

## UNIX シェルユーザのための 適応型ユーザインタフェースの構築とその評価

5 J-5

高田 光男 小高 知宏 小倉 久和

福井大学

### 1 はじめに

UNIX のシェルの一つである tcsh では、コマンドライン上において Emacs あるいは vi エディタと同様な編集ができ、その編集コマンドは約 40 種類にのぼる。しかし、実際に tcsh ユーザを対象に編集コマンドの使用状況を調査したところ、初心者とはとより、ある程度使い慣れたユーザであっても実際に使用しているコマンドは半数にも満たないという結果を得た。[1] このような状況を改善するためには、ユーザ自身が能動的に機能を学習していくことが期待されるが、あるシステムを使いつつ高度な機能を学習していくのは容易なことではない。そこで我々は、ある事柄を学んでいく場合、近くによく知っている人がいて適切なアドバイスをするのが最も効率の良い方法であるという認識から、ユーザとシステム間に立ち有用な機能をユーザへ積極的に提示していく知的インタフェースを構築した。本研究では、tcsh のコマンドライン編集機能に関して、ユーザが必要な機能を必要な場面においてアシストを行うオンライン・ヘルプをより効果的に行うことが目的である。

### 2 適応型ユーザインタフェース

ユーザにとって有益なアシストを行うには、ユーザが学習しやすいタイミングで、しかもその内容はユーザの持つ能力や経験に応じたものでなくてはならない。そうでなければ、場合によっては初心者のユーザに対して理解できないような高度な内容のアシストを行ったり、逆に上級のユーザに対して簡単すぎる無意味なアシストを行うといったことになりかねない。[2] つまり、システムは個々のユーザに合わせて対応を変化させる必要がある。我々はこの点を考慮し、各ユーザの能力・経験に応じたアシストを行うために、ユーザ毎の理解状態および使用履歴を表現したユーザモデル (User Model) を持たせた図 1 のようなシステムを構成した。

本システムでは、「挙動解析 (behavior analysis)」においてユーザのシェルに対する一連のキー入力から編集内容を把握し、その編集方法の評価を行う。その結果、ユーザが効率の悪い編集を行った場合、新しくウィンドウを

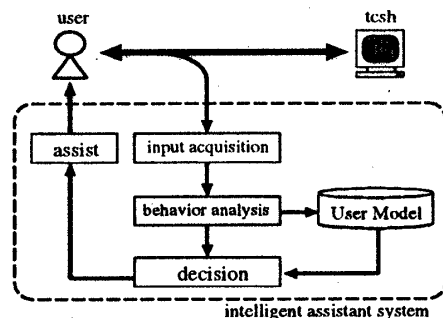


図 1: システムの構成

生成し、より有効なコマンドの提示とその機能の説明を行う。このとき、本システムでは「ユーザモデル (User Model)」を参照することでユーザの能力・経験を考慮した内容のアシストを行うことが可能である。したがって、場合によっては意図的にアシストを抑制することもある。

### 3 ユーザモデルの生成方法

本システムでは、知的 CAI の学習者モデルの一つであるオーバレイモデル [3] を参考に図 2 のようなユーザモデルの構築を行った。オーバレイモデルとは、学習者の知識を教師の部分集合として捉えるモデルであり、その構築方法は学習者が習得した項目にフラグを立てるといった単純なものである。本ユーザモデルにおいては tcsh の編集コマンドの一覧を用意し、システムの判断によりユーザが理解していると推論したコマンドにフラグを立てる。このとき、各コマンド毎に、ユーザが使用するたびに加算し、使用すべきところで使用しなかった場合や長期間

コマンド	得点	最終使用日	フラグの状態
backward word	15	12/25	ON
backward delete word	13	12/23	ON
forward word	2	12/1	OFF
delete word	8	12/13	ON
beginning of line	1	11/28	OFF
history backward	20	12/10	ON
end of line	3	12/25	OFF
	11	12/3	ON

図 2: ユーザモデル

使用していない場合には減点するといった条件で付加した得点を指標としている。つまり、我々の構築したユーザモデルは、オーバレイモデルの一種であるが、フラグの立て方に特徴があり、場合によってはフラグの取り消しがあるなどユーザの使用状況に応じて変化する動的なものである。

## 4 システムの評価

### 4.1 ユーザモデルの評価

ユーザモデルからの情報をアシストの過程において積極的に利用していくためには、そのユーザモデルが信頼性の高いものでなくてはならない。つまり、ユーザモデルがどれだけ正確にユーザの理解状況を把握しているかの検証が不可欠である。この点を踏まえ、我々は実際にユーザが持つ知識とユーザモデルで表現されている理解状態にどれくらいギャップがあるかについて調査した。具体的には、11名のユーザに対し約2ヵ月間に生成されたユーザモデル中でユーザが知っているコマンドとしてフラグが立てられたものと、各ユーザに自分の知っているコマンドを申告してもらったものを比較した。表1がその結果である。これは、例えば被験者Aについては、ユーザの知らないコマンドを知っているものと誤認した割合が0.100であり、逆に知っているコマンドを知らないと誤認した割合が0.025であったことを表している。この結果、本システムでは誤認率0.1以下の信頼性の高いユーザモデルを生成することが可能であると言える。

表1: ユーザモデルの検証結果

被験者	ユーザモデルの誤認率	
	高く評価	低く評価
A	0.100	0.025
B	0.025	0.025
C	0.025	0.000
D	0.000	0.000
E	0.075	0.000
F	0.050	0.025
G	0.025	0.050
H	0.025	0.050
I	0.000	0.050
J	0.000	0.050
K	0.000	0.025
平均	0.030	0.027

### 4.2 アシストの実施結果

図3は、本システムによるアシストを約1ヵ月間行った結果の一例である。このグラフは、ユーザの使用したコマンドの累積度数の変化を表しており、実線は1回以上、破線は5回以上の使用が認められたコマンドの個数である。ただし、実際にアシストを行ったのはグラフ中の縦線以降である。この結果よりアシストの効果が認め

られ、また、被験者からも、いままで知らなかった機能を学ぶことができた肯定的な評価を得ている。

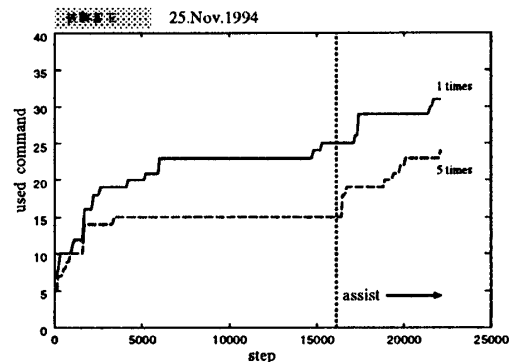


図3: アシストの結果

さらに我々は、ユーザモデルを持たないシステムによるアシストも行った。その結果、ユーザの熟練度の変化に関してはユーザモデルを持つ場合とほぼ同等の効果が得られた。しかし、ユーザがReturnキーを打つまでの入力を1ステップとした場合、ユーザモデルを有するシステムでは平均37.23ステップに1回の割合でアシストが発生するのに対し、ユーザモデル無しのシステムでは平均5.26ステップ毎にアシストが行われている。さらに、アシストの内容に注意を払うことはほとんどないという否定的な意見を出す被験者もいた。すなわち、ユーザモデルの有無の影響はアシストの頻度に現れており、ユーザモデルを持たないシステムによるアシストの大半は、ユーザにとって無意味なものであったと考えられる。

## 5 おわりに

今回、我々は、UNIXのシェルの一つであるtcshを対象に、そのコマンドライン編集機能をユーザにアシストするシステムを構築・評価することで、ユーザモデルを持つ適応型のユーザインタフェースの必要性について考察した。ただし、今回の評価実験は11名のユーザを対象にした試験的なものであったので、今後、多数のユーザを対象にシステムの評価を行う計画である。

## 参考文献

- [1] 高田 光男, 小高 知宏, 小倉 久和: "UNIX シェルユーザの挙動解析とユーザモデル構築の手法", 平成6年度 電気関係学会北陸支部連合大会講演論文集, pp.299(1994)
- [2] Barbara Wasson, Sigmund Akselsen: "An overview of online assistance: from on-line documentation to intelligent help and training", The Knowledge Engineering Review, Vol.7, pp.289-322(1992)
- [3] 溝口 理一郎, 角所 収: "知的CAIにおける学習者モデル", 情報処理学会誌, Vol.29, No.11, pp.1275-1282(1988)