks which combines the benefits of each field.

1. Introduction to neural networks:-

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements called **neurons** working in unison to solve specific problems. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. As, learning in biological systems involves adjustments to the synaptic connections that exist between the neurons which is true for ANNs as well.

1.1 Understanding Neural Networks:-

Artificial Neural Network, as the name implies, is a man made network formed by neurons and is an effort to replicate the animal brain.

Neural networks are a form of multiprocessor computer system, with

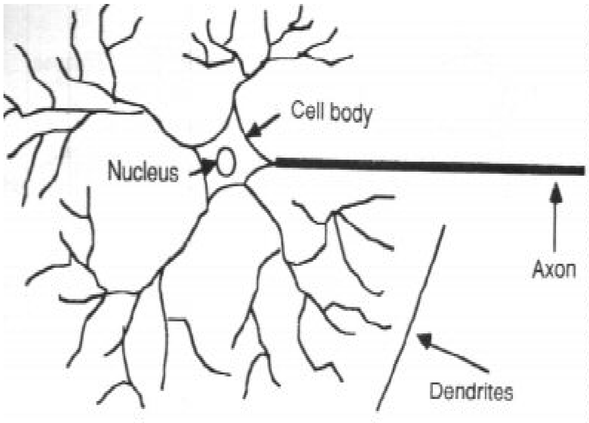
- simple processing elements
- a high degree of interconnection
- simple scalar messages
- adaptive interaction between elements

A biological neuron may have as many as 10,000 different inputs, and may send its output depending on the presence or absence of a short-duration spike to many other neurons.

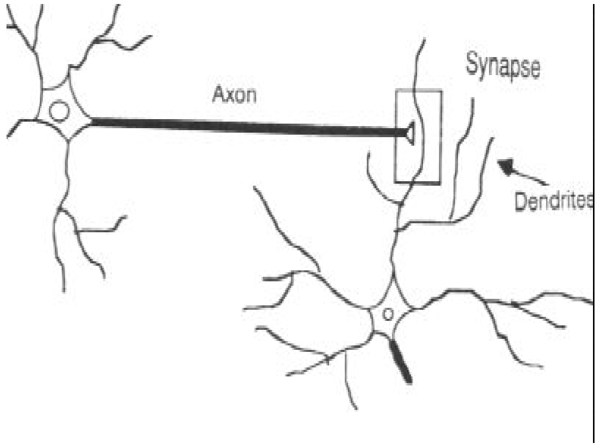
Real brains, however, are orders of magnitude more complex than any artificial neural network.

1.2 How the Human Brain Learns?

Much is still unknown about how the brain trains itself to process information, so theories abound. In the human brain, a typical neuron collects signals from others through a host of fine structures called dendrites. The neuron sends out spikes of electrical activity through a long, thin strand known as an axon, which splits into thousands of branches. At the end of each branch, a structure called a synapse converts the activity from the axon into electrical effects that inhibit or excite activity from the axon into electrical effects that inhibit or excite activity in the connected neurons. When a neuron receives excitatory input that is sufficiently large compared with its inhibitory input, it sends a spike of electrical activity down its axon. Learning occurs by changing the effectiveness of the synapses so that the influence of one neuron on another changes.



Components of a neuron



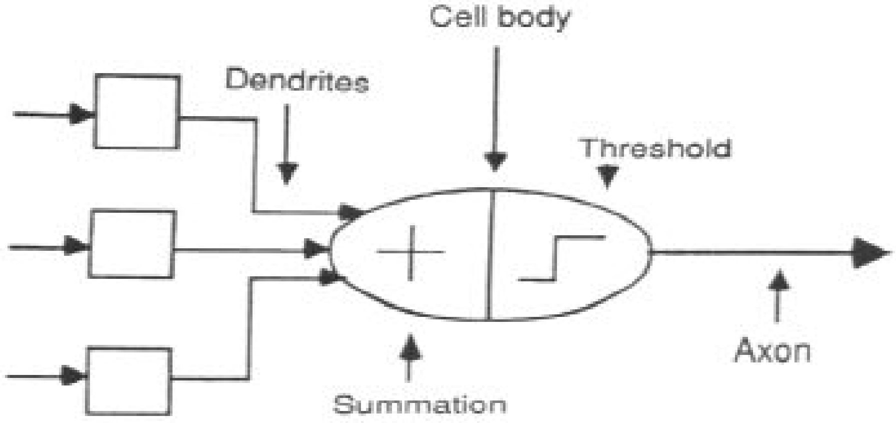
The synapse

1.3 An engineering approach

The Basic Artificial Model

To capture the essence of biological neural systems, an artificial neuron is defined as follows:

- It receives a number of inputs either from original data, or from the output of other neurons. Each input comes via a connection that has a weight; these weights correspond to synaptic efficiency in a biological neuron. Each neuron also has a single threshold value. The weighted sum of the inputs is formed, and the threshold subtracted, to compose the activation of the neuron also known as the post-synaptic potential, or PSP, of the neuron.
- The activation signal is passed through an activation function also known as a transfer function to produce the output of the neuron.

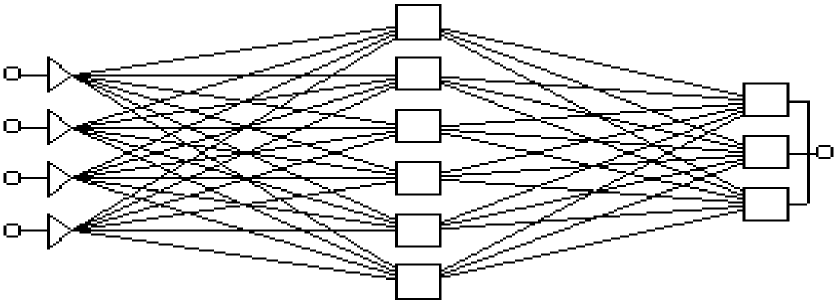


A neuron model

This describes an individual neuron. The next question is: how should neurons be connected together? If a network is to be of any use, there must be inputs which carry the values of variables of interest in the outside world and outputs which form predictions, or control signals. There can be hidden neurons that play an internal role in the network. The input, hidden and output neurons need to be connected together.

The key issue here is feedback .A simple network has a feed forward structure: signals flow from inputs, forwards through any hidden units, eventually reaching the output units. Such a structure has stable behavior.

A typical feed forward network has neurons arranged in a distinct layered topology. The input layer is not really neural at all: these units simply serve to introduce the values of the input variables. The hidden and output layer neurons are each connected to all of the units in the preceding layer.



When the network is executed, the input variable values are placed in the input units, and then the hidden and output layer units are progressively executed. Each of them calculates its activation value by taking the weighted sum of the outputs of the units in the preceding layer, and subtracting the threshold. The activation value is passed through the activation function to produce the output of the neuron. When the entire network has been executed, the outputs of the output layer act as the output of the entire network.

2. Transfer Function

The behavior of a Neural Network depends on both the weights and the input-output function (transfer function) that is specified for the units. This function typically falls into one of three categories:

- **Linear (or ramp):** The output activity is proportional to the total weighted output

- **Threshold:** The output is set at one of two levels, depending on whether the total input is greater than or less than some threshold value.

- **Sigmoid:** The output varies continuously but not linearly as the input changes. Sigmoid units bear a greater resemblance to real neurons than do linear or threshold units, but all three must be considered rough approximations

To make a neural network that performs some specific task, we must choose how the units are connected to one another and we must set the weights on the connections appropriately. The connections determine whether it is possible for one unit to influence another. The weights specify the strength of the influence.

2.1 Training a neural network:-

A common type of training in neural networks is back propagation. With back propagation, a program is set up to automatically change the weights of connections whenever the output is wrong, and strengthens connections whenever the output is correct. When the connections are changed, the program changes the values of certain connections backward from the output nodes to the hidden layer to the input nodes. After further trials, the values of different connections may be changed. This continues until the output is usually correct within a certain degree of freedom. The network never produces outputs that are correct 100% of the time. Once trained, the neural network is a probabilistic system which is correct most of the time.

3. Applications of neural networks:-

Neural networks are applicable in virtually every situation in which a relationship between the predictor variables and predicted variables exists, even when that relationship is very complex and not easy to articulate in the usual terms of "correlations" or "differences between groups."

- **Detection of medical phenomena: -** A variety of health-related indices (e.g., a combination of heart rate, levels of various substances in the blood, respiration rate) can be monitored.
- **Electronic noses:-ANNs** are used experimentally to implement electronic noses. Electronic noses have several potential applications in telemedicine.

Conclusion

The computing world has a lot to gain from neural networks. Their ability to learn by example makes them very flexible and powerful. Furthermore there is no need to devise an algorithm in order to perform a specific task; i.e. there is no need to understand the internal mechanisms of that task. They are also very well suited for real time systems because of their fast response and computational times which are due to their parallel architecture.

Finally, I would like to state that even though neural networks have a huge potential we will only get the best of them when they are integrated with computing, AI, fuzzy logic and related subjects.

4. Fuzzy logic

4.1 Introduction:-

As the complexity of a system increases, it becomes more difficult and eventually impossible to make a precise statement about its behavior, eventually arriving at a point of complexity where the fuzzy logic method born in humans is the only way to get at the problem. Fuzzy logic is a superset of Boolean logic that has been

Extended to handle the concept of partial truth -- truth values between

"Completely true" and "completely false". It was introduced by Dr. Lotfi

Zadeh of UC/Berkeley in the 1960's as a means to model the uncertainty of natural language.

4.2 Fuzzy Logic Analysis and Control:-

Human beings have the ability to take in and evaluate all sorts of information from the physical world they are in contact with and to mentally analyze, average and summarize all this input data into an optimum course of action.

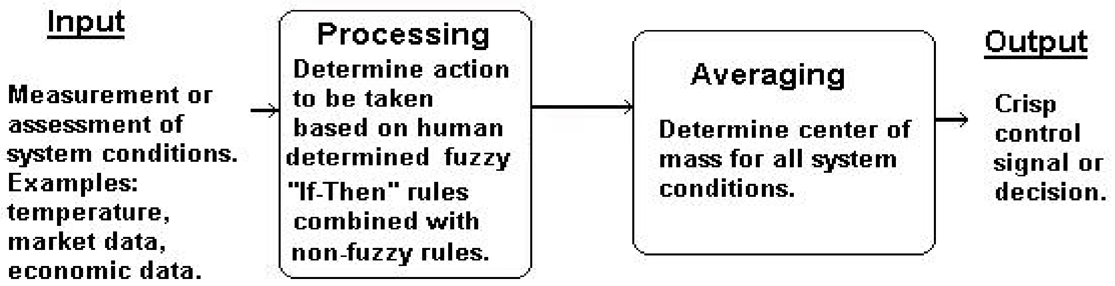
If you think about it, much of the information you take in is not very precisely defined, such as the speed of a vehicle coming up from behind. We call this fuzzy input. However, some of your "input" is reasonably precise and non-fuzzy such as the speedometer reading. Your processing of all this information is not very precisely definable. We call this fuzzy processing. Fuzzy logic theorists would call it using fuzzy algorithms is another word for procedure or program, as in a computer program).

The fuzzy logic analysis and control method is, therefore:

1. Receiving of one, or a large number, of measurement or other assessment of conditions existing in some system we wish to analyze or control.

2. Processing all these inputs according to human based, fuzzy "If-Then" rules, which can be expressed in plain language words, in combination with traditional non-fuzzy processing.

3. Averaging and weighting the resulting outputs from all the individual rules into one single output decision or signal which decides what to do or tells a controlled system what to do. The output signal eventually arrived at is a precise appearing, defuzzified, "crisp" value. Please see the following Fuzzy Logic Control/Analysis Method diagram:


Input
Measurement or assessment of system conditions. Examples: temperature, market data, economic data.
Processing
Determine action to be taken based on human determined fuzzy "If-Then" rules combined with non-fuzzy rules.
Averaging
Determine center of mass for all system conditions.
Output
Crisp control signal or decision.

The Fuzzy Logic Control-Analysis Method

4.3 An engineering approach to fuzzy logic:-

Fuzzy Sets – It is a group of anything that can’t be precisely defined and the where the condition can be given a value between 0 and 1. Example: A woman is 6 feet, 3 inches tall. In my experience, I think she is one of the tallest women I have ever met, so I rate her height at .98. This line of reasoning can go on indefinitely rating a great number of things between 0 and 1.

Universe of Discourse :- a way to say all the objects in the universe of a particular kind, usually designated by one word, that we happen to be talking about or working with in a fuzzy logic solution.

Fuzzy control:-

The purpose of control is to influence the behavior of a system bychanging an input or inputs to that system according to a rule orset of rules that model how the system operates. The system being controlled may be mechanical, electrical, chemical or any combination of these.

Conclusion - Human processing of information is not based on two-valued, off-on, either-or logic. It is based on fuzzy perceptions, fuzzy truths, fuzzy inferences, etc., all resulting in an averaged, summarized, normalized output, which is given by the human a precise number or decision value which he or she verbalizes, writes down or acts on. It is the goal of fuzzy logic control systems to also do this.The input may be large masses of data, but humans can handle it. The ability to manipulate fuzzy sets and the subsequent summarizing capability to arrive at an output we can act on is one of the greatest assets of the human brain. This characteristic is the big difference between humans and digital computers. Emulating this human ability is the challenge facing those who would create computer based artificial intelligence**.**

Speech Recognition System

Pooja Kamra

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

Abstract

The paper describes new automatic service intended for using telephone directory inquiries without human operator. The developed automatic speech recognition system for Russian language solves for processing the voice input and answering the queries of users. The electronic catalogue "Yellow Pages of Saint-Petersburg" containing information about firms and organizations is used for debugging. The development of such system for Russian language has some problems, connected with complex structure of word-formation of Russian language. The developed speech recognition system SIRIUS has one additional level of Russian language representation - morphemic level. As a result the size of vocabulary and time for speech processing are significantly decreased. The specifics of the automatic speech recognition through telephone line are discussed. The various filter techniques are applied for noise reduction and speech enhancement to improve the system performance. The dialogue model for voice control of electronic catalogue "Yellow Pages" and the experimental results are presented in the paper.

1. Introduction

Speech recognition converts spoken words to machine-readable input (for example, to key presses, using the binary code for a string of character codes). The term "voice recognition" is sometimes incorrectly used to refer to speech recognition, when actually referring to speaker recognition, which attempts to identify the person speaking, as opposed to what is being said. Confusingly, journalists and manufacturers of devices that use speech recognition for control commonly use the term Voice Recognition when they mean Speech Recognition.It is also known as **automatic speech recognition** or **computer speech recognition** system.

Speech recognition applications include voice dialing (e.g., "Call home"), call routing (e.g., "I would like to make a collect call"), domotic appliance control and content-based spoken audio search (e.g., find a podcast where particular words were spoken), simple data entry (e.g., entering a credit card number), preparation of structured documents (e.g., a radiology report), speech-to-text processing (e.g., word processors or emails), and in aircraft cockpits (usually termed Direct Voice Input).

One of the most notable domains for the commercial application of speech recognition in the United States has been health care and in particular the work of the medical transcriptionist (MT). According to industry experts, at its inception, speech recognition (SR) was sold as a way to completely eliminate transcription rather than make the transcription process more efficient, hence it was not accepted. It was also the case that SR at that time was often technically deficient. Additionally, to be used effectively, it required changes to the ways physicians worked and documented clinical encounters, which many if not all were reluctant to do. The biggest limitation to speech recognition automating transcription, however, is seen as the software. The nature of narrative dictation is highly interpretive and often requires judgment that may be provided by a real human but not yet by an automated system. Another limitation has been the extensive amount of time required by the user and/or system provider to train the software.

A distinction in ASR is often made between "artificial syntax systems" which are usually domain-specific and "natural language processing" which is usually language-specific. Each of these types of application presents its own particular goals and challenges.

The task behind speech recognition system is getting a computer to understand spoken language.and the understand here means react appropriately and convert the input speech into other medium like text

Articulation is a process which produces sound waves which the ear conveys to brain for processing.

2. Techniques of speech recognition

It involves three main steps:

- Digitization
- Acoustic analysis of the speech signal
- Linguistic interpretation

Digitization

Converting analogue signal into digital representation

- Signal processing
  - Separating speech from background noise
- Phonetics
  - Variability in human speech
- Phonology
  - Recognizing individual sound distinctions (similar phonemes)
- Lexicology and syntax
  - Disambiguating homophones
  - Features of continuous speech
- Syntax and pragmatics
  - Interpreting prosodic features
- Pragmatics
  - Filtering of performance errors (disfluencies)

Digitization involves

- Analogue to digital conversion
- Sampling and quantizing
- Use filters to measure energy levels for various points on the frequency spectrum
- Knowing the relative importance of different frequency bands (for speech) makes this process more efficient
- E.g. high frequency sounds are less informative, so can be sampled using a broader bandwidth (log scale)

Separating speech from background noise which involves no. of steps:

- Noise cancelling microphones
  - Two mics, one facing speaker, the other facing away
  - Ambient noise is roughly same for both mics
- Knowing which bits of the signal relate to speech
  - Spectrograph analysis

Variability in individuals' speech that involves

- Variation among speakers due to
  - Vocal range (f0, and pitch range – see later)
  - Voice quality (growl, whisper, physiological elements such as nasality, adenoidality, etc)
  - ACCENT !!! (especially vowel systems, but also consonants, allophones, etc.)
- Variation within speakers due to
  - Health, emotional state
  - Ambient conditions
- Speech style: formal read vs spontaneous

Speaker independent system that involves

- Speaker-dependent systems
  - Require "training" to "teach" the system your individual idiosyncracies
    - The more the merrier, but typically nowadays 5 or 10 minutes is enough
    - User asked to pronounce some key words which allow computer to infer details of the user's accent and voice
    - Fortunately, languages are generally systematic
  - More robust
  - But less convenient
  - And obviously less portable
- Speaker-independent systems

  - Language coverage is reduced to compensate need to be flexible in phoneme identification
  - Clever compromise is to learn on the fly

Identifying phonemes that involves

- Differences between some phonemes are sometimes very small
  - May be reflected in speech signal (eg vowels have more or less distinctive f1 and f2)
  - Often show up in coarticulation effects (transition to next sound)
    - e.g. aspiration of voiceless stops in English
  - Allophonic variation

Disambiguating homophones that involves

- Mostly differences are recognised by humans by context and need to make sense

(Dis)continuous speech that involves

- Discontinuous speech much easier to recognize
  - Single words tend to be pronounced more clearly
- Continuous speech involves contextual coarticulation effects
  - Weak forms
  - Assimilation
  - Contractions

Interpreting prosodic features that involves

- Pitch, length and loudness are used to indicate “stress”
- All of these are relative
  - On a speaker-by-speaker basis
  - And in relation to context
- Pitch and length are phonemic in some languages

Pitch that involves

- Pitch contour can be extracted from speech signal
  - But pitch differences are relative
  - One man’s high is another (wo)man’s low
  - Pitch range is variable
- Pitch contributes to intonation
  - But has other functions in tone languages
- Intonation can convey meaning

Length that involves

- Length is easy to measure but difficult to interpret
- Again, length is relative
- It is phonemic in many languages
- Speech rate is not constant – slows down at the end of a sentence

Performance errors that involves

- Performance "errors" include
  - Non-speech sounds
  - Hesitations
  - False starts, repetitions
- Filtering implies handling at syntactic level or above
- Some disfluencies are deliberate and have pragmatic effect – this is not something we can handle in the near future

Approaches to Speech Recognition System

- Template matching
- Knowledge-based (or rule-based) approach
- Statistical approach:
  - Noisy channel model + machine learning

Template-based approach is

- Store examples of units (words, phonemes), then find the example that most closely fits the input
- Extract features from speech signal, then it's "just" a complex similarity matching problem, using solutions developed for all sorts of applications
- OK for discrete utterances, and a single user.
- Hard to distinguish very similar templates And quickly degrades when input differs from templates

Therefore needs techniques to mitigate this degradation:

- More subtle matching techniques
- Multiple templates which are aggregated

Rule-based approach is

- Use knowledge of phonetics and linguistics to guide search process
- Templates are replaced by rules expressing everything (anything) that might help to decode:
  - Phonetics, phonology, phonotactics
  - Syntax
  - Pragmatics

  - Typical approach is based on “blackboard” architecture:
  - At each decision point, lay out the possibilities.Apply rules to determine which sequences are permitted
  - Poor performance due to Difficulty to express rules
  - Difficulty to make rules interact
  - Difficulty to know how to improve the system

Statistics-based approach is

- Can be seen as extension of template-based approach, using more powerful mathematical and statistical tools
- Sometimes seen as “anti-linguistic” approach
  - Fred Jelinek (IBM, 1988): “Every time I fire a linguist my system improves”
  - Collect a large corpus of transcribed speech recordings
  - Train the computer to learn the correspondences (“machine learning”)
  - At run time, apply statistical processes to search through the space of all possible solutions, and pick the statistically most likely one

Machine Learning

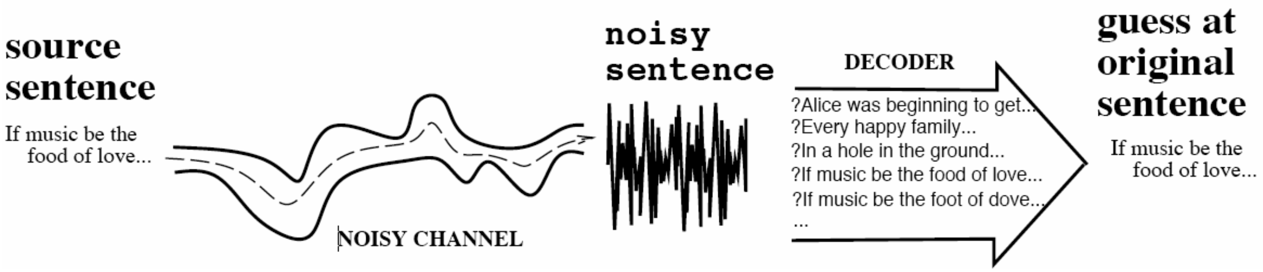
- Acoustic and Lexical Models
  - Analyse training data in terms of relevant features
  - Learn from large amount of data different possibilities
    - different phone sequences for a given word
    - different combinations of elements of the speech signal for a given phone/phoneme
  - Combine these into a Hidden Markov Model expressing the probabilities

Language model is

- Models likelihood of word given previous word(s)
- n-gram models:
  - Build the model by calculating bigram or trigram probabilities from text training corpus
  - Smoothing issues

The Noisy Channel Model

- Search through space of all possible sentences
- Pick the one that is most probable given the waveform


source sentence
If music be the food of love...
NOISY CHANNEL
noisy sentence
DECODER
?Alice was beginning to get...
?Every happy family...
?In a hole in the ground...
?If music be the food of love...
?If music be the foot of dove...
...
guess at original sentence
If music be the food of love...

- Use the acoustic model to give a set of likely phone sequences
- Use the lexical and language models to judge which of these are likely to result in probable word sequences
- The trick is having sophisticated algorithms to juggle the statistics
- A bit like the rule-based approach except that it is all learned automatically from data

Conclusion:

This was a high level overview of how speech recognition works. It’s not nearly enough detail to actually write a speech recognizer, but it exposes the basic concepts. Most speech recognition engines work in a similar manner, although not all of them work this way. If you want more detail you should purchase one of the numerous technical books on speech recognition.

The speech recognizer can now identify what phonemes were spoken. Figuring out what words were spoken should be an easy task. If the user spoke the phonemes, "h eh l oe", then you know they spoke "hello". The recognizer should only have to do a comparison of all the phonemes against a lexicon of pronunciations.

It’s not that simple.

1. The user might have pronounced "hello" as "h uh l oe", which might not be in the lexicon.
2. The recognizer may have made a mistake and recognized "hello" as "h uh l oe".
3. Where does one word end and another begin?
4. Even with all these optimizations, the speech recognition still requires too much CPU.

Speech recognition fundamentally functions as a pipeline that converts PCM (Pulse Code Modulation) digital audio from a sound card into recognized speech. The elements of the pipeline are:

1. Transform the PCM digital audio into a better acoustic representation

2. Apply a "grammar" so the speech recognizer knows what phonemes to expect. A grammar could be anything from a context-free grammar to full-blown English.
3. Figure out which phonemes are spoken.
4. Convert the phonemes into words

References:

[1] Allen, J., Hunnicutt, M. S., & Klatt, D. H. (1987). From Text to Speech: The MITalk System. Cambridge University Press, New York.

[2] Bailly, G. & Benoit, C. (Eds.). (1992). Talking Machines: Theories, Models, and Designs. North Holland, Elvsevier, Amsterdam.

[3] Bell, A. G. (1922). Prehistoric Telephone Days. National Geographic Magazine, 41, 223-242.

[4] Cahn, J. E. (1990). The generation of affect in synthesized speech. Journal of the American Voice Input/Output Society, 8, 1-19.

[5] Carlson, R. & Granström, B. (1976). A text-to-speech system based entirely on rules. Proceedings of the International Conference of Acoustics, Speech, & Signal Processing, ICASSP-76, 686-688.

[6] Cater, John P. (1983). Electronically Speaking: Computer Speech Generation. Howard W. Sams & Co., Inc., Indianapolis, Indiana.

[7] Cooper, F.S., Liberman, A.M., & Borst, J.M. (1951). The interconversion of audible and visible patterns as a basis for research in the perception of speech. Proceedings of the National Academy of Science, 37, 318-325.

Data Mining

Neha Singh

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

Abstract

Data Mining: helps users end users extract useful business information from “large” databases. What makes this definition interesting in the word “large”? If the database were small, you wouldn’t need any new technology to discover useful information. If, for instance this were the early 1800’s and you were the owner of the general store you wouldn’t need to employ data mining since you would probably have only a few hundred customers, each of whom you new by name. You would probably also know, in great detail who they were and what they bought. When you have got only a few hundred customers you really don’t need much of a computer to mine the data. The best database analysis and predictive models that could possibly be done could be done in the shopkeepers head.

Today, however the shopkeeper of 1990s has hundreds to thousand to millions of customers, and for the first time in history the data on these customers is being accumulated at one location where consistent access and consistent storage in being guaranteed; the data warehouse. The metaphors between the data warehousing and data mining can be confusing. The idea that ties them together is that the large data collection in your data warehouse is the data mountain presented to data mining tools. Data warehousing allows you to build that mountain. Data mining allows you to shift that mountain down to the essential information that is useful to your business.

What is so new about extracting information from data to make our business run better. The allure of data mining is that it promises to fix the problem of miscommunication between you and your data and allow you to ask of your data complex questions.

1. The motivation for data mining is tremendous

John, typical cell phone users has just decided to not renew his contract with you, his current cellular provider. Why? Because he was just made an offer by the competing provider for a free phone. Since that Motorola phone that he got from you was really staring to look like old technology and the competition has the same rate plan as you, John opted for there offer. This is good news for John and good news for john’s new cellular provider. But really bad news for john’s old provider (you). You invested in john to the tune of 700$ to land him as the customer less than one year ago. The sad thing is that if you had only known that john was at risk of leaving you surely would have invested the extra 100$ to upgrade john’s phone (John does 350$ worth of calling per month). Now that John has signed the contract with the competition, it is too late. The really aggravating thing is that you have known that john was at risk. If you had used data mining on your customer account database, you could have built a predictive model that would have shown that john, and others like him, are at given risk of attrition. With this predictive model you could have launched a successful and profitable direct marketing campaign to save your valuable customers

2. Continuous Innovation

Although data mining is a relatively new term, the technology is not. Companies have used powerful computers to sift through volumes of supermarket scanner data and analyze market research reports for years. However, continuous innovations in computer processing power, disk storage, and statistical software are dramatically increasing the accuracy of analysis while driving down the cost.

Example

For example, one Midwest grocery chain used the data mining capacity of Oracle software to analyze local buying patterns. They discovered that when men bought diapers on Thursdays and Saturdays, they also tended to buy beer. Further analysis showed that these shoppers typically did their weekly grocery shopping on Saturdays. On Thursdays, however, they only bought a few items.

The retailer concluded that they purchased the beer to have it available for the upcoming weekend. The grocery chain could use this newly discovered information in various ways to increase revenue. For example, they could move the beer display closer to the diaper display. And, they could make sure beer and diapers were sold at full price on Thursdays.

3. Data, Information, and Knowledge

3.1 Data

Data are any facts, numbers, or text that can be processed by a computer. Today, organizations are accumulating vast and growing amounts of data in different formats and different databases. This includes:

- operational or transactional data such as, sales, cost, inventory, payroll, and accounting

- non-operational data, such as industry sales, forecast data, and macro economic data

- meta data - data about the data itself, such as logical database design or data dictionary definitions

3.2 Information

The patterns, associations, or relationships among all this data can provide information. For example, analysis of retail point of sale transaction data can yield information on which products are selling and when.

3.3 Knowledge

Information can be converted into knowledge about historical patterns and future trends. For example, summary information on retail supermarket sales can be analyzed in light of promotional efforts to provide knowledge of consumer buying behavior. Thus, a manufacturer or retailer could determine which items are most susceptible to promotional efforts.

3.4 Data Warehouses

- A decision support database that is maintained separately from the organization's operational databases.
- A data warehouse is a
  - subject-oriented,
  - integrated,
  - time-varying,
  - non-volatile
- collection of data that is used primarily in organizational decision making

Dramatic advances in data capture, processing power, data transmission, and storage capabilities are enabling organizations to integrate their various databases into data warehouses. Data warehousing is defined as a process of centralized data management and retrieval. Data warehousing, like data mining, is a relatively new term although the concept itself has been around for years. Data warehousing represents an ideal vision of maintaining a central repository of all organizational data. Centralization of data is needed to maximize user access and analysis. Dramatic technological advances are making this vision a reality for many companies. And, equally dramatic advances in data analysis software are allowing users to access this data freely. The data analysis software is what supports data mining.

4. Data Mining

- Data Mining (DM) seeks to discover new information or "knowledge" from (very) large databases.
- Knowledge is represented in the form of statistical rules and patterns.
- It differs from machine learning in that it deals with large amounts of data stored primarily on disk (rather than in main memory).
- Knowledge discovered from a database can be represented by a set of rules. Such rules can be discovered using one of two methods:

- User is involved directly in the process of knowledge discovery
- The DM system is responsible for automatically discovering knowledge from the database, by detecting patterns and correlations in the data.

Example

- Credit ratings/targeted marketing:
    - Given a database of 100,000 names, which persons are the least likely to default on their credit cards?
    - Identify likely responders to sales promotions
- Fraud detection
    - Which types of transactions are likely to be fraudulent, given the demographics and transactional history of a particular customer?
- Customer relationship management:
    - Which of my customers are likely to be the most loyal, and which are most likely to leave for a competitor?

4.1 Data Mining helps extract such information

- Process of semi-automatically analyzing large databases to find patterns that are:
    - **valid**: hold on new data with some certainity
    - **novel**: non-obvious to the system
    - **useful**: should be possible to act on the item
    - **understandable**: humans should be able to interpret the pattern
- Also known as Knowledge Discovery in Databases (KDD)

Knowledge Representation using Rules

• General form of rules: antecedent = consequent

• Example: Market Basket Analysis

Market Basket ≡ collection of items purchased by customer in a single customer transaction (single visit to a store or through mail-order catalog)

Idea: Use DM to identify sets of items that are purchased together _ information can be used to improve layout of goods in a store or catalog.

Association Rules

Transid	Custid	Date	Item	Price	Qty
111	201	3/12/1991	pen	35	2
111	201	3/12/1991	ink	2	1
111	201	3/12/1991	Diary	5	3
111	201	3/12/1991	Soap	1	6
112	105	4/2/1991	pen	35	1
112	105	4/2/1991	ink	2	1
112	105	4/2/1991	Diary	5	1
113	106	4/2/1991	pen	35	2
113	106	4/3/1991	Diary	5	1
114	201	5/4/1991	pen	35	2
114	201	5/4/1991	ink	2	2
114	201	5/4/1991	Soap	1	4

Examining set of transactions yields the rule

{pen} = {ink},

i.e., if a pen is purchased in a transaction, it is likely that ink will also be purchased in the transaction.

Such a rule is called an association rule.

General form:

LHS = RHS, where both RHS and LHS are sets of items

Important measures for an association rule:

– Support for a set of items is the percentage of transactions that contain all items in LHS RHS (75% for the above rule).

If support is low (as, e.g., for {diary} = {soap} [25%]), then there is not enough evidence to draw conclusion about correlation between items in LHS and items in RHS.

– Confidence: Consider transactions that contain all items in LHS. The confidence is the percentage of transactions that also contain all items of the RHS (75% for the above rule).

Data Mining represents a process developed to examine large amounts of data routinely collected. The term also refers to a collection of tools used to perform the process. Data mining is used in most areas where data are collected-marketing, health, communications, etc.

For example, retail stores routinely use data mining tools to learn about purchasing habits of its customers.

Amazon.com uses data mining to provide customers with purchase suggestions:

Customers who bought this book also bought:

Seven Methods for Transforming Corporate Data Into Business

Intelligence by Vasant Dhar, Roger Stein

Building Data Mining Applications for CRM by Alex Berson, et al

Data Preparation for Data Mining by Dorian Pyle Kellogg on

Integrated Marketing by Dawn Iacobucci (Editor), et al Multivariate

Data Analysis (5th Edition) by Joseph F. Hair (Editor), et al

The use of association rules has increased sales by 15%. SAS, Inc. developed the process for Amazon.com. Banks and the Federal Reserve use data mining to investigate the flow of money. Federal agencies use data mining to monitor cell phone communications via satellite. Compaq uses data mining to examine calls made to customer service to find patterns of complaints.

5. What can data mining do?

Data mining is primarily used today by companies with a strong consumer focus - retail, financial, communication, and marketing organizations. It enables these companies to determine relationships among "internal" factors such as price, product positioning, or staff skills, and "external" factors such as economic indicators, competition, and customer demographics. And, it enables them to determine the impact on sales, customer satisfaction, and corporate profits. Finally, it enables them to "drill down" into summary information to view detail transactional data.

With data mining, a retailer could use point-of-sale records of customer purchases to send targeted promotions based on an individual's purchase history. By mining demographic data from comment or warranty cards, the retailer could develop products and promotions to appeal to specific customer segments.

For example, Blockbuster Entertainment mines its video rental history database to recommend rentals to individual customers. American Express can suggest products to its cardholders based on analysis of their monthly expenditures.

WalMart is pioneering massive data mining to transform its supplier relationships. WalMart captures point-of-sale transactions from over 2,900 stores in 6 countries and continuously transmits this data to its massive 7.5 terabyte Teradata data warehouse. WalMart allows more than 3,500 suppliers, to access data on their products and perform data analyses. These suppliers use this data to identify customer buying patterns at the store display level. They use this information to manage local store inventory and identify new merchandising opportunities. In 1995, WalMart computers processed over 1 million complex data queries.

The National Basketball Association (NBA) is exploring a data mining application that can be used in conjunction with image recordings of basketball games. The Advanced Scout software analyzes the movements of players to help coaches orchestrate plays and strategies. For example, an analysis of the play-by-play sheet of the game played between the New York Knicks and the Cleveland Cavaliers on January 6, 1995 reveals that when Mark Price played the Guard position, John Williams attempted four jump shots and made each one! Advanced Scout not only finds this pattern, but explains that it is interesting because it differs considerably from the average shooting percentage of 49.30% for the Cavaliers during that game.

By using the NBA universal clock, a coach can automatically bring up the video clips showing each of the jump shots attempted by Williams with Price on the floor, without needing to comb through hours of video footage. Those clips show a very successful pick-and-roll play in which Price draws the Knick's defense and then finds Williams for an open jump shot.

6. How Data Mining Works

While large-scale information technology has been evolving separate transaction and analytical systems, data mining provides the link between the two. Data mining software analyzes relationships and patterns in stored transaction data based on open-ended user queries. Several types of analytical software are available: statistical, machine learning, and neural networks. Generally, any of four types of relationships are sought:

- **Classes**: Stored data is used to locate data in predetermined groups. For example, a restaurant chain could mine customer purchase data to determine when customers visit and what they typically order. This information could be used to increase traffic by having daily specials.

- **Clusters**: Data items are grouped according to logical relationships or consumer preferences. For example, data can be mined to identify market segments or consumer affinities.

- **Associations**: Data can be mined to identify associations. The beer-diaper example is an example of associative mining.

- **Sequential patterns**: Data is mined to anticipate behavior patterns and trends. For example, an outdoor equipment retailer could predict the likelihood of a backpack being purchased based on a consumer's purchase of sleeping bags and hiking shoes.

Data mining consists of five major elements:

- Extract, transform, and load transaction data onto the data warehouse system.

- Store and manage the data in a multidimensional database system.

- Provide data access to business analysts and information technology professionals.

- Analyze the data by application software.

- Present the data in a useful format, such as a graph or table.

Different levels of analysis are available:

- **Artificial neural networks**: Non-linear predictive models that learn through training and resemble biological neural networks in structure.

- **Genetic algorithms**: Optimization techniques that use processes such as genetic combination, mutation, and natural selection in a design based on the concepts of natural evolution.

- **Decision trees**: Tree-shaped structures that represent sets of decisions. These decisions generate rules for the classification of a dataset. Specific decision tree methods include Classification and Regression Trees (CART) and Chi Square Automatic Interaction Detection (CHAID). CART and CHAID are decision tree techniques used for classification of a dataset. They provide a set of rules that you can apply to a new (unclassified) dataset to predict which records will have a given outcome. CART segments a dataset by creating 2-way splits while CHAID segments using chi square tests to create multi-way splits. CART typically requires less data preparation than CHAID.

- **Nearest neighbor method**: A technique that classifies each record in a dataset based on a combination of the classes of the k record(s) most similar to it in a historical dataset (where k 1). Sometimes called the k-nearest neighbor technique.

- **Rule induction**: The extraction of useful if-then rules from data based on statistical significance.

- **Data visualization**: The visual interpretation of complex relationships in multidimensional data. Graphics tools are used to illustrate data relationships.

7. What technological infrastructure is required?

Today, data mining applications are available on all size systems for mainframe, client/server, and PC platforms. System prices range from several thousand dollars for the smallest applications up to $1 million a terabyte for the largest. Enterprise-wide applications generally range in size from 10 gigabytes to over 11 terabytes. NCR has the capacity to deliver applications exceeding 100 terabytes. There are two critical technological drivers:

- **Size of the database**: the more data being processed and maintained, the more powerful the system required.

- **Query complexity**: the more complex the queries and the greater the number of queries being processed, the more powerful the system required.

Relational database storage and management technology is adequate for many data mining applications less than 50 gigabytes. However, this infrastructure needs to be significantly enhanced to support larger applications. Some vendors have added extensive indexing capabilities to improve query performance. Others use new hardware architectures such as Massively Parallel Processors (MPP) to achieve order-of-magnitude improvements in query time. For example, MPP systems from NCR link hundreds of high-speed Pentium processors to achieve performance levels exceeding those of the largest supercomputers.

7.1 Data Mining? Don’t Need It – I’ve Got Statistics

People have been using statistics for better targeting of their marketing efforts for many years now.

Statistics has the same general uses and results as data mining. Regression is used in statistics quite often to create models that are predictive of customer behavior, and these models are built from large stores of historical data. The main difference between data mining and statistics is that data mining is meant to be used by the business end user – not the statistician. Data mining effectively automates the statistical process, thereby relieving the end user of some of the burden. This results in a tool that is easier to use. For instance, it may have occurred to you to ask "If most of the statistics is a matter of making a guess and then checking it out, why don't we just we let the computers make those guesses and then test them automatically".

For this reason data mining tool are often coupled with other tools that make it easier to apply data analysis techniques and understand the results

8. Applications

Banking: loan/credit card approval

---------predict good customers based on old customers

Customer relationship management:

---------identify those who are likely to leave for a competitor.

Targeted marketing:

---------identify likely responders to promotions

Fraud detection: telecommunications, financial transactions

---------from an online stream of event identify fraudulent events

Manufacturing and production:

---------automatically adjust knobs when process parameter changes

Medicine: disease outcome, effectiveness of treatments

--------analyze patient disease history: find relationship between diseases

Molecular/Pharmaceutical:

--------identify new drugs

Scientific data analysis:

--------identify new galaxies by searching for sub clusters

Web site/store design and promotion:

--------find affinity of visitor to pages and modify layout

References

[1] Alex Berson, "Data warehousing, Data Mining and OLAP"

[2] M.Berry and G. Linoff, "Data Mining Techniques", john Wiley 1997.

[3] Usama Fayyad, Gregory Piatetsky-Shapiro & Padhraic Smyth, "Knowledge discovery and data mining: Towards a unifying framework", Proceedings of the International Conference on Knowledge Discovery and Data Mining, pages 82-88, 1996.

[4] Usama Fayyad, Gregory Piatetsky-Shapiro & Padhraic Smyth, "The KDD process for extracting useful knowledge from volumes of data". Journal of the ACM, 39(11): 27-34, 1996.

[5] Usama Fayyad, Gregory Piatetsky-Shapiro & Padhraic Smyth, "Advances in knowledge Discovery and Data Mining", Pages 1-34, AAAI/MIT Press, 1996.

[6] Margraret H. Dunham, "Data Mining: Introductory and Advanced Topics", Pages 8-10, Pearson Education Ltd, 2003.

Proceedings of the National Seminar on Artificial Intelligence - NSAI-09
July 17, 2009

Dronacharya College of Engineering, Khentawas, Farrukh Nagar, Gurgaon-123506

Data Mining and its Applications

Payal Singh

Lecturer, CSE & IT Deptt, DCE, Gurgaon

Abstract: Applying data analysis and discovery algorithms to perform automatic extraction of information from voluminous data is called Data Mining. This process bridges many technical areas, including databases, human-computer interaction, statistical analysis, and machine learning. In this article a brief introduction of Data Mining and its application in various areas is given.

Keywords: Data mining, data warehouse, feature extraction, machine learning

1. Introduction

Today we are living in so called information age. Due to sophisticated technologies like computers, satellites etc. we have been collecting tremendous amount of information. Databases today can range into size of terabytes. Within these voluminous data information of interest may be hidden. Unfortunately, these massive collections of data stored on disparate structures became overwhelming. This initial chaos has led to the creation of structured databases and database management systems (DBMS). Today, we have far more information than it can be handled. Information retrieval is simply not enough anymore for decision-making. Confronted with huge collections of data, we have now created new needs to help us make better managerial choices. These needs are automatic summarization of data, extraction of the "essence" of information stored, and the discovery of patterns in raw data.

2. Data Mining Process

With the enormous amount of data stored in files, databases, and other repositories, it is increasingly important, if not necessary, to develop powerful means in the forms of algorithms for analysis and interpretation of such data and for the extraction of interesting knowledge that could help in decision-making. This whole process is termed as machine learning.

Machine learning can be defined as programming computers to optimize a performance criterion (prediction or decision to act) using example data or past experiences.

Data mining is application of machine learning methods to large databases for inferencing.

The terms knowledge discovery in databases (KDD) and data mining are often used interchangeably. In fact, there have been many other names given to this process of discovering useful (hidden) patterns in data: knowledge extraction, information discovery, exploratory data analysis, information harvesting, and unsupervised pattern recognition [MHD]. Over the last few years KDD has been used to refer to a process consisting of many steps, while data mining is only one of these steps. Following definitions are modified from those found in [FSP1, FSP2].

i. Knowledge discovery in databases (KDD) is the process of finding useful information and pattern in data
ii. Data mining is the use of algorithms to extract the information and patterns derived by the KDD process.

KDD is a process that involves many different steps. The input to this process is the data, and the output is the useful information desired by the users. However, the objective may be unclear or inexact. The process itself is interactive and may require much elapsed time. To ensure the usefulness and accuracy of the results of the process, interaction throughout the process with both domain experts and technical experts might be needed.

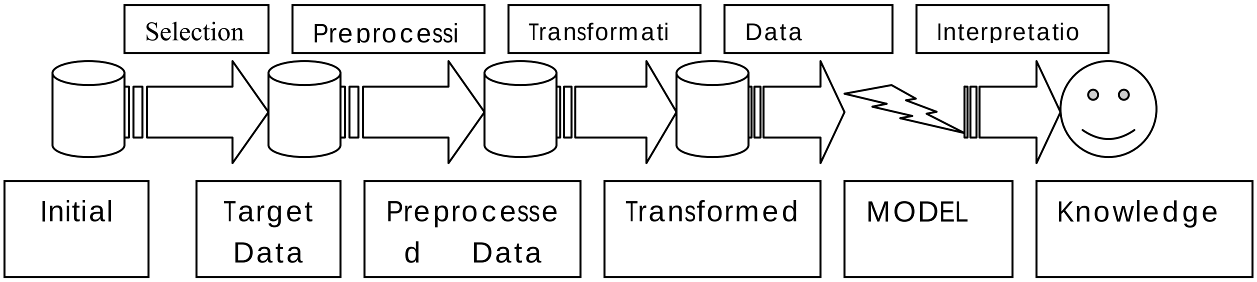


Figure 1.1: KDD Process (modified from [FSP2])

The KDD process consists of the following five steps [FSP3]:

Selection: The data needed for the data mining process may be obtained from many different and heterogeneous data sources such as spreadsheets, document files and other data warehouses.

Data Warehouse is a storehouse; a repository of data collected from multiple data sources and is intended to be used as a whole under the same unified schema.

Preprocessing: The data to be used by the process may have incorrect or missing data. There may be anomalous data from multiple sources involving different data types and metrics. Erroneous data may be corrected or removed, whereas missing data can be supplied or predicted (often using data mining tools).

Transformation: Data from different sources must be converted into a common format for processing. Some data may be encoded or transformed into more usable formats. Data reduction may be used to reduce the number of possible data values being considered

Data mining: Based on the machine learning tasks being performed, this step applies algorithms to the transformed data to generate the desired results.

Interpretation/Evaluation: How the data mining results are presented to the users is extremely important because the usefulness of the results is dependent on it. Various visualization and GUI strategies are used at this last step.

3. Basic Data Mining Model

Data mining involves many different algorithms to accomplish different tasks. All these algorithms attempt to fit a model to the data. The algorithms examine the data and determine a model that is closest to the characteristics of the data being examined. Data mining algorithms can be characterized as consisting of three parts.

Model: The purpose of the algorithm is to fit a model to the data.

Preference: Some criteria must be used to fit one model over another.

Search: All algorithms require some technique to search the data.

As seen in Figure 1.2, the model that is created can be either predictive or descriptive in nature. This figure depicts, under each model type, some of the most common data mining tasks which use that type of model.

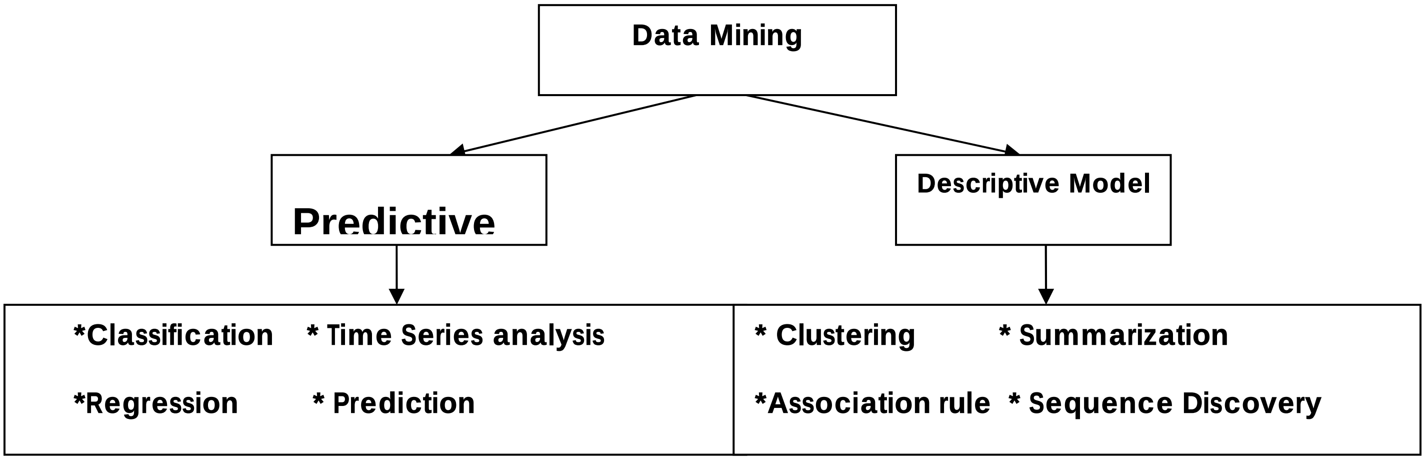


FIGURE 1.2: Data mining models and tasks [BL]

A predictive model makes a prediction about values of data using known results found from different data. Predictive modeling may be made based on the use of other historical data. For example, a credit card use might be refused not because of the user's own credit history, but because the current purchase is similar to earlier purchases that were subsequently found to be made with stolen cards. Predictive model data mining tasks include classification, regression, time series analysis, and prediction.

A descriptive model identifies patterns or relationships in data. Unlike the predictive model, a descriptive model serves as a way to explore the properties of the data examined, not to predict new properties. Clustering, summarization, association rules, and sequence discovery are some of the tasks for these models.

4. Data Mining Tasks

In this part we will briefly explore some of the data mining functions. We follow the basic outline of tasks shown in Figure 1.2. This list is not intended to be exhaustive, but rather illustrative. Of course, these individual tasks may be combined to obtain more sophisticated data mining applications

4.1 Classification

Classification maps data into predefined groups or classes. It is often referred to as supervised learning because the classes are determined before examining the data. Classification algorithms require that the classes be defined based on data attribute values. They often describe these classes by looking at the characteristics of data already known to belong to the classes. Pattern recognition is a type of classification where an input pattern is classified into one of several classes based on its similarity to these predefined classes. A simple example of pattern recognition is given in Example 4.1

Classification Problem can be defined mathematically as

Given a database D= $\{t_1, t_2 \ldots t_n\}$

And a set of classes C= $\{c_1, c_2 \ldots c_m\}$,

The Classification Problem is to define a mapping **f: D C,** such that each t_i is assigned to one class c_j; where i=1, 2…n, and j=1, 2…m.

Example 4.1 An airport security screening station is used to determine if passengers are potential terrorists or criminals. To do this, the face of each passenger is scanned and its basic pattern (distance between eyes, size and shape of mouth, shape of head, etc.) is identified. This pattern is compared to entries in a database to see if it matches any patterns that are associated with known offenders.

4.2 Regression

Regression is used to map a data item to a real valued prediction variable. In actuality, regression involves the learning of the function that does this mapping.

Regression assumes that the target data fit into some known type of function (e.g., linear, logistic, etc.) and then determines the best function of this type that models the given data. Some type of error analysis is used to determine which function is "best" Standard linear regression, as illustrated in Example 4.2, is a simple example of regression.

Example 4.2: A college professor wishes to reach a certain level of savings before her retirement. Periodically, she predicts what her retirement savings will be, based on its current value and several past values. She uses a simple linear regression formula to predict this value by fitting past behavior to a linear function and then using this function to predict the values at points in the future. Based on these values, she then alters her investment portfolio.

4.3 Time Series Analysis

With time series analysis, the value of an attribute is examined as it varies over time. The values usually are obtained as evenly spaced time points (daily, weekly, hourly, etc.). A time series plot (Figure 1.3) is used to visualize the time series of various events. In this figure you can easily see that the plots for Y and Z have variation in their behavior but time series for X shows less variation. There are three basic functions performed in time series analysis. In first case, distance measures are used to determine the similarity between different time series. In the second case, the structure of the line is examined to determine (and perhaps classify) its behavior. In third case it uses the historical time series plot to predict future values. A time series example is given in Example 4.3

Example 4.3: Mr. Smith is trying to determine whether to purchase stock from Companies X, Y, or Z. For a period of one month he charts the daily stock price for each company. Figure 1.3 shows the time series plot that Mr. Smith has generated.

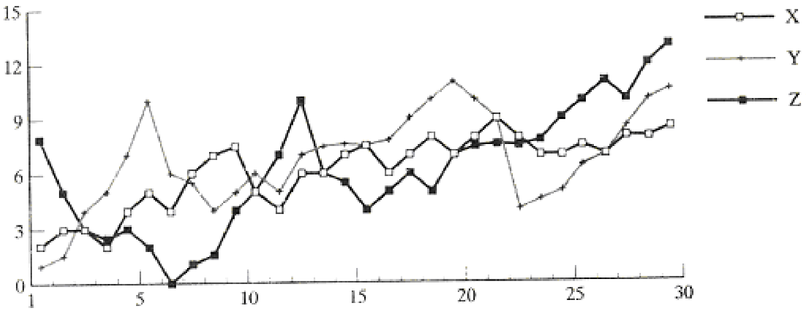


FIGURE 1.3 Time Series plots [BL]

Using this and similar information available from his stockbroker, Mr. Smith decides to purchase stock X because it is less volatile while overall showing a slightly larger relative amount of growth than either of the other stocks. As a matter of fact, the stocks for Y and Z have a similar behavior.

4.4 Prediction

Many real-world data mining applications can be seen as predicting future data states based on past and current data. Prediction can be viewed as a type of classification. The difference is that prediction is predicting a future state rather than a current state. Prediction applications include flooding, speech recognition, machine learning, and pattern recognition. Although future values may be predicted using time series analysis or regression techniques, other approaches may be used as well. Example 4.4 illustrates the process

Example 4.4: Predicting flooding is a difficult problem. One approach uses monitors placed at various points in the river. These monitors collect data relevant to flood prediction: water level, rain amount, time, humidity, and so on. Then the water level at a potential flooding point in the river can be predicted based on the data collected by the sensors upriver from this point. The prediction must be made with respect to the time the data were collected.

4.5 Clustering

Clustering is similar to classification except that the groups are not predefined, but rather defined by the data and algorithm. Clustering is alternatively referred to as unsupervised learning or segmentation. It can be thought of as partitioning or segmenting the data into groups that might or might not be disjointed. The clustering is usually accomplished by determining the similarity among the data on predefined attributes. The most similar data are grouped into clusters. Example 4.5 provides a simple clustering example. Since the clusters are not predefined, a domain expert is often required to interpret the meaning of the created clusters.

Example 4.5 A certain national departmental store chain creates special catalogs targeted to various demographic groups based on attributes such as income, location, and physical characteristics of potential customers (age, height, weight, etc.). To determine the target mailings of the various catalogs and to assist in the creation of new, more specific catalogs, the company performs a clustering of potential customers based on the determined attribute values. The results of the clustering exercise are then used by management to create special catalogs and distribute them to the correct target population based on the cluster for that catalog.

4.6 Association Rules

Link analysis, alternatively referred to as affinity analysis or association, refers to the data-mining task of uncovering relationships among data. The best example of this type of application is to determine association rules. An association rule is a model that identifies specific types of data associations. These associations are often used in the retail sales community to identify items that are frequently purchased together. Example 4.6 illustrates the use of association rules in market basket analysis. Here the data analyzed consist of information about what items a customer purchases

Example 4.6 A grocery store retailer is trying to decide whether to put bread on sale. To help determine the impact of this decision, the retailer generates association rules that show what other products are frequently purchased with bread. He finds that 60% of the times that bread is sold so are pretzels and that 70% of the time jelly is also sold. Based on these facts, he tries to capitalize on the association between bread, pretzels, and jelly by placing some pretzels and jelly at the end of the aisle where the bread is placed. In addition, he decides not to place either of these items on sale at the same time

Users of association rules must be cautioned that these are not causal relationships. They do not represent any relationship inherent in the actual data (as is true with functional dependencies) or in the real world. There probably is no relationship between bread and pretzels that causes them to be purchased together. And there is no guarantee that this association will apply in the future. However, association rules can be used to assist retail store management in effective advertising, marketing, and inventory control.

5. Application of Data Mining

Wide ranges of companies have deployed successful applications of data mining. While early adopters of this technology have tended to be in information-intensive industries such as financial services and direct mail marketing, the technology is applicable to any company looking to leverage a large data warehouse to better manage their customer relationships. Two critical factors for success with data mining are: a large, well integrated data warehouse and a well-defined understanding of the process within which data mining is to be applied. Some successful application areas include:

i. A pharmaceutical company can analyze its recent sales force activity and their results to improve targeting of high-value physicians and determine which marketing activities will have the greatest impact in the next few months.
ii. A credit card company can leverage its vast warehouse of customer transaction data to identify customers most likely to be interested in a new credit product.
iii. A diversified transportation company with a large direct sales force can apply data mining to identify the best prospects for its services.
iv. Data mining tools can be effectively utilized in feature extraction. Feature Extraction is one of the techniques to extract relevant information from database.
v. Data Mining can be used in prediction of unknown attribute value of a new instance when the values of the other attributes of new instance are known and a collection of instances with known value of all the attributes are given.
vi. Classification is forecasting discrete value. Prediction is forecasting continuous value.

Other Applications include:

- Surveillance / Mass surveillance
- National Security Agency
- Customer analytics
- Police-enforced ANPR in the UK
- Stellar wind (code name)

- **Surveillance / Mass surveillance**

Surveillance is the monitoring of the behavior of a person or group of people, often in a surreptitious manner from a distance by means of electronic equipment (such as CCTV cameras), or interception of electronically transmitted information (such as Internet traffic or phone calls).

Data mining & profiling

Data mining is the search for hidden patterns in large amounts of data. Data profiling in this context is the process of assembling information about a particular individual or group in order to generate a profile — that is, a picture of their patterns and behavior. Data profiling can be an extremely powerful tool for psychological and social network analysis. A skilled analyst can discover facts about a person that they might not even be consciously aware of them self.

- **National Security Agency**

The **National Security Agency/Central Security Service** (**NSA/CSS**) is a cryptologic intelligence agency of the United States government. It is responsible for the collection and analysis of foreign communications and foreign signals intelligence, which involves cryptanalysis. In cryptology, Prediction can be used in next bit prediction of key stream generators.

Transaction data mining

NSA is reported to use its computing capability to analyze "transactional" data that it regularly acquires from other government agencies, which gather it under their own jurisdictional authorities. As part of this effort, NSA now monitors huge volumes of records of domestic emails and Internet searches as well as bank transfers, credit-card transactions and travel and telephone records, according to current and former intelligence officials.

- **Customer analytics**

Customer analytics is a process by which data from customer behavior is used to help make key business decisions via market segmentation and predictive analytics. A large consumer package goods company can apply data mining to improve its sales process to retailers.

Data Mining

There are two types of categories of data mining. Predictive models use previous customer interactions to predict future events while segmentation techniques are used to place customers with similar behaviors and attributes into distinct groups. This grouping can help marketers to optimize their campaign management and targeting processes.

- **Police-enforced ANPR in the UK**

The UK has an extensive automatic number plate recognition (ANPR) CCTV network. Police and security services use it to track UK vehicle movements in real time. The resulting data are stored for 5 years in the National ANPR Data Centre to be analyzed for intelligence and to be used as evidence.

Data mining

A major feature of the **National ANPR Data Centre** for car numbers is the ability to data mine. Advanced versatile automated data mining software trawls through the vast amounts of data collected, finding patterns and meaning in the data. Data mining can be used on the records of previous sightings to build up intelligence of a vehicle's movements on the road network or can be used to find cloned vehicles by searching the database for impossibly quick journeys. ANPR can be used on investigations.

- **Stellar wind (code name)**

Stellar Wind is the open secret code name for certain information collection activities performed by the United States' National Security Agency. The information collection activities involved data mining electronic data about tens of millions of American citizens within the United States. This data included information about e-mail communications, phone conversations, financial transactions, and internet activity.

Data Mining

Data Mining is an analytic process designed to explore data in search of consistent patterns and/or systematic relationships between variables, and then to validate the findings by applying the detected patterns to new subsets of data. The ultimate goal of data mining is prediction - and predictive data mining is the most common type of data mining and one that has the most direct business applications.

6. Conclusion

Comprehensive data warehouses that integrate operational data with customer, supplier, and market information have resulted in an explosion of information. Competition requires timely and sophisticated analysis on an integrated view of the data. However, there is a growing gap between more powerful storage and retrieval systems and the users' ability to effectively analyze and act on the information they contain. A new technological leap is needed to structure and prioritize information for specific end-user problems. The data mining tools can make this leap. Quantifiable business benefits have been proven through the integration of data mining with current information systems, and new products are on the horizon that will bring this integration to an even wider audience of users. There are many different data mining applications and algorithms. These algorithms must be carefully applied to be effective. Discovered patterns must be correctly interpreted and properly evaluated to ensure that the resulting information is meaningful and accurate.

References

[1] M.Berry and G. Linoff, “Data Mining Techniques”, john Wiley 1997.

[2] Usama Fayyad, Gregory Piatetsky-Shapiro & Padhraic Smyth, “Knowledge discovery and data mining: Towards a unifying framework”, Proceedings of the International Conference on Knowledge Discovery and Data Mining, pages 82-88, 1996.

[3] Usama Fayyad, Gregory Piatetsky-Shapiro & Padhraic Smyth, “The KDD process for extracting useful knowledge from volumes of data”. Journal of the ACM, 39(11): 27-34, 1996.

[4] Usama Fayyad, Gregory Piatetsky-Shapiro & Padhraic Smyth, “Advances in knowledge Discovery and Data Mining”, Pages 1-34, AAAI/MIT Press, 1996.

[5] Margraret H. Dunham, “Data Mining: Introductory and Advanced Topics”, Pages 8-10, Pearson Education Ltd, 2003.

Natural Language Processing

R. Hema

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

ABSTRACT

Natural Language Processing is a technique where mechine can become more human and there by reducing the distance between human being and the mechine can bereduced. Therefore in sinple sence NLP makes human to communicate withthe mechine easily. There are many applications developed in past few decades in NLP. Most of these are very useful in everyday life for example a mechine that takes intructions by voice. There are lots of research groups working on this topic to develope more practical are useful systems.

1. INTRODUCTION

Language, ablity to speak, write and communication is one of the most fundamental aspects of human behaviour. As the study of human-languages developed the concept of communicating with non-human devices was investigated. This is the origin of natural language processing (NLP). The idea of natural language processing is to design and build a computer system that will analyze , understand and generate natural human-languages. Natural language communication with computers has long been a major goal of artificial intelligence, both for the information it can give about intelligence in general, and for practical utility.

2. WHAT IS NLP?

Natural language processing (**NLP**) is a field of computer science and linguistics concerned with the interactions between computers and human (natural) languages. Natural language generation systems convert information from computer databases into readable human language. Natural language understanding systems convert samples of human language into more formal representations such as parse trees or first order logic that are easier for computer programs to manipulate. Many problems within NLP apply to both generation and understanding; for example, a computer must be able to model morphology (the structure of words) in order to understand an English sentence, and a model of morphology is also needed for producing a grammatically correct English sentence.

2.1 WHY NATURAL LANGUAGE PROCESSING?

- kJfmmfj mmmvvv nnnffn333
- Uj iheale eleee mnster vensi credur
- Baboi oi cestnitze
- Coovoel2^ ekk; ldsllk lkdf vnnjfj?
- Fgmflmllk mlfm kfre xnnn!

COMPUTERS LACK KNOWLEDGE

- Computers “see” text in English the same you have seen the previous text!
- People have no trouble understanding language
  - Common sense knowledge
  - Reasoning capacity
  - Experience
- Computers have
  - No common sense knowledge
  - No reasoning capacity

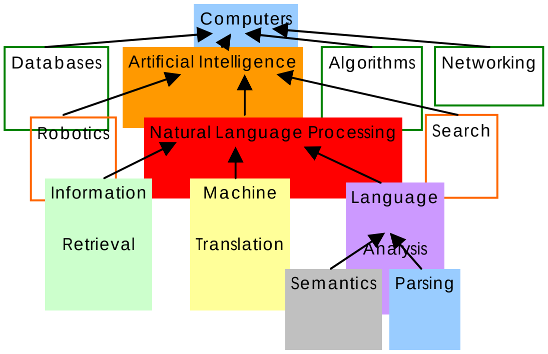
Applications for processing large amounts of texts require NLP expertise.

1. Classify text into categories
2. Index and search large texts
3. Automatic translation
4. Speech understanding
5. Understand phone conversations
6. Information extraction
7. Extract useful information from resumes
8. Automatic summarization
9. Condense 1 book into 1 page
10. Question answering
11. Knowledge acquisition
12. Text generations / dialogues

There are many applications of natural language processing developed over the years. They can be mainly divided into two parts as follows.

- **Text-based applications**
- This envolves applications such as searching for a certain topic or a keyword in a data base, extracting information from a large document, translating one language to another or summarizing text for different purposes.
- **Dialogue based applications**
- Some of the typical examples of this are answering systems that can answer questions, services that can be provided over a telephone without an operator, teaching systems, voice controled mechines (that take instructions by speech) and general problem solving systems.

3. WHERE DOES IT FIX IN THE CS TAXONOMY?



3.1 SUBPROBLEMS

- **Speech segmentation**

  In most spoken languages, the sounds representing successive letters blend into each other, so the conversion of the analog signal to discrete characters can be a very difficult process. Also, in natural speech there are hardly any pauses between successive words; the location of those boundaries usually must take into account grammatical and semantic constraints, as well as the context.

- **Text segmentation**

  Some written languages like Chinese, Japanese and Thai do not have single-word boundaries either, so any significant text parsing usually requires the identification of word boundaries, which is often a non-trivial task.

- **Word sense disambiguation**

  Many words have more than one meaning; we have to select the meaning which makes the most sense in context.

- **Syntactic ambiguity**

  The grammar for natural languages is ambiguous, i.e. there are often multiple possible parse trees for a given sentence. Choosing the most appropriate one usually requires semantic and contextual information. Specific problem components of syntactic ambiguity include sentence boundary disambiguation.

- **Imperfect or irregular input**

  Foreign or regional accents and vocal impediments in speech; typing or grammatical errors, OCR errors in texts.

- **Speech acts and plans**

  A sentence can often be considered an action by the speaker. The sentence structure alone may not contain enough information to define this action. For instance, a question is actually the speaker requesting some sort of response from the listener. The desired response may be verbal, physical, or some combination. For example, "Can you pass the class?" is a request for a simple yes-or-no answer, while "Can you pass the salt?" is requesting a physical action to be performed. It is not appropriate to respond with "Yes, I can pass the salt," without the accompanying action (although "No" or "I can't reach the salt" would explain a lack of action).

4. LADDER OF TRANSFORMATION

- Morphological
- Lexical
- Syntactic
- Semantic
- Discourse
- Pragmatic

4.1 STAGES IN TEXT PROCESSING

Natural Language Processing systems that process text documents (typically unstructured text) involve a number of **stages of processing**.

Cleaning removes unwanted control characters, etc.

Tokenization adds spaces to separate text at boundary points between words and surrounding punctuation, or between different punctuation marks

End-of-sentence detection identifies and marks sentence boundaries

Part-of-speech tagging adds a tag indicating the part of speech for each token

Phrase detection identifies and marks units that consist of multiple words – typically they are noun phrases of some type, but need not be

Entity detection identifies and marks entities, which usually consist of person names, place names, organization or company names and other proper nouns

Categorization identifies and marks what category something belongs to; typically categorization is used primarily for named entities (i.e. proper nouns)

Event detection identifies and marks events, which generally correspond to verbs

Relation detection identifies and marks relations, which are connections between two or more entities or between entities and events

XML or SGML tagging applies the designated tagging scheme used to markup the document for sentences, phrases, entities, categories, events, relations, etc.

Extraction the identified entities, events, relations, and any other identified concepts (like dates) are extracted from the document and stored externally

4.2 NATURAL LANGUAGE PROCESSING (NLP) TECHNIQUES

There are sevaral main techniques used in analysing natural language processing. Some of them can be breafly described as follows.

Pattern matching

The idea here is an approach to natural language processing is to interpret input utterances as a whole father than builing up their interpretation by combining the structure and meaning of words or other lower level constituents. That means the interpretations are obtained by matching patterns of words against the input utterance. For a deep level of analysis in pattern matching a large number of patterns are required even for a restricted domain. This problem can be ameliorated by hierarchical pattern matching in which the input is gradually canonicalized through pattern matching against subphrases. Another way to reduce the number of patterns is by matching with semantic primitives instead of words.

Syntactically driven Parsing

Syntax means ways that words can fit together to form higher level units such as phrases, clauses and sentences. Therefore syntacticaly driven parsing means interpretation of larger groups of words are built up out of the interpretation of their syntacticconstituent words or phrases. In a way this is the opposite of pattern matching as here the interpretation of the input is done as a whole. Syntactic analyses are obtained by application of a grammar that determines what sentenses are legal in the language that is being parsed.

Semantic Grammars

Natural language analysis based on semantic grammar is bit similar to systactically driven parsing except that in semantic grammar the catogaries used are defined semantically and syntactically. There here semantic grammar is also envolved.

Case frame instantiation

Case frame instantiation is one of the major parsing techniques under active research today.

They has some very useful computational properties such as its recursive nature and its ability to combine bottom-up recognition of key.

5. MAJOR TASKS IN NLP

- Automatic summarization
- Foreign language reading aid
- Foreign language writing aid
- Information extraction
- Information retrieval (IR) - IR is concerned with storing, searching and retrieving information.
- Machine translation - Automatically translating from one human language to another.
- Named entity recognition (NER) - Given a stream of text, determining which items in the text map to proper names, such as people or places. Although in English, named entities are marked with capitalized words, many other languages do not use capitalization to distinguish named entities.
- Natural language generation
- Natural language understanding
- Optical character recognition
- anaphora resolution
- Question answering - Given a human language question, the task of producing a human-language answer. The question may be a closed-ended (such as "What is the capital of Canada?") or open-ended (such as "What is the meaning of life?").
- Speech recognition - Given a sound clip of a person or people speaking, the task of producing a text dictation of the speaker(s).
- Spoken dialogue system
- Text simplification
- Text-to-speech
- Text-proofing

6. NATURAL LANGUAGE UNDERSTANDING

(NLU) is an advanced subtopic of Natural language processing that deals with machine reading comprehension.

The process of disassembling and parsing input is more complex than the reverse process of assembling output in natural language generation because of the occurrence of unknown and unexpected features in the input and the need to determine the appropriate syntactic and semantic schemes to apply to it, factors which are pre-determined when outputting language.

6.1 NATURAL LANGUAGE GENERATION

Natural Language Generation (NLG) is the natural language processing task of generating natural language from a machine representation system such as a knowledge base or a logical form.

In a sense, one can say that an NLG system is like a translator that converts a computer based representation into a natural language representation. However, the methods to produce the final language are very different from those of a compiler due to the inherent expressivity of natural languages.

NLG may be viewed as the opposite of natural language understanding. The difference can be put this way: whereas in natural language understanding the system needs to disambiguate the input sentence to produce the machine representation language, in NLG the system needs to make decisions about how to put a concept into words.

The simplest (and perhaps trivial) examples are systems that generate form letters. Such systems do not typically involve grammar rules, but may generate a letter to a consumer, e.g. stating that a credit card spending limit is about to be reached. More complex NLG systems dynamically create texts to meet a communicative goal. As in other areas of natural language processing, this can be done using either explicit models of language (eg, grammars) and the domain, or using statistical models derived by analysing human-written texts.

6.2 INFORMATION RETRIEVAL

This is another area where applications of natural language processing can be seen to extract information required from a large database. There are lots of places where this technique is applied to get things we need faster. Some of them are mentioned below.

Net Owl extractor

NetOwl extractor is an automatic indexing software that identifies names and other important concepts so that users can quickly locate and access the resources they need. The NetOwl extractor technology is particularly appealing to content providers and other organisations who need to add this capability to their own search services. This software makes it possible to recognize and process the names and other special artifacts that trip up most text processing software. Basically this is a data extracting engine that identifies and interprets key elements of free text, particularly names of people, places and organizations IBM, Thomson corporation and US Government are some people who use this software.

Lingsoft's Tools for Indexing And Retrieval

In English, no word has more than a handful of inflectional forms. For instance, the verb walk has four forms: walk, walks, walking and walked. That is why traditional indexing programs designed for English have completely ignored morphology. On the other hand, most other European languages have more complex morphology. A Finnish word may appear in hundreds or thousands of different forms. At each word, your indexing program should store both the word itself and all possible base forms in the index. The base forms are usually stored in a separate field to enable both exact and morphological matches. For instance, when the indexing program encounters the word thought, it should store thought in the exact-match field and the list (think, thought) in the base form field.

The retrieval program should work as follows

- Get a search term (base form), e.g. think or thought.
- Find the records where one of the possible base forms matches the search term. For example, think matches the records think, thinks and thought, while thought matches thought.

6.3 SPEECH RECOGNITION

Since the invention of the typewriter, the keyboard has been the king of human-computer interface, largely because it has been the only one widely available. The search for an alternative method, such as speech, has continued since the 1950s and computers that can be voice conrolled have featured in a number of science fiction films such as '2001:A Space Odyssey'. Speech recognition technology has made significant advances from the limited systems that began emerging in the late 1980s. At last speech recognition technology is a viable tool for both business and home computer users.

Today there are many systems developed using speech recognition technology. Some of them are listed below.

Voice Type Dictation

IBM's VoiceType is a speech recognition system that analyses spoken words and instantly turn them into text on a PC screen, at a typical dictation speeds of 70-100 words per minute with accuracy in access of 90%. It allows users to have hands and eyes free and to talk, rather than type. This works by analysing phonemes - the sound that make up spoken words - and then matching them against the phonemes of words held in the systems's vocabulary. IBM launched VoiceType Dictation 3.0 in June 1996 and is available in the market for under 100 pounds. Some of examples of people who used this are translators, doctors and writers.

Sri International

There are series of projects of SRI International with the goal of creating a technology for understanding spontancous spoken natural languages. This technology combines speech recognition with natural language understanding.

Spoken Language Systems

Spoken language systems perform speech recognition and semantic analysis, attempting to understand users' speech and respond appropriately. SRI's technology provides high accuracy,and real-time performance for large vocabulary tasks. SRI has developed a spoken language system in the air travel planning domain which permits users to ask naturally phrased questions such as "Show me the cheapest flight from Denver to Boston on Saturday after ten in the morning." The system connects to the Official Airline Guide on-line database and provides current information. Some of the other projects done at here are Neural Network, Large Vocabulary Wordspotting, Noise/Channel Robustness, Speech Disfluencies, Speech Machine Translation, Speaker Recognition and Voice Banking.

Dragon System Naturally Speaking

Dragon NaturallySpeaking is the natural way to input text. Almost anyone business professionals, writers and journalists, nontypists, telecommuters, and employees of small offices can find themselves quickly creating documents and reports with ease and accuracy. Dragon NaturallySpeaking spells correctly every time.

7. WHAT CAN WE EXPECT IN THE FUTURE!..........

Well there are so many applications we can dream with NLP techniques. How about robots that understand and follow instructions by human voice or driving by talking to the car like in some science fiction movies. Well they all can be real one day. Imagine we have a computer system that can follow simple human instructions and do what ever we want it to do. How convenient will it be ? But lets leave all that to the FUTURE.........

CONCLUSION

Therefore it is clear that Natural Language Processing takes a very important roll in new machine human interfaces. When we look at some of the products that are based on technologies with NLP we can see that they are very advanced but very useful. But there are many limitations, requiring improvements and developement of NLP oriented systems. For example language we speak is highly ambiguous. This makes it very difficult understand and analyze. Also with so many languages spoken all over the world it is very difficult to design a system that is 100 % accurate. These problems get more complicated when we think of different people speaking the same language with different styles. Therefore most of research on speech recognition is more concentrated on their areas. Information retrieval can be improved to give very accurate results for various searches. This will involve intelligence to find and sort all the results. So such intelligent systems are being experimented right now are we will be able to see improved applications of NLP in the near future.

REFERENCES

[1] Encyclopedia of Artificial Intelligence, Many authors.

[2] Natural Language Understanding, James Allen.

[3] Survey of the Human Language Technology, Ronald A.Cole, Joseph Mariani.

[4] Natural Language Theory and Technology, Ron Kaplan,Jeanette Figueroa.

Requirements Engineering for Data Warehousing

*Shubhra Saggar ** Nidhi Khurana

* Sr. Lecturer, GNIM Delhi shubhrasaggar@yahoo.com

** Lecturer, GNIM Delhi nidhi.khurana@sify.com

1. Abstract

Data Warehouses are used in multiple domains such as management and business process performance evaluation, strategic decision making and business planning, or even to support decisions made in business processes. Several surveys indicate that a significant percentage of data warehouses fail to meet business objectives or are outright failures. One of the reasons for this is that requirement analysis is typically overlooked in real projects Two different perspectives are integrated for requirement analysis: organizational modeling, centered on stakeholders, and decisional modeling, focused on decision makers. . In this paper we propose a goal-oriented approach to requirement analysis for data warehouses, based on the Tropos methodology.

2. Introduction

Building a data warehouse is a very challenging task because it can often involve many organizational units of a company. The main purpose of the Data warehouse is to support the decision making based on the analysis of distributed information. Most of the existing DW development approaches deal with how data should be structured, stored, and managed in DW[3].

Requirements Engineering, or the gathering and analysis of requirements, has been identified as one of the most important and vital aspects of software development. It is the primary means of defining business functionality and shapes the design of any data warehouse.

Hence it is necessary to gather all requirements from the users of a data warehouse, which belong to their analysis views. The design of data warehouse system is highly dependent on these requirements. Very often data warehouses are built without understanding correctly these needs and requirements and consequently fail for that reason.

During the requirement definition process system analysts of the IT department or consultants work together with users to describe the requirements for the data warehouse system. The data warehousing team receives these descriptions, but they have often trouble understanding the business terminology and find the description too informal to use for the implementation. Therefore the data warehousing team writes its system specification from a technical point of view. When the system specification is presented to the users, they do not quite understand because it is too technical. They are, however, forced to accept it in order to move forward.

3. Analyzing existing DW development approaches

Decision makers want to make their decisions on operational data. For this, decision makers can use data marts (DM: these are small DWs) to focus on but also they can directly access the entire DW through complex queries. Developing a DW involves defining the structure of its repository, defining the different operations that allow to feed data in it as well as to exploit it.

The DW development approaches can be categorized according to four perspectives:

(i.) the system perspective, (ii.) the subject perspective, (iii.) the usage perspective, and (iv.) the development perspective.

3.1 The System perspective: logical models

System perspective includes two options: tabular models and dimensional models. Tabular models are same like the relational model while the Dimensional model introduces the concept of Data Cube.

3.2 The Subject perspective: Analysis direction

DW can be developed using two different directions: the top-down direction, and the bottom up direction. In the Top-Down direction, the focus is on the information needed to make decisions prior to the information available at the operational level. This idea of this approach is taken from the Waterfall approach. Bottom-Up direction consists in building Data Marts first, which is faster than building a whole DW. This is basically used for legacy systems.

3.3 The Usage perspective: Analysis approaches

Usage perspective approach is used to differentiate the DW development techniques according to the approach used to analyze the system. The two approaches are: process driven and data driven. Process driven approaches are used in the business processes by which DWs are populated or used as well as the decision processes during which DWs are exploited. Data driven approaches consider first the sources of data upon which decisions are made.

3.4 The Development perspective: Development techniques

Development perspective includes the two development technique used to design the DW schema. There are two development techniques used: the E/R model and the Dimensional model. According to E/R model based approaches, DWs schemas are usually composed of flat entities. The Dimensional model was introduced by Kimball. It includes two tables: “facts table" that dominates the others and the “dimension table” that provide details on the operational data of the facts table.

To sum up, three major observations can be drawn from our review of DW development approaches:

(i.) Two main approaches can be distinguished as:

(a) process-driven approaches and

(b) data-driven bottom-up approaches

(ii.) The majority of approaches are data-driven

(iii.) Very few approaches are requirements-driven; because requirements-driven approaches are considered to be time consuming.

4. The Approach: Tropos Methodology

4.1 REQUIREMENT ANALYSIS

Methodology, based on the conceptual framework where the concepts of **agent, goal,** and **related notions** are used to support all software development phases, Tropos differs from other goal-oriented methodologies since it uses the notions of agent and goal to the early stages of software development.

During early requirement analysis, the requirements engineer identifies the main dimensions and requirements and models them as social actors, who depend on one another for goals to be fulfilled, tasks to be performed, and resources to be furnished

APPLYING METHODOLOGY

The Tropos methodology has been already been successfully applied in different areas. The facts and notations used in Tropos used are:

• **Actors**. An actor represents an enterprise user this can be a physical or virtual user. They are the main users of the system and are the one who interact with the system. They are represented with circle.

• **Dependencies.** A dependency represents a link between two actors, one depending on the other in respect of data the link can be a goal to be fulfilled, a task to be performed, or a resource to be delivered.

• **Actor diagram**. It is a graph of actors related by dependencies used to model how actors depend on each other.

• **Rationale diagram.** It is used to represent the logical relationships that represent the relationships between actors. Logically they are represented with AND, Or, Not symbols goals are decomposed into subgoals, with either AND the goal is achieved or with OR

4.1.1 Organizational Modeling

Organizational modeling consists of three different phases:

(i) Goal analysis, in which actor and rationale diagrams are produced;

(ii) Fact analysis, in which rationale diagrams are extended with facts

(iii) Attribute analysis, in which rationale diagrams are further extended with attributes.

Each phase is a different iterative process taking in input the diagrams produced by the previous one.

4.1.1.1 Goal Analysis

The first step for goal analysis is to represent the relevant stakeholders for the organization and their dependencies. This is done with the help of an actor diagram, in which actors can represent agents, roles, or positions within the organization.

4.1.1.2 Fact Analysis

The objective of fact analysis is to identify all the relevant facts for the organization which help in gathering the requirements. The main task here is that the analyst looks the rationale diagram of each actor and extension it by linking goals with facts that model the set of events to be recorded when goals are achieved.

4.1.1.3 Attribute Analysis

Attribute analysis is aimed at identifying all the attributes that are given a value when facts are recorded

4.1.2 Decisional Modeling

After organization modeling, the methodology proposes a second type of analysis focused on the goals of decision makers, i.e., the actors that play the most relevant role in the decisional process.

Firstly, all the decision makers are identified; then, for each of them, four steps are carried out:

(i) goal analysis, that produces rationale diagrams

(ii) fact analysis, that extends them with facts

(iii) dimension nalysis, that further extends them with dimensions

(iv) measure analysis, that further extends them with measures.

4.1.2.1 Goal Analysis

As for organizational modeling, goal analysis starts by analyzing the actor diagram for the decision makers. Decision makers are identified and initial dependencies between them are established. The goals associated to each decision maker are then decomposed and analyzed in detail, to produce a set of rationale diagrams. Goals may be completely different from those analyzed during organization modeling; indeed they are part of the decision process and might be not included in the operative process of the organization.

The goal analyze transactions is OR-decomposed into analyze debit transactions and analyze withdrawals, which in turn are further decomposed. So, for instance, the goal analyze debit transactions is OR-decomposed into analyze total amount and analyze number of transactions.

4.1.2.2 Fact Analysis

The objective of fact analysis is to identify all the relevant facts for the organization. The analyst navigates the rationale diagram of each actor and extends it by associating goals with facts that model the set of events to be recorded when goals are achieved.

4.1.2.3 Dimension Analysis

In this phase, each fact is related to the dimensions that decision makers consider necessary in order to satisfy their decisional goals. Dimensions are connected to the goals associated to the fact, where dimensions account number and month are associated to goal analyze total amount.

4.1.2.4 Measure Analysis

Finally, the analyst associates a set of measures to each fact previously identified.

5. CONCLUSION

In this paper we have proposed a goal-oriented methodology for requirement analysis in DWs. We believe that the adoption of this methodology can help the designer to reduce the risk of project failure by ensuring that early requirements are properly taken into account and formalized–which ensures a “good” design“.

6. REFERENCES

[1] Mohamed Frendi, Camille Salinesi: Centre de Recherche en Informatique Université de Paris 1, Panthéon Sorbonne

[2] Paolo Giorgini University of Trento - Italy Stefano Rizzi University of Bologna – Italy Maddalena Garzetti University of Trento - Italy

[3] Finkelstein, C.: An Introduction to Information Engineering. Addison Wesley: Sydney etc. 1989.

[4] Brinkkemper, S., Lyytinen K. and Welke R.: Method Engineering, Chapman & Hall, London 1996.

[5] Paul Raj Poonia, "Fundamentals of Data Warehousing", John Wiley & Sons, 2003.

[6] Sam Anahony, "Data Warehousing in the real world: A practical guide for building decision support systems", John Wiley, 2004

[7] W. H. Inmon, "Building the operational data store", 2nd Ed., John Wiley, 1999.

[8] Kamber and Han, "Data Mining Concepts and Techniques", Hartcourt India P. Ltd.,2001

Object Oriented Multidimensional Model for Multidimensional Databases

Suman Mann(Reader) Koyel Datta
Gupta(Lecturer)
Deptt. of computer science Deptt.of computer
science
M.S.I.T, New Delhi M.S.I.T, New
Delhi
Sumanmann200@gmail.com koyel.dg@gmail.com

ABSTRACT

There has recently been an increased interest in multidimensional databases (MDB) and On-line Analytical processing (OLAP) techniques. OLAP systems enforce different requirements than On-line Transactional Processing (OLTP) systems, and therefore, different data models and implementation methods are required for each type of system. Traditional access control models for transactional (relational) databases, based on tables, columns and rows, are not appropriate for Data warehouses (DW). Instead, security and audit rules defined for DWs must be specified based on the multidimensional (MD) modeling used to design data warehouses. Current approaches for the conceptual modeling of DWs do not allow us to specify security and confidentiality constraints in the conceptual modeling phase. There have been several different multidimensional data models proposed recently.

In this paper we are going to propose a object oriented approach for developing the conceptual model in which main focus will be on the properties: 1.Aggregation 2.Generalization 3 Association which are explained by UML.

Key-words: Unified modeling language, object orientation, multidimensional modeling.

1. INTRODUCTION

Data warehouse is used for taking strategic decision which plays a major role in giving effectiveness, performance, satisfaction and success to the organization. For this various Conceptual multidimensional models are used and this is not giving the accuracy of information as desired. Object oriented modeling and design is a new way of thinking about problems using models organized around real-world concept. The fundamental construct is the object, which combines both data structures and behaviors in a single entity .Object oriented models are useful for understanding problem, communicating with application expert ,modeling enterprises, designing program and database. Object oriented approach generally include four aspects: identity, classification, polymorphism and inheritance. Identity means data is quantized into discrete, distinguishable entities called objects. For example employee in a company, Maruti in a car are the examples of object.

Classification means object with the same data structure and behavior are grouped into a class. Polymorphism means many form i.e. Same operation may behave differently on different class. An operation is an action that an object or class perform

Inheritance is the sharing of attributes and operations among classes based on hierarchical relationship.i.e. a class is defined broadly and then refined into finer subclass. Each subclass inherits all the properties of the super class and adds its own unique properties.

1.1. DIMENSIONAL MODELING

The Dimensional modeling generally transform requirement into multidimensional model. This multidimensional is basically the Star schema. In star schema the major components are:

Fact: It is an item of interest for an organization. It is composed of the name of the fact like sale, primary key of every dimension which contribute in fact and measures which are normally of numeric type.

Dimension: Entity that play a major role in fact and attributes that is normally in non numeric type.

Now this Star schema which also considered as multidimensional model has the following major properties:

- Aggregation
- Inheritance
- Symmetric treatment of dimension and fact
- Many-to-many relationship in fact and dimension
- Additive etc.

Now these properties we are trying to consider with the help of object oriented approach

Now let us take object oriented model using UML

2.1 OBJECT ORIENTED MODELING

In conceptual multidimensional as shown in the following figure we are observing the sale, of the product, in particular time and in particular market.

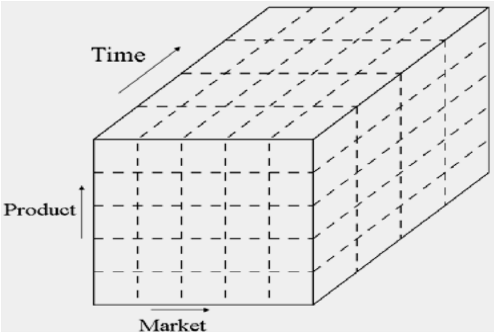


Fig1. Multidimensional model

Now the hierarchical relationship for this is model we can represent as:

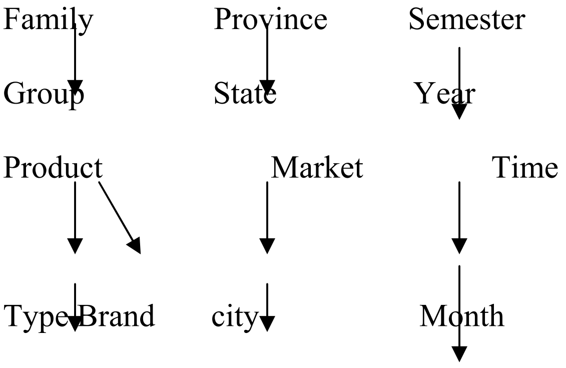


Fig2. Hierarchy according to cube given in fig1.

Object oriented model either is taken form software requirement specification or form the conceptual model. Object modeling for the conceptual model is represented as:

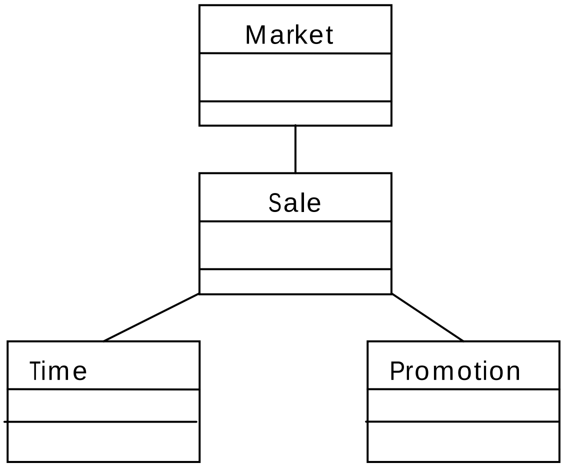


Fig 3.Object modeling

Here Dimensions we are showing with the class names i.e. Sale, promotion, time, market.

Now with the help of this object oriented model we can show the various types of relations which are normally taken as the major type of characteristics of the Conceptual multidimensional model.

1. Aggregation:

In object oriented approach the aggregation relation is just is represented with the help of symbol:

——◆

Here aggregated sale of the product is represented with the help of notation.

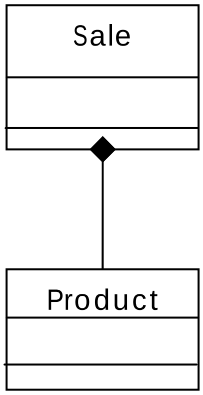
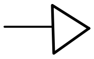


Fig 4.Aggregation

2. Symmetric treatment of dimension and fact: In the object model Dimension and fact are treated at the same level. Most of the dimensions which are here as classes are associated with the Fact which again consider as class. Like in Fig 3 sale class is associated with all the class time, product, and market

3. Inheritance Many times Similarities exist between the classes. When no of classes share the same method then it will be wastage of writing method again and again. The solution is inheritance. Inheritance is that mechanism. Inheritance models "is a" and "is like" relationships, enabling to reuse existing data and code easily. When A inherits from B, we say A is the subclass of B and B is the super class of A. Furthermore, we say we have "pure inheritance" when A inherits all the attributes and methods of B. The UML modeling notation for inheritance is a line with a closed arrowhead pointing from the subclass to the super class



In object modeling the inheritance relationship can be represented as

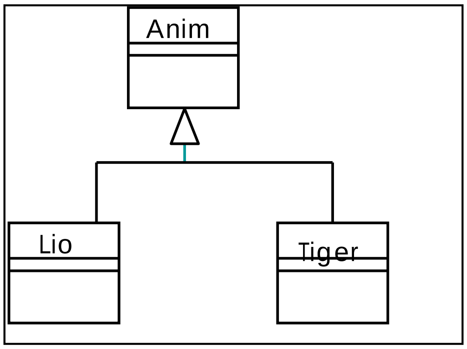


Fig. 5. Inheritance Relation

4. Many-to-many relationship in fact and dimension

Here Many-to-many means multiplicity. Multiplicity refers to the number of objects in one class that can be related to one object in the related class. Given diagram shows that 1or more rooms are related to one building. Thus its very easy to show multiplicity through object modeling which is complex in conceptual modeling.

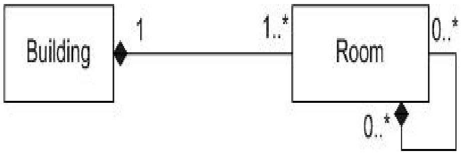


Fig 6. Many-to-Many Relationship

Now the object oriented approach we can easily apply to the whole system in which we can represent all the features which are discussed. The full fledge object model is:

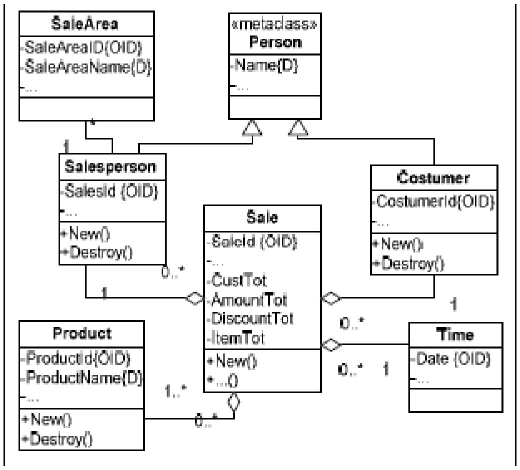


Fig 7. Object oriented model

III. PHASES OF THE DATA WAREHOUSE

According to the proposed object oriented model we can just have the following phases in the architecture of the data warehouse. Now the phases will be:

1. Object Oriented which simply convert the requirement specification into the multidimensional model
2. Logical shows mainly the representation of the data
3. Physical which actually implement the system

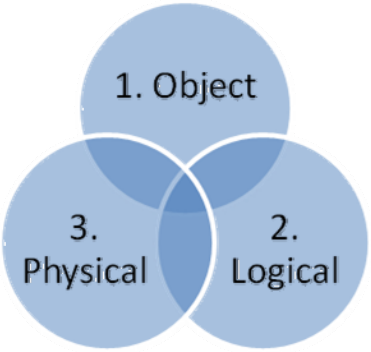


Fig 8. Phases of warehouse

References

[1] BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML, 2000.

[2] BUZYDLOWSKI, J.; SONG, II-Y.; HASSELL, L. A Framework for object-Oriented On-Line Analytic Processing. In Proc. Of the ACM 1st Int. Workshop on Data warehousing and OLAP (DOLAP). Washington DC, USA, 1998.

[3] G.Booch,J.Rumbaugh,and I. Jacobson, The Unified Modeling Language user Guide,Addison Longman Wesley,Reading,Mass.,1998.

[4] J.Trujillo, the GOLD Model: An Object Oriented Conceptual model for the design of OLAP Applications, Doctoral dissertation, Languages and Information systems Dept., Alicante University, Spain, June 2001.

[3] LUJÁN-MORA, S.; TRUJILLO, J.; SONG, I.Y. Extending UML for Multidimensional Modeling. 5th International Conference on the Unified Modeling Language (UML 2002), p. 290-304. Germany, 2002.

[4] SAPIA, C.; BLASCHKA, M.; HÖFLING, G.; DINTER B. Extending the E/R Model for the Multidimensional Paradigm. In: Proc. of the 1st Intl. Workshop on Data Warehouse and Data Mining (DWDM'98), 1998.

[5] TRUJILLO, J.; PALOMAR, M. An Object Oriented Approach to Multidimensional Database Conceptual Modeling (OOMD). 1st Int. Workshop on Data Warehousing and OLAP (DOLAP), Washington DC (USA), 1998.

[6] TRUJILLO, J.; PALOMAR, M.; GÓMEZ, J. The GOLD definition language (GDL): an object oriented formal specification language for multidimensional databases.Symposium on Applied Computing. Proceedings of the 2000 ACM Symposium on Applied Computing. Italy, p.346-350, 2000.

[7] TRUJILLO, J. et al. Designing Data warehouse with OO conceptual models. IEEE, p. 66-75, 2001.

Distributed Artificial Intelligence

Vikas Mathur

Asstt. Professor, CSE & IT Deptt, DCE, Gurgaon

ABSTRACT

Distributed artificial intelligence (DAI) was a subfield of Artificial intelligence research dedicated to the development of distributed solutions for complex problems regarded as requiring intelligence. These days DAI has been largely supplanted by the field of Multi-Agent Systems. DAI research is applicable in areas of Parallel problem solving , Distributed problem solving (DPS), Multi-Agent Based Simulation (MABS, both at macro and micro level) etc. The key concept used in DPS and MABS is the abstraction called software agents.

1. INTRODUCTION

Distributed Artificial Intelligence (DAI) systems can be defined as cooperative systems where a set of agents act together to solve a given problem. These agents are often heterogeneous. Its metaphor of intelligence is based upon social behaviour (as opposed to the metaphor of individual human behavior in classical AI) and its emphasis is on actions and interactions, complementing knowledge representation and inference methods in classical AI. This approach is well suited to face and solve large and complex problems, characterized by physically distributed reasoning, knowledge and data managing. An agent is a real or virtual entity which is emerged in an environment where it can take some actions, which is able to perceive and represent partially this environment, which is able to communicate with the other agents and which possesses an autonomous behaviour that is a consequence of its observations, its knowledge and its interactions with the other agents. DAI systems are based on different technologies like, e.g., distributed expert systems, planning systems or blackboard systems. Different languages are now advocated for describing DAI systems at a conceptual level. As such, DAI systems are therefore recognized as composite systems. A common feature of all these languages relies on the introduction of agents.

2. KNOWLEDGE ACQUISITION

DAI can be exploited in knowledge acquisition for modeling the experts (their cooperation, their conflicts), for modeling the knowledge acquisition process and specially the cooperation during knowledge acquisition from a group of experts. The process of knowledge acquisition can be described as the behaviour of a composite system made of several human agents (experts, knowledge engineers and users) and software agents (the knowledge acquisition tool, the final system and the software where the final system will be integrated). This notion of composite system made of heterogeneous, interacting agents can help to model the main relations between such agents (knowledge transfer, explanation, validation, assistance to problem solving...) and to analyze the cooperation underlying the process of knowledge acquisition. The notion of agent should allow to model the end-user as an agent and ease the description of a knowledge acquisition methodology involving several human agents.

The problems of design of a multi-agent system with multiple interaction levels requires protocols for interaction among such agents. Such a model can be extended in order to guide the construction of a knowledge acquisition tool which is based on a multi-agent architecture and involves several heterogeneous agents: human agents such as the experts, the knowledge engineers, the users, and artificial agents such as the knowledge acquisition too and the cognitive agents composing it.

The logics of intentions and of capabilities can be used for specifying the interactions among multiple cooperating agents that may be either human experts from whom knowledge must be acquired or cognitive agents that will represent them in a knowledge acquisition tool. The use of such logics may enable to formalize research on knowledge acquisition from multiple experts.

2.1 COOPERATION AND EXPLANATIONS

The type of cooperation depends on the organization of the agents: horizontal and vertical organizational structures can be distinguished. In non hierarchical societies, cooperation is based on sharing of tasks and of results, while in a hierarchical society; it relies on commands, bids and competition. In addition, the class of problems studied can influence the kind of cooperation: for example, the study of cooperation among designers influenced the multi-agent architecture implementing such collaboration. In the same way, conflicts among designers are analyzed and exploited for a tool supporting cooperative design.

Cooperation relies on the notion of goal adoption, which implies a goal common to the different agents. Different types of cooperation are presented: accidental cooperation, unilaterally intended cooperation, mutual cooperation. Cooperation is considered as a function of mutual dependency among the agents.

2.2 LEARNING

2.2.1 Single Agent learning

Single agent learning involves improving the performance or increasing the knowledge of a single agent. Single agent learning systems may be classified according to their underlying learning strategies. These strategies are ordered according to the amount of inferencing or the degree of knowledge transformation required by the learning system.

This order also reflects the increasing amount of effort required by the learning system and the decreasing effort required by the teacher.

2.2.2 Multiple Agent Learning

Distributed artificial intelligence (DAI) systems solve problems using multiple, cooperative agents. In these systems, control and information are often distributed among the agents. This reduces the complexity of each agent and allows agents to work in parallel and increases problem solving speed. In addition, a DAI system can continue to operate even if some of its agents cease to operate. This behavior allows the system to degrade gracefully in the event of failure of any of its parts. Also, each agent has resource limitations which could limit the ability of a single agent system to solve large, complex problems. Allowing multiple agents to work on these types of problems may be the only way to realistically solve them.

3. COORDINATION

Coordination is the process by which an agent reasons about its local actions and the (anticipated) actions of others to try and ensure the community acts in a coherent manner, is perhaps the key problem of the discipline of Distributed Artificial Intelligence (DAI). In order to make advances it is important that the theories and principles which guide this central activity are uncovered and analyzed in a systematic and rigorous manner.

Traditional AI Traditional AI has devoted considerable attention to problems of manufacturing scheduling and control (Smith 91). By taking into account semantic information about the domain that does not lend itself to numerical computation; by applying heuristics judiciously and selectively (rather than globally as with dispatch rules), and by adopting a "satisficing" approach that does not insist on a theoretically perfect optimum, symbolic computation has led to systems that are somewhat faster than numerical programming and are more flexible and able to accommodate richer constraints, while yielding results superior to dispatch rules. However, these systems still tend to be large, complex, and specific to a particular installation, thus making them expensive to construct and difficult to maintain and reconfigure. Furthermore, while they are faster than some numerical programming codes, they are not fast enough for a facility whose configuration and load changes daily.

CONCLUSION

Taking into account such research on cooperation among agents, DAI can help to model a cooperative system. The couple system-user can be considered as a couple of two agents that must cooperate in order to perform for example cooperative problem solving. It is then interesting to model the set constituted by the system and by the user as a multi-agent system compound of two agents: decomposition of the global task between both agents, distribution of subtasks, planning of job among them, possible interactions among them, possible communication language they use, possible conflicts and way such conflicts will be solved. This vision can be extended to several users: in this case, a multi-agent system is obtained, with an artificial agent (the cooperative system) and several human agents (the users), such human agents can interact among themselves or with the assistance system. For building a Knowledge Based Systems (KBS), the interaction of the system and of the users can be modeled through a model of agent. Likewise, a model of agent may help to model the explanation process between (possibly heterogeneous) agents. For example, the logics of intentions and of capabilities can be applied to model the behaviour of the couple of agents KBS - user (the former must provide the latter with a cooperative assistance).

REFERENCES:

[1] Carbonell, Jamie G. and Langley, Pat Machine Learning Tutorial from Seventh National Conference on Artificial Intelligence, 1988.

[2] Carbonell, Tom M., Michalski, Ryszard S. and Mitchell, Tom M., eds. Machine Learning: Volume 1. Palo Alto, CA: Tioga Pub. Co., 1983.

[3] Bond,A.H. and Gasser,L. (1988), "An Analysis of Problems and Research in DAI," in A.H.Bond and L.Gasser, eds., Readings in Distributed Artificial Intelligence, Morgan Kaufmann, 3-36.

[4] Applications of Distributed Artificial Intelligence in Industry by Dr. H. Van Dyke PARUNAK, Industrial Technology Institute, eds., Foundations of Distributed Artificial Intelligence. Wiley Inter-Science, 1994.

[5] Coordination Techniques for Distributed Artificial Intelligence by N. R. Jennings, Queen Mary and Westfield College, University of London, Mile End Rd. London E1 4NS UK.

[6] Learning for Distributed Artificial Intelligence Systems, by Michael L. Dowell and Ronald D. Bonnell, Department of Electrical & Computer Engineering University of South Carolina, Columbia, SC 29208

Proceedings of the National Seminar on Artificial Intelligence - NSAI-09
July 17, 2009
Dronacharya College of Engineering, Khentawas, Farrukh Nagar, Gurgaon-123506

Soft Computing: New Dimensions Towards Consumer Appliances (Review)

Vinay Kumar Nassa

Asst Prof (ECE) Dronacharya College of Engg, (Gurgaon), vinay_nassa@rediffmail.com

Abstract

Soft Computing is a complex of methodologies that embraces approximate reasoning, imprecision, uncertainty and partial truth in order to mimic the remarkable human capability of making decisions in real-life, ambiguous environments. Soft Computing has therefore become popular in developing systems that encapsulate human expertise. The applications of soft computing covers a wide range of application areas, including optimization, data analysis and data mining, computer graphics and vision, prediction and diagnosis, design, intelligent control, and traffic and transportation systems. Soft Computing encourages the integration of soft computing techniques and tools into both everyday and advanced applications. By integrating soft computing with other disciplines results in new applications.

Soft computing is causing a paradigm shift (breakthrough) in engineering and science fields since it can solve problems that have not been able to be solved by traditional analytic methods (tractability (TR)).

This paper intends to remove the gap between theory and practical and attempts to learn how to apply soft computing practically to industrial systems from Examples/analogy reviewing many application papers.

Keywords—Industrial applications, soft computing, fuzzy logic, neural networks, evolutionary computation, computational intelligence.

1. Introduction

This paper in fact examines the possibility of incorporating soft computing techniques into various applications. 1st section gives an overview of soft computing, hard computing followed by 2nd section that describes about the constituents of the soft computing. Industrial innovations using soft computing are discussed in section 3rd and applications to consumer appliances are explained in section 4th.Finally future opportunities are outlined in section 5th and conclusions are presented in 6th section.

Physical systems described by multiple variables and multiple parameter models having non-linear coupling occurring the fields of physics, engineering applications, economy (Business) etc. The conventional approaches for understanding and predicting the behaviour of such systems based on analytical techniques can prove to be very difficult, even at the initial stages of establishing an appropriate mathematical model. The computational environment used in such an analytical approach is perhaps too categorical and inflexible in order to cope with intricacy and the complexity of the real world physical systems. It turns out that in dealing with such systems, one has to face a high degree of uncertainty and tolerate imprecision. It is very costly to try to increase eprecision.

Albert Einstein in 1921 said

“So far as laws of mathematics refer to reality, they are not certain and so far as they are certain, they do not refer to reality.”

The statement has a very deep-routed implications bringing out the fact that we can not always talk about precision, accuracy and certainty, as far as complex working systems are concerned. We have to live with uncertainty, then why not to exploit these to our advantages.

The concept of soft computing was introduced by Dr.Lotfi Zadeh in 1991.

Soft computing infect differs from the Hard conventional computing in that, unlike hard computing it is tolerant of impression, uncertainty and partial truth. In effect, the role model for soft computing is the human mind. The guiding principle of soft computing is; exploit the tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness and low solution cost. The basic ideas underlying soft computing in its current incarnation have links to many earlier influences like fuzzy sets; analysis of complex systems and decisions processes; and possibility theory and soft data analysis.

Loosely speaking that computing has been categorized as the form of computing based upon binary logic, crisp sets (exact reasoning) and rigorous mathematical modeling. At this juncture, the principal constituents of soft computing are fuzzy logic (FL), neural network theory (NN). Genetic algorithms (GA) and probabilistic reasoning (PR), with the later subsuming belief networks, genetic algorithms, chaos theory and parts of learning theory. What is important to note is that SC is not a mélange of FL, NN, GA and PR. Rather; it is a partnership in which each of the partners contributes a distinct methodology for addressing problems in its domain. In this perspective, the principal contribution of FL, NN and PR are complementary rather then competitive.

2. Implications of Soft Computing

The methodologies in (SC) Soft computing are complementary and synergistic rather then competitive. The complementarity of FL, NN,GA and PR has an important consequence: in many cases a problem can be solved most effectively by using FL, NN, GA and PR rather then exclusively. Within the soft computing, the main concerns of fuzzy logic, neuro computing and probabilistic computing center on:

FL: appropriate reasoning, information granulation, computing with words

NC: learning adaptation, classification, system modeling and identification

GC: synthesis, tuning an optimization through systematized random search and evolution

PC: management of uncertainty belief networks, prediction, chaotic system

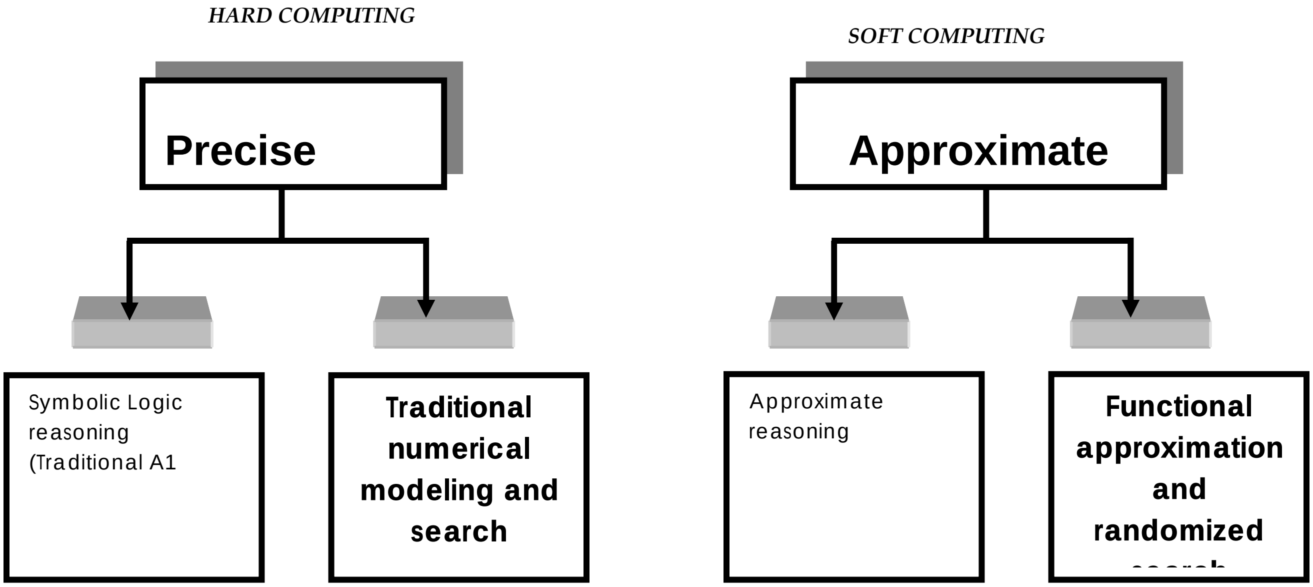

HARD COMPUTING
SOFT COMPUTING
Precise
Approximate
Symbolic Logic reasoning (Traditional A1
Traditional numerical modeling and search
Approximate reasoning
Functional approximation and randomized search

Fig 1: PROBLEM SOLVING TECHNOLOGIES

3. Industrial innovation using Soft Computing

Soft computing (SC) was proposed for construction of new generation artificial intelligence (high machine intelligence quotient (HMIQ), human –like information processing) and for solving non-linear and mathematically unmodeled system (tractability) (TR) [3]. In addition, SC can be implemented at low cost (LC). SC is the fusion or combination of fuzzy, neuro and evolutional computing. Later, chaos computing and immune networks were added [4] to explain so called complex systems [2], cognitive distributed artificial intelligence and reactive distributed artificial intelligence.

It has been proved that non-linear mapping obtained by neural networks can be approximated to any desired accuracy by the use of fuzzy systems [6]. As neural networks have flexible learning capabilities, it is possible to develop non-linear models using only input output data. How ever it is often cumbersome to fine-tune the modeling accuracy of neural networks, because it may be difficult to explain logically the cause and result in the excitation–response relationship. On the other hand, fuzzy systems provide clear advantages in a knowledge representation and acquisition. For example, knowledge is easily introduced in parallel to an adaptive fuzzy neural network by constructing a hierarchal diagnosis structure and modifying rules by available structured knowledge or modifying and adjusting fuzzy interference for pattern recognition with lack of input data by some complementary knowledge [7]. Fuzzy systems, however, have been missing adaptation capabilities for a long time [1]. It has been shown that under the condition of minor restrictions, functional behaviors of radial bases function networks and fuzzy interference system are the same [8]. On the other hand, local models in blended multiple models structures for non-linear systems (fast fuzzy neural networks) have been recently investigated [9] [10] [11].

Also type-I fuzzy systems implemented using Gaussian radial basis function neural networks as local models in blended model structures for non-linear systems. This fuzzy neural network is obtained by replacing the output layer weights with a linear function of the network inputs. Each neuron represents a local linear model with its corresponding validity function (Membership function). Further more, the radial basis function network is normalized like fuzzy membership functions. The side effects of normalizing should be considered, as all validity functions for a specific input combination sum up to one [12]. The Gaussian validity function determines the regions of the input space where each neuron is active. The input space becomes larger when these networks represent dynamic systems. A fast fuzzy neural network with general parameter learning is developed. It is especially suitable for real time fault diagnosis since what we have to do is to observe only changes in a general parameter. Recurrent fuzzy neural networks are recommended as a means to reduce the size of the input space [13]. They are able to yield adaptive self-tuning, self-organizing and automated design functions for non-linear systems and system for each suitable mathematical model are not obtained. They are also used for cognitive (fuzzy decision tree etc.) and reactive (multiagent system coordination etc.) decision making. DSP's and advanced computer systems are at present utilized to implement soft computing.

Neuro computing and evolutionary computation usually need to a lot of computational time, which is the disadvantage of the implementation of soft computing. Recently developed fuzzy neural networks enable solutions to be obtained for problems that have not been able to be solved by traditional analytical methods (hard computing) [14], since function approximation is used rather than parameter optimization (TR). Tractability enables industrial systems to become increasingly innovative. Evolutionary computation has been developed and modified for application of optimization for large scale and complex system as shown in this paper. Fogel proposed intelligence based on Bioinformatics. Data mining for which soft computing is an effective and promising approach, has been attracting the attention of researchers. Data mining is accepted to be applied large-scale process plants and electric power system for decision support and optimization (TR).

Soft computing has recently been playing an important role in advanced knowledge processing. An advanced learning method using a combination of perception and motion has been introduced. Emergent, self organizing, reflective and interactive (among human beings, environment and artificial intelligence) knowledge processing is considered by using soft computing and by borrowing ideas from bio information processing. Soft computing provides rich knowledge representation (symbol and pattern), flexible knowledge acquisition (by learning from data and by interviews with experts), and knowledge processing (interface by interface between symbolic and pattern knowledge). Therefore it is straightforward to construct low cost intelligent system. The various kinds of artificial intelligence (cognitive and reactive AI) make industrial system intelligent.

Such intelligent system has adaptive, autonomous, decision support, optimization and emergent functions (HMIQ). This HMIQ enables innovations in industry fields.

4. Applications in Consumer Appliances

4.1 General View

The field of consumer or home appliances is not a popular research area in the academic community. Almost all such research activities are related to practical product development. Therefore, most of the sparse literature (mainly conference papers) on soft computing in consumer appliances has its origins in industry. Due to commercial confidentiality reasons, these conference papers do not usually give detailed descriptions of algorithms and methods used but, rather, tend to be fairly superficial. Such industrial research and development is particularly active in Japan and South Korea, while corresponding industries in Europe and the United States are only just starting to use soft computing in the control of various consumer appliances. In Japan, even ordinary consumers are aware of the great potential of fuzzy logic, neural networks, and chaos computing which have already brought machine intelligence into their daily lives. There is clearly a demand in developed countries in Asia for intelligent, human-like, and user-friendly control features. The figurative term ”heartware” is sometimes used in respect to such consumer products that support the following general objectives:

- Comfortable way of life;

- Ease of life to manage time and space; and

- Health- and environment-conscious life.

Although the field of research on consumer appliances differs greatly from the other application fields reviewed in this paper, an overview of state-of-the-art appliances

should be given in view of the considerable and rapidly growing monetary value of the intelligent home appliance business. Besides, numerous interesting innovations have

been made in this field during the past ten years.

4.2 Application Areas

Soft computing has been used in consumer appliances since the late 1980s. In the pioneering years, fuzzy logic was clearly the dominating methodology. Then, in the early 90s, neural networks were merged with fuzzy logic to construct various neuro-fuzzy combinations; and soon after that, chaos computing (CC) started to attract interest in the Japanese appliance industry. More recently, evolutionary computation (EC) has also shown considerable potential in this heterogeneous application field. A summary of home appliances and their innovative characteristics that were made possible largely by soft computing is presented in Table 3. All of those characteristics have an obvious connection to the heartware concept mentioned above. Next, we take a closer look at three specific classes of appliances: cooling and heating, washing, and food preparation.

5. Future opportunities

The successful applications of soft computing (SC) suggest that SC will have increasingly greater impact in the coming years. Soft computing is already playing an important role both in science and engineering. In many ways, soft computing represents a significant paradigm shift (breakthrough) in the aim of computing, a shift that reflects the fact that the human mind, unlike state-of-the-art computers, possesses a remarkable ability to store and process information, which is pervasively imprecise, uncertain, and lacking in categorists [2].

Soft computing can be extended to include computing not only from human thinking aspects (mind and brain) but also from bio-informatics aspects. In other words, cognitive and reactive distributed artificial intelligence will be developed and applied to large-scale and complex industrial systems.

In fuzzy systems, computation with words will be investigated increasingly and also evolutionary computation will be emerging. It is expected that they will be applied to the construction of more advanced intelligent industrial systems.

Conclusions

Soft computing is already a major area of academic research. However, the concept is still evolving, and new methodologies, e.g., chaos computing and immune networks are nowadays considered to belong to SC. While this methodological evolution is taking place, the number of successful soft computing-based products is increasing concurrently. In the majority of such products, SC is hidden inside systems or sub-systems, and the end user doesn't necessarily know that soft computing methods are used in control, diagnosis, pattern recognition, signal processing, etc. This is the case when SC is mainly used for improving the performance of conventional hard computing algorithms or even replacing them. However, soft computing is very effective when it is applied to real-world problems that cannot be solved by traditional hard computing.

References

[1] P. J.Werbos, ”Neuro-control and elastic fuzzy logic: Capabilities, concepts, and applications,” IEEE Trans. on Industrial Electronics, vol. 40, no. 2, pp.170-180, 1993.

[2] Y. Dote and R. G. Hoft, Intelligent Control - Power ElectronicsSystems, Oxford, U.K., Oxford University Press, 1998.

[3] L. A. Zadeh, ”Fuzzy logic, neural networks and soft computing,”in Proc. of the IEEE Int. Workshop on Neuro Fuzzy Control,Muroran, Japan, p. 1, 1993.

[4] L. A. Zadeh, ”The role of soft computing and fuzzy logic in theconception, design, and development of intelligent systems,” inProc. of the IEEE Int. Workshop on Soft Computing in Industry,

Muroran, Japan, pp. 136-137, 1996.[5] L. A. Zadeh, ”From computing with numbers to computing withwords -F rom manipulation of measurements to manipulation of

perceptions,” in Proc. of the IEEE Int. Workshop on Soft Computingin Industry, Muroran, Japan, pp. 221-222, 1999.

[6] L. X. Wang, ”Fuzzy systems are universal approximators,” inProc. of the IEEE Int. Conf. on Fuzzy Systems, San Francisco,CA, pp.1163-1172, 1992.

[7] S. Sato, Y. Arai, and K. Hirota, ”Pattern recognition using fuzzyinference with lacked input data,” in Proc. of the IEEE Int. Conf.on Fuzzy Systems, San Antonio, TX, pp.100-104, 2000.

[8] J.-S. R. Jang and C.-T. Sun, ”Functional equivalence betweenradial basis function networks and fuzzy inference systems,” IEEETrans. on Neural Networks, vol. 4, no. 1, pp. 156-159, 1993.

[9] R. Shorten, R. Murray-Smith, R. Bjorgan, and H. Gollee, ”On theinterpretation of models in blended multiple model structures,”Int. Journal of Control, vol.72, no. 7/8, pp. 620-628, 1999.

[10] T. F. Junge and H. Unbehauen, ”On-line identification of nonlinearsystems using structurally adaptive rectangular local linearmodel networks,” in Proc. of the 3rd SIMONET Workshop onRecent Results in System Identification and Modeling, Bochum,

Germany, pp. 1-7, 1997.

[11] O. Nelles, ”Orthogonal basis functions for nonlinear system identificationwith local linear model trees (LOLIMOTO),” in Proc.of the 11th IFAC Symposium on System Identification, Fukuoka,Japan, pp. 667-672, 1997.

[12] R. Murray-Smith and T. A. Johansen (Eds.), Multiple ModelApproaches to Modeling and Control, London, U.K.: Taylor &Francis,1997.

[13] C.-H. Lee and C.-C .Teng, ”Identification and control of dynamicsystems using recurrent fuzzy neural networks,” IEEE Trans. onFuzzy Systems, vol. 8, no. 4, pp. 349-366, 2000.

[14] P. A. Marchi, L. S. Coelho, and A. A. R. Coelho, ”Comparativestudy of parametric and structural methodologies in identificationof an experimental nonlinear process,” in Proc. of the IEEE Int.Conf. on Control Applications, Hawaii, U.S.A., pp. 1062-1067,

1999.

[15] A. J. Calise, ”Neural networks in nonlinear aircraft flight control,”IEEE Aerospace and Electronics Systems Magazine, vol.11, no. 7, pp. 5-10, 1996.

[16] G. Kolumban, M. P. Kennedy, and L. O. Chua, ”The role of synchronization

in digital communications using chaos-Part I: Fundamentals of Digital Communications,” IEEE Trans. on Circuits

and Systems-I: Fundamental Theory and Applications, vol. 44, no. 10, pp. 927-936, 1997.

[17] S. K. Patra and B. Mulgrew, ”Fuzzy implementation of aBayesian equalizer in the presence of inter-symbol and co-channel interference,” IEE Proceedings-Communications, vol. 145, no. 5, pp. 323-330, 1998.

[18] C. E. Cramer and E. Gelenbe, ”Video quality and traffic QoS in learning-based sub-sampled and receiver-interpolated video sequences,” IEEE Journal on Selected Areas in Communication, vol. 18, no. 2, pp. 150-167, 2000.

[19] J. M. Jou and P.-Y. Chen, ”A fast and efficient lossless datacompression method,” IEEE Trans. on Communications, vol. 47, no. 9, pp. 1278-1283, 1999.

[20] E. Gelenbe, I. W. Habib, S. Palazzo, and C. Douligeris, ”Guest editorial: Intelligent techniques in high speed networks,” IEEE Journal on Selected Areas in Communication, vol. 18, no. 2, pp. 145-149, 2000.

[21] G. Chakraborty and B. Chakraborty, ”A genetic algorithm approach to solve channel assignment problem in cellular radio networks,” in Proc. of the IEEE Midnight-Sun Workshop on Soft Computing Methods in Industrial Applications, Kuusamo, Finland,

pp. 34-39, 1999.

[22] B. Dengiz, F. Altiparmak, and A. E. Smith, ”Local search genetic algorithm for optimal design of reliable networks,” IEEE Trans. on Evolutionary Computation, vol. 1, no. 3, pp. 179-188,1997.

[23] X. M. Gao, X. Z. Gao, J. M. A. Tanskanen, and S. J. Ovaska, ”Power prediction in mobile communication systems using an optimal neural-network structure,” IEEE Trans. on Neural Networks, vol. 8, no. 6, pp. 1446-1455, 1997.

[24] J. H. Kim, K. S. Kim, M. S. Sim, K. H. Han, and B. S. Ko, ”An application of fuzzy logic to control the refrigerant distribution for the multi type air conditioner,” in Proc. of the IEEE Int. Fuzzy Systems Conference, Seoul, Korea, vol. 3, pp. 1350-1354, 1999.

[25] R. Zhu, B. Tian, Q. Wang, and G. Dai, ”Application of fuzzy logic in home appliance: Gas heater controller design,” in Proc. Of the IEEE Int. Conf. on Intelligent Processing Systems, Beijing, China, pp. 373-376, 1997.

[26] T. Nitta, ”Applications of neural networks to home appliances,” in Proc. of the IEEE Int. Joint Conf. on Neural Networks, Nagoya, Japan, pp. 1056-1060, 1993.

Table 1 SUMMARY OF APPLICATION OF SOFT COMPUTING

Application	Characteristics	SC Component	Reference
Cooling and Heating	- Stable refrigerant distribution under changing loading condition with multiple indoor	FL(TR)	24
	- Temperature, air capacity and direction of air stream are determined index; pleasant and comfortable living space.	NN (HMIQ)	24
	- On site learning of parameters of neural network – based, temperature controller; adoption to user's habits and preference.	EC, NN (HMIQ)	24
	-The number and locations of occupants are identified using thermal imaging; comfortable room temperature and wind direction	FL, NN (HMIQ)	25
	-Stable temperature control for compensating heat shocks; keeps food fresh for a longer period	FL, NN(TR)	25
	- Learns user's patterns to use the refrigerator (e.g. the frequency of door openings); energy savings and well-regulated temperature.	FL, NN (HMIQ)	25
	- Takes account the complex coupling between temperature and relative humidity; favorable dynamic process behavior under a wide range of operating conditions.	FL (TR)	
	- Robust disturbance handling capability; comfortable and safe bathing conditions.	FL(TR)	26
Washing	Fine tuning of prediction rules; accurate estimation of dish amount	EC, FL, NN (TR)	
	-Determination of washing time using fuzzy inference, and chaotic movement of a two-link nozzle; lower electric power consumption and higher washing efficiency.	CC, FL (HMIQ)	
	- Implicit mimicking physical sensors by neural network based tuning of rough membership functions; fine-tuned but low-cost automatic washing machine.	FL, NN (TR)	26
Food Preparation	- Different phases of the heating process are finely controlled according to traditional practices that are reproduced tastier cooked rice.	FL, NN (TR)	26
	- Fine tuning of estimation rules; accurate estimation of rice amount	EC, FL, NN (TR)	26
	- fine tuning of fuzzy control rules; optional control of the cooking process.	EC, FL, NN (TR)	