A Informática “Não-educativa” Suportando o Desenvolvimento de Habilidades Cognitivas

Adam Victor Nazareth Brandizzi – 07/43291
Universidade de Brasília
Instituto de Psicologia

Departamento de Psicologia Escolar e Desenvolvimento
Desenvolvimento Psicológico e Ensino
Professora Pollianna Galvão
28 de junho de 2007

Resumo

Com a popularização dos computadores, o uso didático dessas ferramentas têm atraído interesse. Entretanto, a maior parte do interesse recai sobre programas didáticos específicos, como jogos educativos ou “máquinas de ensinar” behavioristas. Esse ensaio chama a atenção para o valor didático das ferramentas quotidianas, como suítes de escritório e navegadores Web.

A Informática “Não-educativa” Suportando o Desenvolvimento de Habilidades Cognitivas

Motivação

Segundo a Wikipedia (2007a), *computador* é qualquer máquina que manipule um determinado conjunto de dados de acordo com um conjunto de instruções.

Obviamente, um computador não processa dados despropositados, mas sim dados cujo processamento interesse a alguém, de uma maneira ou de outra. Quando alguém utiliza um processador de texto, como *Microsoft Word*, o texto que edita é o dado que se interessa processar. Quando se cria uma planilha eletrônica, o objetivo é que algum conjunto de dados com significado seja processado, como uma folha de pagamento de uma empresa ou uma planilha de notas de uma turma de alunos. Todo dado processado por um computador, em algum momento, é um dado de interesse para seres humanos. Quando um conjunto de dados é de interesse imediato de um usuário (e. g. um texto, uma planilha ou uma imagem), dizemos que o *software* que o gerencia é um *aplicativo* (Wikipedia, 2007b).

Este ensaio pretende explorar a idéia de que o próprio uso – ao menos um modo de uso específico, mais adequado – do aplicativo pode incrementar as habilidades do usuário na atividade fim do *software*. O uso adequado de um processador de textos, então, exercita as próprias habilidades de redação do editor; o uso adequado de uma planilha de cálculos incrementaria a habilidade do usuário em compreender e

desenvolver modelos financeiros.

Explorações Prévias

A idéia de que o próprio uso de uma ferramenta computacional possa exercitar as habilidades cognitivas de um indivíduo em uma determinada habilidade é, de certa forma, uma extensão de versões fracas da Hipótese de Sapir-Whorf. A Hipótese de Sapir-Whorf postula que existe uma relação íntima entre as categorias gramaticais que uma pessoa usa em sua linguagem e sua maneira de compreender e agir no mundo (Wikipedia, 2007c).

Tradicionalmente, a Hipótese de Sapir-Whorf disserta sobre a influência das linguagens humanas. Como postulado, é uma idéia que faz compreender melhor frases como a clássica afirmação de Heidegger de que “só é possível filosofar em alemão” (Almeida, 2005). Estudos em fenômentos semelhantes são muito comuns também em computação. Iverson (1979) apresenta uma interessante leitura sobre o tema da influência da linguagem de programação no desenvolvimento das habilidades do programador, tema recentemente revisitado por Poe (2006), por exemplo. Entretanto, este não é um tema muito explorado em um nível de abstração mais alto, que é o dos aplicativos.

O Exemplo do Processador de Textos: Ensino Inadequado da Ferramenta

Segundo a proposta deste artigo, a maneira como usualmente vêm se ensinando informática ao usuário final tem sido menos eficaz que o

desejável por ignorar o caráter expressivo das ferramentas aplicativas. Os típicos usuários são mal treinados em aplicativos porque são treinados de maneira a apenas conhecer *os ins and outs* do *software*, não a maneira mais eficiente e poderosa de utilizá-los.

O melhor exemplo deste mau ensino são os processadores de texto WYSIWYG[1], como *Microsoft Word* ou *OpenOffice.org Writer*. Ao contrário de processadores de textos mais antigos, como LaTeX, esses processadores de textos enfatizam não a estrutura e os componentes do texto, como títulos, capítulos, referências etc., mas sim sua formatação. A típica forma de ensino do uso destas ferramentas, então, também foca na formatação do texto: os alunos são ensinados a usar recursos de formatação de textos, como fontes em negrito ou itálico, e não recursos de estruturação semântica, como estilos e modelos.

As conseqüências disto são que usuários não adquirem habilidades em redigir textos bem planejados e estruturados nem habilidades em formatação, mas apenas o conhecimento do uso de funcionalidades pontuais de formatação de uma ferramenta. Os documentos tenderão a não possuir uma consistência de aparência e, se o redator não possuir grandes habilidades de escrita, o computador não poderá auxiliá-lo na estruturação do texto. No final das contas, conseguiu-se apenas o mau ensino de um *processador de texto*, enquanto seria possível obter um

1 *WYSIWYG* é o acrônimo de "*What you see is what you get*". É um adjetivo utilizado para se referir a processadores de textos que apresentam uma constante representação final do texto ao editor.

ótimo ensino de *processamento de textos*.

O Exemplo do Processador de Textos: Ensino Ineficiente da Técnica

Também podemos analisar o ensino de processamento de texto por outra perspectiva.

Considere um professor que esteja ensinando seus alunos os conceitos e técnicas básicas de redação. Ele menciona que a redação tem um título, tem introdução, desenvolvimento e conclusão etc. Dependendo do aluno (especialmente se ele estiver aquém do estágio operatório-formal, ou tiver, por outras razões, dificuldades com conceitos abstratos), os conceitos inerentes à estrutura de uma redação parecerão excessivamente abstratos, obscuros ou de difícil apreensão. Entretanto, como o método tradicional de redação geralmente baseia-se na escrita cursiva *apenas*, é necessário que o aluno apreenda esses conceitos de uma forma ou de outra.

Uma potencial ajuda para a solução desse problema seria um curso de redação que utilizasse tanto redação cursiva quanto redação digitada, através de processadores de texto com suporte a estruturação semântica. Como o *software* tornaria mais explícitos, mais concretos os componentes de uma redação, o aluno poderia ter mais facilidade na apreensão dos conceitos.

A Ferramenta Ensinada com a Técnica

Os problemas no ensino de informática são resultado tanto da

novidade do tema quanto de limitações técnicas e econômicas. Justamente por, durante algum tempo, o computador ter sido visto como um equipamento excepcional que precisasse de todo um ritual para ser utilizado[2], é que se passou a crer que ele teria uma função separada das atividades da vida quotidiana, profissional ou escolar. Entretanto, o ensino das ferramentas de informática à parte do seu uso quotidiano não é só menos eficiente que o ensinar da informática atrelado às atividades mais "mundanas": é um ensino simplesmente sem sentido. Só há sentido em ensinar uma criança ou um adulto a utilizar um processador de texto se for para ele redigir documentos. Só há sentido em ensinar o uso de uma ferramenta para se alcançar um objetivo. E, se se vai ensinar o próprio objetivo, por que não através da ferramenta também?

O Conhecimento Registrado no *Software*

Para finalizar, há um fato curioso e relevante sobre aplicativos que esse ensaio gostaria de fazer notar.

Todo *software* é desenvolvido por alguém ou (e este é ó *modus operandi* mais freqüente atualmente) por uma equipe. O *ideal* é que seja desenvolvido por uma equipe com vários profissionais, incluindo especialistas na área fim do programa. Em caso de programas aplicativos, isto é praticamente inevitável: estamos falando de *softwares* extremamente complexos em sua estrutura (por mais que não pareça)

2 Esta visão deturpada do computador é discutida, de maneira um tanto quanto simbólica e humurística, por Salomon (2005)

que precisam ser plenamente intuitivos para usuários leigos. Estes programas são usualmente desenvolvidos por empresas que não têm porque economizar em equipes, e seu desenvolvimento quase sempre é acompanhado por especialistas da área fim.

Oras, se o desenvolvimento do *software* envolve a atuação de especialistas na área fim, é evidente que o *software* em si possui conhecimento sobre a área fim registrado nele. Por que não tomar este conhecimento e aplicar no ensino? Por que não utilizar o próprio *software* como um livro-texto especial?

Conclusão

Tem havido um maior interesse, atualmente, no uso didático da informática. Vários artigos sobre *softwares* educativos, jogos didáticos etc. vêm sendo redigidos, mas as ferramentas quotidianas estão sendo completamente ignoradas nesta perspectiva. É importante notar o poder didático destas ferramentas também, tanto pelo quanto representam no dia-a-dia quanto pelo investimento de cultura nelas feito.

Referências

Wikipedia, the Free Encyclopedia (27/06/2007). Computer. Acessado em 27 de junho de 2007 em <http://en.wikipedia.org/wiki/Computer>

Wikipedia, the Free Encyclopedia (12/06/2007). Application Software. Acessado em 27 de junho de 2007 em <http://en.wikipedia.org/wiki/Application_software>

Wikipedia, the Free Encyclopedia (27/06/2007). Sapir-Whorf Hypotesis. Acessado em 28 de junho de 2007 em < http://en.wikipedia.org/wiki/Sapir_Whorf>

Almeida, Alexander de. (2005). Por que só é Possível Filosofar em Grego e Alemão? *Leia Livro, Não Ficção*. Acessado em 27 de junho de 2007 em <http://www.leialivro.sp.gov.br/texto.php?uid=6312>

Iverson, Kenneth E. (1979). Notation as a Tool of Though. *1979 Turing Awards Lecture, Association of Computer Machinery*

Poe , Kurtis (2006). Sapir-Whorf is not a Klingon. *ONLamp.com*. Acessado em 27/06/2007 em <http://www.oreillynet.com/onlamp/blog/2006/05/sapirwhorf_is_not_a_klingon.html>

Salomon, Gavriel (2005). O Laboratório do Computador: uma Má Idéia, Atualmente Santificada. *Divertire*. Acessado em 27/06/2007 em http://www.divertire.com.br/educacional/artigos/4.htm