

多桁減算における手続きバグを解消する CAI の開発

Development of CAI that relieves procedural bugs of multi-digit subtraction

社会情報システム学講座 0312000094 柴田大紀

指導教員: 藤原康宏 市川尚 鈴木克明

1. 研究の背景

計算などの一定の手続きに従って答えを求める課題において、誤った手続きを常に適用することにより起こる誤りとして手続きバグがある。この誤りの場合、学習者がどのようなバグを持っているのかを診断し、それを取り除くアプローチが必要となる。

森田¹⁾は、コンピュータに誤りを犯す学び手の役割を担わせ、学習者にはそれを正す教え手の役割を与え、コンピュータが演示する誤りを指摘・修正するという活動を導くための学習環境としてチューコ・システムを開発した。この中で、「学習者の犯す誤りは、それが自己のものであれ他者のものであれ、内省的学習を導く重要な契機となり得る」としている。

本研究では、森田によるチューコ・システムの学習環境を発展させて、多桁減算の計算において誤りを犯す学習者を対象として、学習者のバグを解消する為の CAI の開発を行った。

2. 先行研究

手続きバグに関する代表的なシステムに BUGGY²⁾がある。BUGGY は算術技能に関する学習者の犯すバグを同定する機構を実装している。チューコ・システムはこの BUGGY を基に、引算ができないチューコというねずみに引算を教えてあげるという文脈で、チューコに問題を与え、チューコが間違った計算をした場合に学習者が指摘するというシステムである。チューコ・システムはバグの同定は行わず、BUGGY での誤りを演示する機能を取り入れたシステムといえる。

本システムは学習者の持つバグを解消することを目的とし、チューコ・システムの文脈による学習とバグの演示機能に加え、BUGGY で行っているバグを同定する機能を組み込む。つまりチューコ・システムとの違いは以下の 3 点になる。第1に学習者の誤りを BROWN and VANLEHN³⁾により報告された多桁減算に関するバグのいずれかに同定することが出来る、第2に学習者が持つバグを同定した上でそのバグが必ず生起する問題を自動生成する、第3に学習者が誤った判断・指摘をした際にフィードバックが与えられる。

3. システムの設計

(1) 学習フェーズと問題の分類

本システムを用いた学習は、事前テスト、練習、事後テストの 3 つのフェーズからなる。学習者がどのようなバグを持っているかを同定する為に事前テストを行い、ここで観察されたバグを基に、練習フェーズでコンピュータに学習者の犯した誤りを演示させる。

各フェーズの問題は、繰り下がりが無い「ナシ問題」、一つ上の桁からの繰り下がりがある「一段問題」、二つ上の桁からの繰り下がりがある「二段問題」の 3 つに分類した。これを大分類として、さらにバグが生起する条件ごとにより詳細な問題形式に分類した。

(2) バグの判定と演示

事前テストフェーズの問題は「ナシ問題」に属する問題形式から出題し、計算に誤りがある場合は間違えた問題と同等の問題を出題し同じバグが観察された時にバグを持っていると判定する。

練習フェーズでは、事前テストフェーズにおいて学習者に観察されたバグをモデル化しコンピュータに演示させる。学習者が練習を進める中で演示中のバグが解消されたと判断した際、他のバグが観測されているならばそのバグを演示し、さらに練習を行う。

(3) 練習フェーズの流れ

練習フェーズの流れを図 1 に示す。ステップ 1 では、事前テストフェーズで観測されたバグが必ず生起するような問題をシステム側で生成する。またステップ 2 で、コンピュータの計算にバグがあるにも関わらず学習者が正しいと判断した場合、コンピュータの計算に対して指摘を行うがその指摘が間違っている場合の 2 つの状況においてフィードバックが与えられる。前者の場合、コンピュータが演示したバグと同じバグを学習者が持っていると考えられる為、そのバグの誤っている点を示す。後者の場合は、事前テストフェーズでのバグの選定方法を用いて再度合致するバグを探し、ここで合致したバグに合わせたフィードバックを与える。これ以外のステップは森田¹⁾と同様である。

(4) 文脈の導入

本システムでは、「引算が苦手なウサギのラビと一緒に店を出し、ラビの計算が間違っていたらそれを教えてあげる」という文脈で学習環境を提供する。この

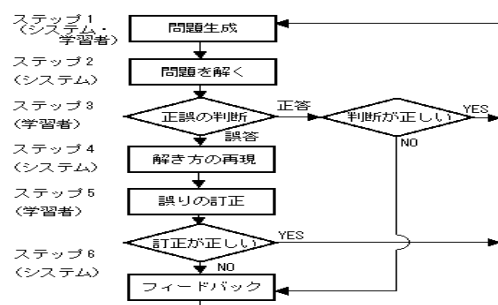


図 1.練習フェーズの流れ

文脈を導入した理由は、システム側で生成した問題を客との関わりにより自然に出題でき、学習者にも納得しやすいものであろうと考えたためである。

本文脈には、ラビ・ラビの父・客の3人が登場する。まず事前テストフェーズでラビの父が登場し、「ラビが店を出すことになったがおつりの計算が出来るか不安なのでラビが間違ったら教えてあげてほしい」という活動の文脈を伝え、「どれくらい引算が出来るか知りたいから次に出す問題を計算してほしい」という流れで事前テストを行い、その後練習フェーズに移る。

図 2 はステップ2における練習フェーズの画面例である。客が来て商品を選び代金を渡してくる。そこから商品の値段を引いておつりを渡すという活動の中で3桁引く2桁の問題を生成し、出題している。この際、ラビの計算過程と計算結果が表示される。繰り下がり際には右下の計算エリアで、桁から桁へと借り入れる数字がアニメーションで提示されるようになっており、10ではなく9を借り入れるバグなどの時でもバグに合わせたアクションが起きる。この後、ラビの計算の正誤を問うボタンが現れ、学習者が間違っていると判断した際は正しい答えを入力する為のボタンが画面上に現れる。この判断が合っているならば次の練習へ移り、間違っているならば3章(3)で述べたと通りのフィードバックが提示される。



図 2.画面例

4. システム開発と形成的評価

本システムをFlashとPerlを用いて実装した。バグ計算に必要な計算ステップをプロダクションルールとしてサブルーチン化し、各バグをこのプロダクションルールで表現しデータベース化した。これにより、バグによる計算結果が求められることができ、学習者の回答と比較することで学習者のバグが判定できる。

本システムを小学校教諭に試用してもらい、提示される文章やボタンが小学校低学年の児童が理解できる物となっているか判断してもらった。評価の結果、事前テストの出題数の多さ、10 円の物に対して 300 円を支払ってくるような普段ではありえない事象の不自然さがあげられた。この結果を元に、事前テストの出題数を少なくし、文脈の導入でおつりの計算の不自然さを納得させるといった改善を行った。また、繰り下がりのアニメーションに注目させるような工夫をすると効果的だろうという意見からこの点も改善した。

これらの改善を行い学童クラブの小学二年生2名と小学三年生1名に協力してもらい、文章やボタン操作が理解できるかを評価した。ボタン操作についてはパソコンの経験がない児童でも簡単に行えていた。文章や語句については小学二年生の児童でも十分理解できていた。しかし、事前テストについては出題数が多いようで、誤りが続き途中で止めてしまう学習者がいた。そこで、出題する問題形式を5種類に絞り、それぞれを1問ずつ出題し、バグを持っていると判定された時点で練習へと移るように改善した。

5.まとめ

本研究では、多桁減算の計算において誤りを犯す学習者を対象として、学習者のバグを解消する為のCAIの開発とインタフェースの評価を行った。評価により挙げられた改善を行い、対象となる小学二年生でも理解できる文章と、容易に操作できるインタフェースが実現できた。今後は、実際にバグを持つ学習者による評価を行い、学習効果を測る必要がある。

参考文献

- 1) 森田英嗣:「誤りから学ぶ」環境の開発研究,日本教育工学雑誌, 18(1), pp.1-13 (1994)
- 2) BROWN, J.S. and BURTON, R.R.:
Diagnostic Models for Procedural Bugs in Basic Mathematical Skills, *Cognitive Science*, (2), pp.155-192 (1978)
- 3) BROWN, J.S. and VANLEHN, K.: Repair Theory: A generative Theory of Bugs in Procedural Skills, *Cognitive Science*, (4), pp.379-426 (1980)