ISSN (Online): 1694-0814
www.IJCSI.org

Predict Success or Failure of Remote Infrastructure Management

Er. Satinder Pal Ahuja[1], Dr. Sanjay P. Sood [2]

[1] **Research Scholar**

[2] **Department of Information Technology, CDAC, Mohali, Punjab, India**

Abstract

Matching IT infrastructure with the needs of a business is a CIO's biggest challenge. The increasing complexity of IT infrastructure and the constant pressure to reduce its costs, forces CIOs to maximize the use of existing resources and to enhance productivity of key technical people. Downtime, however brief, can result in revenue losses, unhappy customers and a loss of productivity. Instead of using productive time to making strategic decisions, key personnel are forced to spend it in routine operations management of IT infrastructure. Remote Infrastructure Management (RIM) involves a combination of near-shore and offshore delivery models. RIM can reduce the costs of operations, thus enabling IT managers to consider investing in new technology. The objective of this paper is to focussed whether a RIM will ultimately lead to improvement in Software Process using ANN. The benefit of this work will be that it will save cost and time in actual implementation of RIM
Keywords–Artificial Neural Networks (ANN), Remote InfrastructureManagement (RIM) etc

1. Introduction

Remote Infrastructure Management Services provide for monitoring and management of all infrastructures pertaining to networks, data centers, servers, storage security, Applications and End user computing, outside the company's offices.

Advantages of RIM include:

- Cost Reduction - Reduced manpower,performance efficiency and capacity planning resulting in 40-60% reduction in costs
- Quality and Process –Process-based approach to solve issues, efficient handling of escalations, and quality certifications to ensure adherence to standards and security controls
- Best practices – Experience with multiple enterprises resulting in standardization through ideas, learning and best practices
- Expertise - Domain experts who aid others in skill development
- Risk Mitigation - Improved risk-mitigation and assured business continuity
- Visibility - Timely reporting providing CIOs with greater visibility, real-time control and analysis of historical trends. Increased automation with integrated tools, providing common framework for operations
- Scalability - Absorb the peaks and troughs of manpower needs
- Service levels - Precise service-level agreements with penalty clauses for downtime
- Pre-emptive problem resolution - Through proactive monitoring and event correlation

2. Operational Issues of RIM

Following are the issues observed in RIM process.

- People related issues
- Poorly defined roles and responsibilities
- Every team does administration, development & support
- No clear responsibility matrix / SLAs. Outsourced support -
  Technology related issues
- When there is no evidence of architecture documents for network, systems and security
- When there are no tools for monitoring & managing the infrastructure
- When there is poor metrics measurement
- When there are single points of failure in Internet connectivity, Load Balancers, Firewalls
- When there is weak security

- When redundancy for applications at the system/ server/ database level is unavailable
- Since actual Implementation of RIM involves large costs and time investments there is a need to develop a system for predicting the success or failure of RIM

3. Objective

The objective of this work is to develop an ANN based system to verify whether a RIM will ultimately lead to improvement in Software Process. The benefit of this work will be that it will save cost and time in actual implementation of RIM

4. Literature Survey

While IT management services represent a mature subject in the IT business arena, the emerging cloud generation of management services require critical enhancements to the current processes and technologies in order to deliver IT management remotely with rapid onboarding and minimal labor involvement from experts, to be affordable and scale up to the promise of the cloud. Traditional Remote Infrastructure Management (RIM) service providers use their own Network Operations Centers (NOC) to remotely monitor and manage customers' IT infrastructure. The primary business value for RIM services is that it helps global enterprises to small and medium businesses (SMB) to outsource the burden of managing their IT infrastructure. Although the IT management service itself delivered this way is more affordable, the RIM customer on-boarding process particularly is not, taking between one to two months of expensive labor.

Management services represent a mature subject in the IT business arena. According to Gartner Dataquest, Remote Infrastructure Management (RIM) is a rapidly growing market growing at a Compound Annual Growth Rate of 36%, and projected to grow from USD $14.3B to $30B by 2010 [[8]]. Typical RIM service providers use their own Network Operations Centers to remotely monitor and manage customers' IT infrastructure elements such as networks, systems' hardware and operating systems, and applications. The primary business value for RIM services is that it helps global enterprises and SMBs to outsource the burden of managing their IT infrastructure, thus, cutting down costs for infrastructure management and gaining access to expert skills. The customers can focus then on their core business, shifting the responsibility for IT management to RIM, while maintaining ownership of their assets.

A RIM solution generally involves monitoring services comprising of NOC support, reporting, incident notification and escalation, while management services cover problem management and root cause analysis, configuration management, change and release management, maintenance and updates installation. Prior to providing any of these RIM services, the customer has first to select what services to subscribe to during a procedure that is called "on-boarding". Although the IT management itself is rendered more affordable when provided remotely as a service, the RIM customer on-boarding process particularly is not, taking between one to two months of expensive labor.

The current process for RIM customer onboarding consists of multiple interviews and interactions with customers to 'discover' their IT environment, identify the resources to be managed and guide the enablement of the environment for remote management. This labor intensive approach (measured in weeks) proves to be unscalable when RIM is to be delivered as Management-as-a-Service from an IT infrastructure management cloud. Cloud computing is an emerging paradigm whereby services and computing resources are delivered to customers over the internet (or intranet) from a service provider who owns and operates the cloud. Cloud-based services characteristically can scale up promptly to meet growing demand. The benefit of this will remain unrealized if RIM onboarding takes weeks as is the standard today. Since the duration to traditionally provision resources for new RIM customers is comparable to the current onboarding duration, there is little incentive to motivate change to the current on-boarding approach.

However, RIM's goal for delivery from the cloud is 'on-boarding in minutes', which means radical revision of the current approach. To this end, we have identified the following on-boarding problems: (1)lack of a standardized approach or automation for the on-boarding operation flow, (2) inaccuracies in manually assessing the environment from the customer's descriptions or semi-updated inventory files, (3) missing configuration data (e.g., credentials, directory paths, key performance indicators -- KPIs) necessary to setup the monitoring systems, (4) overhead for the SMB customer who is expected to perform complex configurations in their environment (e.g., VPN setup, monitoring data agent/collector installations), (5) evaluated price is not commensurate with the cost of the service expected to be provided.

There are many managed services providers in the marketplace. Some are local providers, others regional, and still others global. It is largely the regional and global providers that utilize RIM techniques. They are recruiting IT professionals and making them available to client projects through the use of Global Delivery centers. For the client on-boarding process, these IT professionals have to identify the client's IT environment either manually during multiple interviews with the customer or by providing a template for exchanging inventory information (e.g., a spreadsheet). Sometimes the customer may provide one by filling out his own questionnaire. However, a better option is programmatic discovery using dedicated discovery software. The inventory information and additional configuration details are then used to configure the monitoring and management toolset. Manual information gathering methods are notoriously error-prone – some of these errors are caught during the tool configuration step which engenders more interactions with the same customer. Other causes that drive the inaccuracy of the manual environment assessment are existing inventory out of date, incomplete or invalid data, and untracked configuration changes (e.g., for the credentials, directory paths, KPIs), that may jeopardize the quality of the RIM service. We will use the manual data gathering performances in the comparison of our experimental results. When the IT environment discovery is done programmatically, the typical approaches are via stand-alone products, e.g., TADDM [1] or via services that make use of remote product download, e.g., Paglo [2]. Although more accurate in terms of discovery quality compared to the manual approach, the stand-alone products are not suitable for smalland- medium business or SMBs, which have tens to few hundreds of servers. These customers cannot afford nor need sophisticated tools oriented towards large IT enterprises with thousands of IT elements. SMBs prefer to use a streamlined asset discovery service to get the inventory of their IT environment, without the hassle of installing, configuring and managing a discovery product. However, the current remote discovery service providers still require software to be installed and configured by the SMBs in their environment.

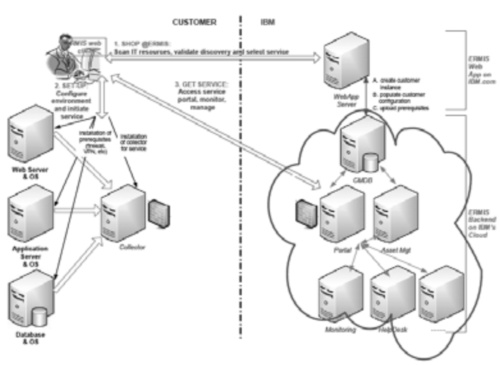

Figure 1. Enhanced on-boarding process for RIM delivered from an IT management cloud.

In [2] the provider offers the discovery tool for free in the context of their monitoring service and the customer has to take care of the discovery tool installation. This is due to the fact that the discovery must take place from a node in the network to be probed, to circumvent firewalls, Network Address Translation (NAT), and other impediments. Many users, especially in the SMB space, prefer not to face the burden of this administrative overhead and may lack the necessary skills required. After the discovery is completed, the IT professionals have typically to manually collect additional configuration details since the discovery provides partial detection of the environment. We call this approach manual onboarding with automated discovery and will compare in Section 5 its characteristics to our approach presented in this paper as well as to the manual approach. Other related solutions that involve remote discovery include [3] which uses a browser to control a discovery process, however, it is unclear whether the provider intends for the "NDM Agent" to be running on the web server or some other machine (or whether it should or could be located in the browser). They also provide "passive discovery" which involves packet sniffing to discover applications running on client machines. JLocator [4] describes a Java applet based network discovery tool. They discover the network topology, but do not examine applications neither services on the identified elements. XAssets [5] is a service comparable to SNAPPiMON [6], where the management tool is also browser based. In [5], the discovery process uses a wide variety of discovery techniques to collect hardware items details, while unrecognized software items lists from customers are sent on a regular basis to the provider staff and these items are manually investigated and added to the discovery database. In [6] the discovery is also a combination of manual discovery, for network and server level items and credentials, and automatic look-up for OS and application configuration. Once the inventory has been discovered and validated, and the additional information on the resources to be managed gathered, the IT professionals proceed to or guide the customer through the enablement of the environment for remote management. This step consists of the installation of data collector or agents into the customer's premises, firewall configuration for siteto- site VPN set-up and NATing of endpoints.

Finally, upon performing all necessary data collection and setup for monitoring and managing the selected items in the customer's environment, the RIM provider prices the offering and starts delivering the IT management service.

Artificial Neural Networks have emerged as as a major paradigm for Data Mining applications. Neural nets have gone through two major development periods -the early 60's and the mid 80's. They were a key development in the field of machine learning. Artificial Neural Networks were inspired by biological findings relating to the behavior of the brain as a network of units called neurons. The human brain is estimated to have around 10 billion neurons each connected on average to 10,000 other neurons. Each neuron receives signals through synapses that control the effects of the signal on the neuron. These synaptic connections are believed to play a key role in the behavior of the brain.

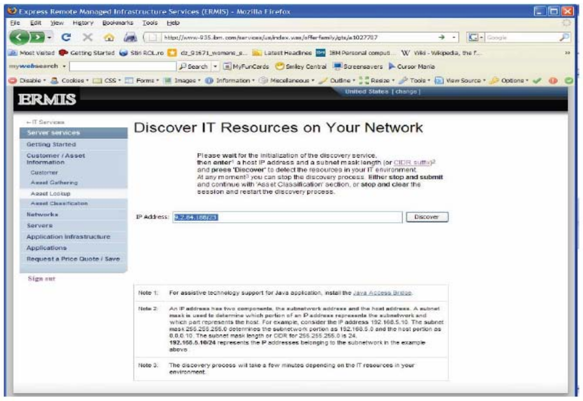

Figure 2. Applet prompts customer for network address range to scan.

Neural networks take a different approach to problem solving than that of conventional computers. Conventional computers use an algorithmic approach i.e. the computer follows a set of instructions in order to solve a problem. Unless the specific steps that the computer needs to follow are known the computer cannot solve the problem. That restricts the problem solving capability of conventional computers to problems that we already understand and know how to solve. But computers would be so much more useful if they could do things that we don't exactly know how to do.

Neural networks process information in a similar way the human brain does. The network is composed of a large number of highly interconnected processing elements(neurones) working in parallel to solve a specific problem. Neural networks learn by example. They cannot be programmed to perform a specific task. The examples must be selected carefully otherwise useful time is wasted or even worse the network might be functioning incorrectly. The disadvantage is that because the network finds out how to solve the problem by itself, its operation can be unpredictable.

On the other hand, conventional computers use a cognitive approach to problem solving; the way the problem is to solved must be known and stated in small unambiguous instructions. These instructions are then converted to a high level language program and then into machine code that the computer can understand. These machines are totally predictable; if anything goes wrong is due to a software or hardware fault.

Neural networks and conventional algorithmic computers are not in competition but complement each other. There are tasks are more suited to an algorithmic approach like arithmetic operations and tasks that are more suited to neural networks. Even more, a large number of tasks, require systems that use a combination of the two approaches (normally a conventional computer is used to supervise the neural network) in order to perform at maximum efficiency.

5. Methodology

Matlab will be used as the simulation tool. Attempt will be made to build a classifier that can predict the success or failure of implementation of RIM. Six parameters of the RIM will be considered. Neural networks have proved themselves as proficient classifiers and are particularly well suited for addressing non-linear problems. Given the non-linear nature of real world phenomena, like predicting success of RIM, neural networks is certainly a good candidate for solving the problem. The six characterstics will act as inputs to a neural network and the prediction of seccess will be the target. Given an input, which constitutes the six measured values for the parameters of the matrix, the neural network is expected to identify if the RIM process will produce success or not. This is achieved by presenting previously recorded RIM parameters to a neural network and then tuning it to produce the desired target outputs. This process is called neural network training. The samples will be divided into training, validation and test sets. The training set is used to teach the network. Training continues as long as the network continues improving on the validation set. The test set provides a completely independent measure of network accuracy. The trained neural network will be tested with the testing samples. The network response will be compared against the desired target response to build the classification matrix which will provide a comprehensive picture of a system performance.

The training data

The training data set includes a number of cases, each containing values for a range of input and output variables. The first decisions you will need to make are: which variables to use, and how many (and which) cases to gather.

The choice of variables (at least initially) is guided by intuition. Expertise in the problem domain will give you some idea of which input variables are likely to be influential. As a first pass, you should include any variables that you think could have an influence - part of the design process will be to whittle this set down.

Neural networks process numeric data in a fairly limited range. This presents a problem if data is in an unusual range, if there is missing data, or if data is non-numeric. Fortunately, there are methods to deal with each of these problems. Numeric data is scaled into an appropriate range for the network, and missing values can be substituted for using the mean value (or other statistic) of that variable across the other available training cases.

Handling non-numeric data is more difficult. The most common form of non-numeric data consists of nominal-value variables such as Outcomer= {Success, Failure}. Nominal-valued variables can be represented numerically. However, neural networks do not tend to perform well with nominal variables that have a large number of possible values.

For example, consider a neural network being trained to estimate the value of houses. The price of houses depends critically on the area of a city in which they are located. A particular city might be subdivided into dozens of named locations, and so it might seem natural to use a nominal-valued variable representing these locations. Unfortunately, it would be very difficult to train a neural network under these circumstances, and a more credible approach would be to assign ratings (based on expert knowledge) to each area; for example, you might assign ratings for the quality of local schools, convenient access to leisure facilities, etc. Other kinds of non-numeric data must either be converted to numeric form, or discarded. Dates and times, if important, can be converted to an offset value from a starting date/time. Currency values can easily be converted. Unconstrained text fields (such as names) cannot be handled and should be discarded.

The number of cases required for neural network training frequently presents difficulties. There are some heuristic guidelines, which relate the number of cases needed to the size of the network (the simplest of these says that there should be ten times as many cases as connections in the network). Actually, the number needed is also related to the (unknown) complexity of the underlying function which the network is trying to model, and to the variance of the additive noise. As the number of variables increases, the number of cases required increases nonlinearly, so that with even a fairly small number of variables (perhaps fifty or less) a huge number of cases are required. This problem is known as "the curse of dimensionality," and is discussed further later.

For most practical problem domains, the number of cases required will be hundreds or thousands. For very complex problems more may be required, but it would be a rare (even trivial) problem which required less than a hundred cases. If your data is sparser than this, you really don't have enough information to train a network, and the best you can do is probably to fit a linear model. If you have a larger, but still restricted, data set, you can compensate to some extent by forming an ensemble of networks, each trained using a different resampling of the available data, and then average across the predictions of the networks in the ensemble.

Many practical problems suffer from data that is unreliable: some variables may be corrupted by noise, or values may be missing altogether. Neural networks are also noise tolerant. However, there is a limit to this tolerance; if there are occasional outliers far outside the range of normal values for a variable, they may bias the training. The best approach to such outliers is to identify and remove them (either discarding the case, or converting the outlier into a missing value). If outliers are difficult to detect, a city block error function may be used, but this outlier-tolerant training is generally less effective than the standard approach.

Pre- and Post-processing

All neural networks take numeric input and produce numeric output. The transfer function of a unit is typically chosen so that it can accept input in any range, and produces output in a strictly limited range (it has a squashing effect). Although the input can be in any range, there is a saturation effect so that the unit is only sensitive to inputs within a fairly limited range. The illustration below shows one of the most common transfer functions, the logistic function (also sometimes referred to as the sigmoid function, although strictly speaking it is only one example of a sigmoid - S-shaped - function). In this case, the output is in the range (0,1), and the input is sensitive in a range not much larger than (-1,+1). The function is also smooth and easily differentiable, facts that are critical in allowing the network training algorithms to operate.

The limited numeric response range, together with the fact that information has to be in numeric form, implies that neural solutions require preprocessing and post-processing stages to be used in real applications. Two issues will need to be addressed:

Scaling. Numeric values have to be scaled into a range that is appropriate for the network. Typically, raw variable values are scaled linearly. In some circumstances, non-linear scaling may be appropriate (for example, if you know that a variable is exponentially distributed, you might take the logarithm). Non-linear scaling is not supported in ST Neural Networks. Instead, you should scale the variable using STATISTICA's data transformation facilities before transferring the data to ST Neural Networks.

Nominal variables. Nominal variables may be two-state (e.g., Outcome ={Success ,Failure }) or many-state (i.e., more than two states). A two-state nominal variable is easily represented by transformation into a numeric value (e.g., Success =0, Failure =1). Many-state nominal variables are more difficult to handle. They can be represented using an ordinal encoding but this implies a (probably) false ordering on the nominal. A better approach, known as one-of-N encoding, is to use a number of numeric variables to represent the single nominal variable. The number of numeric variables equals the number of possible values; one of the N variables is set, and the others cleared. ST Neural Networks has facilities to convert both two-state and many-state nominal variables for use in the neural network. Unfortunately, a nominal variable with a large number of states would require a prohibitive number of numeric variables for one-of-N encoding, driving up the network size and making training difficult. In such a case it is possible (although unsatisfactory) to model the nominal variable using a single numeric ordinal; a better approach is to look for a different way to represent the information. In classification, the objective is to determine to which of a number of discrete classes a given input case belongs. The most common classification tasks are two-state, although many-state tasks are also not unknown. Neural networks can actually perform a number of classification tasks at once, although commonly each network performs only one. In this case the network will have a single output variable.

Multilayer Perceptrons is the type of network in which the units each perform a biased weighted sum of their inputs and pass this activation level through a transfer function to produce their output, and the units are arranged in a layered feedforward topology. The network thus has a simple interpretation as a form of input-output model, with the weights and thresholds (biases) the free parameters of the model. Such networks can model functions of almost arbitrary complexity, with the number of layers, and the number of units in each layer, determining the function complexity. Important issues in Multilayer Perceptrons (MLP) design include specification of the number of hidden layers and the number of units in these layers.

The number of input and output units is defined by the problem (there may be some uncertainty about precisely which inputs to use, a point to which we will return later. However, for the moment we will assume that the input variables are intuitively selected and are all meaningful). The number of hidden units to use is far from clear. As good a starting point as any is to use one hidden layer, with the number of units equal to half the sum of the number of input and output units.

We are going to use unary encoding in this simulation to perform symbol translation. The first six columns of data will represent the emails characterstics. The 7th column represents whether the RIM is successful or not. This data will be randomly generated. The next step will be to preprocess the data into a form that can be used with a neural network. The next step is to create a neural network that will learn to identify if the RIMprocess will cause improvement or not.

The assumed samples will be automatically divided into training, validation and test sets. The training set will be used to teach the network. Training will continue long as the network continues improving on the validation set. The test set will provide a completely independent measure of neural network accuracy to detect success. The trained neural network will be tested with the testing samples. This will give a sense of how well the network will do when applied to data from the real world.

6. Conclusion

Due to vast changes in IT Infrastructure & technologies, ANN is useful to predict success or failure of RIM.

Neural networks have proved themselves as proficient classifiers and are particularly well suited for addressing non-linear problems. Given the non-linear nature of real world phenomena, like predicting success of RIM, neural networks is certainly a good candidate for solving the problem.

7. References

1. Manager(TADDM), http://www.ibm.com/software/tivoli/products/taddm
2. Paglo,http://www.paglo.com/opensource/paglocrawler
3. Design of Hybrid Network Discovery Module for Detecting Client Applications an ActiveX Controls, http://www.springerlink.com/content/y51p5g76k25578g1/
4. JLocator, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3 9.5030
5. XAssets, http://www.xassets.com/discovery.aspx
6. IBM SNAPPiMON, http://www.snappimon.com/
7. Head, M.R.; Sailer, A.; Shaikh, H.; Viswanathan, M.; , "Taking IT Management Services to a Cloud," Cloud Computing, 2009. CLOUD '09. IEEE International Conference on , vol., no., pp.175-182, 21-25 Sept. 2009
8. Aljunaid, A.B.; AbuElMaaly, I.; Sagahyroon, A.; , "Using ANN To Predict The Best HUB Location," Circuits and Systems, 2006. APCCAS 2006. IEEE Asia Pacific Conference on , vol., no., pp.317-320, 4-7 Dec. 2006
9. Nmap, http://nmap.org/
10. Gartner Dataquest, August 2006, http://www.gartner.com/it/ products/research/dataquest.jsp