

# 競争型情報推薦システム Recommendation Battlers とその挙動

Competitive Information Recommendation System "Recommendation Battlers" and Its Behavior

阪本 俊樹\*1      北村 泰彦\*1      辰巳 昭治\*1  
Toshiki Sakamoto      Yasuhiko Kitamura      Shoji Tatsumi

\*1 大阪市立大学大学院 工学研究科  
Graduate School of Engineering, Osaka City University

Recommendation systems draw the attention of practitioners in B-to-C electronic commerce. In an independent recommendation system such as used in amazon.com, a user cannot compare the information with that in other information sources. In a broker-mediated recommendation system, the broker takes the initiative of recommendation, so the information provider cannot recommend its information directly to the user.

In this paper, we propose a competitive information recommendation system in which multiple character agents recommend their information competitively, and apply it to the field of restaurant recommendation. Each agent recommends restaurants from its own point of view, the user tells good or bad about them. In our competitive information recommendation system, a user can compare recommended information, and the information providers can recommend their information directly to the user. Consequently, this system has an advantage for both of the user and the information providers.

## 1. はじめに

近年、B-to-C の電子商取引を支援する手段として推薦システムが注目されている。従来の情報推薦システムには、amazon.com\*1のように情報提供者が単一の情報源から利用者へ推薦を行う独立型情報推薦システムや、DealTime\*2のようにブローカが複数の情報源から情報収集して推薦を行うブローカ型情報推薦システムがある [2]。

独立型情報推薦システムでは、情報源が単一であるため、利用者は他の情報源との比較はできない。これに対しブローカ型情報推薦システムでは、利用者は複数の情報源の情報推薦をブローカから受け取ることができ複数の情報を比較できる。しかしブローカ型情報推薦システムでは、情報提供者にあった情報推薦の主導権がブローカに握られているため、情報提供者が持つ推薦基準が情報推薦に反映されず、情報提供者にとって十分なメリットがあるとは言えない。

本研究では、複数の情報源がキャラクタエージェントを介して直接利用者と相互作用を行い、競争的に利用者へ推薦を行う、競争型情報推薦システムを提案し、レストラン推薦に応用する。また競争型情報推薦システムにおいて、エージェントが非合理的な提案を行わないことを保証する合理的提案アルゴリズムを提案する。競争型情報推薦システムでは、情報提供者は独自にエージェントを開発することで独自の基準で推薦を行うことができ、また利用者は複数の情報源からの情報を比較することができる。このことから競争型情報推薦システムは、情報提供者と利用者の双方にメリットのあるシステムといえる。

以下、第2章で競争型情報推薦システムについて、第3章で合理的提案アルゴリズムについて、第4章で競争型情報推薦システムの例題であるレストラン推薦システムについて述べ、そして第5章でシミュレーションによる競争型情報推薦システムの挙動についての評価を行う。第6章では関連研

究について述べ、第7章でまとめとする。

## 2. 競争型情報推薦システム

### 2.1 システム構成

競争型情報推薦システムは、推薦ブラックボードとエージェント、ブラウザとキャラクタインタフェースによって構成される。システム構成を図1に示す。

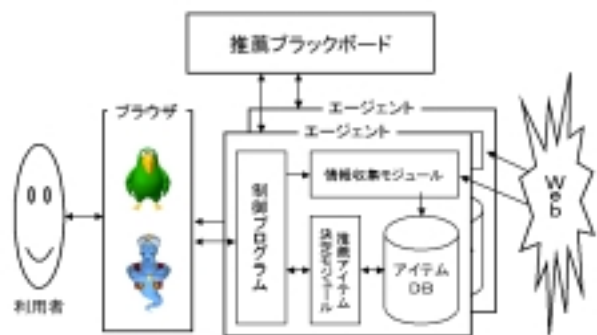


図1: 競争型情報推薦システムの構成

このシステムでは各エージェントは競争的に情報推薦を行う。競争的とは、他のエージェントが推薦した情報よりも優れた情報を推薦するということである。このためには、各エージェントは他のエージェントが推薦した情報を知る必要があり、その情報は推薦ブラックボードを参照することで得られる。推薦ブラックボードはエージェントが各情報を共有する場である。また、エージェントは関連付けられた情報源から情報集を行い利用者へ情報推薦を行う。

### 2.2 アイテム

エージェントが推薦する情報をアイテムと呼ぶことにする。一般に、買い手は商品を購入する場合、価格、性能、見た目(色や形など)の属性を考慮する必要がある。多属性効用理論 [4] では、アイテム  $C_i$  に対してその属性を  $X_1, X_2, \dots, X_n$

連絡先: 阪本俊樹, 大阪市立大学大学院工学研究科, 大阪市住吉区杉本 3-3-138, Tel:06-6605-2778, Fax:06-6605-2778, toshi@kdel.info.eng.osaka-cu.ac.jp

\*1 <http://www.amazon.com/>

\*2 <http://www.dealtime.co.jp/>

とし、それらの属性値を  $x_{i,1}, x_{i,2}, \dots, x_{i,n}$  とする。例えば、アイテム  $C_i$  の属性  $X_1$  に関する値は  $x_{i,1}$  である。また、売り手は商品を販売するとき、それによって得られる利益を考慮する。アイテム  $C_i$  に対してその属性を  $X_{n+1}$  とし、その属性値を  $x_{i,n+1}$  とする。よってアイテム  $C_i$  は式 (1) で表される。

$$C_i = \langle x_{i,1}, x_{i,2}, \dots, x_{i,n}, x_{i,n+1} \rangle \quad (1)$$

本研究では、売り手に対する属性は利益のみの 1 属性とし、買い手の属性は多属性であるとする。また、買い手の属性数が  $n$  であるようなアイテムを、 $n$  属性アイテムと呼ぶことにする。

### 2.3 利用者のモデル

商品の属性には、買い手が重点を置く属性とそうでない属性がある。これを買い手の好みと呼ぶ。買い手の持つアイテム  $C_i$  の各属性  $X_1, X_2, \dots, X_n$  に対する好みを  $w_1, w_2, \dots, w_n$  (ただし、 $\sum_{j=1}^n w_j = 1$ ,  $0 \leq w_j \leq 1$ ) とすると、アイテム  $C_i$  の買い手に対する効用  $U(C_i)$  は式 (2) で表される。

$$U(C_i) = \sum_{j=1}^n w_j u_j(x_{i,j}) \quad (2)$$

効用関数  $u_j(x_{i,j})$  は属性  $X_j$  の最悪値  $x_{worst,j}$  において最小値  $u_j(x_{worst,j}) = 0$  となり、最良値  $x_{best,j}$  において最大値  $u_j(x_{best,j}) = 1$  となるような関数である。

競争型情報推薦システムにおいて、利用者はエージェントから推薦されたアイテムに対して、効用が大きければ Good、小さければ NoGood の評価を行うことができる。

### 2.4 推薦ブラックボード

推薦ブラックボードでは、エージェント間で共有される以下の 3 つの情報を保持する。

#### (1) 利用者の要求

利用者がシステムに対して自分の好みを入力したものを、利用者の要求とする。例えば、レストラン推薦においては、「天王寺でお好み焼きを食べたい。」というのが利用者の要求である。

#### (2) 推薦アイテム

エージェントによって推薦されたアイテムである。例えばレストラン推薦においては、レストラン名と、レストランの価値を決定付ける構成要素である、平均予算、最寄駅からの距離、などの属性に対する属性値である。

#### (3) 利用者の評価

利用者が推薦アイテムに対して自分の好みに合うものであれば Good、そうでなければ NoGood と評価を行ったものである。

### 2.5 ブラウザとキャラクタインタフェース

ブラウザとキャラクタインタフェースの制御については、各エージェントによって行われる。ブラウザにはエージェントにより推薦された情報を表示し、その情報についての利用者との相互作用をキャラクタインタフェースにより行う。ブラウザやキャラクタインタフェースを介する情報は、同時に推薦ブラックボードに書き込まれ、エージェント間で共有される。

また、キャラクタインタフェースは情報推薦の視覚化に役立っている。つまり文字だけの情報推薦に比べて、キャラクタが登場して発言することにより、情報推薦の過程を利用者は簡単に見て取ることができる。またキャラクタの動きや発言内容により、利用者の興味を引くことにも役立つ。

## 2.6 エージェント

### (1) 情報収集

各エージェントは、関連付けられた情報源から利用者の要求を満たす情報を収集してアイテムデータベースの作成を行う。Web からの情報収集には、MetaCommander[1] と TemplateFilter を用いる。

MetaCommander とは、スクリプトにより Web からの情報を取得するためのツールである。これにより各エージェントは利用者の要求から独自のスクリプトを生成し、Web ページのダウンロードを行う。

TemplateFilter とはフォーマットの決まったページから情報抽出する際に、あらかじめ用意しておいたテンプレートファイルと比較することで、指定した部分の情報のみを抜き出すことのできるツールである。これにより各エージェントはダウンロードした Web ページから必要な情報を抽出する。

### (2) 推薦アイテムの決定

エージェントが推薦を行う目的には次の 2 つがある。

- 可能な限り得られる利益を大きくする
- 他のエージェントに打ち勝つ

利益を大きくするには、2.2 節で述べた  $x_{i,n+1}$  に対する効用ができるだけ大きな値となるアイテムを推薦すればよい。アイテム  $C_i$  に対するエージェントの効用  $U_s(C_i)$  は式 (3) で表される。

$$U_s(C_i) = u_s(x_{i,n+1}) \quad (3)$$

また情報提供者は独自にエージェントを開発できるので、他のエージェントに打ち勝つ戦略を独自に作成できる。

### (3) ブラウザとキャラクタインタフェースの制御

エージェントは推薦情報を利用者に表示するため、ブラウザを制御し画面の切り替えを行う必要がある。1 つのエージェントが一度に推薦できる情報は 1 つだけであり、新たな情報が推薦されるたびに画面は切り替えられていくことになる。

エージェントはブラウザの制御をするとともに、キャラクタインタフェースの制御も行う。これはキャラクタに推薦情報についてのコメントを発言させるものである。また、推薦情報についての利用者の評価を得る必要がある場合、キャラクタは利用者へ評価を問う。

## 2.7 処理の流れ

次にこのシステムにおける情報推薦の流れを示す。

1. 利用者からの要求を受け取り、それを推薦ブラックボードに書き込む
2. 各エージェントは推薦ブラックボードを参照し、利用者の要求を獲得する
3. 各エージェントは関連付けられた情報源から情報収集を行う
4. いずれかのエージェントが戦略に基づいてアイテムの推薦を行う
5. 必要があれば推薦アイテムに対して利用者が評価を行う
6. 各エージェントは推薦ブラックボードを参照し、推薦アイテムの内容とそれに対する利用者の評価を獲得する
7. 4 へもどる

なお、このシステムが終了するのはいずれかのエージェントがアイテムを推薦することができなくなったときである。

### 3. 合理的提案アルゴリズム

#### 3.1 合理的提案アルゴリズム

競争型情報推薦システムでは、情報推薦を行うエージェントが複数存在し、それらが自律的に提案を行うことから非合理的な提案が行われる可能性がある。そこで、それぞれのエージェントが提案内容の合理性を判断し、合理的な提案を行うための合理的提案アルゴリズムについて説明する。合理的な提案とは、エージェントが提案を行う際に、利用者にとって明らかに不要なアイテムの提案を行わないということである。

合理的提案アルゴリズムでは、アイテムを合理性に基づく複数の領域へ分類し、アイテムが属する領域を参照した上で、提案を行う。ただし、提案を行う度に領域は変化するので、適切に更新して行く必要がある。

ここでは、2属性アイテム問題を扱っている。また問題を簡単化するために、以下の2点を仮定する。

- 属性値はすべて正の値をとり、属性値が低いほどそのアイテムの効用が高い
- 利用者は提案に対して、Good と NoGood の評価のみを行う。

次に2属性アイテム問題における合理的提案手法について述べる。

##### (1) 3領域の定義

アイテムを分類する3領域を以下のように定義する。

領域  $R$  : 受理可能なアイテムが属する領域

領域  $I$  : 受理不可能なアイテムが属する領域

領域  $U$  : 受理可能性が不明なアイテムが属する領域

ここでの受理可能なアイテムとは、推薦アイテムがすでに推薦されたアイテムと比較したとき、すべての属性において優れたアイテムのことであり、また受理不可能とは、すべての属性において劣ったアイテムのことであり、例えば、レストラン情報においてすでに予算 = 2000 円、距離 = 150 m のアイテムが推薦されているとすると、予算 = 1500 円、距離 = 100 m のアイテムは受理可能なアイテムであり、領域  $R$  に属する。予算 = 3000 円、距離 = 200 m のアイテムは受理不可能なアイテムであり領域  $I$  に属する。また予算 = 1500 円、距離 = 200 m アイテムは受理可能性が不明なアイテムであり領域  $U$  に属する。

領域  $I$  のアイテムは利用者にとって明らかに不要であるため、提案は  $R$  と  $U$  の2領域から行われる。また  $R$  からの提案は利用者にとって明らかに有用であるため、利用者へ評価を問う必要がない。

##### (2) 合理的提案アルゴリズム

以下に、合理的提案アルゴリズムを示す。この合理的提案アルゴリズムは、2.7 節の 4. から 7. に相当する。

アイテム  $P$  が2属性を有し、それぞれの属性値が  $x, y$  であるとき、アイテム  $P$  を  $(x, y)$  で表す。また利用者が Good の評価を下したものを受理アイテムと呼び、そのうち領域の境界となるアイテムを境界アイテムと呼ぶ。

なお、すでに  $n-1$  個のアイテムが提案されている時の領域をそれぞれ  $R_{n-1}, I_{n-1}, U_{n-1}$  とする。 $P_{A_1}, P_{A_2}, \dots, P_{A_m}$

は境界アイテム、 $m$  は境界アイテムの個数、また新たに提案するアイテムを  $P_n = (x_n, y_n)$  とする。

1. 領域  $R$  又は  $U$  から戦略に基づいてアイテムを選択し、提案を行う。
2. 領域  $U$  からの提案については利用者に評価を問う。
3. 領域状態を更新する。

##### (a) 領域 $I$ の更新

$$I_n = I_{n-1} \cup \{(x, y) | (x \geq x_n) \wedge (y \geq y_n)\} - \{(x_n, y_n)\}$$

##### (b) 領域 $R$ の更新

$$i. P_n \in R_{n-1}$$

$$R_n = \{(x, y) | (x \leq x_{A_1}) \wedge (y \leq y_{A_1})\} \cup \dots \cup \{(x, y) | (x \leq x_{A_m}) \wedge (y \leq y_{A_m})\}$$

$$ii. P_n \in U_{n-1}$$

##### A. 利用者の評価が「Good」

$$R_n = R_{n-1} \cup \{(x, y) | (x \leq x_n) \wedge (y \leq y_n)\}$$

##### B. 利用者の評価が「NoGood」

$$R_n = R_{n-1}$$

##### (c) 領域 $U$ の更新

$$U_n = \overline{R_n \cup I_n}$$

4. 1. に戻る。

##### (3) 領域の推移

アイテムの提案が行われた時の領域の推移についてレストラン推薦を例にして考える。レストラン情報は  $x$  を予算 (円)、 $y$  を距離 (m) とし  $(x, y)$  で表すものとする。

##### 領域 $R$ からの提案

すでに、 $P_0 = (2000, 150)$  のレストランが提案されており  $P_0$  が境界アイテムであるとき、 $P_1 = (1200, 60)$  のレストランを提案するときの領域の推移を考える。 $P_1$  は  $P_0$  と比べて、2属性の両方もが優れているので、領域  $R$  に属している。よって利用者へ評価を問う必要はない。 $P_1$  を提案すると領域は図2のようになる。 $P_1$  の提案により、 $(x \geq 1200)$  かつ  $(y \geq 60)$  の領域が領域  $I$  となり、 $(x \leq 1200)$  かつ  $(y \leq 60)$  の領域が領域  $R$  となり、それ以外は領域  $U$  となる。

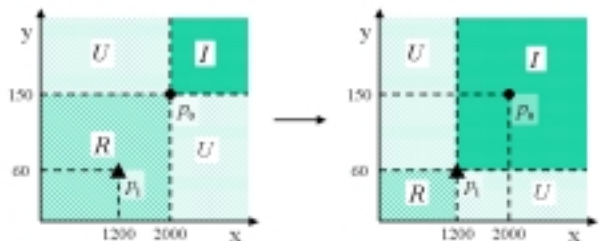


図 2: 領域  $R$  からの提案による領域の推移

##### 領域 $U$ からの提案

すでに、 $P_2 = (1200, 150)$  のレストランが提案されており  $P_2$  が境界アイテムであるとき、 $P_3 = (2400, 90)$  のレス

トランを提案するときの領域の推移を考える． $P_3$  は  $P_2$  と比べて予算は劣っているが距離は優れている．よって  $P_3$  は領域  $U$  に属している．利用者の評価が Good の場合の領域は図 3 のようになる． $P_3$  の提案により， $(x \geq 2400)$  かつ  $(y \geq 90)$  の領域が領域  $I$  に， $(x \leq 2400)$  かつ  $(y \leq 90)$  の領域が領域  $R$  に加えられる．

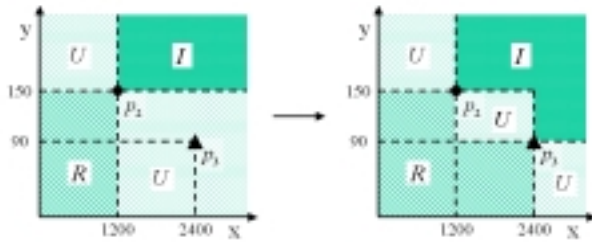


図 3: 領域  $U$  からの提案 (利用者の評価が Good の場合)

利用者の評価が NoGood であった場合の領域は図 4 のようになる． $P_3$  が利用者に受理されなかったためこの場合の領域  $R$  の変化はない．また，領域  $I$  の推移は，利用者の評価が Good であった場合と同じである．

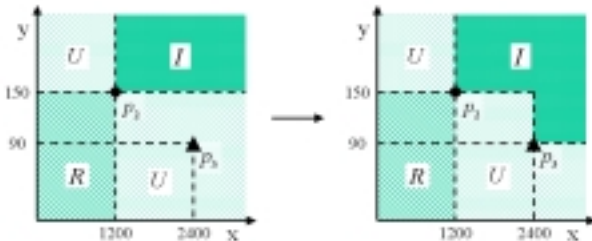


図 4: 領域  $U$  からの提案 (利用者の評価が NoGood の場合)

## 4. レストラン推薦システム

競争型情報推薦システムの例題として，レストラン推薦システムを示す．このシステムでは，利用者が「何処で何を食いたい」という要求を送ることにより，2体のエージェントが競争しながら次々と情報推薦を行うものである．

### 4.1 設定

レストラン推薦システムにおいて，エージェントは2体とし，それぞれの情報源を Yahoo!グルメ<sup>\*3</sup>と関西ウォーカー<sup>\*4</sup>の2つのサイトとする．エージェントが推薦するレストラン情報の属性を，「平均予算」と「最寄駅からの距離」の2属性とし，それぞれ数値の低いものほど効用が大きいものとする．以下，2つの属性の属性名を「予算」と「距離」と呼ぶ．また2体のエージェントはいずれも，受理不可能でないアイテムの中で，予算の最も高いレストランを推薦する，という戦略をとる．

\*3 <http://gourmet.yahoo.co.jp/>

\*4 <http://www.walkerplus.com/kansai/gourmet/>

## 4.2 動作例

次にレストラン推薦システムの動作例を示す．また，動作画面の一例を図 5 に示す．

1. 利用者は「心齋橋でカレーを食べたい」という要求を送る．
2. 関西ウォーカー・エージェントは「予算 3000 円，距離 91 m」のレストランを推薦する．
3. 利用者は Good を返答する．
4. Yahoo! グルメ・エージェントは「予算 1000 円，距離 336 m」のレストランを推薦する．このレストランは1のレストランと比較して，予算は優れているが距離は劣っている．よって利用者への問い合わせを行う．
5. 利用者は NoGood を返答する．
6. 利用者の返答が NoGood であったので，Yahoo! グルメ・エージェントは先ほどのレストランよりも距離の短い「予算 1000 円，距離 209 m」のレストランを推薦する．利用者への問い合わせを行う．
7. 利用者は Good を返答する．
8. 関西ウォーカー・エージェントは「予算 500 円，距離 952 m」のレストランを推薦する．このレストランは1のレストラン，5のレストランと比較して，予算は優れているが距離は劣っている．よって利用者への問い合わせを行う．
9. 利用者は Good を返答する．
10. Yahoo! グルメ・エージェントは「予算 800 円，距離 170 m」のレストランを推薦する．このレストランは5のレストランと比較して，予算・距離の両方が優れている．よって利用者は Good を返答すると予想できるので，利用者への問い合わせは行わない．
11. 関西ウォーカー・エージェントの推薦するレストランがなくなった．これによりシステムは終了する．



図 5: レストラン推薦システムの動作画面の一例

## 5. 評価シミュレーション

### 5.1 実験の目的

競争型情報推薦システムでは，エージェントの数によってその挙動が変化すると予想される．まずエージェントが1体の場合であるが，それは独立型と同等であることは自明である．このシステムでは，エージェントの数が多くなるに従っ

て競争が激しくなることから、利用者の効用が大きくなり、またエージェントにとっては競争相手が増えることによって効用が小さくなるのが予想される。つまり、エージェントの数が多くなるにつれブローカ型に近づくことが考えられる。また全体のアイテム数が多ければ多いほど、利用者にとってより効用の大きいアイテムが存在する可能性が大きくなり、利用者にとって有利になることが考えられる。本章では、エージェントの数、アイテムの総数を変化させたときに、上記のことが示されるかを確認する。

## 5.2 設定

各アイテムは買い手用の属性と売り手用の属性を持つことはすでに述べたが、ここでは各アイテムは2属性アイテムであるとし、それぞれの属性値は0から1まで0.1きざみの値をとるものとし、またそれぞれの属性は互いに独立であるとする。利用者は、各属性に同等の好みを持つものとし、各属性の属性値がその属性の効用値であるとする、その効用は式(4)で表される。

$$U(C_i) = \sum_{j=1}^n \frac{1}{n} x_{i,j} \quad (4)$$

エージェントの効用は式(5)で表される。

$$U_s(C_i) = x_{i,n+1} \quad (5)$$

またエージェントは、各自の利益が最大となるアイテムの中からランダムに提示アイテムを選ぶものとする。

実験は次のように行った。

1. まずアイテム数とエージェント数を決定し、各エージェントに平等にアイテムを分配する。例えば、アイテム数 = 1000 で、エージェント数 = 2 なら、各エージェントは 500 ずつアイテムを持ち、同アイテム数で、エージェント数 = 5 なら、各エージェントは 200 ずつアイテムを持つことになる。
2. 利用者によって最終的に選ばれた推薦アイテムに対する利用者とのエージェントのそれぞれの効用を調べる。
3. 各場合において、これを 10000 回繰り返し、その平均値を求める。

なおブローカ型は、利用者の効用のみを考慮して推薦を行うことから、全アイテムの中から利用者の効用が最も大きいものを推薦アイテムとした。

## 5.3 結果

競争型情報推薦において、エージェントの数とアイテムの数を変化させたときの、利用者の効用を図6に、エージェントの効用を図7に示す。

図6では、エージェントの数が増えるに従い、利用者の効用が増加していき、ブローカ型に近づいている。図7では、エージェントの数が増えるに従い、エージェントの効用が低下していき、ブローカ型に近づいている。

## 5.4 考察

エージェント数とアイテム数による競争型情報推薦システムの挙動について、エージェントが1体のみであるなら、独立型情報推薦システムと等価であることはすでに述べた。実験の結果より、エージェントが複数である場合、エージェントの数を増加させるにつれて、利用者の効用は増加し、エージェントの効用が低下していることから、エージェントの数が多ければ多いほどブローカ型情報推薦システムに近いも

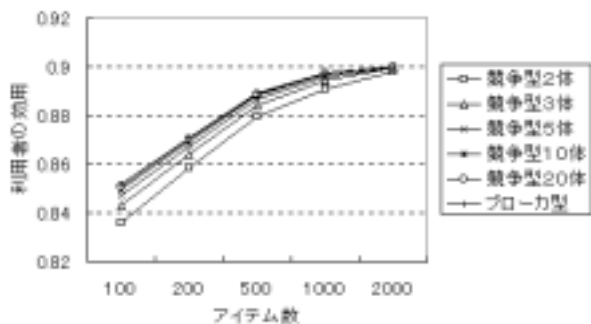


図6: 利用者の効用

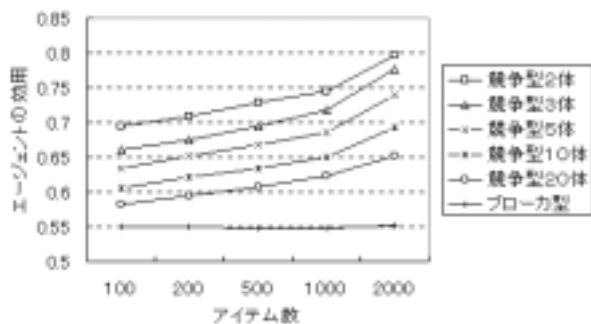


図7: エージェントの効用

のとなることが示された。また、アイテム数が増えるにつれて利用者とエージェントの両方の効用が増加しているが、これは全体のアイテム数が増えることによって、利用者とのエージェント双方にとってより効用の高いアイテムが存在する可能性が高くなるためである。

同様の実験を3属性アイテムについても行った。その結果は上記の2属性の場合に類似していた。競争型情報推薦システムにおいて、属性数が変化してもその動作は変わらない。

## 6. 関連研究

いくつかの研究は、推薦システムにおける対話インターフェース問題に照準を当てている。

ATA [5] は利用者に最適な、もしくはそれに近い旅行プランニングをアシストする。そのシステムでは、CCA(candidate/critique agent) が利用者とのインタラクションから、利用者のモデルを学習していく。CCAは利用者のモデルを多属性における荷重をかけた制約関数の和として表現しており、利用者とのインタラクションから荷重を調整し、制約を付加もしくは更新することで利用者の真のモデルに近づけている。またここではインタラクションの改善のため、単なる解の順序付けだけでなく、利用者のモデルの適当な初期値の設定、支配された解や類似した解の提示は行わない、1つの属性を最適化する解を提示する、といったアイデアが付加されている。

Exper1Clerk [6] は、人間の店員をまねて商品データ

ベースで Web 上の買物客を案内する。そのシステムでは、キャラクタエージェントが買物客と自然言語で対話し、NBA(Navigation by asking) と NBP(Navigation by proposing) という、2つの案内モードを用いてインタラクションの効率化を図っている。NBA は質問に概念的な階層構造をもたせて、それぞれの質問の持つ情報量から質問によって得られる情報利得を最大化する。NBP は対照的な3つのサンプル商品を提示し、それに対する買物客の評価を次の提案に反映する。

ATA, ExpertClerk は、ともにキャラクタエージェントを用い利用者とのインタラクションを繰り返すことで、利用者が満足するアイテムを提示することを目的とした推薦システムである、という点で競争型情報推薦システムと類似している。また、いずれもインタラクションにより利用者の真の好みをどれだけ早く、どれだけ正確に推論できるか、ということに重点を置いている。ただエージェントは単体であり、複数のエージェントによる競争的な情報推薦を目的とはしていない。

## 7. おわりに

複数の情報提供者がそれぞれのエージェントを介して利用者と相互作用を行うことにより、競争的に情報を推薦する競争型情報推薦システムを提案し、レストラン推薦に応用した。競争型情報推薦システムは、利用者は複数の情報を比較でき、また情報提供者は独自のエージェントを用いることにより自分の推薦基準を反映することができるため、利用者と情報提供者の双方にメリットのあるシステムである。またシミュレーションにより、競争型情報推薦システムの挙動についても示し、エージェントの数を増やすことで、その挙動がブローカ型情報推薦システムに類似したものになることも示した。

また競争型情報推薦システムでは、複数のエージェントが自律的に情報推薦を行うため、非合理的な提案が行われる可能性が生じた。そこでエージェントがアイテムの合理性を判断して合理的な提案を行うための合理的提案アルゴリズムを提案した。

今後の課題として、以下の3点について考える必要がある。情報推薦における推薦回数の考慮

推薦システムにおいて、推薦回数が多すぎることは利用者にとってはひどくわずらわしい。これはエージェントのとの戦略によって減らせることが可能である。例えば、自分の利益を省みずに利用者の効用が最も高いと予想されるアイテムを推薦するエージェントがいたとすると、利用者とのインタラクションは非常に少なくすむと考えられる。競争型情報推薦システムでは、各エージェントが自分の利益を最大限に得ることを考えると、インタラクションがひどく長引くことが予想されるので、これを避けるために推薦アイテムを決定する際の取り決めを考案する必要がある。

マルチエージェントと多属性アイテムによる実装

2体のエージェントによる2属性アイテムでの競争型情報推薦システムの実装例として、レストラン推薦システムを示した。これをエージェント数を3体以上、またアイテムの属性数を3属性以上にする。属性数が増えることにより、必然的に領域  $U$  からの提案が多くなり、それに伴って利用者の問い合わせ回数が増加し、また利用者の好みもより複雑になることが予想されるので、それへの対応が問題となる。

エージェントの分散化

レストラン推薦システムでは、1台のサーバで2体のエージェントを動かしているが、これを複数のサーバで個別にエージェントを動作させ、それらを結びつけての情報推薦の実現を目指す。これによりインターネット上で様々なエージェントによる競争型情報推薦が可能になる。

## 参考文献

- [1] 北村泰彦, 野崎哲也, 辰巳昭治, "スクリプトに基づく WWW 情報統合支援システムとゲノムデータベースへの応用", 電子情報通信学会論文誌, J81-D-I(5):451-459, 1998.
- [2] 北村泰彦, "インターネットエージェント", bit, 31(2):7-14, 1999.
- [3] Sonia Bergamasch, Domenico Beneventano, "Integration of Information from Multiple Sources of Textual Data", Matthias Llusch(Ed.), Intelligent Information Agents, pp.53-77, Springer Verlag, 1999.
- [4] Ralph L. Keeney, Howard Raiffa, "Decisions with Multiple Objectives: Preference and Value Trade-offs.", Cambridge Univ. Press, 1993.
- [5] Greg Linden, Steve Hanks, Neal Lesh, "Interactive Assessment of User Preference Models: The Automated Travel Assistant", In Anthony Jameson, Cecile Paris, and Carlo Tasso (Eds.) User Modeling: Proceedings of the Sixth International Conference, UM97 (pp. 67-78). Vienna, New York: Springer Wien New York, 1997.
- [6] Hideo Shimazu "ExpertClerk: Navigating Shoppers' Buying Process with the Combination of Asking and Proposing, Seventeenth International Joint Conference on Artificial Intelligence, IJCAI-01 (pp. 1443-1448), Seattle, Washington, USA, August 4-10, 2001.