**Converting Graphic Relationships into Conditional Probabilities in Bayesian Network**

Loc Nguyen

Sunflower Soft Company, An Giang, Vietnam

**Abstract**

Bayesian network is a powerful mathematical tool for prediction and diagnosis applications. A large Bayesian network can be constituted of many simple networks which in turn are constructed from simple graphs. A simple graph consists of one child node and many parent nodes. The strength of each relationship between a child node and a parent node is quantified by a weight and all relationships share the same semantics such as prerequisite, diagnostic, and aggregation. The research focuses on converting graphic relationships into conditional probabilities in order to construct a simple Bayesian network from a graph. Diagnostic relationship is the main research object, in which sufficient diagnostic proposition is proposed for validating diagnostic relationship. Relationship conversion is adhered to logic gates such as AND, OR, and XOR, which is essential feature of the research.

**Keywords**: diagnostic relationship, Bayesian network, transformation coefficient

**1. Introduction**

Bayesian network (BN) is a directed acyclic graph (DAG) consisting of a set of nodes and a set of arcs. Each node is a random variable. Each arc represents a relationship between two nodes. The strength of relationship in a graph can be quantified by a number called *weight*. There are some important relationships such as prerequisite, diagnostic, and aggregation. The difference between BN and normal graph is that the strength of every relationship in BN is represented by a conditional probability table (CPT) whose entries are conditional probabilities of a child node given parent nodes. There are two main approaches to construct a BN:

* The first approach aims to learn BN from training data by learning machine algorithms.
* The second approach is that experts define some graph patterns according to specific relationships and then, BN is constructed based on such patterns along with determined CPT (s).

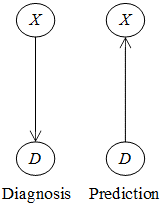
This research focuses on the second approach in which relationships are converted into CPT (s). Essentially, relationship conversion aims to determined conditional probabilities based on weights and meanings of relationships. We will have different ways to convert graphic weights into CPT (s) for different relationships. It is impossible to convert all relationships but some of them such as diagnostic, aggregation, and prerequisite are mandatory ones that we must specify as computable CPT (s) of BN. Especially, these relationships are adhered to logic X-gates (Wikipedia, 2016) such as AND-gate, OR-gate, and SIGMA-gate. The X-gate inference in this research is derived and inspired from noisy OR-gate described in the book “Learning Bayesian Networks” by author (Neapolitan, 2003, pp. 157-159). Authors (Díez & Druzdzel, 2007) also researched OR/MAX, AND/MIN, noisy XOR inferences but they focused on canonical models, deterministic models, ICI models whereas I focus on logic gate and graphic relationships. So their research is different from mine but we share the same result that is AND-gate model. In general, my research focuses on applied probability adhered to Bayesian network, logic gates, and Bayesian user modeling (Millán & Pérez-de-la-Cruz, A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation, 2002). The scientific results are shared with authors Eva Millán and José Luis Pérez-de-la-Cruz (Millán & Pérez-de-la-Cruz, A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation, 2002).

Factor graph (Wikipedia, Factor graph, 2015) represents factorization of a global function into many partial functions. If joint distribution of BN is considered as the global function and CPT (s) are considered as partial functions, the sum-product algorithm (Kschischang, Frey, & Loeliger, 2001) of factor graph is applied into calculating posterior probabilities of variables in BN. Pearl’s propagation algorithm (Pearl, 1986) is very successful in BN inference. The application of factor graph into BN is only realized if all CPT (s) of BN are already determined whereas this research focuses on defining such CPT (s) firstly. I did not use factor graph for constructing BN. The concept “X-gate inference” only implies how to convert simple graph into BN. However the arrange sum with a fixed variable mentioned in this research is the “*not-sum*” (Kschischang, Frey, & Loeliger, 2001, p. 499) of factor graph. Essentially, X-gate probability shown in formula 3.4 is as same as *λ* message in the Pearl’s algorithm (Kschischang, Frey, & Loeliger, 2001, p. 518) but I use the most basic way to prove the X-gate probability.

As default, the research is applied in learning context in which BN is used to assess students’ knowledge. Evidences are tests, exams, exercises, etc. and hypotheses are learning concepts, knowledge items, etc. Note that diagnostic relationship is very important to Bayesian evaluation in learning context because it is used to evaluate student’s mastery of concepts (knowledge items) over entire BN. As a convention, all equations are called formulas having particular titles. Now we start relationship conversion with a research on diagnostic relationship in the next section.

**2. Diagnostic relationship**

In some opinions like mine, the diagnostic relationship should be from hypothesis to evidence. For example, disease is hypothesis and symptom is evidence. The symptom must be conditionally dependent on disease. Given a symptom, calculating the posterior probability of disease is essentially to diagnose likelihood of such disease (Millán, Loboda, & Pérez-de-la-Cruz, Bayesian networks for student model engineering, 2010, p. 1666). Inversely, the arc from evidence to hypothesis implies prediction where evidence and hypothesis represent observation and event, respectively. Given an observation, calculating the posterior probability of the event is essentially to predict/assert such event (Millán, Loboda, & Pérez-de-la-Cruz, Bayesian networks for student model engineering, 2010, p. 1666). Figure 2.1 shows diagnosis and prediction.



**Figure 2.1.** Diagnosis and prediction with hypothesis *X* and evidence *D*

The weight *w* of relationship between *X* and *D* is 1. Figure 2.1 depicts simplest graph with two random variables. We need to convert diagnostic relationship into conditional probabilities in order to construct a simplest BN from the simplest graph. Note that hypothesis is binary but evidence can be numerical. In learning context, evidence *D* can be test, exam, exercise, etc. The conditional probability of *D* given *X* (likelihood function) is *P*(*D*|*X*). The posterior probability of *X* is *P*(*X*|*D*), which is used to evaluate student’s mastery over concept (hypothesis) *X* given evidence *D*. Formula 2.1 specifies CPT of *D* when *D* is binary (0 and 1):

**Formula 2.1.** CPT of binary evidence of diagnostic relationship

Formula 2.1 is our first relationship conversion. It implies

Evidence *D* can be used to diagnose hypothesis *X* if the so-called *sufficient* *diagnostic proposition* is satisfied, as seen in table 2.1.

|  |
| --- |
| *D* is equivalent to *X* in diagnostic relationship if *P*(*X*|*D*) = *kP*(*D*|*X*) given uniform distribution of *X* and the *transformation coefficient k* is independent from *D*. In other words, *k* is constant with regards to *D* andso *D* is called *sufficient evidence*. |

**Table 2.1.** Sufficient diagnostic proposition

The concept of sufficient evidence is borrowed from the concept of sufficient statistics and it is inspired from equivalence of variables *T* and *T’* in the research (Millán & Pérez-de-la-Cruz, A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation, 2002, pp. 292-295). The proposition can be restated that evidence *D* is only used to assess hypotheses if it is sufficient evidence. As a convention, the proposition is called *diagnostic condition* and hypotheses have uniform distribution. The assumption of hypothetic uniform distribution (*P*(*X*=1) = *P*(*X*=0)) implies that we cannot assert whether or not given hypothesis is true before we observe its evidence.

In learning context, *D* can be totally used to assess student’s mastery of *X* if diagnostic condition is satisfied. Derived from such condition, formula 2.2 specifies transformation coefficient *k* given uniform distribution of *X*.

**Formula 2.2.** Transformation coefficient given uniform distribution of *X*

We need to prove that formula 2.1 satisfies diagnostic condition. Suppose the prior probability of *X* is uniform:

We have:

It is easy to infer that the transformation coefficient *k* is 1 if *D* is binary. In practice, evidence *D* is often a test whose grade ranges within an interval {0, 1, 2,…, *η*}. Formula 2.3 specifies CPT of *D* in this case:

Where,

**Formula 2.3.** CPT of integer evidence ranging in {0, 1, 2,…, *η*} of diagnostic relationship

As a convention, . Formula 2.3 implies that if student has mastered concept (*X*=1), the probability that she/he completes the exercise/test *D* is proportional to her/his mark on *D* . We also have:

We need to prove that formula 2.3 satisfies diagnostic condition. Suppose the prior probability of *X* is uniform:

The assumption of prior uniform distribution of *X* implies that we do not determine if student has mastered *X* yet. Similarly, we have:

So, the transformation coefficient *k* is if *D* ranges in {0, 1, 2,…, *η*}.

In the most general case, discrete evidence *D* ranges within an arbitrary integer interval . In other words, *D* is bounded integer variable whose lower bound and upper bound are *a* and *b*, respectively. Formula 2.4 specifies CPT of *D* where .

Where,

**Formula 2.4.** CPT of general discrete evidence of diagnostic relationship

Note, . According to the diagnostic condition, we need to prove the equality where

Similarly, we have:

If evidence *D* is continuous in the real interval [*a*, *b*] with note that *a* and *b* are real numbers, formula 2.5 specifies probability density function (PDF) of continuous evidence . The PDF replaces CPT in case of continuous random variable.

Where,

**Formula 2.5.** CPT of continuous evidence of diagnostic relationship

As a convention, [*a*, *b*] is called domain of continuous evidence, which can be replaced by open or half-open intervals such as (*a*, *b*), (*a*, *b*], and [*a*, *b*). Of course we have . In learning context, evidence *D* is often a test whose grade ranges within real interval [*a*, *b*].

Functions and are valid PDF (s) due to:

According to the diagnostic condition, we need to prove the equality

Where,

When *D* is continuous, its probability is calculated in *ε*-vicinity where *ε* is very small number. As usual, *ε* is bias if *D* is measure values produced from equipment. The probability of *D* given *X* where *D*+*ε* [*a*, *b*] and *D*–*ε* [*a*, *b*] is:

In fact, we have:

In general, formula 2.6 summarizes CPT of evidence of single diagnosis relationship.

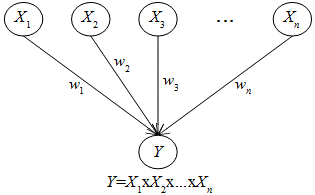
Where,

**Formula 2.6.** CPT of evidence of diagnostic relationship

In general, if the conditional probability *P*(*D|X*) is specified by formula 2.6, the diagnostic condition will be satisfied. Note that the CPT *P*(*D|X*) is the PDF *p*(*D|X*) in case of continuous evidence. The diagnostic relationship will be extended with more than one hypothesis. The next section will mention how to determine CPT (s) of a simple graph with one child node and many parent nodes based on X-gate inferences.

**3. X-gate inferences**

Given a simple graph consisting of one child variable *Y* and *n* parent variables *Xi*, as shown in figure 3.1. Each relationship from *Xi* to *Y* is quantified by a normalized weight *wi* where 0 ≤ *wi* ≤ 1. A large graph is an integration of many simple graphs. Figure 3.1 shows the DAG of a simple BN. As aforementioned, the essence of constructing simple BN is to convert graphic relationships of simple graph into CPT (s) of simple BN.



**Figure 3.1.** Simple graph or simple network

Child variable *Y* is called target and parent variables *Xi* (s) are called sources. Especially, these relationships are adhere to X-gates such as AND-gate, OR-gate, and SIGMA-gate. These gates are originated from logic gate (Wikipedia, 2016). For instance, AND-gate and OR-gate represent prerequisite relationship. SIGMA-gate represents aggregation relationship. Therefore, relationship conversion is to determined X-gate inference. The simple graph shown in figure 3.1 is also called X-gate graph or X-gate network. Please distinguish the letter “X” in the term “X-gate inference” which implies logic operators (AND, OR, XOR, etc.) from the “variable *X*”.

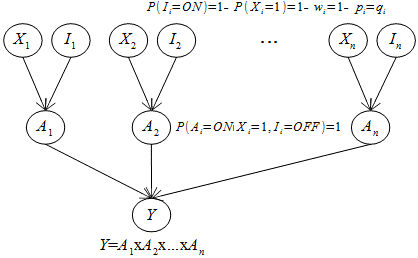
All variables are binary and they represent events. The probability *P*(*X*) indicates event *X* occurs. Thus, *P*(*X*) implicates *P*(*X*=1) and *P*(not(*X*)) implicates *P*(*X*=0). Formula 3.1 specifies the simple NOT-gate inference.

**Formula 3.1.** NOT-gate inference

X-gate inference is based on three assumptions mentioned in (Neapolitan, 2003, p. 157):

* *X-gate inhibition*: Given a relationship from source *Xi* to target *Y*, there is a factor *Ii* that inhibits *Xi* from integrated into *Y*. Factor *Ii* is called inhibition of *Xi*. That the inhibition *Ii* is turned off is prerequisite of *Xi* integrated into *Y*.
* *Inhibition independence*: Inhibitions are mutually independent. For example, inhibition *I*1 of *X*1 is independent from inhibition *I*2 of *X*2.
* *Accountability*: X-gate network is established by accountable variables *Ai* for *Xi* and *Ii*. Each X-gate inference owns particular combination of *Ai* (s).

Figure 3.2 shows the extended X-gate network with accountable variables *Ai* (s) (Neapolitan, 2003, p. 158).



**Figure 3.2.** Extended X-gate network with accountable variables *Ai* (s)

The strength of each relationship from source *Xi* to target *Y* is quantified by a weight 0 ≤ *wi* ≤ 1. According to the assumption of inhibition, probability of *Ii*=*OFF* is *pi*, which is set to be the weight *wi*.

If notation *wi* is used, we focus on strength of relationship. If notation *pi* is used, we focus on probability of *OFF* inhibition. In probabilistic inference, *pi* is also prior probability of *Xi*=1. However, we will assume each *Xi* has uniform distribution later on. Formula 3.2 specifies probabilities of inhibitions *Ii* (s) and accountable variables *Ai* (s).

**Formula 3.2.** Probabilities of inhibitions *Ii* (s) and accountable variables *Ai* (s)

According to formula 3.2, given probability *P*(*Ai*=*ON* | *Xi*=1, *Ii*=*OFF*), it is assured 100% confident that accountable variables *Ai* is turned on if source *Xi* is 1 and inhibition *Ii* is turned off. Formula 3.3 specifies conditional probability of accountable variables *Ai* (s) given *Xi* (s), which is corollary of formula 3.2.

**Formula 3.3.** Conditional probability of accountable variables

Following is proof of formula 3.3.

(By applying formula 3.2)

It implies

As a definition, the set of all *Xi* (s) is complete if and only if

The set of all *Xi* (s) is mutually exclusive if and only if

For each *Xi* there is only one *Ai* and vice versa, which establishes a bijection between *Xi* (s) and *Ai* (s). Obviously, the fact that the set of all *Xi* (s) is complete is equivalent to the fact that the set of all *Ai* (s) is complete. We will prove by contradiction that “the fact that the set of all *Xi* (s) is mutually exclusive is equivalent to the fact that the set of all *Ai* (s) is mutually exclusive”. Suppose but : . Let be preimage of *B*. Due to and , we have and , which causes that . There is a contradiction and so we have:

By similar proof, we have:

The extended X-gate network shown in figure 3.2 is interpretation of simple network shown in figure 3.1. Specifying CPT of the simple network is to determine the conditional probability *P*(*Y*=1 | *X*1, *X*2,…, *Xn*) based on extended X-gate network. The X-gate inference is represented by such probability *P*(*Y*=1 | *X*1, *X*2,…, *Xn*) specified by formula 3.4 (Neapolitan, 2003, p. 159).

**Formula 3.4.** X-gate probability

Following is the proof of formula 3.4.

(Due to Bayes’ rule)

(Due to total probability rule)

(Because *Y* is conditionally independent from *Xi* (s) given *Ai* (s))

(Due to Bayes’ rule)

(Because *Ai* (s) are mutually independent)

(Because each *Ai* is only dependent on *Xi*)

It is necessary to make some mathematical notations because formula 3.4 is complicated, which is relevant to arrangements of *Xi* (s). Given the set Ω = {*X*1, *X*2,…, *Xn*} where all variables are binary, table 3.1 specifies binary arrangements of Ω.

|  |
| --- |
| Given Ω = {*X*1, *X*2,…, *Xn*} where |Ω|=*n* is cardinality of Ω.  Let *a*(Ω) be an arrangement of Ω which is a set of *n* instances {*X*1=*x*1, *X*2=*x*2,…, *Xn*=*xn*} where *xi* is 1 or 0. The number of all *a*(Ω) is 2|Ω|. For instance, given Ω = {*X*1, *X*2}, there are 22=4 arrangements as follows:  *a*(Ω) = {*X*1=1, *X*2=1}, *a*(Ω) = {*X*1=1, *X*2=0}, *a*(Ω) = {*X*1=0, *X*2=1}, *a*(Ω) = {*X*1=0, *X*2=0}.  Let *a*(Ω:{*Xi*}) be the arrangement of Ω with fixed *Xi*. The number of all *a*(Ω:{*Xi*}) is 2|Ω|–1. Similarly, for instance, *a*(Ω:{*X*1, *X*2, *X*3}) is an arrangement of Ω with fixed *X*1, *X*2, *X*3. The number of all *a*(Ω:{*X*1, *X*2, *X*3}) is 2|Ω|–3.  Let *c*(Ω) and *c*(Ω:{*Xi*}) be the number of arrangements *a*(Ω) and *a*(Ω:{*Xi*}), respectively. Such *c*(Ω) and *c*(Ω:{*Xi*}) are called arrangement counters. As usual, counters *c*(Ω) and *c*(Ω:{*Xi*}) are equal to 2|Ω| and 2|Ω|–1, respectively but they will vary according to specific cases.  Let and denote sum and product of values generated from function *F* acting on every *a*(Ω). The number of arrangements on which *F* acts is *c*(Ω).  Let x denote the X-gate operator, for instance, for AND-gate, for OR-gate, for NAND-gate, for NOR-gate, for XOR-gate, for XNOR-gate, for U-gate, for SIGMA-gate. Given an x-operator, let *s*(Ω:{*Xi*}) and *s*(Ω) be sum of all through every arrangement of Ω with and without fixed *Xi*, respectively.  For example, *s*(Ω) and *s*(Ω:{*Xi*}) for OR-gate are:  Such *s*(Ω) and *s*(Ω:{*Xi*}) are called arrangement sum. They are acting function *F*.  Note that Ω can be any set of binary variables. |

**Table 3.1.** Binary arrangements

It is not easy to produce all binary arrangements of Ω. Table 3.2 shows a code snippet written by Java programming language for producing such all arrangements.

|  |
| --- |
| public class ArrangementGenerator {  private ArrayList<int[]> arrangements;  private int n;  private int r;  private ArrangementGenerator(int n, int r) {  this.n = n;  this.r = r;  this.arrangements = new ArrayList();  }  private void create(int[] a, int i) {  for(int j = 0; j < n; j++) {  a[i] = j;  if(i < r - 1)  create(a, i + 1);  else if(i == r -1) {  int[] b = new int[a.length];  for(int k = 0; k < a.length; k++) b[k] = a[k];  arrangements.add(b);  }  }  }  public int[] get(int i) {  return arrangements.get(i);  }  public long size() {  return arrangements.size();  }  public static ArrangementGenerator parse(int n, int r) {  ArrangementGenerator arr =  new ArrangementGenerator(n, r);  int[] a = new int[r];  for(int i=0; i<r; i++) a[i] = -1;  arr.create(a, 0);  return arr;  }  } |

**Table 3.2.** Code snippet generating all binary arrangements

Each element of the list “*arrangements*” is a binary arrangement *a*(Ω) presented by an array of bits (0 and 1). The method “*create*(*int*[] *a*, *int i*)” which is recursive method, is the main one that generates arrangements. The method call “*ArrangementGenerator*.*parse*(2, *n*)” will list all possible binary arrangements.

Formula 3.5 specifies the connection between *s*(Ω:{*Xi*=1}) and *s*(Ω:{*Xi*=0}), between *c*(Ω:{*Xi*=1}) and *c*(Ω:{*Xi*=0}).

**Formula 3.5.** Connection among arrangement sum and counter

It is easy to draw formula 3.5 when the set of all arrangements *a*(Ω:{*Xi*=1) is complement of the set of all arrangements *a*(Ω:{*Xi*=0).

Let *K* be a set of *Xi* (s) whose values are 1 and let *L* be a set of *Xi* (s) whose values are 0. *K* and *L* are mutually complementary. Formula 3.6 determines sets *K* and *L*.

**Formula 3.6.** Sets *K* and *L*

The **AND-gate** inference represents prerequisite relationship satisfying AND-gate condition specified by formula 3.7.

**Formula 3.7.** AND-gate condition

From formula 3.4, we have

(Due to formula 3.3)

In general, formula 3.8 specifies AND-gate inference.

**Formula 3.8.** AND-gate inference

The AND-gate formula was also described in (Díez & Druzdzel, 2007, p. 33). Formula 3.8 varies according to two cases whose arrangement counters are listed as follows:

The **OR-gate** inference represents prerequisite relationship satisfying OR-gate condition specified by formula 3.9 (Neapolitan, 2003, p. 157).

**Formula 3.9.** OR-gate condition

The OR-gate condition implies

From formula 3.4, we have (Neapolitan, 2003, p. 159):

(Due to formula 3.3)

In general, formula 3.10 specifies OR-gate inference.

**Formula 3.10.** OR-gate inference

Where *K* is the set of *Xi* (s) whose values are 1. The OR-gate formula was mentioned in (Neapolitan, 2003, p. 158) and (Díez & Druzdzel, 2007, p. 20). Formula 3.10 varies according to two cases whose arrangement counters are listed as follows:

According to De Morgan’s rule with regard to AND-gate and OR-gate, we have:

(Due to formula 3.10)

According to formula 3.8, we also have:

In general, formula 3.11 specifies NAND-gate inference and NOR-gate inference derived from AND-gate and OR-gate:

**Formula 3.11.** NAND-gate inference and NOR-gate inference

Where *K* and *L* are the sets of *Xi* (s) whose values are 1 and 0, respectively.

Suppose the number of sources *Xi* (s) is even. Let *O* be the set of *Xi* (s) whose indices are odd. Let *O*1 and *O*2 be subsets of *O*, in which all *Xi* (s) are 1 and 0, respectively. Let *E* be the set of *Xi* (s) whose indices are even. Let *E*1 and *E*2 be subsets of *E*, in which all *Xi* (s) are 1 and 0, respectively.

Thus, *O*1 and *E*1 are subsets of *K*. Sources *Xi* (s) and target *Y* follow **XOR-gate** if one of two XOR-gate conditions specified by formula 3.12 is satisfied.

**Formula 3.12.** Two XOR-gate conditions

From formula 3.4, we have:

If both XOR-gate conditions are not satisfied then,

If the first XOR-gate condition is satisfied, we have:

We have

(Due to formula 3.3)

We also have

(Due to formula 3.3)

Given the first XOR-gate condition, it implies

Similarly, given the second XOR-gate condition, we have:

If one of XOR-gate conditions is satisfied then,

This implies formula 3.13 to specify XOR-gate inference.

Where,

**Formula 3.13.** XOR-gate inference

Given *n* ≥ 2 and *n* is even, formula 3.13 varies according to six cases whose arrangement counters are listed as follows:

Suppose the number of sources *Xi* (s) is even. According to **XNOR-gate** inference (Wikipedia, 2016), the output is on if all inputs get the same value 1 (or 0). Sources *Xi* (s) and target *Y* follow XNOR-gate if one of two XNOR-gate conditions specified by formula 3.14 is satisfied.

**Formula 3.14.** Two XNOR-gate conditions

From formula 3.4, we have:

If both XNOR-gate conditions are not satisfied then,

If *Ai*=*ON* for all *i*, we have:

(Please see similar proof in AND-gate inference)

If *Ai*=*OFF* for all *i*, we have:

(Please see similar proof in OR-gate inference)

If one of XNOR-gate conditions is satisfied then,

This implies formula 3.15 to specify XNOR-gate inference.

**Formula 3.15.** XNOR-gate inference

Where *K* and *L* are the sets of *Xi* (s) whose values are 1 and 0, respectively. Formula 3.15 varies according to three cases whose arrangement counters are listed as follows:

Let *U* be a set of indices such that *Ai*=*ON* and let *α* ≥ 0 and *β* ≥ 0 be predefined numbers. The U-gate inference is defined based on *α*, *β* and cardinality of *U*. Formula 3.16 specifies four common U-gate conditions.

|  |  |
| --- | --- |
| |*U*|=*α* | if there are exactly *α* variables *Ai* = *ON* (s). Otherwise, . |
| |*U*|≥*α* | if there are at least *α* variables *Ai* = *ON* (s). Otherwise, . |
| |*U*|≤*β* | if there are at most *β* variables *Ai* = *ON* (s). Otherwise, . |
| *α*≤|*U*|≤*β* | if the number of *Ai* = *ON* (s) is from *α* to *β*. Otherwise, . |

**Formula 3.16.** U-gate conditions

Your attention please, U-gate condition on |*U*| can be arbitrary and it is only relevant to *Ai* (s) (*ON* or *OFF*) and the way to combine *Ai* (s). For example, AND-gate and OR-gate are specific cases of U-gate with |*U*|=*n* and |*U*|≥1, respectively. XOR-gate and XNOR-gate are also specific cases of U-gate with specific conditions on *Ai* (s). However, it must be assured that there is at least one combination of *Ai* (s) satisfying the predefined U-gate condition, which causes that U-gate probability is not always equal to 0. In this research, U-gate is the most general nonlinear gate where U-gate probability contains products of weights (see formula 3.17). Later on, we will research a so-called SIGMA-gate that contains only linear combination of weights (sum of weights, see formula 3.19). Shortly, each X-gate is a pattern owning a particular X-gate inference that is X-gate probability *P*(*X*1x*X*2x…x*Xn*). Each X-gate inference is based on particular X-gate condition (s) relevant to only variables *Ai* (s).

From formula 3.4, we have:

Let be the set of all possible *U* (s), we have:

If then,

This implies all sets *U* (s) must be subsets of *K*. The U-gate probability is rewritten as follows:

(Due to formula 3.3)

Let *PU* be the U-gate probability, formula 3.17 specifies U-gate inference and cardinality of where is the set of subsets (*U*) of *K*.

|  |  |
| --- | --- |
| Let,  As a convention, | |
| |*U*|=0 |  |
| |*U*|≥0 | The case |*U*|≥0 is the same to the case |*U*|≤*n* |
| |*U*|=*n* |  |
| |*U*|=*α*  0<*α*<*n* |  |
| |*U*|≥*α*  0<*α*<*n* |  |
| |*U*|≤*β*  0<*β*<*n* |  |
| *α*≤|*U*|≤*β*  0<*α*<*n*  0<*β*<*n* |  |

**Formula 3.17.** U-gate inference

Note that the notation denotes the number of combinations of *j* elements taken from *n* elements.

Arrangement counters relevant to U-gate inference and the set *K* are listed as follows:

The **SIGMA-gate** inference (Nguyen, 2016) represents aggregation relationship satisfying SIGMA-gate condition specified by formula 3.18.

Where the set of *Ai* (s) is complete and mutually exclusive.

**Formula 3.18.** SIGMA-gate condition

The sigma sum indicates that *Y* is exclusive union of *Ai* (s) and here, it does not express arithmetical additions.

This implies:

The sigma sum now expresses arithmetical additions of probabilities *P*(*Ai*).

SIGMA-gate inference requires the set of *Ai* (s) is complete and mutually exclusive, which means that the set of *Xi* (s) is complete and mutually exclusive too. The SIGMA-gate probability is (Nguyen, 2016):

It implies

(Due to formula 3.3)

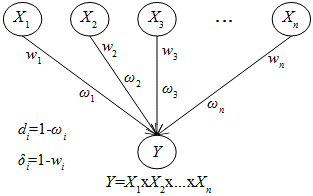
In general, formula 3.19 specifies the theorem of SIGMA-gate inference (Nguyen, 2016). The base of this theorem was mentioned by authors (Millán & Pérez-de-la-Cruz, A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation, 2002, pp. 292-295).

Where the set of *Xi* (s) is complete and mutually exclusive.

**Formula 3.19.** SIGMA-gate inference

The arrangement counters of SIGMA-gate inference are *c*(Ω:{*Xi*=1}) = *c*(Ω:{*Xi*=0}) = 2*n*–1, *c*(Ω) = 2*n*.

Formula 3.3 specifies the “clockwise” strength of relationship between *Xi* and *Y*. Event *Xi*=1 causes event *Ai*=*ON* with “clockwise” weight *wi*. There is a question “given *Xi*=0, how likely the event *Ai*=*OFF* is”. In order to solve this problem, I define a so-called “counterclockwise” strength of relationship between *Xi* and *Y* denoted *ωi*. Event *Xi*=0 causes event *Ai*=*OFF* with “counterclockwise” weight *ωi*. In other words, each arc in simple graph is associated with a clockwise weight *wi* and a counterclockwise weight *ωi*. Such graph is called *bi-weight simple graph* shown in figure 3.3.



**Figure 3.3.** Bi-weight simple graph

With bi-weight simple graph, all X-gate inferences are extended as so-called X-gate bi-inferences. Derived from formula 3.3, formula 3.20 specifies conditional probability of accountable variables with regard to bi-weight graph.

**Formula 3.20.** Conditional probability of accountable variables with regard to bi-weight graph

The probabilities *P*(*Ai*=*ON* | *Xi*=0) and *P*(*Ai*=*OFF* | *Xi*=1) are called clockwise adder *di* and counterclockwise adder *δi*. As usual, *di* and *δi* are smaller than *wi* and *ωi*. When *di*=0, bi-weight graph becomes normal simple graph.

The total clockwise weight or total counterclockwise weight is defined as sum of clockwise weight and clockwise adder or sum of counterclockwise weight and counterclockwise adder. Formula 3.21 specifies such total weights *Wi* and . These weights are also called relationship powers.

Where,

**Formula 3.21.** Total clockwise and counterclockwise weights

Given formula 3.21, the set of all *Ai* (s) is complete if and only if .

By extending aforementioned X-gate inferences, we get bi-inferences for AND-gate, OR-gate, NAND-gate, NOR-gate, XOR-gate, XNOR-gate, and U-gate as shown in table 3.3.

|  |
| --- |
| There are four common conditions of *U*: |*U*|=*α*, |*U*|≥*α*, |*U*|≤*β*, and *α*≤|*U*|≤*β*. Note that is the complement of *U*,  The largest cardinality of is: |

**Table 3.3.** Bi-inferences for AND-gate, OR-gate, NAND-gate, NOR-gate, XOR-gate, XNOR-gate, and U-gate

The largest cardinalities of *K* (*L*) are 2*n*–1 and 2*n* with and without fixed *Xi*. Thus, it is possible to calculate arrangement counters. As a convention, the product of probabilities is 1 if indices set is empty.

With regard to SIGMA-gate bi-inference, the sum of all total clockwise weights must be 1 as follows:

Derived from formula 3.19, the SIGMA-gate probability for bi-weight graph is:

Shortly, formula 3.22 specifies SIGMA-gate bi-inference.

Where the set of *Xi* (s) is complete and mutually exclusive.

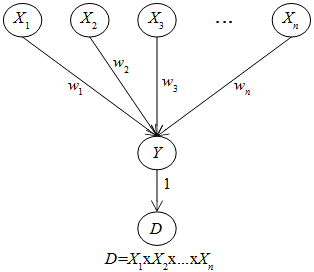
**Formula 3.22.** SIGMA-gate bi-inference

The next section will research diagnostic relationship which adheres to X-gate inference.

**4. Multi-hypothesis diagnostic relationship**

Given a simple graph shown in figure 3.1, if we replace the target source *Y* by an evidence *D*, we get a so-called *multi-hypothesis diagnostic relationship* whose property adheres to X-gate inference. Maybe there are other diagnostic relationships in which X-gate inference is not concerned. However, this research focuses on X-gate inference and so multi-hypothesis diagnostic relationship is called *X-gate diagnostic relationship*. Sources *X*1, *X*2,…, *Xn* become hypotheses*.* As a convention, these hypotheses have prior uniform distribution.

According to aforementioned X-gate network shown in figures 3.1 and 3.2, the target variable must be binary whereas evidence *D* can be numeric. It is impossible to establish the evidence *D* as direct target variable. Thus, the solution of this problem is to add an augmented target binary variable *Y* and then, the evidence *D* is connected directly to *Y*. In other words, the *X-gate diagnostic network* have *n* sources {*X*1, *X*2,…, *Xn*}, one augmented hypothesis *Y*, and one evidence *D*. As a convention, X-gate diagnostic network is called *X-D network*. The CPT (s) of the entire network are determined based on combination of diagnostic relationship and X-gate inference mentioned in previous sections. Figure 4.1 depicts the augmented X-D network. Note that variables *X*1, *X*2,…, *Xn*, and *Y* are always binary.



**Figure 4.1.** Augmented X-D network

The joint probability of augmented X-D network shown in figure 4.1 is

The joint probability of X-D network is

By applying total probability rule into X-D network, we have:

(Due to Bayes’ rule)

(Due to total probability rule)

(Because *D* is conditionally independent from all *Xi* (s) given *Y*)

(Due to Bayes’ rule)

It is implied that the augmented X-D network is equivalent to X-D network with regard to variables *X*1, *X*2,…, *Xn* and *D*. As a convention, augmented X-D network is considered as same as X-D network.

The simplest case of X-D network is NOT-D network having one hypothesis *X*1 and one evidence *D*, equipped with NOT-gate inference. NOT-D network satisfies diagnostic condition because it essentially represents the single diagnostic relationship. Inferred from formulas 2.1 and 3.1, the conditional probability *P*(*D|X*1) and posterior probability *P*(*X*1*|D*) of NOT-D network are:

(Due to Bayes’ rule and uniform distribution of *X*1)

It implies NOT-D network satisfies diagnostic condition. Let

We will validate whether the CPT of diagnostic relationship, *P*(*D|X*) specified by formula 2.6, still satisfies diagnostic condition within general case, X-D network. In other words, X-D network is general case of single diagnostic relationship.

Recall from dependencies shown in figure 4.1, formula 4.1 specifies the joint probability of X-D network.

Where Ω = {*X*1, *X*2,…, *Xn*}.

**Formula 4.1.** Joint probability of X-D network

Formula 4.2 specifies the conditional probability of *D* given *Xi* (likelihood function) and the posterior probability of *Xi* given *D*.

**Formula 4.2.** Conditional probability *P*(*D*|*Xi*) and posterior probability *P*(*Xi*|*D*)

Where Ω = {*X*1, *X*2,…, *Xn*} and the sign “\” denotes the subtraction (excluding) operator in set theory (Wikipedia, Set (mathematics), 2014).

Given uniform distribution of *Xi* (s), we have:

The joint probability becomes

The joint probability of *Xi* and *D* is:

(Due to formula 2.6)

The marginal probability of *D* is:

By applying table 3.1, the joint probability *P*(*Xi*, *D*) is determined as follows:

Similarly, formula 4.3 specifies the joint probability *P*(*Xi*, *D*) and the marginal probability *P*(*D*) given uniform distribution of all sources.

**Formula 4.3.** Joint probability *P*(*Xi*, *D*) and marginal probability *P*(*D*) given uniform distribution of all sources

Where *s*(Ω) and *s*(Ω:{*Xi*}) are specified in table 3.1. In general, formula 4.4 specifies conditional probability *P*(*D|Xi*), posterior probability *P*(*Xi|D*), and transformation coefficient for X-gate inference.

**Formula 4.4.** Conditional probability, posterior probability, and transformation coefficient of X-D network

The transformation coefficient is rewritten as follows:

Note that *S*, *D*, *M* are abstract symbols and there is no proportional connection between 2*n*–1*S* and *D* for all *D*, specified by formula 2.6. Assuming that such proportional connection 2*n*–1*S* = *aDj* exists for all *D* where *a* is arbitrary constant. Given binary case when *D*=0 and *S*=1, we have:

There is a contradiction, which implies that it is impossible to reduce *k* into the following form:

Therefore, if *k* is constant with regard to *D* then,

Where *C* is constant. We have:

It is implied that

This holds

Assuming *ND*=*S* we have:

There is a contradiction because *M* is maximum value of *D*. Therefore, if *k* is constant with regard to *D* then *s*(Ω) = 2*n*–1. Inversely, if *s*(Ω) = 2*n*–1 then *k* is:

In general, the event that *k* is constant with regard to *D* is equivalent to the event *s*(Ω) = 2*n*–1. This implies *diagnostic theorem* stated in table 4.1.

|  |
| --- |
| Given X-D network is combination of diagnostic relationship and X-gate inference:  The diagnostic condition of X-D network is satisfied if and only if  At that time, the transformation coefficient becomes:  Note that weights *pi*=*wi* and *ρi*=*ωi*, which are inputs of *s*(Ω), are abstract variables. Thus, the equality *s*(Ω) = 2|Ω|–1 implies all abstract variables are removed and so *s*(Ω) does not depend on weights. |

**Table 4.1.** Diagnostic theorem

The diagnostic theorem is the optimal way to validate the diagnostic condition.

The formula 4.4 becomes simple with AND-gate inference. Recall that formula 3.8 specified AND-gate inference as follows:

Due to only one case *X*1 = *X*2 =…= *Xn* = 1, we have:

Due to *Xi* = 0, we have:

Derived from formula 4.4, formula 4.5 specifies conditional probability *P*(*D|Xi*), posterior probability *P*(*Xi|D*), and transformation coefficient according to X-D network with AND-gate reference called *AND-D network*.

**Formula 4.5.** Conditional probability and posterior probability of AND-D network

For convenience, we validate diagnostic condition with a case of two sources Ω = {*X*1, *X*2}, *p*1 = *p*2 = *w*1 = *w*2 = 0.5, . According to diagnostic theorem stated in table 4.1, if *s*(Ω) ≠ 2 for given X-gate then, such X-gate does not satisfy diagnostic condition.

Given AND-gate inference, by applying formula 3.8, we have:

Given OR-gate inference, by applying formula 3.10, we have:

Given XOR-gate inference, by applying formula 3.13, we have:

Given XNOR-gate inference, by applying formula 3.15, we have:

Given SIGMA-gate inference, by applying formula 3.19, we have:

It is asserted that AND-gate, OR-gate, XOR-gate, and XNOR-gate do not satisfy diagnostic condition and so they should not be used to assess hypotheses. However, it is not asserted if U-gate and SIGMA-gate satisfy such diagnostic condition. It is necessary to expend formula for SIGMA-gate diagnostic network (called *SIGMA-D network*) in order to validate it.

In case of SIGMA-gate inference, by applying formula 3.19, we have:

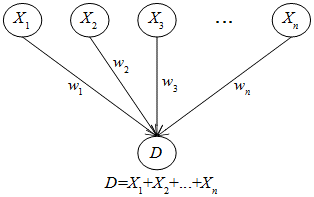
It is necessary to validate SIGMA-D network with SIGMA-gate bi-inference. By applying formula 3.22, we recalculate these quantities as follows:

Obviously, quantities *s*(Ω), *s*(Ω:{*Xi*=1}), and *s*(Ω:{*Xi*=0}) are kept intact. According to diagnostic theorem, we conclude that SIGMA-D network does satisfy diagnostic condition due to *s*(Ω)=2*n*–1. Thus, SIGMA-D network can be used to assess hypotheses.

Formula 4.6, an immediate consequence of formula 4.4, specifies conditional probability *P*(*D|Xi*), posterior probability *P*(*Xi|D*), and transformation coefficient for SIGMA-D network.

**Formula 4.6.** Conditional probability, posterior probability, and transformation coefficient of SIGMA-D network

In case of SIGMA-gate, the augmented variable *Y* can be removed from X-D network. The evidence *D* is now established as direct target variable. Figure 4.2 shows a so-called *direct SIGMA-gate diagnostic network* (direct SIGMA-D network).



**Figure 4.2.** Direct SIGMA-gate diagnostic network (direct SIGMA-D network)

Derived from formula 3.19, the CPT of direct SIGMA-D network is determined by formula 4.7.

Where the set of *Xi* (s) is complete and mutually exclusive.

**Formula 4.7.** CPT of direct SIGMA-D network

The formula 4.7 specifies valid CPT due to:

From dependencies shown in figure 4.2, formula 4.8 specifies the joint probability of direct SIGMA-D network.

**Formula 4.8.** Joint probability of direct SIGMA-D network

Inferred from formula 4.3, formula 4.9 specifies the joint probability *P*(*Xi*, *D*) and the marginal probability *P*(*D*) of direct SIGMA-D network, given uniform distribution of all sources.

**Formula 4.9.** Joint probability *P*(*Xi*, *D*) and marginal probabilities *P*(*Xi*), *P*(*D*) of direct SIGMA-D network

Where *s*(Ω) and *s*(Ω:{*Xi*}) are specified in table 3.1.

By browsing all variables of direct SIGMA-D network, we have:

Similarly, we have:

By applying formula 4.9, *s*(Ω:{*Xi*=0}), *s*(Ω:{*Xi*=1}), and *s*(Ω), we get the same result with formula 4.6.

Therefore, it is possible to use direct SIGMA-D network to assess hypotheses. It is asserted that SIGMA-D network satisfy diagnostic condition when single relationship, NOT-D network, direct SIGMA-D network are specific cases of SIGMA-D network. There is a question: Does an X-D network which is different from SIGMA-D network and not aforementioned exist such that it satisfies diagnostic condition?

Recall that each X-D network is a pattern owning a particular X-gate inference which in turn is based on particular X-gate condition (s) relevant to only variables *Ai* (s). The most general nonlinear X-D network is U-D network whereas SIGMA-D network is linear one. The U-gate inference given arbitrary condition on *U* is:

Let *f* be the arrangement sum of U-gate inference.

The function *f* is sum of many large expressions and each expression is product of four possible sub-products (Π) as follows:

In any case of degradation, there always exist expression *Expr* (s) having at least 2 sub-products (Π), for example:

Consequently, there always exist *Expr* (s) having at least 5 terms relevant to *pi* and *ρi* if *n* ≥ 5, for example:

Thus, degree of *f* will be larger than or equal to 5 given *n* ≥ 5. According to diagnostic theorem, U-gate network satisfies diagnostic condition if and only if *f(pi*, *ρi*) = 2*n*–1 for all *n* ≥ 1 and for all abstract variables *pi* and *ρi*. Without loss of generality, each *pi* or *ρi* is sum of variable *x* and a variable *ai* or *bi*, respectively. Note that all *pi*, *ρi*, *ai* are *bi* are abstract variables.

The equation *f*–2*n*–1 = 0 becomes equation *g*(*x*) = 0 whose degree is *m* ≥ 5 if *n* ≥ 5.

Where coefficients *Ci* (s) are functions of *ai* and *bi* (s). According to Abel-Ruffini theorem (Wikipedia, Abel-Ruffini theorem, 2016), equation *g*(*x*) = 0 has no algebraic solution when *m* ≥ 5. Thus, abstract variables *pi* and *ρi* cannot be eliminated entirely from *g*(*x*)=0, which causes that there is no specification of U-gate inference *P*(*X*1x*X*2x…x*Xn*) so that diagnostic condition is satisfied.

It is concluded that there is no nonlinear X-D network satisfying diagnostic condition but a new question is raised: Does there exist the general linear X-D network satisfying diagnostic condition? Such linear network is called GL-D network and SIGMA-D network is specific case of GL-D network. The GL-gate probability must be linear combination of weights.

Where *C* is arbitrary constant.

The GL-gate inference is singular if *αi* and *βi* are functions of only *Xi* as follows:

The functions *hi* and *gi* are not relevant to *Ai* because the final formula of GL-gate inference is only relevant to *Xi* (s) and weights (s). Because GL-D network is a pattern, we only survey singular GL-gate. Mentioned GL-gate is singular by default and it is dependent on how to define functions *hi* and *gi*. The arrangement sum with regard to GL-gate is:

Suppose *hi* and *gi* are probability mass functions with regard to *Xi*. For all *i*, we have:

The arrangement sum becomes:

GL-D network satisfies diagnostic condition if

Suppose the set of *Xi* (s) is complete.

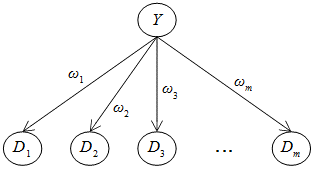
This implies *C*=0. Shortly, formula 4.10 specifies the singular GL-gate inference so that GL-D network satisfies diagnostic condition.

Where *hi* and *gi* are probability mass functions and the set of *Xi* (s) is complete.

**Formula 4.10.** Singular GL-gate inference

Functions *hi*(*Xi*) and *gi*(*Xi*) are always linear due to *Xim*=*Xi* for all *m*≥1 when *Xi* is binary. It is easy to infer that SIGMA-D network is GL-D network with following definition of functions *hi* and *gi*.

According to authors (Millán & Pérez-de-la-Cruz, A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation, 2002), a hypothesis can have multiple evidences as seen in figure 4.3. This is *multi-evidence diagnostic relationship* opposite to aforementioned multi-hypothesis diagnostic relationship.



**Figure 4.3.** Diagnostic relationship with multiple evidences (M-E-D network)

Figure 4.3 depicts the multi-evidence diagnostic network called M-E-D network in which there are *m* evidences *D*1, *D*2,…, *Dm* and one hypothesis *Y*. Note that *Y* has uniform distribution.

In simplest case where all evidences are binary, the joint probability of M-E-D network is:

The product is denoted as likelihood function as follows:

The posterior probability *P*(*Y* | *D*1, *D*2,…, *Dm*) given uniform distribution of *Y* is:

The possible transformation coefficient is:

M-E-D network will satisfy diagnostic condition if *k* = 1 because all hypotheses and evidence are binary, which leads that following equation specified by formula 4.11 has 2*m* real roots *P*(*Dj|Y*) for all *m* ≥ 2.

**Formula 4.11.** Diagnostic condition equation of binary M-E-D network

The equation 4.11 has no real root given *m*=2 according to following proof. Suppose equation 4.11 has 4 real roots as follows:

From equation 4.11, it holds:

The final equation leads a contradiction (*b*1=2 or *b*1=1.5) and so it is impossible to apply the sufficient diagnostic proposition into M-E-D network. Such proposition is only used for one-evidence network. Moreover, X-gate inference absorbs many sources and then produces out of one targeted result whereas the M-E-D network essentially splits one source into many results. It is impossible to model M-E-D network by X-gates. The potential solution for this problem is to group many evidences *D*1, *D*2,…, *Dm* into one representative evidence *D* which in turn is dependent on hypothesis *Y* but this solution will be inaccurate in specifying conditional probabilities because directions of dependencies become inconsistent (relationships from *Dj* to *D* and from *Y* to *D*) except that all *Dj* (s) are removed and *D* becomes a vector. However evidence vector does not simplify the hazardous problem and it changes the current problem into a new problem.

Another solution is to reverse the direction of relationship, in which the hypothesis is dependent on evidences so as to take advantages of X-gate inference as usual. However, the reversion method violates the viewpoint in this research where diagnostic relationship must be from hypothesis to evidence. In other words, we should change the viewpoint.

Another solution is based on a so-called *partial diagnostic condition* that is a loose case of diagnostic condition for M-E-D network, which is defined as follows:

Where *k* is constant with regard to *Dj*. The joint probability is:

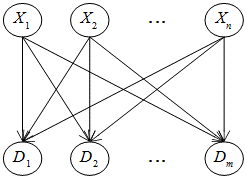
M-E-D network satisfies partial diagnostic condition. In fact, given all variables are binary, we have:

(Let Ψ = {*D*1, *D*2,…, *Dm*})

(Due to uniform distribution of *Y*)

Partial diagnostic condition expresses a different viewpoint. It is not an optimal solution because we cannot test a disease based on only one symptom while ignoring other obvious symptoms, for example. The equality *P*(*Y|Dj*) = 0.5*P*(*Dj|Y*) indicates the accuracy is decreased two times. However, Bayesian network provides inference mechanism based on personal belief. It is subjective. You can use partial diagnostic condition if you think that such condition is appropriate to your application.

If we are successful in specifying conditional probabilities of M-E-D network, it is possible to define an extended network which is constituted of *n* hypotheses *X*1, *X*2,…, *Xn* and *m* evidences *D*1, *D*2,…, *Dm*. Such extended network represents *multi-hypothesis multi-evidence diagnostic relationship*, called M-HE-D network. Figure 4.1 depicts M-HE-D network.



**Figure 4.1.** M-HE-D network

The M-HE-D network is the most general case of diagnostic network, which was mentioned in (Millán & Pérez-de-la-Cruz, A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation, 2002, p. 297). We can construct any large diagnostic BN from M-HE-D networks and so the research is still open.

**5. Conclusion**

In short, relationship conversion is to determine conditional probabilities based on logic gates that is adhered to semantics of relationships. The weak point of logic gates is to require that all variables must be binary. For example, in learning context, it is inconvenient for expert to create an assessment BN with studying exercises (evidences) whose marks are only 0 and 1. In order to lessen the impact of such weak point, I use numeric evidence for extending capacity of simple Bayesian network. However, combination of binary hypothesis and numeric evidence leads to errors or biases in inference. For example, given a student gets maximum grade for an exercise but the built-in inference results out that she/he has not mastered fully the associated learning concept (hypothesis). Therefore, I propose the sufficient diagnostic proposition so as to confirm that numeric evidence is adequate to make complicated inference tasks in BN. The probabilistic reasoning based on evidence is always accurate. Application of the research can go beyond learning context whenever probabilistic deduction relevant to constraints of semantic relationships is required. A large BN can be constituted of many simple BN (s). Inference in large BN is hazardous problem and there are many optimal algorithms for solving such problem. In future, I will research effective inference methods for the special BN that is constituted of X-gate BN (s) mentioned in this research because X-gate BN (s) have precise and useful features of which we should take advantages. For instance, their CPT (s) are simple in some cases and the meanings of their relationships are mandatory in many applications. Moreover, I try my best to research deeply M-E-D network and M-HE-D network whose problems I cannot solve absolutely now.

Two main documents I referred to do this research are the book “Learning Bayesian Networks” (Neapolitan, 2003) by the author Richard E. Neapolitan and the article “A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation” (Millán & Pérez-de-la-Cruz, A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation, 2002) by authors Eva Millán and José Luis Pérez-de-la-Cruz. Especially, the SIGMA-gate inference is based on and derived from the work of the authors Eva Millán and José Luis Pérez-de-la-Cruz. This research is originated from my PhD research “A User Modeling System for Adaptive Learning” (Nguyen, A User Modeling System for Adaptive Learning, 2014). Other references relevant to user modeling, overlay model, and Bayesian network are (Fröschl, 2005), (De Bra, Smits, & Stash, 2006), (Murphy, 1998), and (Heckerman, 1995). Please concern these references.

**References**

De Bra, P., Smits, D., & Stash, N. (2006). The Design of AHA! In U. K. Wiil, P. J. Nürnberg, & J. Rubart (Ed.), *Proceedings of the seventeenth ACM Hypertext Conference on Hypertext and hypermedia (Hypertext '06)* (pp. 171-195). Odense, Denmark: ACM Press.

Díez, F. J., & Druzdzel, M. J. (2007). *Canonical Probabilistic Models.* National University for Distance Education, Department of Inteligencia Artificial. Madrid: Research Centre on Intelligent Decision-Support Systems. Retrieved May 9, 2016, from http://www.cisiad.uned.es/techreports/canonical.pdf

Fröschl, C. (2005). *User Modeling and User Profiling in Adaptive E-learning Systems.* Master Thesis, Graz University of Technology, Austria.

Heckerman, D. (1995). *A Tutorial on Learning With Bayesian Networks.* Microsoft Corporation, Microsoft Research. Redmond: Microsoft Research. Retrieved from ftp://ftp.research.microsoft.com/pub/dtg/david/tutorial.ps

Kschischang, F. R., Frey, B. J., & Loeliger, H.-A. (2001, February). Factor Graphs and the Sum-Product Algorithm. *IEEE Transactions on Information Theory, 47*(2), 498-519. doi:10.1109/18.910572

Millán, E., & Pérez-de-la-Cruz, J. L. (2002, June). A Bayesian Diagnostic Algorithm for Student Modeling and its Evaluation. (A. Kobsa, Ed.) *User Modeling and User-Adapted Interaction, 12*(2-3), 281-330. doi:10.1023/A:1015027822614

Millán, E., Loboda, T., & Pérez-de-la-Cruz, J. L. (2010, July 29). Bayesian networks for student model engineering. (R. S. Heller, J. D. Underwood, & C.-C. Tsai, Eds.) *Computers & Education, 55*(4), 1663-1683. doi:10.1016/j.compedu.2010.07.010

Murphy, K. P. (1998). *A Brief Introduction to Graphical Models and Bayesian Networks.* Retrieved 2008, from Kevin P. Murphy's home page: http://www.cs.ubc.ca/~murphyk/Bayes/bnintro.html

Neapolitan, R. E. (2003). *Learning Bayesian Networks.* Upper Saddle River, New Jersey, USA: Prentice Hall.

Nguyen, L. (2014, April). *A User Modeling System for Adaptive Learning.* University of Science, Ho Chi Minh city, Vietnam. Abuja, Nigeria: Standard Research Journals. Retrieved from http://standresjournals.org/journals/SSRE/Abstract/2014/april/Loc.html

Nguyen, L. (2016, March 28). Theorem of SIGMA-gate Inference in Bayesian Network. (V. S. Franz, Ed.) *Wulfenia Journal, 23*(3), 280-289.

Pearl, J. (1986, September). Fusion, propagation, and structuring in belief networks. *Artificial Intelligence, 29*(3), 241-288. doi:10.1016/0004-3702(86)90072-X

Wikipedia. (2014, October 10). *Set (mathematics)*. (A. Rubin, Editor, & Wikimedia Foundation) Retrieved October 11, 2014, from Wikipedia website: http://en.wikipedia.org/wiki/Set\_(mathematics)

Wikipedia. (2015, November 22). *Factor graph*. (Wikimedia Foundation) Retrieved February 8, 2017, from Wikipedia website: https://en.wikipedia.org/wiki/Factor\_graph

Wikipedia. (2016, June 10). *Abel-Ruffini theorem*. (Wikimedia Foundation) Retrieved June 26, 2016, from Wikipedia website: https://en.wikipedia.org/wiki/Abel%E2%80%93Ruffini\_theorem

Wikipedia. (2016, June 2). *Logic gate*. (Wikimedia Foundation) Retrieved June 4, 2016, from Wikipedia website: https://en.wikipedia.org/wiki/Logic\_gate