Causal Support as Trust and Reputation Mechanism

Lourembam Suraj Singh
System Analyst, Dept. of Computer Centre,
Mizoram University, Mizoram, India

Sarangthem Ibotombi Singh
Assistant Professor, Dept. of CSE,
Tezpur University, Assam, India

Abstract

With the advancement of technologies in the field of web technology, trust and reputation mechanism have become indispensable with our day-to-day lives. Every day, we refer to the reputation or the trust values while making decisions such as online marketing, downloading movies and documents etc. Thus, trust and reputation mechanisms play a very important role in shaping our day-to-day decisions. In this paper, we present a trust and reputation mechanism based on 'Causal Support' and check if incorporating such a mechanism really helps in making better decisions. Causal support describes how human beings draw decisions based on the available data. It gives an idea of how a potential cause is attributed as the cause of an event. We also try to reflect the closeness of the relationships between the clients by considering the fact that we give more emphasis on the recommendations from our trusted friends than the recommendations from the unknown source.

Key Words: Trust, Reputation, Causal Support, Causal Model, Cause and Effect.

1. Introduction

In any social network, trust plays an important role. Without trust, it will be almost impossible for any social network to function. Let us take the example of purchasing an item from an online site. Before making any purchase we assume that the site sells genuine products; and items will be delivered as expected. This kind of trust can be inferred by our past experience i.e. if we have already purchased items from the site. If we are ignorant of the site, we can use the reputation of the site to shape our decision. Nowadays, there are many large scale distributed systems such as peer to peer networks, web services, e-commerce etc. These systems consist of a large number of agents. In order for these systems to function properly, the agents must be interacting with each other in an efficient and reliable manner; and these interactions depend on the interpersonal trust among the agents. But trust cannot be assigned vaguely as some agents are kind and benevolent providing good and reliable services while some agents are bad agents that try to harm other agents. So, the mechanism of assigning trust and reputation values becomes very important in this kind of scenario where traditional security measures fail to account for [10].

The rest of the paper is organized as follows—section 2 explains the causal model and causal graphical model, section 3 gives an idea of the causal support, we propose the working of the model in section 4, section 5 gives the experiment and design while we give conclusion in section 6.

2. Causal graphical model

Causal graphical model provides formalism for learning and reasoning about causal relationship [9] [11]. In this causal model, events are viewed as cause and effects. It is based on the assumption that cause precedes an effect i.e. effect occurs because of the occurrence of a cause.

In the causal graphical model, all the related events are represented as nodes. If an event produces a cause then a directed edge is drawn from the cause to effect. Thus, a directed edge between the nodes indicates that a causal relationship exists between the corresponding nodes. For example, let us consider a case. When you reached home after work and enter your room, you find out that your cooler was not working. Then there are two possible causes—

- The switch is broken
- There is no power supply.

Here the states of the cooler, switch and power supply are the variables being considered, so each is assigned a node. The links from switch and supply to cooler represent an indication that the states of switch and supply are direct causes of the cooler not working. This can be represented in the graphical model as—

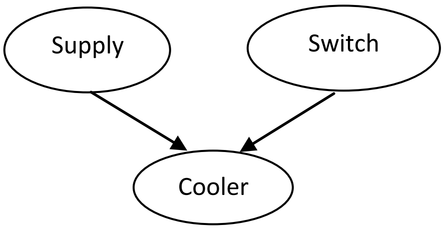


Figure 1: Causal Graphical model showing the relationship between cooler and occurrence of supply cut or broken switch or both.

3. Causal Support

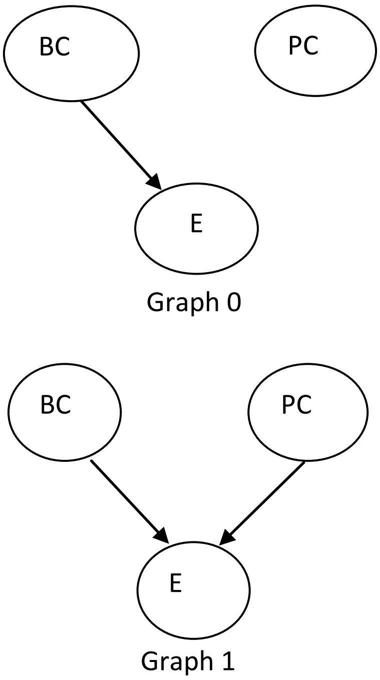


Figure 2: Graphical representations of three variables BC, PC and E.

The above figure shows graphical representations over three variables— an effect E, a potential cause PC, and a background cause BC. Each graph represents a hypothesis about the causal relations that could hold among these variables. In Graph 0, BC causes E, but PC has no relationship to either BC or E. In Graph 1, both BC and PC cause E.

The graph structure in causal graphical model simply says whether there is a relationship between the cause and effect. For example, if there is a relationship between PC and E, then the graph will be G_1 otherwise the graph will be G_0. In many cases we are interested in knowing whether a given set of data 'D' is better accounted by G_0 or G_1. This measure can be provided by Causal Support [7] and is given as the evidence that the data 'D' provide in favor of G_1 over G_0. Mathematically, it can be presented as—

$$Support = log\frac{P(D|Graph_1)}{P(D|Graph_0)} \qquad (1)$$

Mechanism to compute causal support is given in detail in [7]. We will be using the mechanism given in [7] to compute the causal support.

4. Proposed Model

4.1. Scenario

We design a network where some nodes acts as clients while some nodes acts as the service provider. Clients seek services from the service provider. For simplicity, we assume that all the service providers can provide same type of services and a client can interact with any other client or any service provider at any time and every client is truthful in providing the recommendation value. Every client develops two kind of trust – trust in the service providers' competence in providing the service and trust in other client's reliability in providing reliable recommendations.

4.2. Working model

For our model, we assume that the trust level of a service provider can be divided into four levels—

i) Very Low (VL)
ii) Low (L)
iii) Medium (M)
iv) High (H)

We also assume that at any given time 't', the trust level of the service provider can be at any of the four possible states. We then represent these trust levels as nodes and given the trust level at time 't' and set of data 'D', we calculate the evidence that the service provider will provide satisfactory services at time 't+1'.

The transition of the state at timestamp 't' to timestamp 't+1' is then represented as a causal graphical model assuming that the transition is likely to take place considering the state 't' and a set of data 'D'. For example, let the state of a service provider be 'A' at time 't' and at time 't+1', its state is 'B'. This can be graphically represented as—

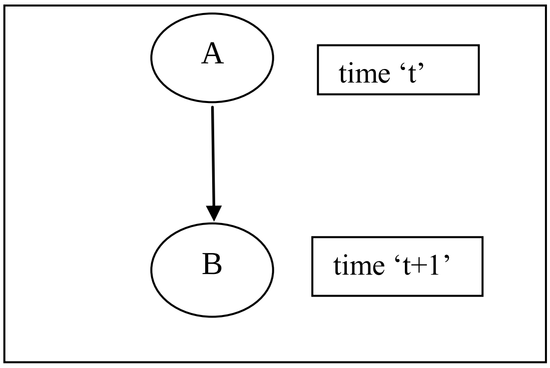


Figure 3: Graphically representing the transition of state at timestamp 't' to timestamp 't+1'.

As there are four trust levels, the service provider can make transition to any of the possible four states. Graphically it can be represented as—

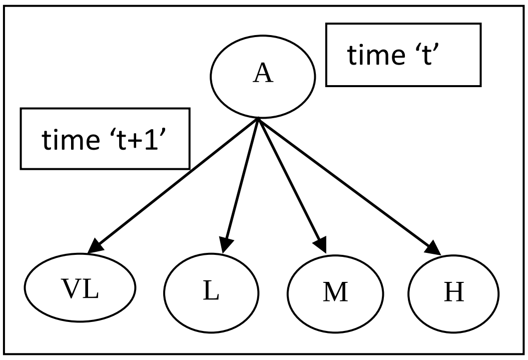


Figure 4: Graphically representing all the possible transitions

In order to separate the good services from bad services, we define the transitions into states {VL, L} as bad services and the transitions into states {M, H} as good services. For, example, if 'A' and 'B' represents the states of the service provider at timestamp 't' and 't+1' respectively, then an interaction is said to be a successful interaction if the transition is A→B such that A∈{VL, L, M, H} and B∈ {M, H}. So, we say a transition A→B at timestamp 't' to timestamp 't+1' is indicating a bad service if B∈ {VL, L} and if B∈ {M, H}, we say that the transition is indicating a good service. Thus, every transition is classified as belonging to either set S_1 or S_2 where

S_1 ={(VL→VL), (VL→L), (L→VL), (L→L), (M→VL), (M→L), (H→VL), (H→L)}
={A→B such that A∈{VL, L, M, H} and B∈{VL, L}}

S_2 ={(VL→M), (VL→H), (L→M), (L→H), (M→M), (M→H), (H→M), (H→H)}
={A→B such that A∈{VL, L, M, H} and B∈{M, H}}

So, if any transition belongs to set S_1 then we infer that the interaction was unsuccessful otherwise we infer that the interaction was successful.

In addition to maintaining the previous interacting state of every server, each client also maintains a contingency table where it stores all the past interactions. The contingency table is organized as a matrix of size 4x4 and is represented as—

TABLE 1: CONTINGENCY TABLE

n(VL→VL)	*n(VL→L)*	*n(VL→M)*	*n(VL→H)*
n(L→VL)	*n(L→L)*	*n(L→M)*	*n(L→H)*
n(M→VL)	*n(M→L)*	*n(M→M)*	*n(M→H)*
n(H→VL)	*n(H→L)*	*n(M→L)*	*n(H→H)*

4.3. Calculating the evidence value

We assume that all the servers provide the same type of services and a client can interact with any server at any time. So, the client has to select the most reliable server. The client can select such a server based on the trust and the reputation values. When a client wants to use the service provided by the server, it arranges the servers based on the trust and reputation values and select the server with the most reliable value. By this mechanism, the client improves the chance of having a successful interaction.

In order to predict the best reliable server, we use the framework defined in the figure 2. To understand the working of the model properly, let us assume at time 't', the state of the server 'j' is M and the data available in the contingency table is represented by the set 'D'. With the help of this model, we will find the evidence of a successful interaction at time 't+1'. We discuss two approaches as described below.

4.3.1. Model 1:

We assume that BC={VL, L, H}; PC = {M} and E = {M, H}

We are taking E as {M, H} because we are interested only in high trust levels. Graphically it can be represented as—

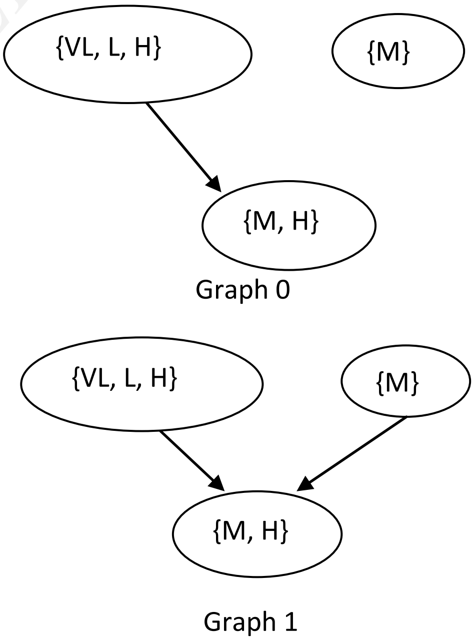


Figure 5: Graphically representing Causal Support model 1

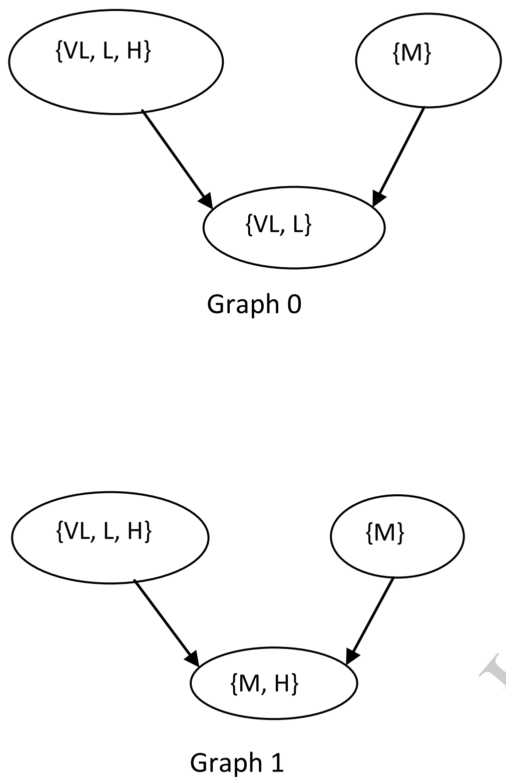
We calculate P(D|Graph 1) and P(D|Graph 0) as given by equation 1 and find out causal support as—

$$support = log\frac{P(D|Graph\ 1)}{P(D|Graph\ 0)}$$

This support is then used to find the trust level of the server at time ‘t+1’.

4.3.2. Model 2:

Let us assume that the trust level of a server at time ‘t’ is ‘M’. Now, with the help of this model we will try to evaluate what will be the trust level at time ‘t+1’.



Graph 0

Graph 1

Figure 6: Graphically representing Causal Support model 2

We calculate P(D|Graph 1) and P(D|Graph 0) as given by equation 1 and find out causal support as—

$$support = log\frac{P(D|Graph\ 1)}{P(D|Graph\ 0)}$$

This support is then used to find the trust level of the server at time ‘t+1’.

If Agent *‘i’* wants to access services from the servers, *‘i’* has to first find out the last interacting state of each server *‘j’*, before each interaction. Each agent stores the last interacting state of each of the server. After getting the last interacting state, agent *‘i’* then looks up in its contingency table and calculates trusts value for going to {M, H} level. If $t1_{ij}$ and $t2_{ij}$ denotes the trust that agent *‘i’* has in the server *‘j’* for getting good services obtained from model 1 and model 2 respectively, then trust values can be calculated as—

Case a: If Agent *‘i’* has not interacted with server *‘j’*:
In this case, $t1_{ij}=0$ and $t2_{ij} = 0$.

Case b: If Agent *‘i’* has interacted with server *‘j’*:
In this case $t1_{ij}$ *and* $t2_{ij}$ is given as—

$$t1_{ij} = \begin{cases} -\left|log\frac{P(D|Graph\ 1)}{P(D|Graph\ 0)}\right|; if\ P(D|Graph\ 1) < 0 \\ \left|log\frac{P(D|Graph\ 1)}{P(D|Graph\ 0)}\right|; otherwise \end{cases}; \quad (2)$$

Where P(D|Graph 1) and P(D|Graph 0) are obtained as in model 1.

$$t2_{ij} = \begin{cases} -\left|log\frac{P(D|Graph\ 1)}{P(D|Graph\ 0)}\right|; if\ P(D|Graph\ 1) < 0 \\ \left|log\frac{P(D|Graph\ 1)}{P(D|Graph\ 0)}\right|; otherwise \end{cases}; \quad (3)$$

Where P(D|Graph 1) and P(D|Graph 0) are obtained as in model 2.

Mathematically it can be expressed as—

$$t1_{ij} = \begin{cases} 0; if\ i\ has\ not\ interacted\ with\ j \\ t1_{ij} as\ obtained\ from\ (2); otherwise \end{cases} \quad (4)$$

$$t2_{ij} = \begin{cases} 0; if\ i\ has\ not\ interacted\ with\ j \\ t2_{ij} as\ obtained\ from\ (3); otherwise \end{cases} \quad (5)$$

Agent *‘i’* then sends reputation requests to all the other agents for server *‘j’*. On receiving the reputation request, each agent *‘k’*, then finds out the last interacting state with server *‘j’* (say *Y*). . If $t1_{kj}$ and $t2_{kj}$ denotes the trust that agent *‘k’* has in the server *‘j’* for getting good services obtained from model 1 and model 2 respectively, then—

$$t1_{kj} = \begin{cases} 0; if\ k\ has\ not\ interacted\ with\ j \\ t1_{kj} as\ obtained\ from\ (4); otherwise \end{cases} \quad (6)$$

$$t2_{kj} = \begin{cases} 0; if\ k\ has\ not\ interacted\ with\ j \\ t2_{kj} as\ obtained\ from\ (5); otherwise \end{cases} \quad (7)$$

Thus, total trust that agent *‘i’* has in the j^{th} server is given by—

$$T1_{ij} = t1_{ij} + \frac{\sum_{k=1}^{n} W_{ik} * t1_{kj}}{\sum_{k=1}^{n} W_{ik}} \quad (8)$$

$$T2_{ij} = t2_{ij} + \frac{\sum_{k=1}^{n} W_{ik} * t2_{kj}}{\sum_{k=1}^{n} W_{ik}} \quad (9)$$

Where W_{ik} is the weight that agent *‘i’* has for the k^{th} agent. $t1_{kj}$ is the trust that k^{th} agent has in the server *‘j’* (obtained by (6)) . $t1_{ij}$ is the trust that agent *‘i’* has in the j^{th} server (obtained by (4)). $T1_{ij}$ is the total trust that agent ‘i’ has for the j^{th}

server for model 1. $t2_{kj}$ is the trust that k^{th} agent has in the server *'j'* (obtained by (7)) . $t2_{ij}$ is the trust that agent *'i'* has in the j^{th} server (obtained by (5)). $T2_{ij}$ is the total trust that agent 'i' has for the j^{th} server for model 2. Thus agent *'i'* calculates trust values for all *j*, i.e. for all servers. It then selects the server with the largest value of trust and interacts with it.

4.4. Evaluating an interaction:

In order to evaluate the result of an interaction, we employ the same method as described in [13]. In this approach, every server is given a value (say 'x') before every interaction and based on the type of the server; the server returns a value (say 'y') to the interacting agent. There are four types of the server— Honest Server, Dishonest Server, Random Server and Neutral Server. After each interaction, the client then looks up for the actual value 'x' and depending on the difference of the values 'x' and 'y', the trust level of the interaction with the server is calculated, which in turn is used to determine whether the interaction was successful or unsuccessful. The evaluation mechanism in detail can be obtained from [13].

4.5. Handling other agents:

One important feature of the reputation is that, agents tend to give more emphasis on the recommendations from its cliques than the recommendations from the unknown source. To account for this idea, each agent assigns weight to each recommendations from other agents based on the past history of the recommendations provided by each agent. If the recommendation provided by the agent is matching with the outcome, then the weight of the agent is increased otherwise the weight is decreased. The detail procedure of handling other agents' recommendations can be obtained from [13].

5. Experiment:

We design an experimental environment consisting of 50 agents and 10 service providers or servers. We made assumption that all the servers can provide same type of services and any agent can interact with any server or any other agent at any time. The weight of each agent is initialized at the beginning of each simulation by assuming that some agents have already made interactions with some other agents. For example, if agent 'i' has made interactions with agent 'j' then we initialize 'w_{ik}' = 1 otherwise we initialize 'w_{ik}' = 0.5. We run the simulations for 100 numbers of times and each configuration is run for 10 times. After getting the result, we plot the graphs taking means of each result.

We design the first experiment to see if trust and reputation model based on causal support helps in selecting servers with better reliability. Thus, we design three models—

- Trust and reputation model based on model 1 [section 4.3.1], we call it model M11
- Trust and reputation model based on model 1 [section 4.3.2] we call it model M12
- A simple model that selects server randomly, we call it Random Model.

After the experiment, we plot a graph of the models with the percentage of successful number of interactions against the total number of interactions.

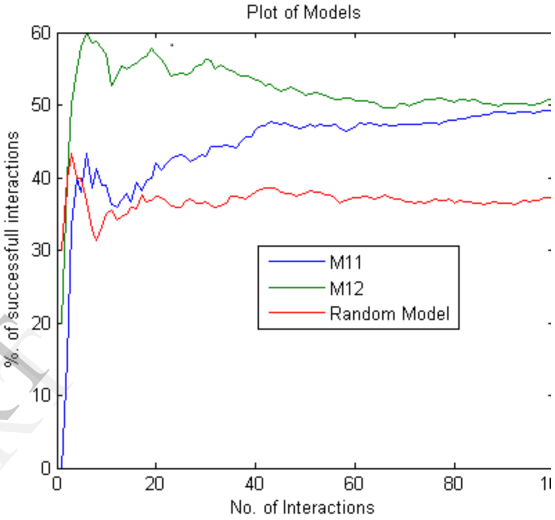


Figure 7: Trust and reputation models v/s agent selecting server in a random manner.

From the above graph, it is clear that the agents employing the trust and reputation mechanism performs better than the agents that do not employ such mechanism.

We design the second experiment in such a way to see if the reputation method i.e. taking the recommendations from other agents really improves the selection of the server with better reliability. Thus, we design four models—

- Trust and reputation model based on model 1 [section 4.3.1] we call it model M21
- Trust and reputation model based on model 2 [section 4.3.2] we call it model M22
- Trust model without reputation based on model 1 [section 4.3.1] we call it model M23
- Trust model without reputation based on model 2 [section 4.3.2] we call it model M24

The graph plotted by taking percentage of the successful interactions against the total number of interactions is shown below.

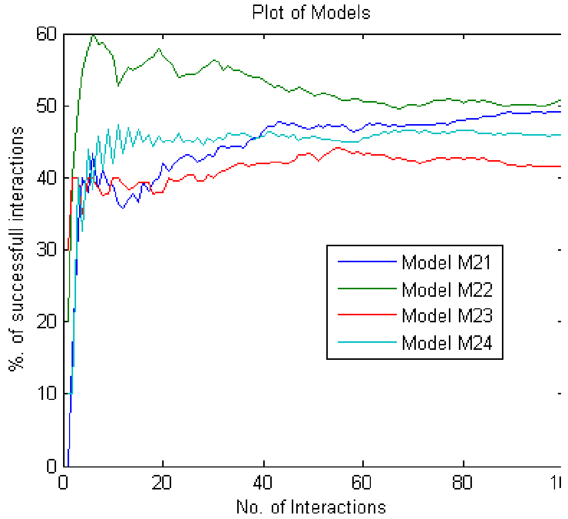


Figure 8: Trust and Reputation models v/s Trust model without reputation.

From the above graph, we see that the agents employing reputation mechanisms perform better than the agents that do not employ reputation mechanism.

6. Conclusions:

From the above experiment, we can conclude that trust and reputation model really helps in selecting server with better reliability. We can also deduce that the reputation mechanism i.e. taking help from other agents helps in making better decisions.

References:

[1] H.M. Jenkins, W.C. Ward, "Judgment of contingency between responses and outcomes", Psychological Monographs, 79, (1965).

[2] L. G. Allan, "A note on measurement of contingency between two binary variables in judgment tasks", Bulletin of the Psychonomic Society, 15 (1980), 147-149.

[3] L. G. Allan, "Human contingency judgments: Rule based or associative?", Psychological Bulletin, 114(1993), 435-448.

[4] L. G. Allan, H.M. Jenkins, "The effect of representations of binary variables on judgment of influence", Learning and Motivation 14(1983), 381-405.

[5] P.W. Cheng, K.J Holyoak, "Complex adaptive systems as intuitive statisticians: causality, contingency, and prediction" Comparative approaches to Cognitive Science, H.L Roitblat and J.-A Meyer (Edns), Cambridge, MA:MIT Press, (1995), 271-302.

[6] E.R. Melz, P.W. Cheng, K.J. Holyoak, and M.R. Waldmann, "Cue competition in humane categorization : Contingency or the Rescorla-Wagner rule, comments on Shanks, Journal of Experimental Psychology: Learning, Memory, and Cognition, 19(1991), 1398-1410.

[7] T.L. Griffiths, J.B. Tenenbaum, "Structure and strength in causal induction", Cognitive Psychology 51(2005), 334-384.

[8] P. Cheng, "From co variation to causation" A causal power theory", Psychological Review 104(1997), 367-405.

[9] P. Spirtes, C. Glymour, and R. Schienes, "Causation prediction and search" Springer-Verlag, New York, (1993).

[10] Y. Wang, J. Vassileva, "Bayesian Network-Based Trust Model", Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence (WI 2003), Halifax, Canada, 13-16 October 2003, pp 372-378.

[11] J. Pearl, Causality: Models, reasoning and inference. Cambridge, UK: Cambridge University Press (2000).

[12] C. Kemp, N. D. Goodman, J. B. Tenenbaum, "Learning to Learn Causal Models", Cognitive Science (2010), 1-59.

[13] L. Suraj Singh, H. R. Singh, A. Neelima, S. I. Singh, "Trust and Reputation Model based on Causal Induction Method", International Journal of Computer Science and Technology, Vol-3, Issue-3, July-Sep.(2012),pp 11-16.

[14] C. G. Lucas, T.L. Griffiths, "Learning the Form of Causal Relationships Using Heirarchical Bayesian Models", Cognitive Science 34 (2010) 113-147.