知識獲得支援のためのグループウェア GRAPEにおける仮説構造化

上田晴康 國藤進 (富士通(株) 国際情報社会科学研究所)
井深克憲 須永知之 岩内雅直 ((株) 富士通ソーシアルサイエンスラボラトリ)

1 はじめに

本論文では、知識システム構築最大のボトルネックである知識獲得ボトルネック [LPS86] を解消するために構築中のグループウェア GRAPE(Group knowledge Acquiring & Processing Environment) の仮説構造化の方法について報告する。GRAPE は、前報 [國藤 89a, 國藤 89b] で述べたように、初期知識ベース獲得機能と計画問題支援機能からなる。前者は現在インプリメンテーション中で、後者は基本概念を検討中である。GRAPE の初期知識ベース獲得機能は基本的に三つのモジュール (仮説構造化、属性構造化、および評価構造決定) からなる。本シンポジウムでは、このうち属性構造化を次報 [國藤 90] で、仮設構造化を本報で詳述する。

GRAPE では、評価構造決定部に AHP(Analytic Hierachy Process) [Saa80, 刀根 86] を用いる。AHP を利用する理由は、評価構造全体の整合性をとるメジャーが与えられていることである。AHP の欠点は、階層構造決定の方法論やツールを提供していないこと、特に評価対象である仮説の数が多くなった場合の方法論を明示していないことである。そこで、GRAPE の初期知識ベース獲得機能の第一モジュールとして仮説の構造化を支援することにした。

仮説構造化モジュールでは、仮説が葉についた木構造の知識獲得を支援する。木構造の分枝部で AHP を行なえば、多数の仮説を扱えないという問題点は回避できる。このとき獲得される木構造は、直観的に分かりやすく、再利用可能な知識であることが望ましい。GRAPE ではこの条件を満たす知識として、is-a 関係からなる階層木を獲得する。

グループの各参画者に自由に is-a 構造やそのクラス名を入力させた場合、後からの合意形成は困難であると予想される。そこで、GRAPE の仮説構造化モジュールではファジイ類似度行列を用いた手法を用いる。is-a 階層のクラス名は、次の属性構造化モジュール [國藤 90] で入力させる。

本報では、仮説構造化モジュールで用いたファジイ類似度行列の性質と、複数のファジイ類似度行列の合成アルゴリズムについて詳述する。

2 仮説構造化

仮説を属性と値で表現したとき、値の等しい属性が多いほど類似していると考えられる。属性の値の継承関係が is-a 関係なので、仮説間の類似性から is-a 関係からなる階層木を作ることができる。GRAPE の仮説構造化モジュールでは、全参画者から類似性を入力し、それをもとにグループ全体の意見を反映した階層木を作る。

類似性を入力をする際に、以下の制限を考える。

1. 全員の意見が反映されること。
2. 入力の回数が、極度に多くならないこと。

類似性を入力する手法として、2つのやり方が考えられる。

- KJ 法のように近い仮説同士をくくり出して、木を直接作るやり方
- 類似性を数値にして行列として入力し、行列から木を構成するやり方

WYSIWIS 原則 [SFB$^+$87] にのっとって木構造をつくり出していくのは、参画者が仮説数の回数の操作をすれば木が構成され、直観的で早い。しかし入力や修正を各人が行なうため、修正権の設定などが必要となり、全員の意見を反映するのが難しい。

類似性を表現する行列で入力する方法は、平均を求めたり差を見つけて交渉点を決めるなどの操作をしやすい。しかし、仮説数の2乗個の入力データが必要であり、多くの入力を必要とする。また、木構造への変換方法は、自明ではない。

GRAPE ではファジイ類似度行列を用いる。ファジイ類似度行列が以下の性質を持っているためである。

1. 木構造に対して相互に変換可能。
2. 強い制約があって、仮説の個数程度の回数の入力で行列全体の値が決まる。

また、

3. 複数の行列をマージして一つの行列を求めるアルゴリズムを開発できる。

3 類似度

3.1 ファジイ類似度行列

ファジイ類似度行列は、次の4つの条件を満たしたものである。 • 全要素が0から1の値をとる • 推移性を満たしている • 対角成分が1 • 対称行列。

定義 1 (推移性) ある行列 $\{a_{ij}\}$ が推移性を満たすとは次式が成り立つ場合である。

$$\forall ij\ (a_{ij} \geq \max_k(\min(a_{ik}, a_{kj}))) \tag{1}$$

□

推移性は次のようにいい換えることもできる。

$$\forall ij\ (\ \forall k(a_{ij} \geq \min(a_{ik}, a_{kj}))) \tag{2}$$

推移性は、類似度行列を木構造に変形するために必要な条件である。

3.2 類似度行列と木構造

ファジイ類似度行列と木構造は、相互に変換することができる [水元88]。

• 類似度行列から木構造を作るには、まず任意のレベル $\alpha \in [0,1]$ で α カットしたレベル行列を作り、次に同じ行を持つ行同士をクラスタリングして、木構造にする。

各 α におけるクラスタリングの包含関係から木を作ることができる。

• 木構造から行列を作るには、各分岐点におけるレベル行列を作り、分解定理を適用する。

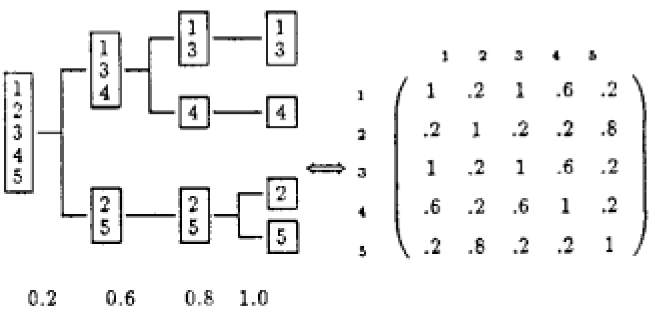


図 1: 類似度行列と木構造の変換例

3.3 類似度行列の性質

ファジイ類似度行列には、自由度が行列の次数しかないとう性質がある。これは、推移性が極めて強い制約となっているためである。

定理 1 類似度行列は、その要素の値として高々 n 種類の値しか採らない。ただし、n は行列の次数である。

証明 類似度行列は、一意に木構造にすることができる。こうして作られた木構造の深さは高々 n である。この木構造からもとの類似度行列を構成することを考える。α- レベル行列は高々 n 個でき、分解定理を用いて、これらの α- レベル行列のファジイ行列の和を作る。ファジイ行列の和の性質より、この行列は要素として高々 n 個の α のいずれかの値をとる。 ∎

さらに推移性をインクリメンタルに調べるために、次の定理が有用である。

定理 2 類似度行列 $\{a_{ij}\}$ では、次式が成り立つ。

$$\forall ijk\ (\quad (a_{ij} = a_{jk} = a_{ki}) \vee (a_{ij} = a_{jk} < a_{ki}) \vee \\ (a_{jk} = a_{ki} < a_{ij}) \vee (a_{ki} = a_{ij} < a_{jk})) \tag{3}$$

すなわち a_{ij}, a_{jk}, a_{ki} のうち少なくとも二つは値が等しく、他の一つはこれらと等しいか大きい。

証明 a_{ij}, a_{jk}, a_{ki} を適当に並べ変えて式 (4) を満たすようにする。

$$a_1 \leq a_2 \leq a_3 \tag{4}$$

ここで、 $a_1 \geq a_2$ を示せば良い。$\{a_{ij}\}$ が対称行列なので、式 (2) より、

$$a_1 \geq min(a_2, a_3) \tag{5}$$
$$a_2 \geq min(a_1, a_3) \tag{6}$$
$$a_3 \geq min(a_1, a_2) \tag{7}$$

がいえる。

式 (7) は式 (4) より明らかに真。式 (6) も $a_1 \leq a_3$ なので、$a_2 \geq min(a_1, a_3) = a_1$ となり真。

式 (5) については、$a_1 \geq a_2$ なので、

$$a_1 \geq min(a_2, a_3) = a_2 \tag{8}$$

よって、

$$a_1 = a_2 \leq a_3 \tag{9}$$

∎

定理2 より、a_2, a_3 の値が決まっていて、$a_2 \neq a_3$ ならば、$a_1 = min(a_2, a_3)$ として、値が定まる。また、$a_2 = a_3$ ならば、$a_1 < a_2$ でなくてはならない。

このように推移性を満たそうとすると多くの値は自動的に定まる。

大切なのは要素の値を大きい方から順に行列に埋めて、そのたびに推移性から推論される値を埋めていき、推移性を満たした行列を構成することである。

定理 3 類似度行列があって、その行列の要素が部分的に値が定まっている。この行列は推移性を満たしていて、定理 *2* で決まる値はすべて定まっているとする。

行列の要素のどれよりも大きくない値を新たに定めると、定理 *2* で決まる値をすべて定めることができる。

そこで、できた行列は再び推移性を満たしたまま定理 *2* で決まる値はすべて定まっている類似度行列となる。

略証 a_{ij} の値を定めるとすると、定理2 より a_{jk},a_{ki} の値が決まる。

a_{jk},a_{ki} の組合せはつぎの3通りしかなく、いずれも定理2 に反しない。

- a_{jk},a_{ki} ともに値が定まっていない。
- 片方のみ定まっている。
- $a_{jk} = a_{ki}$。

$a_{jk} \neq a_{ki}$ にはならないことを背理法を用いて示す。$a_{jk} \neq a_{ki}$ とすると、定理2より $a_{ij} = min(a_{jk}, a_{ki})$ として a_{ij} の値が定まり、a_{ij} が定まっていないことに矛盾する。 ∎

定理 4 定理 *3* の方法で値を定めたあとは、残りのすべての要素を *0* にしても必ず推移性を満足する。

定理3 のように類似度の大きい方から行列を埋めるのは、KJ 法でもっとも類似した仮説からまとめてボトムアップに木をつくってゆくことに相当する。

4 マージアルゴリズム

複数の類似度行列を合成して、一つの類似度行列を作る方法は自明ではない。参画者によって類似度のとらえ方や仮説のイメージが異なることと、純粋にマージを行なう際の計算上の困難があるためである。

前者については、*バーゲニングやグループ*での合意形成支援の手法を必要とするが、本稿では以下の2つの仮定をおき扱わない。

- 参画者全員が共通の問題意識(目標)を持っていて、同一の類似度の基準を用いる。
- 仮説は明確なもののみからなり、また参画者は知らない仮説についての類似度を入力しないで良い。

後者については次のような問題点がある。

1. 主観的な数値は、参画者によって基準が違うために、単純な比較や合成ができない。
2. 不確かな知識は入力させないので、何らかの値を補わなくてはならない。どんな値で補えば良いか。
3. 推移性のように強い制約は、単に行列の平均をとるなどの操作では満たすことができない。

以上3点を考慮して複数の類似度行列をマージして新しい類似度行列を作るアルゴリズムを開発した(付録A参照)。

4.1 マージアルゴリズムの概要

まず、全参画者の入力した類似度行列の算術平均をとった行列を作る。

算術平均行列の要素の値の大きい方から順にコピーして類似度行列を作る。ただし、そのときに定理2 を用いて、値の定まるところは定めてしまう。

定理2 では a_{ij},a_{jk} の二つの値が定まっても $a_{ij} = a_{jk}$ の場合 $a_{ki} > a_{ij}$ ということしかわからないが、本アルゴリズムでは $a_{ki} = a_{ij}$ として、値を定めることができる。これは、a_{ij} がまだ定まっていないことより、算術平均行列の ki 要素は a_{ij} よりも小さいことがわかるためである。

以上の方法で定理3 より推移性を満たした類似度行列ができる。

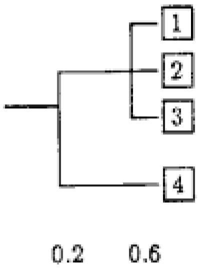
図2にマージアルゴリズムを用いて合成した例を示す。

4.1.1 知識が不確かな場合

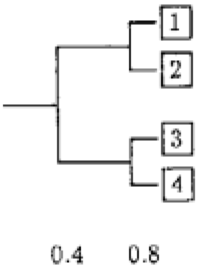
不確かな知識を無理に入力させて全体の知識の質を落とすべきではない。入力がなかった場合のデフォルトの値としては、推移性を満たす値を用いなくてはならない。定理4 より、0は推移性を必ず満たすので、GRAPE では 0をデフォルト値として用いる。このことは、参画者の良く知らない仮説に関する類似度として0を主張したことに相当する。

4.2 マージアルゴリズムの性質

このアルゴリズムで構造の同じ木をマージすると、同じ構造の木をつくり出す。木の同型性を定義した後、同じ構

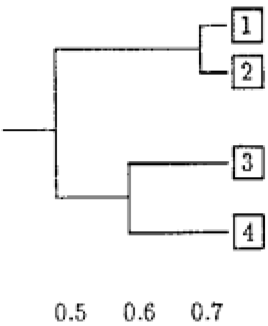


$$= \begin{pmatrix} 1 & .6 & .6 & .2 \\ & 1 & .6 & .2 \\ & & 1 & .2 \\ & & & 1 \end{pmatrix}$$



$$= \begin{pmatrix} 1 & .8 & .4 & .4 \\ & 1 & .4 & .4 \\ & & 1 & .8 \\ & & & 1 \end{pmatrix}$$

$\Downarrow$ 算術平均 $\begin{pmatrix} 1 & .7 & .5 & .3 \\ & 1 & .5 & .3 \\ & & 1 & .6 \\ & & & 1 \end{pmatrix}$



$$= \begin{pmatrix} 1 & .7 & .5 & .5 \\ & 1 & .5 & .5 \\ & & 1 & .6 \\ & & & 1 \end{pmatrix}$$

図 2: 類似度行列のマージの例

造の木をマージしたときには、定理 3 を使わないですべての値をうめることができることを示す。

定義 2 (木構造の同型性) 下図のような *2* つの木 A と B が同型である *(*$A \approx B$ と表記する*)* とは、以下の二つのいずれかを満たすものである。

- A も B も葉で、葉についた名前が一致する。
- A が α というレベルで *2* つの部分木 A_1, A_2 を持ち、B が β というレベルで *2* つの部分木 B_1, B_2 を持つとき、次の *2* つの条件を満たす。

1. $A_1 \approx B_1, A_2 \approx B_2$,

2. α, β はそれぞれの部分木 A_1, B_2 および B_1, B_2 の中のすべての類似度より小さいか等しい。

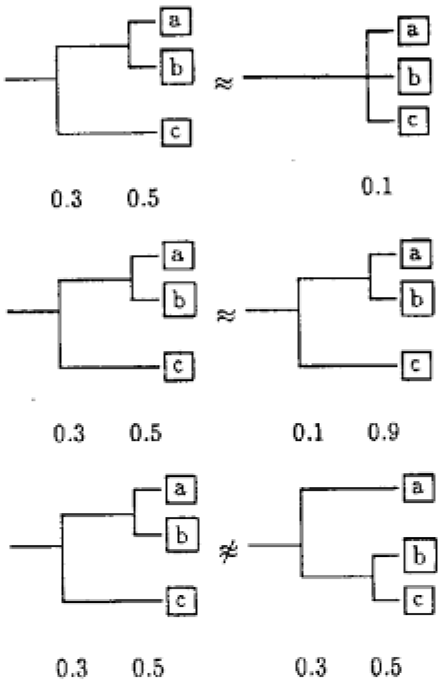
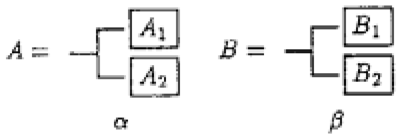
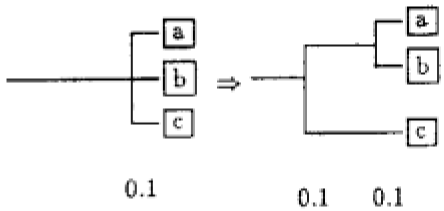


図 3: 木構造の同型性

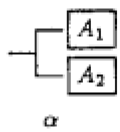
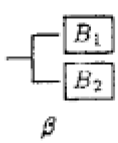


なお、n 分木と二分木の同型性は、n 分木を同じレベルからなる複数の二分木に変換することで調べる。



定理 5 複数の同型な木を表現する類似度行列の算術平均をとると、もとの木と同型な木になる。

証明 二つの木の場合で考える。

二つの木 $A =$  と $B =$  で、

$$\Updownarrow \qquad\qquad \Updownarrow$$

$$\begin{pmatrix} A_1 & \alpha \\ \alpha & A_2 \end{pmatrix} \qquad \begin{pmatrix} B_1 & \beta \\ \beta & B_2 \end{pmatrix}$$

類似度行列の算術平均を考えると、α, β は、それぞ

れ A,B の類似度のうちの最小の値である。よって、$\frac{\alpha+\beta}{2}$ は、$\frac{A_2+B_2}{2}, \frac{A_1+B_1}{2}$ のどの要素よりも大きくはない。そこで、できた行列は推移性をみたした類似度行列となる。 ∎

定理 6 付録 *A* のアルゴリズムを用いて複数の類似度行列をマージするとき定理 *2* によって値を定めた場合は、もとの類似度行列は同型ではない。

略証 定理 2 によって値を定めたならば算術平均された行列は、推移性を満たしていない。定理 5 の対偶より、もとの類似度行列は同型ではない。 ∎

5 まとめ

仮説数の多いときに AHP を適用すると手間ひまがかかりすぎ、また一貫性を保持することがほとんど不可能になるという問題があったが、仮説を木に構造化して AHP の適用範囲の中に収めることにより解決した。

木構造を作るための知識は、グループから仮説間の類似性として獲得される。グループからの複数の知識を一つの知識にまとめあげるのは難しいが、本稿では、類似度行列のマージに関するアルゴリズムを開発した。

得られた木構造は汎用性に富んだ is-a 階層を表現する知識であり、GRAPE の計画問題支援機能でも使えると考えられる。

今後の問題点として、以下のものがあげられる。

- 不確かな値のデフォルトとして、0 を用いた。こうすると信頼度が下がるとともに類似度も下がっていく。

もう一つの方法として、平均をとるときに、入力のあった類似度のみを使うという方法がある。この方法では、信頼度が下がっても類似度は変わらない。

信頼度の低下をどのように補間するか、信頼度を下げずにマージするアルゴリズムがないかが今後の課題である。

謝辞

本研究の一部は、第 5 世代コンピュータプロジェクトの一環として行なわれました。なお、貴重な助言を京都大学の西田助教授と嶋田助手にいただきました。

参考文献

[LPS86] D. Lenat, M. Prakash, and M. Shepherd. CYC: Using Common Sense Knowledge to Overcome Brittleness and Knowledge Acquisition Bottlenecks. *AI Magazine*, Vol. 6, No. 4, pp. 65–85, 1986.

[Saa80] Thomas L. Saaty. *The Analytic Hierarchy Process*. British Library Cataloguing in Publication Data. McGraw-Hill, 1980.

[SFB+87] M. Stefik, G. Foster, D. G. Bobrow, K. Kahn, S. Lanning, and L. Suchman. Beyond the chalkboard: Computer support for collaboration and problem solving in meetings. *CACM*, Vol. 30, No. 1, pp. 32–47, January 1987.

[國藤 89a] 國藤進, 上田晴康, 須永知之, 井深克憲, 岩内雅直. グループ知識獲得支援ツール GRAPE 構想. 第 10 回知識工学シンポジウム, 北海道大学, October 1989. 計測自動制御学会.

[國藤 89b] 國藤進, 上田晴康, 須永知之, 井深克憲, 岩内雅直. グループ知識獲得支援システム GRAPE における初期知識ベース獲得機能. 人工知能学会ヒューマンインターフェースと認知モデル研究会 (第 8 回), アルカディア市ヶ谷私学会館, December 1989. 人工知能学会.

[國藤 90] 國藤進, 上田晴康, 須永知之, 井深克憲, 岩内雅直. 知識獲得支援のためのグループウェア GRAPE における属性構造化. 第 11 回知識・知能システムシンポジウム, 国立教育会館, March 1990. 計測自動制御学会.

[刀根 86] 刀根薫. ゲーム感覚意思決定法. 日科技連, 1986.

[水元 88] 水元雅晴. ファジイ類似関係, 3.4 節, pp. 53–66. サイエンス社, 1988.

付録

A 類似度行列マージアルゴリズム

Step 1 n 個の次数 m の類似度行列を $A_k = \{a_{k_{ij}}\}$, $k = [1 \ldots n]$, $i, j = [1 \ldots m]$ とする。
ただし、$a_{k_{ij}}$ の値が定まっていないことがあれば、既定値として 0 を用いる。

Step 2 新らしい行列 C を作る。C の要素 c_{ij} は、$c_{ij} = \frac{\sum_{k=1}^{n} a_{k_{ij}}}{n}$ によって定める。

Step 3 C の全要素を調べて大きい順に並べてリストを作る。

Step 4 新しい行列 D を作る。D の各要素を定めるために、Step 3 で作られたリストの大きい方から順に取り出して x とし、Step 4.1 〜 Step 4.2 を繰り返す。

Step 4.1 x と同じ値を持つ要素を行列 C から全て探しだし、D の同じ添え字の要素に x を代入する。

Step 4.2 Step 4.1 で D に代入された全要素に関して、推移性をチェックして値の定まる部分については値を代入する。(d_{ij} に値をいれたときには、全ての k に対して d_{ik} を調べ、d_{ik} に値が定まっていたら d_{jk} の値を定める。)
値の定め方は次の通り。

d_{jk} の値も定まっている場合:
推移性を満足している。

d_{jk} の値が未定で $d_{ij} = d_{ik}$ の場合:
行列 D は、行列 C の値の大きい要素から順に転写して定めるので、d_{ik} の値が定まっていて d_{jk} の値が未定のときは、$c_{jk} < x = d_{ik}$。
一方定理 2 より、推移性を満足するには $d_{jk} \geq d_{ik} = d_{ij}$ でなくてはならない。
このため c_{jk} を d_{jk} の値とすることはできない。
そこで d_{jk} をその採りうる最小の値にする。すなわち $d_{jk} = d_{ij} = d_{ik}$。

d_{jk} の値が未定で $d_{ij} \neq d_{ik}$ の場合:
定理 2 より $d_{jk} = min(d_{ij}, d_{ik})$。

Step 5 終了