

# アプリケーション使用時におけるメンタルモデルの修正の検討

## The modification of mental model for users who use application

姚昕<sup>†</sup>, 松林 翔太<sup>†</sup>, 三輪 和久<sup>†</sup>

Xin Yao, Shota Matsubayashi, Kazuhisa Miwa

<sup>†</sup>名古屋大学

Nagoya University

youshin@cog.human.nagoya-u.ac.jp

### 概要

本研究は、ユーザがスマートフォンでアプリケーションを使用する場面を考え、その時行われるメンタルモデルの修正を検討する。方法として、同じ機能を持つが、機能構造が異なる家計簿アプリケーションを2つ設計した。実験では、参加者に異なるアプリを提示して使用させることにより、メンタルモデルの修正を行わせる。その後、カードソーティングを行うことで、メンタルモデルを測定し、その修正の程度を定量的に分析する。また、アプリ操作中のヒューマンエラーを分析し、ヒューマンエラーとメンタルモデルの修正がどう関連するかを検討する。

キーワード:メンタルモデル, ヒューマンエラー, カードソーティング

## 1. 導入

### 1.1 メンタルモデルの修正

技術の進歩に伴い、私たちが日常的に使用するコンピュータやスマートフォンなどの情報機器は高機能化している。それに伴い、その利用方法も益々複雑になる傾向がある。先行研究により、これらの製品を操作する際に、メンタルモデルが重要な役割を担っていることが明らかになっている[1]。メンタルモデルとは、システムが何を含んでいるか、システムがどのように動くか、システムがなぜそう動くかに関して、ユーザの理解を反映した緻密な構造である[2]。ユーザに適切なメンタルモデルの構築を促すことにより、システムの理解が促進され、より正確な操作が期待できる[3]。

従来の研究では、メンタルモデルの「構築」に注目することが多かった。例えば、メンタルモデルの構築に影響する要因、メンタルモデルの構築における動的なプロセス、メンタルモデルの構築度合いがユーザパフォーマンスに与える影響などは研究のテーマになっている[3]。その一方、メンタルモデルの「修正」について検討した研究は少ない。すなわち、不適切なメンタルモデルを持っているユーザがどのようにメンタルモデルを修正するかについての検討はほとんど行われていない。現実には達成すべき目標は同じであっても、アプリケーションによって操作が異なるため、メンタ

ルモデルの修正が必要になることがある。例えば、コンビニによってコピー機の種類が異なる、AndroidアプリとiOSアプリで操作が異なる、文書作成ソフトのMicrosoft WordとmacOS Pagesで操作方法が異なるなどは日頃よく経験する。これらの製品を適切に利用する過程で、メンタルモデルの修正が必要だと考えられる。

そこで、本研究は、ユーザがアプリケーションを使用する場面を取り上げ、その時行われるメンタルモデルの修正を検討する。そのひとつの場面として、近年急速に利用が増えているスマートフォンアプリケーションを用いる。

### 1.2 カードソーティング

メンタルモデルの修正を検討するために、ユーザが持つメンタルモデルを測定することが必要である。従来の研究により、スケッチ法、思考発話、概念リスト、カードソーティングなどの方法が開発された。そのうち、カードソーティングとは、参加者に、システムの機能や情報を書いてある一連のカードを与え、自分の理解に基づいて分類させる課題である[4]。その分類結果からユーザのメンタルモデルを洞察することができる。この手法は、研究の分野だけでなく、システムをデザインする観点でもよく利用されている。また、他の方法と比べ、カードソーティングの結果を定量的に分析することができるという特徴を持つ。

そこで、本研究は、カードソーティングを用いて、メンタルモデルを測定する。カードソーティングの結果を分析することにより、実験参加者のメンタルモデルとアプリケーションの機能構造とのギャップを定量的に示すことができる。すなわち、あるアプリケーションに対して、参加者のメンタルモデルがどの程度不正確であるかを定量的に表すことが可能である。

### 1.3 メンタルモデルとヒューマンエラー

ヒューマンエラーについての研究は、様々な分野で

行われている。ヒューマンエラーとは、意図しなかった結果に至った行為のことである[8]。

IT 製品を使用する際に起こったヒューマンエラーは、タスクが正しく行われるのを妨げ、タスクの遅延やタスクの失敗をもたらす[7]。Norman は、IT 製品を使用する際に問題が発生する根本的な原因は、ユーザのメンタルモデルと製品の間のギャップであると主張した[1]。つまり、ユーザに適切なメンタルモデルの構築を促すことにより、より正確な操作が期待できると考えられる[3]。

本研究は、不適切なメンタルモデルがヒューマンエラーを引き起こし、これらのヒューマンエラーがメンタルモデルの修正に関連するかを検討する。具体的には、どのようなヒューマンエラーを引き起こすか、またこれらのヒューマンエラーがメンタルモデルの修正にどう影響するかを検討する。

#### 1.4 目的

本研究は、ユーザがアプリケーションを使用する場面を考え、その時行われるメンタルモデルの修正を検討する。具体的には、不適切なメンタルモデルによって起こったヒューマンエラーの種類と、そのヒューマンエラーがメンタルモデルの修正に与える影響を検討する。

そこで、同じ機能をもつが機能構造が異なる家計簿アプリケーションを二つ設計した（以下 APP1、APP2 と呼ぶ）。実験では、参加者に APP2 を提示してそのメンタルモデルを構築させた後に APP1 を使用させ、APP2 から APP1 へのメンタルモデルの修正を行わせる。その後、カードソーティングを用いて、参加者のメンタルモデルの修正について定量的に分析する。また、アプリの操作からヒューマンエラーに関する分析を行い、ヒューマンエラーとメンタルモデルとの関連を検討する。

実験 1a では、参加者に APP1 を提示してそのメンタルモデルを構築させた後に APP1 を使用させ、適切なメンタルモデルを持っているユーザがアプリケーションを使用する際に、どのようなヒューマンエラーが起こるかを確認する。次に実験 1b では、参加者に APP2 を提示してそのメンタルモデルを構築させた後に APP1 を使用させ、アプリを使用する際のヒューマンエラーと、APP2 から APP1 へのメンタルモデルの修正を関連して分析する。

適切なメンタルモデルを持っていても、必ずしも適

切なパフォーマンスができるとは限らないことが、いくつかの研究で示されている[4]。そのために、アプリケーションを操作する際に、適切なメンタルモデルを持っているユーザでも、ヒューマンエラーを起こす可能性があると考えられる。つまり、これと比較することで、不適切なメンタルモデルを構築した実験 1b の参加者にとって、タスク中起こったヒューマンエラーのうち、どのようなエラーが適切でないメンタルモデルのため起こったのかを確認することができる。

## 2. 実験 1a

実験 1a の目的は、適切なメンタルモデルを持っているユーザがアプリケーションを使用する際に、ヒューマンエラーが起こるか、またどのようなヒューマンエラーが起こるかを確認することである。

### 2.1 実験装置と刺激

実験 1a で使われる家計簿アプリケーションは APP1 で、支出・収入の入力、入力した項目の検索および分析などの機能を持っている。APP1 (図 1) は、スマートフォン (HUAWEI P20 lite 5.84 インチ) で提示される。



図 1 APP1 インタフェース例

### 2.2 手続き

実験 1a は 4 つのフェーズから構成される。

フェーズ 1 :

先行研究により、メンタルモデルの構築に影響を及ぼす要因はいくつかあることが明らかになっている[3]。これらの要因は、メンタルモデルの修正過程にも

影響する可能性があると考えられる。そこでフェーズ1で、ユーザのメンタルモデルに影響を及ぼす個人特性を把握するために、アンケート調査を行う。具体的には、参加者の年齢・学歴・専攻・スマートフォンまた家計簿アプリケーションの使用経験を測定する。スマートフォンおよび家計簿アプリケーションの利用頻度は5段階評定を行う。

フェーズ2:

参加者が最初に構築するメンタルモデルをコントロールするために、フェーズ2でAPP1に関するトレーニングを実施する。具体的には、参加者に、APP1のインタフェースを提示して、その構造を学習させる。その後、カードソーティング課題により、APP1のメンタルモデルが構築されたか操作チェックを行う。つまり、期待されたメンタルモデルが正確に構築されたことを確認した上で、次のフェーズに入る。

フェーズ3:

参加者にアプリケーションを操作させ、ヒューマンエラーが起こるか、また、どのようなヒューマンエラーが起こるかを確認する。参加者は、トレーニングしたAPP1を操作してタスクを行う。タスクの時間制限は15分である。タスク中の操作は記録される。

フェーズ4:

参加者は、APP1について、再度カードソーティング課題を行う。

### 3. 実験 1b

実験 1b の目的は、ヒューマンエラーとメンタルモデルの修正との関連を検討することである。

#### 3.1 実験装置と課題

実験 1b では、APP1 と、APP1 と同様の機能を持つが機能構造が異なる APP2 の、2つの家計簿アプリケーションを用いる。つまり、APP1 と APP2 で、同じ目標を達成するための操作方法が異なる。例えば、「食事」という支出項目を検索するという目標を達成するために、APP1 (図 1) では、「履歴」メニューから「絞り込み」画面にアクセスしてから項目を検索する。一方、APP2 (図 2) では、「その他」メニューから「カテゴリ」画面にアクセスして、「食事」カテゴリを選択した後に検索する。

#### 3.2 手続き

実験 1b は、以下の4つのフェーズから構成される。

フェーズ1:

実験 1a と同様、ユーザのメンタルモデルに影響を及ぼす個人特性を把握するために、アンケート調査を行う。

フェーズ2:

参加者が最初に構築するメンタルモデルをコントロールするために、フェーズ2でAPP2に関するトレーニングを実施する。具体的には、参加者に、APP2のイ

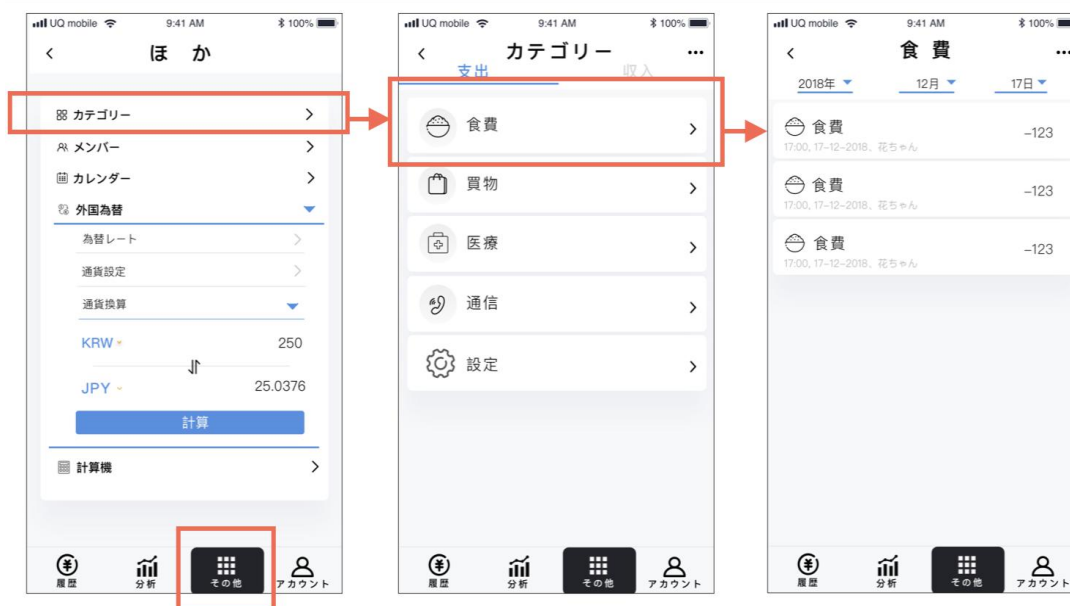


図2 APP2 インタフェース例

ンタフェースを提示して、その構造を学習させる。その後、カードソーティング課題により、参加者が構築したメンタルモデルを測定する。

フェーズ3:

参加者は、トレーニングした APP2 とは機能構造が異なるアプリケーション (APP1) を操作してタスクを行う。タスクの内容と操作したアプリケーションは、実験 1a と同様である。

ここで、参加者は APP1 の操作を、APP2 のメンタルモデル、つまり不適切なメンタルモデルに基づいて行うと考えられる。その場合、参加者は、タスク遂行を通して、APP2 から APP1 へメンタルモデルに修正することが期待される。

フェーズ4:

参加者が修正したメンタルモデルを測定するために、APP1 の機能構造についてカードソーティング課題を行う。

## 4. 分析方針

### 4.1 ヒューマンエラーの分析

実験 1a は、適切なメンタルモデルを持っている参加者がアプリケーションを使用する際に起こったヒューマンエラーの種類と数を確認する。

実験 1b は、不適切なメンタルモデルを構築した参加者が、アプリケーションを使用する際に起こったヒューマンエラーの種類と数を分析する。

実験 1a と実験 1b におけるフェーズ3のタスク遂行中のアプリにおける操作データをそれぞれ分析した後、比較する。そこで、どのようなエラーが適切でないメンタルモデルのために起こったのかを確認する。

#### 4.1.1 ヒューマンエラー分類例

先行研究により、参加者の操作リストを、規範となる操作リストと比較して、その違いを Error of omission, Error of commission, Extraneous acts に分類する[7]。Error of omission は、するべきことをしない行為を、Error of commission は、するべきことを誤ってした行為を、Extraneous acts は、必要のないことをした行為を指す。以上の基準に従ってヒューマンエラーを分類した後、各基準のエラーの数を求める。

以下に、アプリケーションの操作時に起こりうるエラーを英文字で表す。規範となる操作リストは abcdef である場合を想定する。このときの参加者の操作リストとエラーの分類の例を表1に示す。

表1 ヒューマンエラー分類例

ヒューマンエラー	操作リスト	説明
Error of omission	規範: abcdef 参加者: ab_def	操作 c をしなかった
Error of commission	規範: abcdef 参加者: ab <sup>c</sup> def	操作 c をしたが、誤ってした
Extraneous acts	規範: abcdef 参加者: abgdef	操作 c の代わりに他の操作をした

### 4.2 メンタルモデルの修正程度

実験 1b では、不適切なメンタルモデルがどのような種類のヒューマンエラーを引き起こすかを検証した後、違う種類のエラーがメンタルモデルのどのような修正に影響するかを検討する。そのために、まず実験 1b におけるフェーズ2とフェーズ4のカードソーティング課題の結果から、メンタルモデルの修正程度を確認する。

本研究において、メンタルモデルの修正度は、実験 1b におけるフェーズ2の不一致度とフェーズ4の不一致度の差分を指す。フェーズ2で得たカードソーティングの結果を、APP1 の機能構造における規範と比較して、不一致度 (Mismatch Score) を計算する[4]。不一致度は、参加者のメンタルモデルとアプリケーションの機能構造との差分を定量的な形で求めたものを指す。その後、フェーズ4におけるカードソーティングの結果に対しても、同様に不一致度を計算し、フェーズ2との差分を測る。

実験 1b では、メンタルモデルの修正度を分析した後、ヒューマンエラーとメンタルモデルの修正程度との関連を検証する。

## 5. 予測

先行研究により、不適切なメンタルモデルはヒューマンエラーを引き起こすため、実験 1a と実験 1b におけるヒューマンエラーの数は有意な差があると予測する。また、3つのヒューマンエラーのうち、Extraneous acts は、実験 1a より実験 1b の方が有意に多いと予測する。これは、操作対象についてのメンタルモデルが適切でない場合に、タスクの遂行計画をうまく立てられないため、アプリを操作する際に必要のない機能も

探索するからだと考えられる。

また、3つのヒューマンエラーのうち、Extraneous actsとError of omissionは、ユーザに明確な外在的フィードバックがあるため、その検出がされやすく、メンタルモデルの修正程度に正の影響を与えると予測する。例えば、Error of omissionが起こる際に、すべきことをしなかったため、タスクを進められない場合、負のフィードバックが生じる。さらに、Error of commissionは、ユーザに対して明確な負のフィードバックが生じにくいいため、メンタルモデルの修正との関連がないと予測する。

## 6. 今後の検討

メンタルモデルの修正程度については、メンタルモデルの修正に関する新たな知見を発見するために、定性的な分析も行う予定である。具体的に、実験1bにおけるフェーズ4のカードソーティングの結果から、複数の参加者に共通したメンタルモデルのパターンを抽出する。ここでは、類似度 (Similarity Score) を計算することにより[4]、実験参加者全員のカードソーティングの結果を比較して、いくつかの共通した修正パターンを抽出することができると期待される。

## 文献

- [1] Norman,D.A., (1983) “Some observations on Mental Models”,University of California, San Diego Press.
- [2] Carroll,J.M. and Olson,J.R., (1987) “Mental models in human-computer interaction: Research issues about what the user of software knows. Committee on Human Factors”, Commission on Behavioral and Social Sciences and Education, National Research Council. Washington, DC: National Academy Press.
- [3] Yan Zhang, (2013) “The development of users' mental models of MedlinePlus in information searching”, Library & Information Science Research,Vol. 35, No. 2, pp. 159-170.
- [4] M.Schmettow and J.Sommer, (2016) “Linking card sorting to browsing performance—are congruent municipal websites more efficient to use?”, Behaviour & Information Technology, Vol. 35, No. 6, pp. 452-470.
- [5] Capra,M., (2005) “Factor analysis of card sort data:an alternative to hierarchical cluster analysis”, Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 49, No. 5, pp. 691-695.
- [6] S.Pocock, M.Harrison, P.Wright, and P.Johnson, (2001) “Thea – A technique for human error assessment early in design”, Proceeding of 8th IFIP TC . 13 Conference on Human-Computer Interaction, pp. 247-254.
- [7] Robert W. Reeder and Roy A.Maxion, (2005) “User Interface Dependability through Goal-Error Prevention”, Proceedings of the 2005 International Conference on Dependable Systems and Networks, pp. 60-69.
- [8] Bin Zhao and Olivera,F., (2006) “ERROR REPORTING IN

ORGANIZATIONS”, The Academy of Management Review  
Vol. 31, No. 4, pp. 1012-1030.