3 The Medical Records-based Knowledge Acquisition

It is well known that knowledge acquisition is the problem need to be solved in many knowledge systems, of course including expert systems in the field of TCM. As a typical representative of Chinese traditional culture, TCM has its unique field characteristics, which makes it difficult to obtain knowledge from various TCM data, especially in medical records. The obtaining of the academic thinking and clinical experience of learned Chinese doctors from medical records is key to the inheritance and development of TCM theory. Therefore, how to discover knowledge from many medical records by elder learned Chinese doctors is a question badly in need of a solution in the information processing of TCM. This chapter provides several feasible ideas for extracting knowledge from TCM records for some specific problems.

3.1 Centrality Research on the Traditional Chinese Medicine Network

It is the key to sum up and pass on the old famous TCM doctors' academic knowledge and their clinical experiences that analyzing the relationship among concepts of treatments in the TCM. These concepts have been associated with each other by the construction of ontology, making their relationship into a complex network. In this section, a reasonable solution based on graph data mining is proposed, in which graphs are composed out of the TCM networks by making the records of distinct disease to match the ontology in the network.

As a formal language, graph is a kind of tool describing network and its features, to formalize the network data, and to quantize characteristics of a real network. Here, graph is used to formally describe the TCM network, while spot is used to stand for medicine, symptom, evidence, and line is used to represent the relationship among spots. Figure 3-1 shows a first diagnosed symptom network. The centrality algorithm referred is based on graph algorithm to analyze the TCM network.

The method of describing the real network as graphs before data mining is embodied in many applications. For example, in sociology, scholars characterize social relationship by graph which has many vertexes and lines. A vertex represents an actor that may be a person, an organization, or a group, etc. The line connecting two vertexes represents a social joint. In the

field of biology, the protein structure is described as graph in which a vertex is an atom and an edge is a valence. By mining data on the protein structure graph, we can find the internal relationships of protein structure.

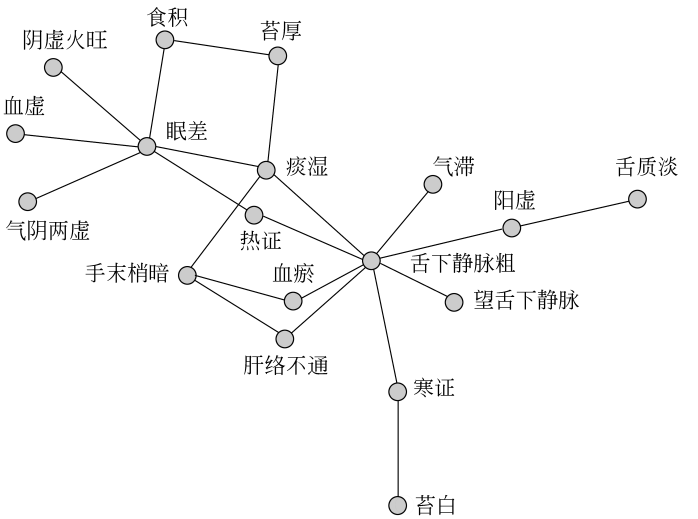

Figure 3-1　A first diagnosed symptom network

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named "Heading 1", "Heading 2", "Heading 3", and "Heading 4" are prescribed.

The centrality mark of the vertex plays an essential role in analysis on network structure which we can determine the key vertex of the whole network. In the graph of social network, centrality mark can define social rights according to relationship, which shows whether individuals or organizations are most important in social or not according to its position whether in a joining center of not. Similarly, in TCM, we can quantize the importance of a disease, a medicine or a symptom with measurement of its centrality. In a whole treatment, a higher centrality medicine plays more primary role, while a higher centrality symptom repeats more times in the diagnoses.

3.1.1　Basic Thought and Concept

Relevant Concepts

The network can be conveniently described as a graph G = (V, E). In the TCM networks,

the vertex set V represents medicines, symptoms or evidences, while the edge set E represents relationships between vertexes, medicine, and symptom.

As knowledge on graph is used in centrality study, the relevant concepts are introduced first.

(1) Path: a route in which each line or point doesn't repeat. In network analysis, more attention is paid on paths than lines.

(2) Length: the number of lines constructing the path.

(3) Geodesic: the shortest path between two points. If there are some shortest paths between two points, they are all geodesic.

(4) Distance: the length of geodesic of two points. So as to say, the distance between two points is the shortest length to connect them. $d_G(\boldsymbol{s}, \boldsymbol{t})$ is used to represent the distance between point $\boldsymbol{s}$ and $\boldsymbol{t}$.

Betweenness Centrality, Centrality Based on the Shortest Path

Classifying by different portray mark, centrality measure method is of different kinds as follows: Degree Centrality, Closeness Centrality and Betweenness Centrality. Degree Centrality uses the number of edges close to vertex. Closeness Centrality reflects the centrality degree of vertex in the network, because it defines the derivative of sum of the distance from this vertex to all the others. Betweenness Centrality reflects the control of the vertex onto others. Here we analyze TCM by studying the Betweenness Centrality arithmetic.

Graph in this section is undirected and no-weighting. Suppose ω is weighting function defined on the edge, then $\omega(e)= 1$.

Suppose σ_{st} is the number of shortest paths from vertex $\boldsymbol{s} \in V$ to vertex $\boldsymbol{t} \in V$, and $\sigma_{st}(\boldsymbol{v})$ is the number of paths passing the vertex $\boldsymbol{v}$ in the shortest path. Eq.(3-1) is a standard measurement formula of Betweenness Centrality.

$$C_B(\boldsymbol{v}) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(\boldsymbol{v})}{\sigma_{st}} \quad \text{(Freeman,1977)} \tag{3-1}$$

What gives in Eq. (3-1) is the expression of Absolute Centrality Degree. In order to be easy to compare between centrality from different graphs, we give the mark of Relative Centrality Degree, which is the standard of absolute centrality degree mark. The method to calculate the Relative Centrality Degree of a point is to divide the Absolute Centrality Degree of the point by a sum result which is got by summing up each possible maximal value of all the other points in the same graph. The discussion below is on the calculation of centrality.

Lemma 3-1: (Bellman-Ford Algorithm) Vertex $\boldsymbol{v} \in V$ lies in the shortest path from vertex $\boldsymbol{s} \in V$ to vertex $\boldsymbol{t} \in V$, if and only if $d_G(\boldsymbol{s}, \boldsymbol{t}) = d_G(\boldsymbol{s}, \boldsymbol{v}) + d_G(\boldsymbol{v}, \boldsymbol{t})$.

Pair-dependence of vertexes $\boldsymbol{s}, \boldsymbol{t} \in V$ of medium point $\boldsymbol{v}$ is $\delta_{st}(\boldsymbol{v}) = \frac{\sigma_{st}(\boldsymbol{v})}{\sigma_{st}}$, that is the

possibility of vertex v lie in the shortest path between s and t. In order to get the Betweenness Centrality mark of vertex v, it is needed to calculate all the pair-dependence of the vertex $C_B(v)=\sum_{s\neq v\neq t\in V}\delta_{st}(v)$.

Therefore, the Betweenness Centrality should be calculated by two steps:

(1) Calculating the length and number of the shortest path between each pair of vertexes.

(2) Calculating each pair-independence.

3.1.2 Method to Calculate Betweenness Centrality

Calculating the Length and Number of the Shortest Path between Each Pair of Vertexes

The calculation begins from vertex s using Board First Search Algorithm and Dijkstra's Single Source Shortest Path Algorithm. Dijkstra's algorithm is applied to get the shortest path from any vertex to any other one.

$$P_s(v)=\{u\in V:\{u,v\}\in E,d_G(s,v)=d_G(s,u)+\omega(s,v)\} \tag{3-2}$$

Lemma 3-2: (combinatorial calculate the shortest path) For $s\neq v\in V$, $\sigma_{st}=\sum_{u\in P_S(v)}\sigma_{su}$.

Calculating Each Pair-independence

In order to elaborate calculation demands of all pair-independence, we first introduce the concept of independence of vertex $s\in V$ on single vertex $v\in V$.

Theorem 3-1: The independence of $s\in V$ on any vertex $v\in V$ agrees with the relationship:

$$\delta_{s\cdot}(v)=\sum_{w:v\in P_s(w)}\frac{\sigma_{sv}}{\sigma_{sw}}\left(1+\delta_{s\cdot}(w)\right) \tag{3-3}$$

An edge $\{v, w\}$ exists where vertex v lying on at least one shortest path from s to t, $\delta_{st}(v)>0$ and $v\in P_s(w)$. This is a little more complex, as Figure 3-2 shows.

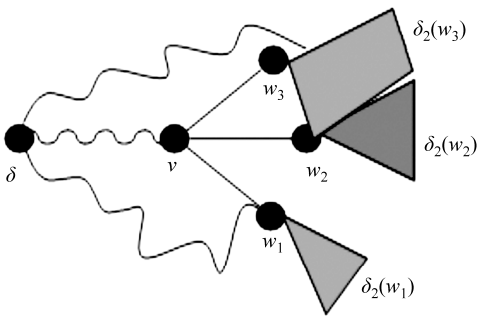

Figure 3-2　Condition theorem

3.1.3 Betweenness Centrality Algorithm

Here are some variables and arrays at the beginning of the Algorithm 3-1. Array *CB* stores centrality, $\sigma[t]$ stores number of t on the shortest path. Chain array $P[w]$ store sets of all vertexes w on the shortest path of vertex s.

In Algorithm 3-1, line 09−22 do BFS traverse, line 13−16 judge whether w is visited for the first time, line 17−20 judge whether v is one point on the shortest path to w, if so then add the path into array $P[w]$. And finally pop out the stack line 24−29 and calculate the centrality.

Algorithm 3-1

```
01  CB[v] ← 0,v∈V;
02  for s∈V do
03      S ← empty stack;
04      P[w] ← empty list, w∈V;
05      σ[t] ← 0,t∈V; σ[s] ← 1;
06      d[t] ← −1, t∈V; d[s] ← 0;
07      Q ← empty queue;
08      enqueue s → Q;
09      while Q not empty do
10          dequeue v ← Q;
11          push v → S;
12          for each neighbor w of v do
13              if d[w] < 0 then
14                  enqueue w → Q;
15                  d[w] ← d[v] + 1;
16              end
17              if d[w] = d[v] + 1 then
18                  σ[w] ← σ[w] + σ[v];
```

```
19                    append v → P[w];
20                end
21            end
22        end
23        δ[v] ← 0, v ∈ V;
24        while S not empty do
25            pop w ← S;
26            for v ∈ P[w] do
27            δ[v] ←δ[v] + σ[v].(1 + δ[w])/ σ[w];
28            if w ≠ s then CB[w] ← CB[w] + δ[w];
29        end
30    end.
```

3.1.4 Example Analyses

There is the calculation result showing in Table 3-1, which is made out by applying the algorithm above to analysis the TCM network that has 32 vertexes. Betweenness Centrality is used to measure the ability of a symptom serving as the medium in first diagnoses. That is if there isn't the symptom, no relation exits between other symptoms. More times such roles a symptom plays, higher Betweenness Centrality it is, and then more other symptoms are connected by it. In another word, this symptom is more important part during the treatment.

Table 3-1　Result of analyzing centrality

Vertexes	Betweenness	nBetweenness
1. Coarse sublingual vein	334.367	71.907
2. Phlegm dampness	169.267	36.401
3. Yang deficiency	168.000	36.401
4. Whitish tongue	150.000	32.258
5. Chill syndrome	130.000	27.957

Continued

Vertexes	Betweenness	nBetweenness
6. White tongue	114.000	24.516
7. Problems of sleep quality	105.700	22.731
8. Thick tongue coating	91.967	19.778
9. Heat syndrome	42.733	9.190
10. Gravy palm	38.967	8.380
11. Obstruction of liver collaterals	11.833	2.545
12. Blood stasis	11.833	2.545
13. Dyspepsia	9.333	2.007
14. Qi-yin deficiency	0	0
15. Qi stagnation	0	0
16. Sublingual vein	0	0
17. Blood deficiency	0	0
18. Yin defic fire hyperact	0	0

The result of analyzing centrality algorithm is shown by data, as Table 3-1. To make it more clear and understandable, a visual picture is given below, as Figure 3-3. Because the centrality of the symptom named coarse sublingual vein is the highest, we can conclude that it was the most possible medium of other symptoms. That is to say curing the coarse sublingual vein is of great importance during the treatment.

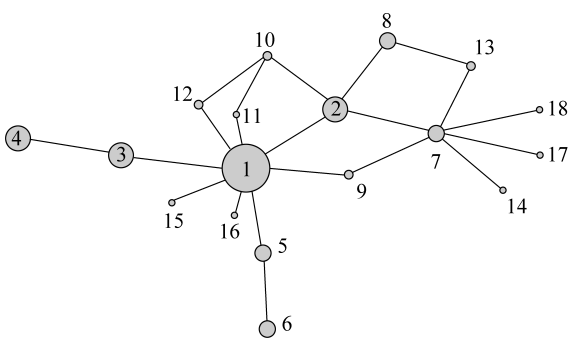

Figure 3-3 The result of centrality algorithm

3.1.5 Conclusions

In this section, we discussed a new way of data mining in the area of the TCM and solved the problem that complex relationship can not be searched by the data mining based on set. First, make the concepts extracted from records of diseases match pair with entries in ontology library, and then compose a graph structure out of the TCM network. Second, analyze the structure of the network on this graph by processing it with the centrality algorithm. Last, get the data mining result.

3.2 Cognitive Induction Based Knowledge Acquisition

Inductive Logic is the logic on probable inference: it is unnecessary but improbable that the conclusion is false while the premises are all true. The cognition, especially experience knowledge acquisition, is closely related to Inductive Logic. Inductive Learning is a mature technique in Machine Learning, aiming at extracting common rules and patterns for decision from an amount of experience data. Inductive Logic and Inductive Learning formed and develops both correlatively and independently. The possibility of finite inductive machine demonstrates the possibility of applying Inductive Logic theory into Inductive Learning, which provide a kind of theory foundation for Inductive Learning. Modern Inductive Logic and later-developed Machine Learning and Inductive Learning are consistent in research objects and methods, but their differences between research directions and targets indicates that it is infeasible to copy Inductive Logic into Inductive Learning. On basis of cognitive mechanism, some theories and methods of Inductive Logic are introduced according to requirements and conditions of specific problems, which help to form Inductive Learning techniques. On basis of Cognitive mechanism and Probabilistic Logic System theory of Hintikka of Finland School, a kind of example-based inductive learning technique is built and it is applied to TCM differentiation and typing in this section.

3.2.1 Data Preprocessing

Carnap solved the measurement problem of logic probability by confirmation function C^*, of which in condition of infinite domain of individual and finite evidence, the confirmation degree of universal factual sentence is 0, as state description is taken as basis of logic semantics in Carnap's inductive theory; while in Hintikka's Conponent theory possible situations of the world is described in different specific degree. There is specific description in

state description theory: each individual in universe is with-or -without property $P_i(i=1,\ldots,\pi)$; While there is general description in structure description theory: frequency of each Q-predicate in N individuals is given. In Carnap's theory, to get the inductive confirmation degree of one sentence (evidence) to the other(hypothesis), it's necessary to know the sentence value(prior probability); to get m of one sentence , it's necessary to know both number of state descriptions in the logic domain of the sentence and m of each state description; to get m of state description, it's necessary to know the total number of state descriptions in the whole system. However, if the individual number in universe is infinite, the total number of state description is unaccountably infinite, which poses problems in calculation. Therefore, Hintikka's Conponent theory is used to describe the possible world to avoid the problem.

Definition 3-1: π one-predicate $P_1(x),\ldots,P_\pi(x)$given, Q-predicate or property component is conjunctions of those predicates:$(\pm)P_1(x) \wedge (\pm)P_2(x) \wedge \ldots \wedge (\pm)P_\pi(x)$

Definition 3-2: Q-predicate can use to form component, C_w is:

$$\exists xQ_1(x)\wedge\exists xQ_2(x)\wedge\ldots\wedge\exists xQ_w(x)\wedge\forall x[Q_1(x)\vee Q_2(x)\vee\ldots\vee Q_w(x)]$$

Component has the most generality in describing a possible world. There are several different Q predicates in one component. For instance, if the number is w, there are only these w Q predicates having illustrations, which means there are individuals having the property represented by these Q predicates, while the other K-w Q predicate has not.

C_w can be regarded as the combination of possible propositions. Suppose n individuals compose evidence 3, of which there are c Q predicates have illustrations, and one of the component C_c（w=c）contains the c Q predicates mentioned above, then the component is consistent with the evidence. If C_w (w<c) is falsified by evidence, the component is inconsistent with the evidence, because in this situation, component C_w indicates there are only less than c Q predicates have illustrations while evidence e indicates there are c Q predicates have illustrations. C_w (w>c) isn't falsified, because although there are c Q predicates with illustrations. Actually C_w (w>c) isn't falsified here, so component C_w is consistent with evidence e, because although evidence e indicates that there are c Q predicates have illustrations, but new Q predicate with illustrations will be observed further.

Here each symptom is taken as initial predicate $P_i(x)$, of several possible symptoms is taken as predicate Q_j , each set of symptoms will be consistent with one element in Q predicate, suppose after inductive learning of train set of mass medical cases, there will be just one component C_w for each symbol and no more else in further observation, that is, C_w is complete description on all possible propositions of this symbol.

Training data set can be structured as: training sample space X={x}, each sample has property set P (x) ={$P_1(x),P_2(x),\ldots,P_\pi(x)$} and diagnosis decision J_j, so each sample can be described as binary group ($P(x)$,J_j).

3.2.2 Inductive Logic Based Inductive Learning Algorithm

Theory of Modern Inductive Logic

According to further study on rules of inductive acceptance and relations of inductive learning and knowledge, many scholars pointed out that if the confirmation degree of one hypothesis *h* is bigger than 1–*ε*, with the relevant universal hypothesis unacceptable, *h* is also unacceptable. Therefore, it's essential to consider confirmation degree of universal hypothesis $P(C_w/e)$ as using confirmation degree of singular hypothesis $P(h/e)$.

In Hintikka's theory, the number of components (components not yet falsified) consistent with evidence *e* is $\sum_{i=0}^{K-c}\binom{K-c}{i}$, where $\binom{K-c}{i}$ is symbol of the combination, also recorded as C_{K-c}^{i}; *K* is number of *Q* predicate, *c* is number of *Q* predicates illustrated in evidence *e*. For instance, if there are 2 unary predicate, *K*=4, which means there are 4 *Q* predicates in system, if 2 *Q* predicates is illustrated, *c*=2, so number of components haven't yet been falsified is: 1+2+1=4.

What's the confirmation degree of a component C_w not having been falsified according to evidence *e*? It can be calculated by Bayes Theorem as follows:

$$P(C_w/e) = P(C_w)P(e/C_w)/\sum_{i=0}^{K-c}\binom{K-c}{i}P(C_{c+i})P(e/C_{c+i}) \tag{3-4}$$

In Eq.(3-4), $P(C_w)$ stands for prior probability of component C_w, $P(e/C_w)$, also named likelihood, stands for probability of evidence *e* if component C_w is true; $P(C_{c+i})$ stands for probability of any component C_{c+i} not yet falsified, $P(e/C_{c+i})$ stand for prior probability of evidence *e* if C_{c+i} is true. The whole formula is obvious, of which how to establish $P(C_w)$ and $P(e/C_w)$ are the key.

Hintikka successively takes the following two methods to establish $P(C_w)$ and $P(e/C_w)$:

1) Algorithm of equal probability confirmation degree

Assign equal prior probability to each component, and equally assign probability of each component to all conditions which make the component true. The prior probability of each component is $1/2^k$. As $P(C_w)$ and $P(C_{c+i})$ are all $1/2^k$, Eq.(3-4) can be simplified as:

$$P(C_w/e) = P(e/C_w)/\sum_{i=0}^{K-c}\binom{K-c}{i}P(e/C_{c+i}) \tag{3-5}$$

When the individual number *N* is infinite, Hintikka takes $(1/w)^n$as approximation of $P(e/C_w)$:

$$P\left(\frac{e}{C_w}\right) = (1/w)^n \tag{3-6}$$

$$P\left(\frac{e}{C_{c+i}}\right) = [1/(c+i)]^n \tag{3-7}$$

Substitute value of $P\left(\frac{e}{C_w}\right)$ and $P\left(\frac{e}{C_{c+i}}\right)$into Eq.(3-5):

$$P(C_w/e) = (1/w)^n / \sum_{i=0}^{K-c} \binom{K-c}{i} [1/(c+i)]^n \tag{3-8}$$

According to Eq.(3-8), when evidence *e* infinitely increase（$n\to\infty$), the confirmation degree of the posterior probability of the component C_c, consistent with evidence, incline to 1, the posterior probability of all the other component $C_w(w > c)$incline to 0. Thus Hintikka initially solve the problem of conforming universal sentences in infinite individual domain. There are two faults of this algorithm. First, when *n* is very small, $P(C_c/e)$ may be very big, which reflects the over-optimistic induction that small amount of experience observation lead to high confirmation degree by induction. For instance, suppose universal sentence $K = 4$, $c = 3$, $P(C_c/e) = 1/[1 + (3/4)^n]$, can obtain high confirmation degree after observing a few cases. Secondly, the posterior probability of singular prediction hypothesis (the next to-be-observed individual has property Q_i), obtained by assigning probability value, isn't nature, and the confirmation degree is always *l*/*c*:

$$P(h/e) = l/c \tag{3-9}$$

The determination of posterior probability only consider which *Q* predicate are observed in evidence, not considering the frequencies of different *Q* predicate.

2) Algorithm of Confirmation Degree of Joint Probability

There are 3 steps to get confirmation degree of singular hypothesis *h*: first assign equal non-zero prior probability to each component, then equally assign the probability of each component to all structure descriptions which make the component true, and then equally assign the probability of each structure description to all status descriptions which make the structure description true. Then the confirmation degree of *h* is:

$$P(h/e) = (n_j + 1)/(n + c) \tag{3-10}$$

In Eq.(3-10) $P(h/e)$ represents the confirmation degree of evidence *e* onto hypothesis *h*. Obviously, confirmation degree of singular hypothesis, 'the next to-be-observed individual has property Q_i', are affected by two factors. One is frequent n_j/n of Q_i predicate in evidence *e*, the other is number *c* of *Q* predicates in evidence.

In conventional application, confirmation degree $P(Q_i/e)$(i=1,...,c) of Q_i predicate in the component is obtained, in order to better described the different contribution degrees of different Q_i predicates onto different components, $P(Q_i/e)$ is normalized, and the result is

represented by $P(Q_i/e)$ as:

$$P(Q/e) = P(Q_i/e) / \sum_{i=0}^{c} P(Q_i/e) \tag{3-11}$$

Hintikka proposed formula of contributions in ∂ - λ two-dimension system as:

$$P(C_w/e) = \pi\left(\partial, \frac{w\lambda}{K}\right) \prod_{j=1}^{c} \pi\left(n_j, \frac{\lambda}{w}\right) / \sum_{i=0}^{K-c} \left\{ \binom{K-c}{i} \pi\left(\partial, \frac{(c+i)\lambda}{K}\right) \prod_{j=1}^{c} \pi\left[n_j, \frac{\lambda}{c+i}\right] \right\} \tag{3-12}$$

K is possible number of Q predicate, w is number of Q predicates in component C_w, c is number of Q predicate illustrated in component C_w, ∂ is an adjustable parameter, aiming at avoid the high confirmation degree of induction as observation samples are a few. n is number of samples in e, n_i is number of samples which is consistent with Q_i predicate and has conclusions of symptom subgroups described by component C_w, w is number of Q predicate in component C_w. w is used to describe confirmation degree of different symbols set Q_i, as evidence e given, onto the symptom subgroups of C_w; it will change as the increasing of e, that is changing of human being's cognitions.

If ∂ =0, λ =w, confirmation degree of universal summarize sentence of Hintikka's joint system is

$$P(C_w/e) = n_1! \times n_2! \times \ldots \times n_c! / \sum_{i=0}^{K-c} \left\{ \binom{K-c}{i} \prod_{j=1}^{c} \pi\left[n_j, w/(c+i)\right] \right\} \tag{3-13}$$

$P(C_w/e)$ still didn't get rid of the 1st fault of the first method, but the great number of records may help to weaken the influence of the fault, so it is possible to get the reasonable result. There is a hypothesis for this method in application: suppose after mass individuals be observed, all possible Q predicate w will be illustrated as w=c, that isn't any new Q predicate illustrated in further observation.

Inductive Learning Algorithm

This algorithm mainly adopts inductive learning to acquire universal confirmation degree of each differential diagnostics and of Q predicates illustrated in each differential diagnostics from mass medical cases, normalizes the confirmation degree, and then do aggregations, thus reasonably remove those Q predicates and P predicates with less confirmation degree on certain differential diagnostics.

The storage structure of knowledge base is defined as sextuple $(Q_i, P(Q_i/e), J_j, C_i, n_i, n)$, J_j is diagnostic decision. Steps of inductive learning algorithm are given.

Step 1: Calculate out symptom subgroups with structure medical records from database of

medical records. If there are s symptom subgroups, samples should be divided into s classes, represented by $C_1,\ldots,C_s$; each individual in component should be taken out from medical records library and stored into their own dataset rec_i.

Step 2: Calculate total number n of samples in each component, Q_i predicates of component and the number n_i from dataset of each component, and record the relevant diagnosis conclusions.

Step 3: Calculate number c of Q predicate illustrated in each component, suppose $w=c$, applying Eq.(3-13) to get confirmation degree $P(C_c/e)$ of existing medical records on corresponding differential diagnostics. If $P(C_c/e)>1-\varepsilon$, go to step 4, else report "Samples in medical records library too scattered, continue collect medical records!", finish learning.

Step 4: Obtain confirmation degree $P(Q_i/e)$ of Q_i predicate (representing one kind of medical records) in component by applying Eq.(3-11).

Step 5: Store above Q_i, $P(Q_i/e)$, J_j, C_i, n_i, n into relevant sextuple of knowledge base 1st.

Step 6: Operate on knowledge base 1, extracted records accord with C_s, if $Q_i \subset Q_j$ and $P(Q_i/e) \geqslant P(Q_j/e)$ or if the following three conditions occurred simultaneously: $Q_k \subset Q_j$ and $P(Q_k/e) \geqslant P(Q_j/e)$ and $P(Q_i/e) \geqslant P(Q_k/e)$, then $P(Q_i/e)= P(Q_j/e)+P(Q_i/e)$, and Q_j related records should be deleted. Q_j should be selected in descending order of $P(Q_j/e)$. By comparing Q_j and Q predicates with higher confirmation degree, Q_j with the highest confirmation degree and contained in Q_j will be selected and operated as above steps. And the processing result should be stored in knowledge base 2 finally. This is the end of Inductive Learning Algorithm on basis of modern inductive logic.

3.2.3 Graph−based Inductive Learning Algorithm

After operation on medical records with inductive learning algorithms on basis of modern inductive logics, symptoms affecting differential diagnostics are aggregated to certain extend in vertically, and may be aggregated in horizontally by removing certain symptom hardly affecting differential diagnostics. Therefore, graph-based inductive learning algorithms are put forward here.

Inductive Learning Algorithm Graph Introduction

(1) The graph, composition of several given points and lines connecting the points, is commonly used to describe specified relations between objects, in which points represent objects, lines connecting two points represent relations between corresponding two objects. Thus the semantic relations between concepts and terms in TCM medical records compose a complicated network graph.

In directed graph, vertex pair $<\boldsymbol{v}_j,\boldsymbol{v}_t>$ is orderly, named directed edge from vertex $\boldsymbol{v}_j$ to

vertex v_t. Thus<v_j,v_t>and <v_t,v_j>are two different edges. In <v_j,v_t>, v_j is the start point with v_t the end. In undirected graph, (v_j,v_t) are orderless, named vertex and vertex related edge, without specified direction and the same graph with (v_t,v_j).

(2) Weight.

Some edges have relevant value, named weight. These weight can represent different things from one vertex to the other, for instance, distance, cost, time, times, etc, the weighted graph is named network.

(3) Adjacent Vertex.

If (u,v) is an edge of $E(G)$, u and v are mutually named adjacent vertex, edge (u,v) adhere to vertex u and vertex v. If <u,v> is one of the directed edge, vertex u adjacent to vertex v and vertex v adjacent to vertex u, edge <u,v>is associated with vertex u and v.

(4) Subgraph.

Suppose two subgraphs G=<V,E> and G'=<V',E' >. If $V'\subseteq V$ and $E'\subseteq E$, G' is the subgraph of G. There are several subgraph of G1 in Figure 3-4. As subgraph mining is important in graph mining, it's crucial to find important subgraph from graph with TCM medicines and symbols as nods.

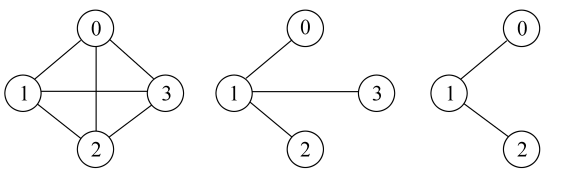

Figure 3-4　Graph and subgraph

(5) Degree of Vertex.

Suppose E is set of edges in undirected graph, V and V' are adjacent vertex in the graph. Degree of vertex V is $D(V)$, representing number of edges associated with V. If E is set of arcs in directed graph, in-degree of V is ID(V), representing number of arcs with V as arc head; out-degree of V is $OD(V)$, representing number of arcs with V as arc end. The degree of vertex V is $D(V)$=ID(V)+OD(V).

(6) Storage Structure of Graph.

There are two common storage methods for graph storage: array and adjacency list. Array is considered to store TCM network for sake of algorithm efficiency. Files are read in array first, transformed into adjacent matrix of graph, and then processed by theories proposed above.

As for weighted graph, adjacency matrix is defined as:

$$A[i][j] = \begin{cases} w(i,j), & i! = j, < i,j > \in E \parallel (i,j) \in E \\ \infty, & \text{else, } i! = j \\ 0, & \text{diagoline, } i = j \end{cases} \quad (3\text{-}14)$$

Adjacency matrix is the reflection of logic structure of graphs, facilitates the calculation of vertex degree of graphs, as well as some simple calculations, which benefit further process.

(7) K-Core.

K-core, also the cohesive subgroup on basis of node degree, is the subgraph satisfies the following condition: any point directly connected to at least *k* other nodes, this implies that the degree of any point in k-core is not less than *k*.

According to k-core theory, by deleting nodes with less ties with others from exterior to interior, symptoms with greater influence on differential diagnostics in the whole medical records will be acquired.

Graph-based Inductive Learning Algorithm

According to graph-based inductive learning algorithm, induction is done by removing symptoms with less influence on differential diagnostics from certain *Q* predicate.

Step 1. Do operation on knowledge base 2, extracted records accord with C_s, get all illustrated P_j (j=0,…,158) from all relevant Q_i , record if P_j and Q_i are related and the relations between Q_i and C_s , save into database.

Step 2. Do data mining on graphs in Step 1 by k-core algorithm to remove some symptoms with little contribution on diagnosis.

Step 3. Operations of Step 1 and Step 3 lead to relative centralized symptom set for certain differential diagnostics. Take $P(Q_i/e)$ as weight from Q_i to Q_i. Calculate that occurrence number of each predicate, assign contribution degree of Q_i to one-predicate P_j of Q_i according to proportions of occurrence number of P_j of Q_i. Take sum of all contributions degree of P_j as contribution degree of P_j to certain differential diagnostic.

Step 4. Save the left one-predicate P_j and its contribution degree on certain differential diagnosis $P(P_j/e)$ into knowledge base 3, when there isn't any conclusion for new medical records in knowledge base 1, knowledge base 3 can be referred to.

3.2.4 Application of Inductive Learning Algorithm

Medical records are saved into the below medical records library named collectable (Figure 3-5) according to medical acquisition system, the common symptoms are divided into 12 fields, including cold and heat, sweat, head and body, urination and defecation, food and drink, chest and abdomen, deafness, menstruation and leucorrhea, tongue proper, tongue body, tongue fur, pulse. There are 12 fields to store the 12 symptoms in collection, in which

character string is used to represent situations of different symptoms.

编号	寒热	汗	头身	二便	饮食	胸腹	聋	经带	舌质	舌体	舌苔	脉	诊断结论
1	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000001000	100000
2	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
10	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
11	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
12	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
13	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
14	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
15	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
16	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
17	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
18	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000
19	0100	0000	000000100000000000000000000	0010000000000101	000000000010000000000	000000000000000000	000	000000000000000000000	000000010	00100	10000000000	00000000000000000	100000

Figure 3-5　Medical cases library: collectable

0 indicates that the patient doesn't have the symptoms, while 1 is opposite. Field zzjl represents diagnosis results.

Knowledge base 1, as shown in Figure 3-6, is obtained according to inductive algorithm. The record number of deficiency of Qi and Yin decreased from 203 to 47, the confirmation degree of each kind of Q predicate to deficiency of Qi and Yin, as field $P(Q_i/e)$ shown in Figure 3-7. The second record has the highest confirmation degree of 0.38799977.

id	Qi	ni	p(Qi/e)
1	0100000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000000000000001000	1	0.0079999957
2	0100000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000000000000000000	96	0.38799977
3	0100000000000000000000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000000000000000000	3	0.015999991
4	0000000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000000000000000000	5	0.023999985
5	0000000001000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000000000000100000	1	0.0079999957
6	0000000001000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000000000001000000	1	0.0079999957
7	0100000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000001000000000	33	0.13599993
8	0100000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000000100000000	1	0.0079999957
9	0100000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000001000000000000000	1	0.0079999957
10	0100000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000001000000000000	1	0.0079999957
11	0100000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000100000000000	1	0.0079999957
12	0100100001000000101000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000001000000000000000000	8	0.035999976
13	0100000000000000100000000000000000000000100000000000010100000000001000000000000000000000000000000000000000000000000000000000000000000000000000010001001000000000000000000001000000000	12	0.051999971

Figure 3-6　Knowledge base 1

Knowledge base 2, as shown in Figure 3-7 is also obtained according to inductive algorithm in Step 5. Obviously, the confirmation degree of Q predicate, with ID of 2, increased from 0.38799977 to 0.7360003207, one-predicate p decreased from 54 kinds to 24 kinds, which includes: burning sensation of five centers, spontaneous perspiration, pale face, less verbal communication, limb fatigue, limb pain, gum bleeding, dry feces, dark urine, oliguresis, tastelessness, dry mouth with little saliva, cough, metrorrhagia, sublingual vein cyanosis, varicose sublingual vein, corpulent tongue, teeth-printed tongue, shrunk tongue, thin white fur of tongue, tongue fur with little saliva, yellowish fur, thready and slippery pulse. It indicates that symptoms contribute to differential diagnostics is aggregated to certain extend.

id	Qi	ni	p(Qi/e)
3	0100000000000001000000000000000000000001000000000001010000000000100000000000000000000000000000000000000000000000000000000000000000000000001000100100000000000000000000000000000000	164	0.7360003207
4	0000000000000001000000000000000000000001000000000001010000000000100000000000000000000000000000000000000000000000000000000000000000000000001000100100000000000000000000000000000000	7	0.0399998027
16	0100000000000001000100000000000000000001000000000001010000000000100000010000000000000001000000000000000000000000000000000000000000000000001000100100000100000000000000010000000000	1	0.0079999957
20	0100000000000001000001000000000000000001000000000001010000000000100000000000000000000000000000000000000000000000000000000000000000000000001000100000000100000000000000000000000000	1	0.0079999957
21	0100000000000000000000000000000000000001000000000001000000000000100000000000000000000000000000000000000000000000000000000000000000000000001000100100000000000000000000000000000000	9	0.0599999668
35	0100000000000001000000000000000000000001000000000001010000000000000000000000000000000000000000000000000000000000000000000000000000000000000100100000000000000000000000000000000	8	0.0599999672
36	0100000000000001000000000000000000000000000000000001010000000000000000000000000000000000000000000000000000000000000000000000000000000000000100100000000000000000000000000000000	1	0.0079999957
37	0000000000000001000000000000000000000001000000000001010000000000000000000000000000000000000000000000000000000000000000000000000000000000001000100100000000000000000000000000000000	1	0.0079999957
39	0100000000000001000000000000000000000000000000000001010000000000100000000000000000000000000000000000000000000000000000000000000000000000001000100000000000000000000000000000000000	2	0.0159999914
41	0100000000000001000000000000000000000001000000000001010000000000100000000000000000000000000000000000000000000000000000000000000000000000001000000100000000000000000000000000000000	2	0.0159999914
45	0100000000000001000000000000000000000001000000000001000000000000100000000000000000000000000000000000000000000000000000000000000000000000001000100100000000000000000000000000000000	2	0.011999993
46	0100000000000001000000000000000000000000000000000001010000000000100000000000000000000000000000000000000000000000000000000000000000000000001000100100000000000000000000000000000000	2	0.011999993
47	0100100001000000001000000000000000000001000000000001010000000000000000000000000000000000000000000000000000000000000001000000000000000000000000100100000000000000000000010000000000	3	0.015999991

Figure 3-7 Knowledge base 2

The graph-based inductive learning algorithm aims at centralizing symptoms more affective to differential diagnostics. Step 1 of the algorithm can lead to adjacent matrix shown in Figure 3-8, among which rows represent *Q* predicates, columns represent *P* predicates.

$$
\boldsymbol{A}=\begin{bmatrix}
1\,0\,0\,1\,0\,0\,0\,0\,1\,1\,1\,1\,0\,0\,0\,0\,1\,0\,1\,0\,1\,0\,0\,0\,0 \\
0\,0\,0\,1\,0\,0\,0\,0\,1\,1\,1\,1\,0\,0\,0\,0\,1\,0\,1\,0\,1\,0\,0\,0\,0 \\
1\,0\,0\,1\,0\,1\,0\,1\,1\,1\,1\,1\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0\,1\,0 \\
1\,0\,0\,1\,0\,0\,1\,1\,1\,1\,1\,0\,0\,0\,0\,1\,0\,1\,0\,0\,0\,1\,0\,0 \\
1\,0\,0\,0\,0\,0\,0\,1\,1\,0\,1\,0\,0\,0\,0\,1\,0\,1\,0\,1\,0\,0\,0\,0 \\
1\,0\,0\,0\,0\,0\,0\,1\,1\,0\,1\,0\,0\,0\,0\,1\,0\,1\,0\,1\,0\,0\,0\,0 \\
1\,0\,0\,1\,0\,0\,0\,1\,1\,1\,0\,0\,0\,0\,0\,0\,0\,1\,0\,1\,0\,0\,0\,0 \\
1\,0\,0\,1\,0\,0\,0\,0\,1\,1\,0\,0\,0\,0\,0\,0\,0\,1\,0\,1\,0\,0\,0\,0 \\
0\,0\,0\,1\,0\,0\,0\,1\,1\,1\,0\,0\,0\,0\,0\,1\,0\,1\,0\,1\,0\,0\,0\,0 \\
1\,0\,0\,1\,0\,0\,0\,0\,1\,1\,1\,0\,0\,0\,0\,1\,0\,1\,0\,0\,0\,0\,0\,0 \\
1\,0\,0\,1\,0\,0\,0\,1\,1\,1\,1\,0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,0\,0\,0 \\
1\,0\,0\,1\,0\,0\,0\,1\,0\,1\,1\,0\,0\,0\,0\,1\,0\,1\,0\,1\,0\,0\,0\,0 \\
1\,0\,0\,1\,0\,0\,0\,0\,1\,1\,1\,0\,0\,0\,0\,1\,0\,1\,0\,1\,0\,0\,0\,0 \\
1\,1\,1\,0\,1\,0\,0\,1\,1\,1\,0\,0\,0\,1\,0\,0\,0\,1\,0\,1\,0\,0\,0\,1
\end{bmatrix}\quad (0\leqslant i\leqslant 12,\ 0\leqslant j\leqslant 23)
$$

Figure 3-8 Knowledge base 3

According to *K*-core algorithm in Step 2, adjacent matrix, by removing nodes and edges with core less than 2, becomes matrix shown in Figure 3-9, symptoms with more affect on differential diagnostics(Difference of Qi and Yin) decreased from 24 to 9, including: burning sensation of five centers, less verbal communication, dry feces, dark urine, oliguresis, tastelessness, varicose sublingual vein, teeth-printed tongue, thin white fur of tongue. The result is consistent with literal-based differential diagnostics criteria of postoperative patients with breast cancer on *Contemporary Mastology in TCM*, page 737, which indicates the rationality of the algorithm.

According to Step 3, after assigning the confirmation degree of *Q* predicate Q_i to one-predicate P_j of Q_i according to occurrence number of P_j, adjacent matrix of weight graph is obtained as Figure 3-10 shown.

$$A=\begin{bmatrix}
1&0&0&1&0&0&0&1&1&1&1&0&0&0&0&1&0&1&0&1&0&0&0&0\\
0&0&0&1&0&0&0&1&1&1&1&0&0&0&0&1&0&1&0&1&0&0&0&0\\
1&0&0&1&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&1&0&0&0&1&1&1&1&0&0&0&0&1&0&1&0&0&0&0&0&0\\
1&0&0&0&0&0&0&1&1&0&1&0&1&0&0&0&0&1&0&1&0&0&0&0\\
1&0&0&1&0&0&0&1&1&1&1&0&0&0&0&0&0&0&0&1&0&1&0&0\\
1&0&0&1&0&0&0&0&1&1&0&0&0&0&0&0&0&1&0&1&0&0&0&0\\
0&0&0&1&0&0&0&1&1&1&1&0&0&0&0&0&1&0&1&0&0&0&0&0\\
1&0&0&1&0&0&0&0&1&1&1&0&0&0&0&1&0&1&0&0&0&0&0&0\\
1&0&0&1&0&0&0&1&1&1&1&0&0&0&0&1&0&0&0&1&0&0&0&0\\
1&0&0&1&0&0&0&1&1&1&1&0&0&0&0&1&0&0&0&1&0&0&0&0\\
1&0&0&1&0&0&0&1&0&1&1&0&0&0&0&1&0&1&0&1&0&0&0&0\\
1&0&0&1&0&0&0&0&1&1&1&0&0&0&0&1&0&1&0&1&0&0&0&0\\
1&0&0&0&0&0&0&1&1&0&1&0&0&0&0&0&0&1&0&1&0&0&0&0
\end{bmatrix}\quad(0\leqslant i\leqslant 12,\ 0\leqslant j\leqslant 23)$$

Figure 3-9　Adjacent matrix after *K*-core algorithm

$$A=\begin{bmatrix}
0.0852&0&0&0.0852&0&0&0&0.0775&0.0930&0.0930&0.0697&0&0&0&0&0.0697&0&0.0852&0&0.0775&0&0&0&0\\
0&0&0&0.0052&0&0&0&0.0048&0.0057&0.0057&0.0043&0&0&0&0&0.0043&0&0.0052&0&0.0048&0&0&0&0\\
0.0014&0&0&0.0014&0&0&0&0.0012&0.0015&0.0015&0.0011&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0.0010&0&0&0.0010&0&0&0&0.0009&0.0011&0.0011&0.0008&0&0&0&0&0.0008&0&0.0010&0&0&0&0&0&0\\
0.0092&0&0&0&0&0&0&0.0083&0.0100&0&0.0075&0&0&0&0&0.0075&0&0.0092&0&0.0083&0&0&0&0\\
0.0086&0&0&0.0086&0&0&0&0.0078&0.0094&0.0094&0&0&0&0&0&0&0&0.0086&0&0.0078&0&0&0&0\\
0.0013&0&0&0.0013&0&0&0&0&0.0014&0.0014&0&0&0&0&0&0&0&0.0013&0&0.0012&0&0&0&0\\
0&0&0&0.0012&0&0&0&0.0011&0.0013&0.0013&0&0&0&0&0&0.0010&0&0.0012&0&0.0011&0&0&0&0\\
0.0023&0&0&0.0023&0&0&0&0&0.0026&0.0026&0.0019&0&0&0&0&0.0019&0&0.0023&0&0&0&0&0&0\\
0.0021&0&0&0.0021&0&0&0&0.0019&0.0023&0.0023&0.0017&0&0&0&0&0.0017&0&0&0&0.0019&0&0&0&0\\
0.0016&0&0&0.0016&0&0&0&0.0014&0&0.0017&0.0013&0&0&0&0&0.0013&0&0.0016&0&0.0014&0&0&0&0\\
0.0016&0&0&0.0016&0&0&0&0&0.0017&0.0017&0.0013&0&0&0&0&0.0013&0&0.0016&0&0.0014&0&0&0&0\\
0.0027&0&0&0&0&0&0&0.0024&0.0029&0.0029&0&0&0&0&0&0&0&0.0027&0&0.0024&0&0&0&0
\end{bmatrix}\quad(0\leqslant i\leqslant 12,\ 0\leqslant j\leqslant 23)$$

Figure 3-10　Weight adjacent matrix

By respectively sum and normalize the value in corresponding column of one-predicate *P*, the confirmation degree of the 9 *P* predicates to differential diagnostics (difficiency of Qi and Yin) is obtained, among which the 9 *P* predicates are respectively corresponding to 9 symptoms as follows: burning sensation of five centers, less verbal communication, dry feces, dark urine, oliguresis, tastelessness, varicose sublingual vein, teeth-printed tongue, thin white fur of tongue, as Table 3-2 shows.

Table 3-2　Symptoms and one-predicate *P*

Symptom	One-predicate (*P*)
burning sensation of five centers	0.1170
less verbal communication	0.1115
dry feces	0.1073
dark urine	0.132

Continued

Symptom	One-predicate (*P*)
oliguresis	0.1246
tastelessness	0.0896
varicose sublingual vein	0.0894
teeth-printed tongue	0.1199
thin white fur of tongue	0.1078

The result indicates that urine condition is of greater influence on diagnosis of deficiency of Qi and Yin. The above data will further be applied into reasoning in following reasoning module.

3.3 Analysis on Interactive Structure of Knowledge Acquisition

During knowledge base construction of expert system, interview is the main method to acquire experience and knowledge of domain expert. However, influenced by various human factors, the knowledge acquired by interview is usually of relatively serious features of casualness, incompleteness and skew distribution. Literature analysis indicates that, the existing knowledge acquisition thoughts attach grate importance to mining, as the problems of knowledge acquisition are partly solved by data mining and knowledge discovery, and on basis of basic research on association rule data mining, data mining scale gradually expand to various application fields of complex type data. But there isn't any efficient method for acquisition of knowledge, which is of multi-level relations, and complex structures. Meanwhile, it is necessary to prepare knowledge environment before the mining and the enlightened simulations for the process. In view of cognitive science and knowledge engineering, it is more effective to acquire knowledge based on the idea of construction, i.e. making full use of existing knowledge system or meta knowledge system to acquire the unknown relations. Beginning from structural analysis, knowledge construction is accomplished dynamically by knowledge structural modeling. On basis of existing meta-knowledge, dynamic interactions are applied to new knowledge acquisition.

3.3.1 Relevant Work

Data mining is the main technique to acquire knowledge, research on mining of structural knowledge is relatively mature, while multi-level, structural knowledge is mainly acquired by graph-based algorithms, which are obviously stemed from idea of data mining. According to Common KADS, knowledge engineering is regarded as modeling activities, and modeling is the description of just several aspects of knowledge. Conventional knowledge construction methods include expert interview and model-based knowledge acquisition; another kind of methods are ontology-based, i.e. using ontology knowledge base as the existing knowledge, acquire knowledge by information abstraction. These methods obtained relatively good results in their own adaptation range, but are limited for knowledge of deep-level or implicit or small sample data, and there isn't any other effect method till now.

From the prospect of knowledge construction, knowledge which represents objective facts construct complex knowledge network, and reflect the relations between concepts. The process of knowledge construction and knowledge acquisition is the increasing of new concept nodes and concept relations. There are difficulties for the existing data mining techniques to acquire knowledge which has knowledge structure. According to the theory of system structural modeling, it is effective to study the relations between concepts in knowledge structure by system analysis. Results of structural analysis embody in the structure model, which represent the evidentization of cognition results or complex knowledge structure. Structure model describes relations between system compositions, and records qualitative cognition of human beings, and as a media in the cognition process, it also promotes further cognition. As to knowledge structure analysis, it is a process of analyzing, studying and acquiring knowledge.

Construction of certain disease knowledge can help to reappear the clinical scene of famous veteran doctors of TCM, explore syndrome differention process, and then acquire diagnosis and treatment knowledge. Structural modeling is the process of knowledge acquisition by structural analysis, that is a process of transforming implicit thinking into dominant knowledge construction and acquisition, and the commonly used structural transformations methods include decomposition, enumeration, aggregation, structuralization, etc.. The analysis of structural modeling is similar in principles to TCM records interpretation. According to relations between internal knowledge nodes in TCM records, relations between other unknown knowledge nodes can be deduced. Here structural modeling analysis is applied to analysis of TCM records, and the effectiveness of this method is verified by experimental research of implicit knowledge acquisition.