

## 相違度を利用したリスト解答型問題のスコアリング法

小方 博之<sup>\*1</sup>, 山下 実<sup>\*2</sup>

### Scoring Method of List Response Questions Based on Dissimilarity of Answers

Hiroyuki OGATA<sup>\*1</sup>, Minoru YAMASHITA<sup>\*2</sup>

**ABSTRACT** : This paper discusses on a scoring method of list response question of which response is a list of elements, and the elements are completely ordered. List response questions include questions such as sequencing and application-operation type. Few papers focus attention on efficient scoring method when the correct answers are combinationally redundant, or on scoring method considering seriousness of mistakes. This paper deals with these problems in the case that the dissimilarity criterion is used for scoring. To realize an efficient scoring when plural correct answers exist, a method to construct a scoring operation graph from the correct answers and the examinee's answer, and to reduce it to solve the minimal cost problem is proposed. A heuristic function to solve the minimal cost problem using A\* algorithm is also presented. When considering the seriousness of mistakes, deciding the degrees of penalty is an important problem. Here, we consider estimating the degrees from human scores, and decided them using regression model. A cooking task is taken for an example to verify the validity of our method.

**Keywords** : list response questions, sequencing, scoring using dissimilarity criterion, scoring operation graph, minimal cost problem

(Received September 18, 2013)

### 1. はじめに

ものづくりの分野においては、熟練技能者の減少が深刻であり、その技能を後継者にいかに効率的に伝承するかが問題となっている<sup>9,10</sup>。企業では、技能のマニュアル化やビデオ映像としての記録化を進めているところもある。しかし、技能をマスターするためには、単に学習者がマニュアルやビデオに目を通すだけでなく、課題を実践し、評価され、フィードバックされることによってその技能を高めていく必要がある。

技能はさまざまな要素から構成されるが、重要視される要素のひとつに段取りがある<sup>9</sup>。「段取り八分」という言葉があることから、その重要性がうかがえるが、段取りは、熟練者にとっては習慣的作業となっており、無意識に行っていることも珍しくない。

本論文では、段取りのような技能を評価することをね

らい、リストの形で解答を行う「リスト解答型」の問題を扱う。最も典型的な問題としては、課題に対して、それを達成する順序を選択肢から解答するものが挙げられる。

リスト解答型問題は並べ替え形式、端末操作形式、一部の役割演技形式や実技形式の問題を含むものと捉えられる。課題に対する解答が選ばれた選択肢のシーケンスになる点で、一般の選択問題の単純な拡張であるが、選択問題では困難な様々な出題が可能になる。例えば、一般の選択問題では、実用の上で選択肢の数に制限を設けねばならず、選択肢を見て正解を導出できてしまう問題には不向きだが、リスト解答型問題では選択肢を選ぶ組み合わせによって膨大な解答パターンを生成可能なので、そのような制限にとらわれにくい。このような特徴を活かして、リスト解答型問題は、作業に対する受験者の段取りのよさをはじめ、文章の読解力、状況判断・意思決定能力などのより精確な評価に活用できる。

しかし、リスト解答型問題は、マークセンス方式や

\*1 : システムデザイン学科教授 (ogata@st.seikei.ac.jp)

\*2 : エレクトロメカニクス学科卒業生 (現 THK (株))

computer based testing (CBT)等での出題が容易なのに対して、スコアリングの自動化が困難なことから、スコアリングの簡単な一部の特殊なケースの他には普及するには至っていない。並べ替え形式等は手作業の採点を行うテストでは有効なものとして一般に認識されている方法であることから、自動的なスコアリング法を確立できれば、マークセンス方式やCBTで計測できる能力の幅が広がることが期待される。

リスト解答型問題のアルゴリズム的なスコアリング法として、これまでもいくつかの方法が提案されているが、本論文ではそれらにおいて残された課題を指摘し、それらを解決する方法を論じる。

## 2. リスト解答型問題

本論文では、課題に対する解答が選ばれた選択肢のリスト構造として表現できる問題をリスト解答型問題と名付ける。リスト構造になっているということは、解答が単に選ばれた選択肢の集合ではなく、選ばれた選択肢の間に完全な順序性があることが特徴である。また、本論文ではリストの長さは固定されているという制約はおかない。

リスト解答型問題に関連して本論文で使用する用語の定義は以下の通りである。

答案：受験者が提出し、スコアリングの対象となる解答  
 正解：採点者側の用意する正しい解答

要素：答案や正解のリストを構成する元

選択肢：要素の取り得る値・記号等の候補

選択肢集合：ある要素に対して受験者に提示される選択肢の集合

リスト解答型とは、得られる解答の表現に着目した分類といえる。受験者が解答を行うインタフェースに着目した場合には以下のような形式の問題を含めることができる。

### ・並べ替え形式

選択肢の一部または全部を正しい順序に並べることで解答を構成する形式。例えば、選択肢として示された英単語や語句を並べ替えることで正しい英文を作る問題が挙げられる。

### ・端末操作形式

アプリケーション上で、キーボードやマウスを操作して、指示された作業を遂行する形式。受験者はマウスでメニューからの項目選択、ボタンクリックなどを行い、答案を作成する。MOS (Microsoft Office Specialist)<sup>10)</sup>など、情報系の資格試験でよく見られる形

式である。一連の操作が評価対象となる場合にはリスト解答型問題と言える。

### ・役割演技(role playing)形式

場面ごとにプレーヤーに意思決定が求められ、その選択に応じて次に提示される場面が決定される形式。OSCE (Objective Structured Clinical Examination; 客観的臨床能力試験)<sup>15)</sup>などがこの範疇に入る。意思決定場面で選択肢が提示される場合にはリスト解答型問題に含められる。

### ・実技形式

受験者が実環境で作業を行い、その実行過程を記録し評価する形式の実技試験の中にも、リスト解答型とみなせるものが幾つかある。例えば、受験者のとる姿勢をCCDカメラで順次認識し、その結果として得られる手順から心肺蘇生法の知識を評価するシステム<sup>9)</sup>や、茶器の使用順序をICタグリーダで認識し、中国茶の淹れ方の知識を評価するシステム<sup>11)</sup>などがこの範疇に属する。

端末操作形式の問題では、各時点で操作可能なwidget (ボタン、チェックボックス等) が選択肢の役割を果たす。一般の並べ替え形式と比較して選択肢集合が大きいことが特徴と言える。

並べ替え形式の問題ではすべての要素に対する当初の選択肢集合は同じであり、どこから解答してもよい。一方、端末操作形式や役割演技形式では、解答は先頭から順次行っていく必要があり、その履歴に依存して選択肢集合が変化する特徴がある。

また、並べ替え形式等では一般に選択肢集合が明示されており、受験者はその中から能動的に選択肢を選ぶ。一方、文献9)の実技評価システムでは、受験者の姿勢が、あらかじめ用意されたカテゴリに分類されることによって認識が行われる。ここでは、各カテゴリが選択肢に相当し、選択肢集合は受験者には明示されず、選択は受動的に行われると解釈できる。

## 3. 関連研究

### 3.1 従来のスコアリング法

アルゴリズム的にスコアリングを行う方法は、これまでいくつかの種類の問題に対して提案されている。

文献7)では英作文問題に対して、正解と答案との近似度を定義し、それをを用いてスコアリングする方法が示されている。具体的には、2つの文の間で一致する単語の間に線を引き、交差せずに引ける最大本数を近似度とし

ている。

文献1)では文の並べ替え問題に対して次の5種類のスコアリング法を挙げ、後ろの4つの方法について信頼性と妥当性の検証を行っている。

- 方法1：答案が正解と完全に一致する場合にのみスコアを与える方法
- 方法2：答案の要素の位置が正解のそれと一致する数に基づきスコアを与える方法 (exact match)
- 方法3：答案と正解に共通に出現する2個の要素の並びの数に基づきスコアを与える方法 (full pair)
- 方法4：要素の位置が正しい位置にある場合に1点を与え、更に直前と直後の要素が正解のものと同じ場合にそれぞれ1点ずつ加算する方法。先頭と末尾の要素にはそれぞれ直前・直後の要素が存在しないので、正しく先頭・末尾にあることに対してそれぞれ1点ずつ加算する (classic)
- 方法5：classicからexact matchを引いてスコアとする方法 (addval)

また、文献10)では文献1)の方法を以下のように若干手直しして、英単語並べ替え問題に対して適用し、信頼性と妥当性の検証を行っている。

- Binary採点法：文献1)の方法1
- Exact採点法：文献1)の方法2
- Pair採点法：文献1)の方法3に、先頭と末尾の要素が正しく先頭・末尾にあることに対してそれぞれ1点ずつ加算する方法
- Combined採点法：Exact採点法とPair採点法を合計したものを得点とする方法

文献9)では、正解と答案との間で後述の相違度を定義し、DPマッチングを用いてスコアリングする方法が示されている。

これらのスコアリング法のうちbinary採点法は最も素朴な方法である。また、

- exact採点法は選択肢の位置、
- full pair採点法は選択肢の接続関係、
- 近似度を用いた方法は選択肢の前後関係、
- 相違度を用いた方法は選択肢の並び方

の正しさをそれぞれ重視したスコアリング法であり、課題の性質に応じて使い分けられるべきものといえる。本論文ではこのうち相違度を用いたスコアリング法に着目して論じる。

### 3.2 スコアリング法の課題

これらのスコアリング法においては、共通の課題が残されている。

一つは、正解が複数存在する場合のスコアリング法の効率化である。文献7)でも指摘されているように、課題によっては正解が指数オーダーのバリエーションを持つ場合が存在する。最も単純な例は、一部の要素に対して別の選択肢を当てはめた場合も正解となるケースである。これが箇所存在する場合には正解は個存在するので、正解と答案を一つ一つ比較する単純な方法では非効率である。

また、実技形式の問題等で、ある操作が任意回許容される場合が存在する。文献9)ではポットによる湯の継ぎ足し操作がそれに相当する。この際、受験者がポットを持ち上げるたびにICタグリーダがそれを検知し、要素が解答に追加されていくが、繰り返し回数が極端に多い場合でも不正解とはいえず、このような場合でも正しいスコアリングを行うためには何らかの工夫が必要である。

例に挙げたような単純なケースの場合は、それぞれアドホックな対処も可能だが、より複雑なケースを考えると、汎用かつ効率的な比較方法が求められる。

もう一つはスコアリングにおけるミスの重み付けである。ミスには軽微なものもあれば、重大なものも存在する。採点者の手作業の場合には、それを勘案して総合的にスコアをつけるのが普通である。それに対して、従来のアルゴリズム的なスコアリング法では、それが考慮されていない。

本論文では相違度を用いたスコアリング法においてこれらの課題に対処する方法を検討する。

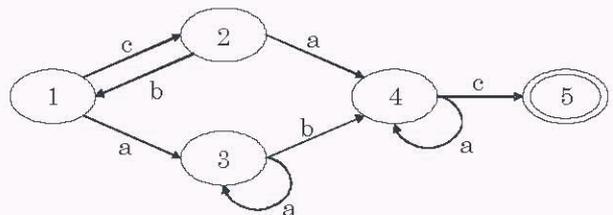


Fig.1 A directed graph of making tea



Fig.2 A directed graph of an answer of an examinee

## 4. 答案と正解のグラフ表現

答案はリスト構造なので、要素をノードとしたシーケンシャルな有向グラフで同等の表現が可能である。正解は、単一の場合はやはりシーケンシャルな有向グラフとなるが、複数存在する場合は分岐や閉路を有する一般的な有向グラフとなる。

例えば、Fig. 1 は紅茶を淹れる作業を有向グラフで表現したものである。ここでは、操作後に得られた状態が解答の要素として観測されることを前提としている。a, b, cはそれぞれお湯を注ぐ操作、ティーバッグをカップに入れる操作、ティーバッグを取り出す操作であり、1~5の数字は各操作の結果として生じる状態を表す。このような簡単な例でも分岐や閉路が随所に見られる。Fig. 2 に受験者の答案の有向グラフ表現例を示す。

答案のグラフでは出次数が0のノードが1つのみ存在し、それが終了ノード、すなわち答案の末尾となる。正解のグラフでは出次数が0のノードは必ず終了ノードとなるが、出次数が1以上のノードで終了ノードとなるものも存在する可能性がある。その場合は終了ノードの集合を明示する必要がある。Fig. 1 では終了ノードを二重丸で表示している。

以後、答案、正解を表現したグラフをそれぞれ答案グラフ、正解グラフと呼ぶ。

## 5. 相違度の計算

### 5.1 相違度の定義

相違度の考え方はstring-to-string correction problem<sup>13)</sup>に基づく。

String-to-string correction problemとは、ワードプロセッサやテキストエディタ等で、ある文字列を別の文字列に修正するのに必要な最小のキー操作数を求める問題であり、この最小操作数によって文字列間の相違度が定義される。ここで考慮されているキー操作は文字の挿入、削除、置換の3種類である。

答案と単一の正解との間の相違度も、文字列を答案または正解、文字を解答の要素に置き換えることにより定義可能である。すなわち、答案グラフを正解グラフと同一にする場合のノードの挿入、削除、置換の最小操作数が相違度である。

正解が複数存在する場合には、正解として考えられるすべての並びと答案とを比較した場合に最小となる操作数を相違度と定義する。ここでは、相違度が最小となった並びが、受験者が意図したものに最も近い正解だと仮定している。

### 5.2 答案操作グラフ

答案に対して挿入・削除・置換を行い、正解と同一にするまでの操作数は、先頭から要素のスキャンを行っていくことでアルゴリズム的に勘定できる。例えば答案「12」（最初の要素が1、次が2）を正解「13」にするための操

作数は以下のように勘定する。

- (1) 先頭の要素「1」と「1」の比較。要素が同じなので、答案、正解に対する操作は何も行わず、次の要素に進む。この時点での操作数は0である。
- (2) 次の要素「2」と「3」の比較。要素が異なるので、挿入・削除・置換のいずれかの操作が必要となる。挿入操作の場合には答案の「2」の前に「3」を挿入し、「2」は正解の次の要素との比較に回す。削除操作の場合には答案の「2」の要素を削除し、正解の「3」は答案の次の選択肢との比較に回す。置換操作の場合には答案の「2」を「3」に置換する。いずれの場合も、この時点で操作数は1となる。置換を行った場合は答案、正解いずれにも要素は残っていないので、この時点でスキャンが終了する。
- (3) 次の要素の比較。(2)で挿入を行った場合は答案の「2」と比較する正解側の要素がないので、行えるのは削除操作のみである。削除を行った場合は正解の「3」と比較する答案側の要素がないので、行えるのは挿入操作のみである。これら2つの場合には、この時点で操作数は2となる。

結局、(2)の時点で置換操作を行った場合に、操作数が最小となるので、答案と正解の間の相違度は1となる。

この操作を有向グラフで表現するとFig. 3 のようになる。各ノードはどの要素同士を比較しているかを表し、リンクは行っている操作を表す。相違度を得ることは、このグラフで、終端ノード(endで表記)までのno operation以外のリンク数の最小値を得ることに他ならない。本論文ではこのグラフを答案操作グラフと称する。

答案操作グラフは一般には非循環有向グラフ(DAG)になる。例えば、答案が「112」、正解が「12」の場合はFig. 4のような合流部分が生じるが循環はしない。

正解が複数存在する場合にも同様に答案操作グラフの構築が可能である。

例えば、上の例で正解が「13」および「14」の2通りあった場合、正解グラフはFig. 5 のようになる。最初に先頭の要素「1」と「1」を比較することは上の例と同じだが、その後ろで正解グラフが「3」と「4」に分岐しているため、次には「2」と「3」の比較と「2」と「4」の比較を行う場合が出てくる。したがって、答案操作グラフもここで分岐が生じ、Fig. 6 のようになる。

正解グラフがFig. 7 のような閉路を持つ場合にはFig. 8 のように展開する。この場合、答案操作グラフはFig. 8 のように無限の大きさをもつグラフとなる。

5.3 探索による相違度の算出

答案操作グラフ上で、終端ノードまでのno operation以外のリンクの数の最小値を求めることは最小コスト問題の枠組みで可能である。最小コスト問題とは、有向グラフにおいて  $n$  個のノード

$$E = \{e_1, \dots, e_n\}$$

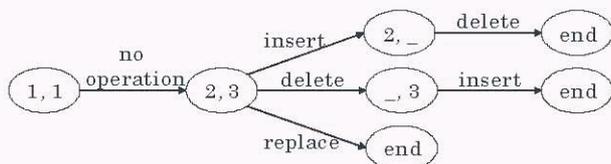


Fig. 3 An operation graph for changing "12" to "13"

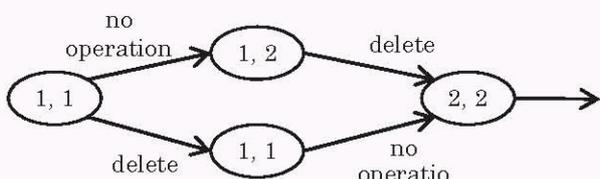


Fig. 4 A partial operation graph in a directed acyclic form

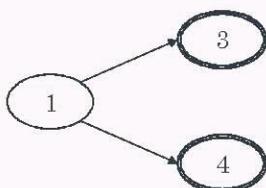


Fig. 5 A graph indicating the right answer is "13" or "14"

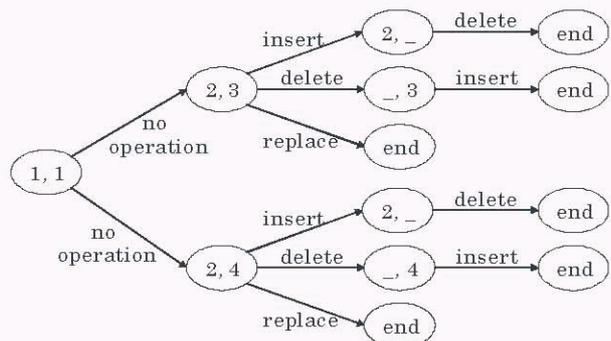


Fig. 6 An operation graph when the right answer is "13" or "14"

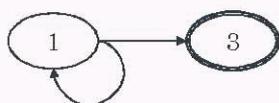


Fig. 7 An example of right-answer graph containing a loop

が存在し、ノード  $e_i$  から  $e_j$  への直結リンク  $v_{ij}$  のコストが与えられている場合（直結リンクが存在しない場合は便宜的に  $v_{ij} = \infty$  とする）、指定されたノード  $e_{start}$  から  $e_{goal}$  までのコストの合計が最小となる経路を探索する問題である。解法は確立しており、カーナビでの経路生成（例えば 文献 8）などに広く応用されている。本論文の問題に適用する場合は、no operation リンクのコストを 0、挿入・削除・置換リンクのコストを 1 と置けばよい。

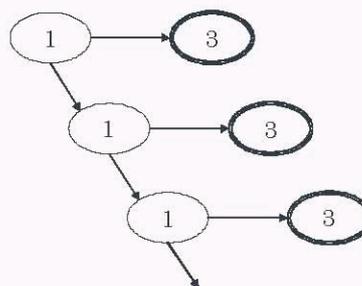


Fig. 8 An extensive form of the graph in Fig. 7

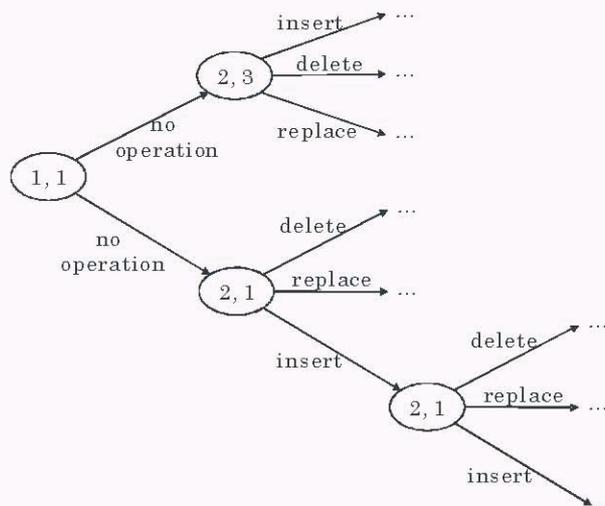


Fig. 9 An operation graph when the right answer is as in Fig. 7

本研究における答案操作グラフのような規模で、コストが非負となるグラフにおける最小コスト問題の効率的解法としては A\* アルゴリズム<sup>3), 4)</sup> やそのサブセットである Dijkstra のアルゴリズム<sup>2)</sup> が知られている。これらのアルゴリズムでは、ノードを OPEN, CLOSED, およびそれ以外の 3 種類に分類する。  $e_{start}$  からノード  $e_j$  までのコストは計算されているが、その先のノードまでのコストが計算されていない場合、  $e_j$  は OPEN に分類される。他の既探索ノードは CLOSED に分類される。未探索ノードはそれ以外に分類される。これらのアルゴリズムでは探索の順番を工夫し、OPEN, CLOSED に属す

るノードを探索対象から除外することで、効率化が行われる。初期状態では

$$OPEN = \{e_{start}\}$$

$$CLOSED = \{\}$$

である。

A\*アルゴリズムでは

$$e_j^* = \arg \min_{e_j} (g(e_j) + \hat{h}(e_j))$$

$$e_j \in \{e_j \mid v_{ij} < \infty, e_i \in OPEN, e_j \notin OPEN \cup CLOSED\}$$

なるノードを優先的に探索する。ここで、 $\hat{h}(e_j)$ は $e_j$ から $e_{goal}$ までの最小コストの推定値を表すヒューリスティック関数である。 $e_{goal}$ が複数ある場合は、それらの値の最小値となる。

実際の最小コストを $h(e_j)$ とした場合、

$$\hat{h}(e_j) \leq h(e_j) \tag{1}$$

が常に成り立つ時に限って、得られた経路がコスト最小になることが保証される。また、 $\hat{h}(e_j)$ が $h(e_j)$ に近いほど探索の効率がよいことが知られている。

Dijkstraのアルゴリズムはヒューリスティック関数が $\hat{h}(e_j) = 0$ なる定数関数である場合に相当する。したがって、適切なヒューリスティック関数を見つけられればA\*アルゴリズムを適用し、見つけられない場合にはDijkstraのアルゴリズムを用いる。

5. 4 ヒューリスティック関数の導出

答案操作グラフで終了ノードまでの最小コストを推定するための手掛かりにできるのは、 $e_j$ において参照している答案と正解の要素の位置である。答案と正解において、未参照の要素数をそれぞれ $m_j, n_j$ とすると、以後の操作数の最小値は $|m_j - n_j|$ と見積れる。これは、要素が一致するのはたかだか $\min(m_j, n_j)$ 個なので、最低でも残りの $|m_j - n_j|$ の要素に対して削除または挿入操作が必要だからである。したがって、

$$\hat{h}(e_j) = |m_j - n_j|$$

と設定すると、式(1)の条件を満たし、A\*アルゴリズムのヒューリスティック関数として使用できる。正解が複数ある場合には、原則的に $|m_j - n_j|$ の最小値をヒューリスティック関数とするが、求めるのが不可能な場合や、計算時間を要する場合には、Dijkstraのアルゴリズムを用いる。

なお、文献9)の正解が1通りの場合にDPマッチングを用いる方法はDijkstraのアルゴリズムを用いる方法の特殊なケースとなっている。

5. 5 グラフの展開と探索との同時実行

組み合わせ的に生成される答案操作グラフは一般に規模が大きい。また、正解グラフが開路を有するときには答案操作グラフは無限の大きさとなる。

したがって、答案操作グラフを展開した後に探索を行う逐次的な方法は効率が悪い。この場合、現実的な方法として、探索時にグラフの必要な部分のみを展開する方法を採るのが一般的である。

OPEN, CLOSEDに分類されるノードは既探索であり、グラフの展開が行われている部分である。その他の未探索ノード部分は展開がなされていない。初期状態では $OPEN = \{e_{start}\}, CLOSED = \{\}$ なので、 $e_{start}$ のみが展開されている状態である。探索時は、OPENに属するノードに直結したノードを調べていくので、その部分のみを展開する。

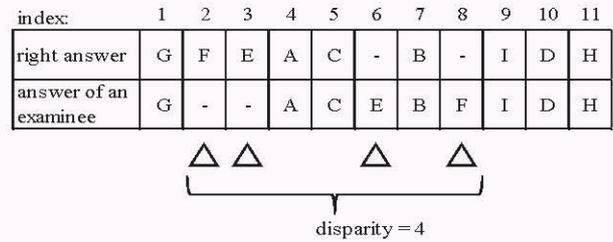


Fig.10 Comparative chart of the right and the examinee's answer

6. ミスの軽重の考慮

答案と正解の間の相違度を求めた後、ミスの軽重を考慮することでスコアリングを行う。

そのために、ここではFig. 10のように上段に相違度が最小となった正解を、下段に答案を並べ、インデックス番号を付与する。このとき、対応する要素が置換操作を行った要素が上下で並ぶようにする。また、挿入操作を行った要素については答案側にスペース (Fig. 10ではハイフンで表記) を挿入し、削除操作を行った要素については正解側にスペースを挿入する。

以下の実験では調理作業を例題とするが、調理作業では答案における要素の位置が正解の位置から離れているほどミスの度合いが大きいと考えられる。そこで、その位置の差に要素に応じた重み付けを行って総和

$$G = \sum_{i=1}^n w_i \cdot |c_i - r_i| \tag{2}$$

を求め、満点からこの値を引くことでスコアを算出する。ここで、 $n$ は要素の数、 $c_i, r_i$ はそれぞれ正解側第*i*要

素に対応する正解側インデックス番号と答案側インデックス番号,  $w_i$  は正解側第  $i$  要素に対するミスの重みである。

Fig. 10 の例でE, Fに対するミスの重みをそれぞれ 2, 3 とすると,  $G = 2 \times 3 + 3 \times 6 = 24$  となる。

ミスの重みを決定するにあたって参考になるのは, 人間の採点者の用いる値である。しかし, 人間の採点者は明示的に重みを決定して用いているのではなく, 潜在的な重みで総合的にスコアをつけているものと考えられる。そこで, 人間の採点者のスコアリングモデルも式(2)に従うものと仮定して, 採点者のスコアのサンプルから,  $w_i$  を推定することにする。これは満点からスコアを引いた値を目的変数とし, 定数項を 0 に固定した重回帰分析を用いることで可能である。

相違度が最小の正解が複数あった場合, ミスの軽重を考慮して計算したスコアが異なってくるので問題になる。これへの対処法としては

- A. 計算したスコアの中で最大または最小のものをとる
- B. 相違度が最小となる正解と答案を比較して, そのミスの発生可能性が最も高くなる正解を選び出す。その上でスコアを計算する
- C. スコアの平均値または期待値をとる

などが考えられる。重回帰分析を用いる場合, Cでは説明変数の独立性がなくなるため使用できない。BはAと比較すると, より厳密だが, ミスの発生可能性を評価する必要がある。ミスの発生可能性の評価を行わない場合には, 簡便な方法としてAを採用することが考えられる。Aでは最大・最小のどちらを採る方法も考えられるが, ここでは最大のものを採ることにする。

Table 1 Ingredients list given to the subjects

A. beef	B. eggplant
C. potato	D. tomato
E. carrot	F. onion
G. salad oil	H. curry roux
I. water	

## 7. 実験

### 7.1 実験1

最初の実験課題として正解が1通りのみのカレーの調理作業手順の評価を取り上げる。受験者にはあらかじめ, 調理に使用, 鍋への投入順序を解答してもらう。食材は

あらかじめ下記のように示されているとする。本課題で与えた食材リストをTable 1 に示す。

調理作業を課題として選択した理由は以下の通りである。

- ・調理作業には段取りの巧みさが求められ, 今回の受験者にとっても適度な難易度がある。
- ・結果(その投入順序によってどの程度おいしい料理ができるか)が, ある程度料理の知識のある素人にも判定しやすく, スコアリングの妥当性を確認しやすい。
- ・ミスに軽重が存在する。例えば, 火の通りやすい食材を先に入れると煮崩れるのでミスの程度が重い。一方, 火の通りにくい食材同士の投入順序が逆転しているのはミスの程度としては些細である。

なお, ここで正解とした手順は

G. サラダ油	→	F. タマネギ	→	E. ニンジン	→
A. 牛肉	→	C. ジャガイモ	→	B. ナス	→
I. 水	→	D. トマト	→	H. ルー	

である。

まず, 重回帰モデルを用いて重みの推定を行った。料理経験の様々な20代から40代までの男女43名に出題し, 答案のサンプルを得た。

これらの答案に対して1人の採点者によるスコアリングを行った。採点は料理経験5年以上で, 普段週に3回以上料理をしている者が担当した。採点は通常時と同様に, 正解と答案のみを見て行う。スコアリングの規準は以下の通りである。

優	…	1	良	…	2
可	…	3	不可	…	4

正解と答案の間の相違度を計算し, Fig. 10 と同様な表を作成し, 食材についてのインデックス番号差を計算することで説明変数の値を求めた。

これを重み推定用の30データと確認用の13データに分け, 前者に対して重回帰分析を行い, 重みを推定した結果をTable 2 に掲げる。重決定係数は0.85 である。また, 採点者のつけたスコアと, 回帰式から得られたスコアとの関係をプロットしたものをFig. 11 に示す。相関係数は0.84 であり, 比較的高い相関が見られた。

重みの値を見ると, 表におけるタマネギ, 肉の位置がスコアに大きな影響を与えていることがわかる。実際, みじん切りにしたタマネギや肉を他の食材の後に炒め始

めた場合の評価は悪くなり、採点者のスコアリング規則と一致していた。サラダ油の重みは負の値という妙な結果になっているが、標準誤差の値から、真値は正である可能性が大きい。標準誤差が大きいのは、サラダ油を最初に投入していない答案が30データ中1データしか含まれていないことが原因と考えられる。

結果の汎用性を確認するために、残りの13データに対して、採点者のつけたスコアと回帰式によって計算したスコアとの比較を行った。そのプロットをFig. 12に示す。相関係数は0.79となった。

次に、本論文で提案したスコアリング法と、相違度のみを用いたスコアリング法との比較を示す。重みを求め

るのに使用した30個の答案の正解との相違度と、採点者のスコアとの関係をプロットしたものをFig. 13に示す。相関係数は-0.41と、絶対値が提案手法のものと比較して小さな値となる。また、確認用の13個の答案の正解との相違度と、採点者のスコアとの関係をプロットしたものをFig. 14に示す。この場合も、相関係数が-0.60となり、絶対値が提案手法のものと比較して小さな値となった。

7.2 実験2

次に、正解が複数ある場合の例として「椎茸ときょうりのスープ」を調理する作業を取り上げた。(レシピは文献12)による)

Table 2 Weights calculated for each ingredient

ingredient	salad oil	onion	carrot	meet	potato	eggplant	water	tomato	roux
weight	-0.08	0.27*	0.13	0.30*	0.18*	0.12	0.08	0.14	0.03
s.d.	0.35	0.07	0.11	0.09	0.08	0.12	0.11	0.11	0.08

\*p<0.05

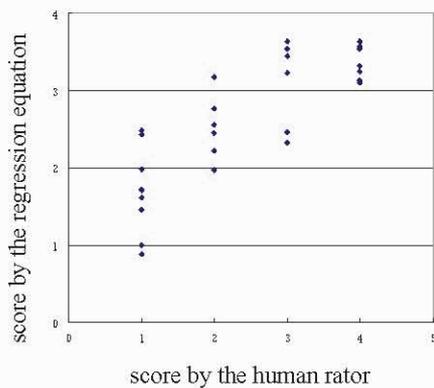


Fig. 11 Comparison of the score by the human rator and by the regression equation, when using the data used to calculate weights in experiment 1

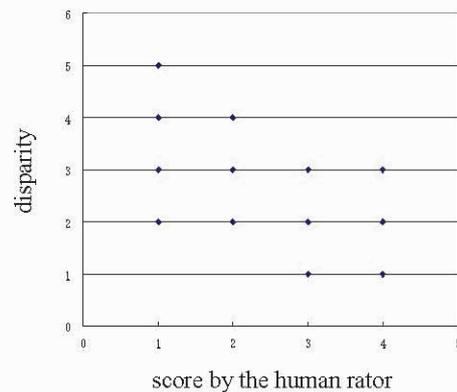


Fig. 13 Comparison of the score by the human rator and the disparity, when using the data used to calculate weights in experiment 1

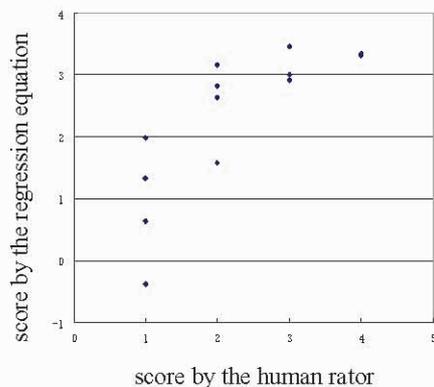


Fig. 12 Comparison of the score by the human rator and by the regression equation, when using the reference data in experiment 1

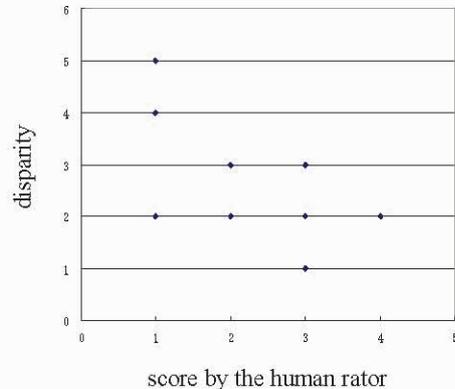


Fig. 14 Comparison of the score by the human rator and the disparity, when using the reference data in experiment 1

Table 3 Weights obtained from experiment 2

task	add mushroom	add cucumber	pour soaking liquid	boil	add meat
weight	0.14	0.13	0.09	0.29*	0.19*
s.d.	0.16	0.17	0.13	0.05	0.03

\*p<0.05

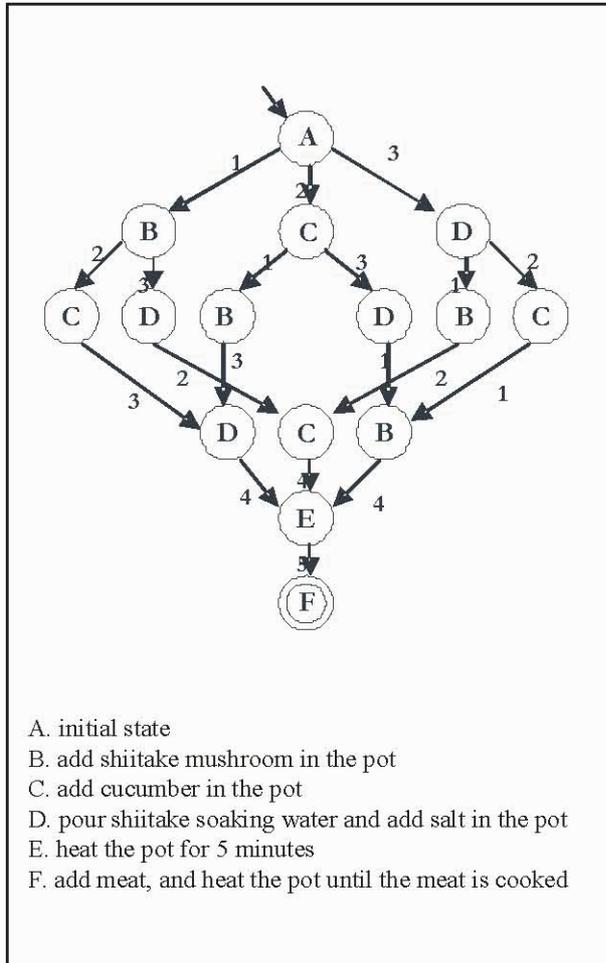


Fig. 15 Set of the alternatives for the “shiitake mushroom and cucumber soup” and the right-answer graph

選択肢集合と正解グラフをFig. 15に示す。正解グラフは椎茸ときゅうりと椎茸の戻し汁を入れる順番は入れ替わってもよいことを表している。

20代から40代までの男女41名に出題し、答案のサンプルを取得した。

採点者によるスコアリングは、実験1と同様1~4の4段階で行った。A\*アルゴリズムやDijkstraのアルゴリズムを用いて相違度を算出すると、相違度と同時に答案に最も近い正解を求めることができる。その正解と答案からFig. 10と同様な表を作成し、食材についてのインデックス番号差を計算することで説明変数の値を求めた。3

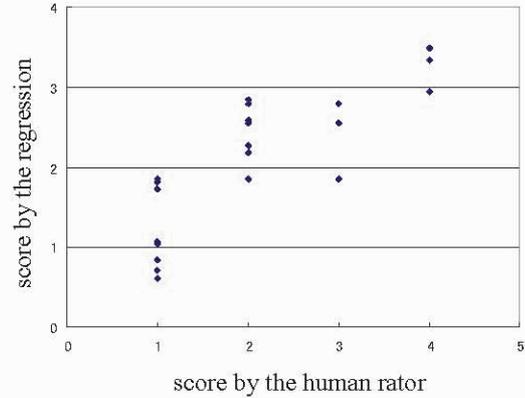


Fig. 16 Comparison of the score by the human rator and by the regression equation, when using the data used to calculate weights in experiment 2

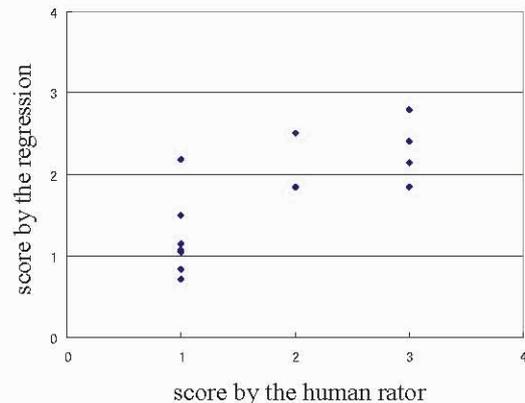


Fig. 17 Comparison of the score by the human rator and by the regression equation, when using the reference data in experiment 2

0データについて採点者のスコアを目的変数とし、重回帰分析によって推定した重みをTable 3に掲げる。重決定係数は0.72である。採点者のつけたスコアと、回帰式から得られたスコアとの関係をプロットしたものをFig. 16に示す。相関係数は0.86であった。相違度のみを用いた場合の相関係数は-0.77であった。

また、残りの11データに対して採点者と回帰式のスコアとを比較したところ、Fig. 17のようになった。相関係数は0.57であった。相違度のみを用いた場合は相関係数は-0.55であり、わずかながらミスの軽重を考慮した方が値が大きくなった。

## 8. おわりに

本論文では段取りのような技能の評価を自動的に行うことを目的に、解答がリストの形となる「リスト解答型」問題のスコアリング法について検討した。

リスト解答型問題で正解が複数ある場合、その総数は組合せ的に大きい場合が多く、効率的なスコアリング法を考える必要がある。また、採点者が手作業でスコアリングする場合には、ミスの軽重を考慮して行うことが一般的だが、従来のアルゴリズム的なスコアリング法では考慮されてこなかった。

本論文では相違度を用いた採点法を対象とし、前者の問題に対しては、答案と正解から答案操作グラフを作成し、最小コスト問題に帰着させることでA\*アルゴリズムやDijkstraのアルゴリズムで効率的にスコアを算出する方法を提案した。また、後者に対しては調理作業のような課題を想定し、採点者のスコアサンプルからミスの軽重を重回帰モデルで推定しスコアリングに利用する方法を提案した。

調理作業を例に実験を行い、採点者のスコアと回帰式から計算したスコアの間で高い相関を得ることができた。また、正解が複数存在する問題において、Dijkstraのアルゴリズムで正しく相違度を求め、スコアリングができることを確認した。

## 参考文献

- 1) J. Charles Alderson, Richard Percsich, Gabor Szabo: Sequencing as an item type, *Language Testing*, 17, 4 (2000) 423.
- 2) Edsger W. Dijkstra: A note on two problems in connexion with graphs, *Numerische Mathematik*, 1 (1959) 269.
- 3) P. E. Hart, N. J. Nilsson, B. Raphael: A Formal Basis for the Heuristic Determination of Minimum Cost Paths, *IEEE Transactions on Systems Science and Cybernetics*, 4, 2 (1968) 100.
- 4) P. E. Hart, N. J. Nilsson, B. Raphael: Correction to "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", *SIGART Newsletter*, 37 (1972) 28.
- 5) 稲崎一郎: 研削加工のスキルフリー化に向けて, *精密工学会誌*, 75, 7 (2009) 813.
- 6) 小松 研治, 小郷 直言: 段取りについての一考察, *高岡短期大学紀要*, 16 (2001) 187.
- 7) 西村 則久, 安村 通晃: 外国語作文における自動添削手法について, *情報処理学会研究報告*, 1999, 5 (1999) 1.
- 8) 西村 茂樹, 津田 新吾, 大橋 紳悟, 香川 浩司, 長尾 真伸, 松本達治: ナビゲーションシステム, *日本ロボット学会誌*, 17, 3 (1999) 29.
- 9) 小方 博之, 五十嵐 俊介: CCDカメラによる作業手順認識を利用した実技評価システム, *日本テスト学会誌*, 1, 1 (2005) 41.
- 10) 静 哲人, 竹内 理, 八島 智子, 吉澤 清美: 並べ替え問題における異なる採点法の信頼性・妥当性への影響, *全国英語教育学会全国大会* (2002).
- 11) 高村 悠, 小方 博之: ICタグを用いた実技試験の自動化, *精密工学会春季大会学術講演会予稿集*, (2005) CD-ROM 収録.
- 12) 白田 幸世: 20分で晩ごはん, *NHK 今日の料理*, 日本放送出版協会 (1996).
- 13) Robert A. Wagner, Michael J. Fischer: The String-to-String Correction Problem, *Journal of the Association for Computing Machinery*, 21, 1 (1974) 168.
- 14) 綿貫 啓一: バーチャルトレーニングと OJT を融合した鑄造技能伝承および人材育成, *精密工学会誌*, 76, 4 (2010) 382.
- 15) R. M. Harden, F. A. Gleeson: Assessment of clinical competence using an objective structured clinical examination (OSCE), *Medical Education*, 13, 1 (1979) 39.
- 16) <http://mos.odyssey-com.co.jp/>