

The State of the Art in Automating Usability Evaluation of User Interfaces

MELODY Y.IVORY AND MARTI A.HEARST

May 31, 2004

3 OVERVIEW OF USABILITY EVALUATION METHODS

私達はWIMP インタフェースに利用されている UE 法 75 と Web インタフェースに利用されている UE 法 57 についてサーベイした。これら 132 の方法のうち 29 の手法だけが Web, WIMP UIs の両方ともに利用できる。私達はインタフェースのタイプに基づいて、文字通り評価に利用されている、それぞれの手法の適用可能性を判断した。つまり、そのそれぞれの手法というものが、別のタイプのインタフェースの評価にも適用可能であるかを判断した。表 1 は二つのタイプのインタフェースに対するサーベイの結果を組み合わせたものだ。メソッドクラスを第一行目に太字のエントリとして記してある。メソッドタイプはそれぞれのメソッドタイプに対し第一行目に (太字ではない) エントリとして記してある。第二行から第五行までの項目はそれぞれ、具体的な UE 法について自動化のサポートの度合いと自動化を実施するにあたって必要な労力と共に記述している。

表の説明文の訳。括弧の中の数は特定のメソッドタイプとオートメーションタイプに分類された UE 法の数を表す。それぞれの方式におけるエフォートレベルは次のように表記されている。ミニマルエフォートは空白, formal は F, informal は I, model は M で記した。*印は formal または informal なインタフェース使用が必要であることを示している。† 記号はその方式がモデルを使用してもしなくてもどちらでも良いことを示している。

幾つかの UE 法については複数の観点から議論する。従って、エフォートレベルの横にサーベイした UE 法の数括弧で括って示してある。幾つかのアプローチは複数のメソッドタイプに対して自動化をサポートしている (付録 A を参照のこと)。

付録 A Table XVIII の説明の訳。4つのソフトウェアツールは複数のメソッドタイプにおいて自動化をサポートしている。DRUM は performance measurement と log file analysis を。AMME は log file analysis と Petri net modeling を。KALDI は performance measurement と log file analysis と remote testing を。UsAGE は performance measurement と log file analysis を。

表 1 は 110 の方式を含んでいるが、これは、幾つかの方式が複数のメソッドタイプに対して適用可能だからだ。また、私たちは WIMP と Web UI の両方において適用可能な方式しか書いていない。Table 2 は全てのメソッドタイプについての説明が記されている。5つのメソッドクラスは、自動化のサポートの点で

大きな違いがある。全体的に、WIMP と Web インタフェースにおいて自動化のパターンは似通っている。ただ、analytical modeling と simulation においては異なっており、WIMP インタフェースにおいては 16 の方式が存在するのに対し、Web 領域では 2 方式しかない。

表 1 は総じて、自動化は大いに探索中であることを示している。自動化をサポートしない方式は 67% であり、自動化をサポートするものはトータルで 33% である。この 33% のうち、自動取得する方式は 13%、自動分析を行う方式は 18%、批評まで行う方式は 2% であった。自動取得をする方式のうち、二つの方式以外は、あるレベルのインタフェース使用を必要とする。即ち genetic algorithms と information scent modeling は共に、その後の分析のための使用データを生成するため、シミュレーションを使用する。総じて、たったの 29% の方式 (自動化されていない方式も含む) のみが、評価にあたって formal または informal なインタフェース使用を免れている。

完全な自動化を達成するには、その評価ソフトは formal または informal な使用を課さずに、インタフェースを批評するものであるべきだろう。私たちがサーベイした限りでは、たった一つのメソッドタイプのみがこの水準に達していた。それらは、guideline review における Farenc and Palanque [1999], Lowgren and Nordvist [1992], そして Laskowski [1998] である。Guideline review 法は、自動的にユーザビリティ違反を発見しレポートする。そして、それらをどう直せば良いかという提案をする (第 5 章で詳しく議論している)。

次に、自動解析の段階にある方式というのは、表 1 に示されているように、analytical modeling と simulation が大勢を占めている。これらの方式の殆どは、informal または formal なインタフェース使用を必要としない。

続く章では、様々な UE 法とその自動化について詳細に議論する。幾つかの方式は WIMP と Web interface 両方に適用可能である。しかしながらその適用可能性については、必要な時しか言及はしない。そして議論には、自動化の方式について次に示す基準に則った私たち自身による評価も含まれている。

- 有効性：いかに良くその方式がユーザビリティ問題を発見できるか
- 使いやすさ：実施するにあたってどれくらい簡単か
- 学習しやすさ：どれだけ学習するのが簡単か
- 適用性：その方式が、元々評価対象としているインタフェース以外に、どれほど広く適用できるか

私たちはそれぞれのメソッドクラスにおける議論の中で、これらの基準に焦点をあて議論する。Ivory [2001] では、サーベイしたインタフェース全てについて詳細な議論がなされている。

10 Conclusions

この論文で私達は UE 自動化の概観の提供と、様々な手法を比較するための分類法を提案した。私達はまた、WIMP インタフェースと Web インタフェース評価における自動化の利用に関する広範囲に渡るサーベイを提供した。その結果、サーベイした手法のうち自動化が使用されていたのは、全体のわずか 33% であった。

こともあろうに、サーベイした手法のうち、たった 29%だけが、formal または informal なインタフェース使用の必要から免れていた。guideline review を除外した、formal または informal なインタフェース使用を必要としない全ての手法は analytical modeling または simulation に基づいていた。

UE 自動化は、重要な定性的・主観的な情報 (例えば、ユーザの好みや思い違い) を収集することはできないということを頭に入れておくのは重要なことであり、それらは usability testing や heuristic evaluation や他の標準的な inquiry methods を通じてのみ明らかにされうるものである。それでもやはり、simulation や analytical modeling というのは、多大な開発コストをコミットすることなくデザイナーがデザイン代替案を選択する助けになるはずである。

更に、評価者は自動化を heuristic evaluation や usability testing といった大抵は自動化されていない手法と併用することができる。例えば、heuristic evaluation を行っている評価者は、UI 内において自動的に生成された使用記録を取得することができる。

UE へオートメーションを付加することは多くの潜在的な利点がある。その利点としては、自動化されていない手法のコストの削減、デザイン代替案の選択の援助、UE 評価の一貫性の改善が挙げられる。analytical modeling や guideline review, log file analysis といった手法の更なる進展のための研究は新たな効率的な自動化された手法に帰着するかもしれない。

Table 1: 表 2

Method Class Method Type	概要
Testing	
Thinking-Aloud Protocol	テストの間ユーザが発話する
Question-Asking Protocol	テスターがユーザに質問をする
Shadowing Method	専門家がユーザの動作をテスターに説明する
Coaching Method	ユーザは専門家に質問することができる
Teaching Method	熟達ユーザが初心者ユーザを教える
Codiscovery Learning	二人のユーザが共同作業を行う
Performance Measurement	テストの間、テスターが使用記録をとる
Log File Analysis	テスターが使用記録を解析する
Retrospective Testing	テスターがユーザと一緒にビデオテープをレビューする
Remote Testing	テストの間、ユーザとテスターが同じ場所にいない
Inspection	
Guideline Review	専門家がガイドラインとの一致をチェックする
Cognitive Walkthrough	専門家がユーザの問題解決をシミュレートする
Pluralistic Walkthrough	多数人で認知的 Walkthrough を行う
Heuristic Evaluation	専門家がヒューリスティクス違反の同定を行う
Perspective-Based Inspection	専門家が焦点を絞ったヒューリスティック評価を行う
Feature Inspection	専門家がプロダクトの機能を評価する
Formal Usability Inspection	専門家が形式的なヒューリスティック評価を行う
Consistency Inspection	専門家がプロダクトを通じた一貫性チェックする
Standards Inspection	専門家が基準を満たしているかをチェックする
Inquiry	
Contextual Inquiry	インタビュアーがユーザの環境についてユーザに質問する
Field Observation	インタビュアーがユーザの環境においてシステムの使用を観察する
Focus Groups	多数のユーザがディスカッションに参加する
Interviews	一人のユーザがディスカッションに参加する
Surveys	インタビュアーがユーザ個別の質問について尋ねる
Questionnaires	ユーザが個別の質問に対して回答を行う
Self-Reporting Logs	ユーザが UI 操作の記録をとる
Screen Snapshots	ユーザが UI のスクリーンショットをとる
User Feedback	ユーザがコメントをつける
Analytical Modeling	
GOMS Analysis	実行と学習に要する時間を予測する
UIDE Analysis	UIDE の中で GOMS analysis を行う
Cognitive Task Analysis	ユーザビリティ問題を予測する
Task-Environment Analysis	UI タスクにおけるユーザのゴールの対応を評価する
Knowledge Analysis	学習可能性を予測する
Design Analysis	デザインの複雑さを評価する
Programmable User Models	ユーザのように動作するプログラムを書く
Simulation	
Information Proc. Modeling	ユーザインタラクションを真似る
Petri Net Modeling	使用記録からユーザインタラクションを真似る
Genetic Algorithm Modeling	初心ユーザのインタラクションを真似る
Information Scent Modeling	Web サイト ⁴ ナビゲーションを真似る