

Caderno Escolar Electrónico Adaptativo

Resultados Preliminares

Luís Alexandre

LabSI²/ESTIG, Instituto Politécnico de Beja, Portugal
luis.alexandre@ipbeja.pt

Salvador Abreu

Universidade de Évora e CENTRIA FCT/UNL, Portugal
spa@di.uevora.pt

Resumo

Neste artigo descrevemos a construção de uma interface adaptativa para o Caderno Escolar Electrónico, capaz de prever qual a tarefa mais provável que o utilizador irá efectuar no sistema, com base num conhecimento prévio das acções já efectuadas, assim como no tempo e percurso efectuado até determinada altura. Esta adaptabilidade visa a melhoria da interacção entre o sistema e o seu utilizador, reduzindo assim o número de interacções explícitas necessárias para efectuar uma determinada tarefa. Para conseguir prever as tarefas mais prováveis em determinada situação, o sistema recorre a algoritmos de classificação, baseados em árvores de decisão e redes de Markov. Os utilizadores com necessidades especiais, são o principal alvo deste projecto, mas o sistema poderá também dar um grande contributo a outros utilizadores em idade escolar, dado que esta ferramenta favorece a adopção de um processo metódico para recolha e organização de notas em contexto de sala de aula. Por ter sido desenvolvido com recurso a técnicas de desenho para sistemas interactivos, encontra-se totalmente adaptado aos utilizadores alvos.

Palavras-chave

Sistemas Interactivos, Pessoas com Necessidades Especiais, Tecnologias de Apoio, Interacção Pessoa-computador, Inteligência Artificial, Predição de Tarefas.

1. INTRODUÇÃO

O Caderno Escolar electrónico [Alexandre2007] (Figura 1) tem vindo a assumir-se como uma ferramenta de grande utilidade para crianças com dificuldades motoras, dado que para além de permitir a recolha e organização de notas em contexto de aula, armazena e organiza de uma forma lógica todos os documentos referentes a uma determinada disciplina e determinada aula.

O Caderno Escolar electrónico (CE-e) em conjunto com outras tecnologias de apoio, como o Eugénio [Garcia2003] apoiam estas crianças, permitindo que estas possam realizar as tarefas que envolvem escrita, num espaço de tempo muito semelhante ao dispendido pelas crianças sem necessidades especiais. Assim, esta ferramenta ao favorecer a adopção de um processo metódico de recolha de notas em sala de aula poderá introduzir benefícios assinaláveis no sucesso escolar [Boyle2001].

O principio orientador para o sistema segue uma filosofia DWIM (Do What I Mean) [Teitelman1984], ou seja, o sistema deve comportar-se exactamente de acordo com o objectivo do utilizador, em vez de ter um comportamento coerente com indicações por si dadas, resultantes de um

comando mal executado. Sendo este um objectivo inatingível, é no entanto norteador da nossa intenção.

Mas o CE-e poderá prestar um maior apoio a estes alunos, se reconhecer qual a tarefa em curso, ajudando assim os utilizadores a concluir os passos seguintes da mesma. Em algumas situações é difícil determinar as intenções que se encontram por detrás de uma sequência de acções do utilizador, mas noutros casos, quer recorrendo a conhecimento sobre o domínio [Dix2004] quer observando o comportamento do utilizador em situações anteriores [Maes1994], pode-se conseguir prever qual a operação que pretende realizar. Um caso óbvio será a abertura de uma nova lição no caderno de uma disciplina quando se aproxima a hora dessa aula. Além de prestarem apoio a alunos com dificuldades motoras estes mecanismos também poderão beneficiar alunos com dificuldades cognitivas que com este tipo de ajuda poderão trabalhar de forma mais autónoma em determinadas situações. Este tipo de mecanismos já tem vindo a ser experimentado em algumas ferramentas mais comuns, como por exemplo programas de e-mail [Maes1994] com o intuito libertar o utilizador da execução de tarefas rotineiras através da delegação da responsabilidade pela sua concretização no sistema. Esta abordagem também já foi experimentada no

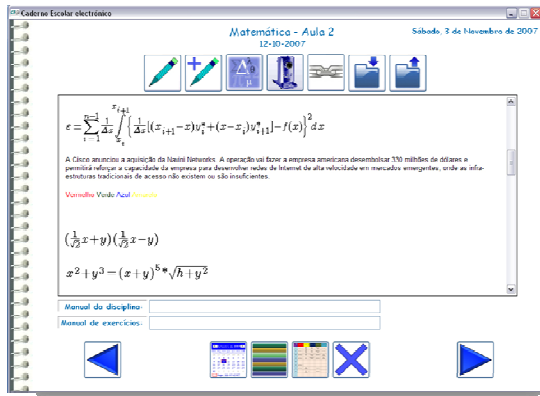


Figura 1 – Interface de *notetaking* do CE-e

desenvolvimento de um sistema de apoio ao desenvolvimento de configurações de teclado para o Eugénio [Garcia2003, Rodrigues2007].

Este projecto tem como objectivo, implementar e avaliar uma interface inteligente para o CE-e que consiga fornecer um maior nível de suporte aos alunos com necessidades especiais na realização de diversas tarefas.

Mas estas ferramentas necessitam de conter modelos da aplicação e do utilizador [Dix2004] para que seja possível identificar as intenções deste último e assim fornecer ajuda na realização de uma tarefa no sistema.

Assim numa primeira fase deste projecto procedemos à criação de uma base de dados que contem o conhecimento sobre as várias tarefas realizadas pelo utilizador no CE-e, as diferentes acções que as constituem e o contexto onde foram realizadas. Para a caracterização do contexto registamos e armazenamos um conjunto de informação temporal como a data, dia da semana e hora, bem como as acções associadas. Estes dados foram recolhidos no decurso das aulas, com alunos a utilizar o caderno electrónico para o registo das várias actividades lectivas. Os alunos em causa possuem paralisia cerebral, estando apenas afectados na parte física. As actividades, nas quais os registos foram efectuados, decorreram em salas de aula, com turmas entre o terceiro e o sétimo ano de escolaridade, em computadores pessoais com o CE-e instalado. A utilização do sistema foi realizada com recurso aos tradicionais dispositivos de *input/output*, sem a intervenção de quaisquer outros dispositivos adaptados de interacção (e.g. ratos adaptados ou interruptores).

Esta base de dados e de conhecimento foi então utilizada na fase seguinte do projecto, onde estão a ser criados e avaliados mecanismos de apoio ao utilizador na realização de diversas tarefas no CE-e, que entraram em linha de conta com todas as dimensões do contexto. Esta é a principal contribuição deste projecto, dado que se o sistema conseguir interpretar correctamente todas as dimensões do contexto, poderá evoluir de um conjunto de regras *hardcoded* que fornecem um conjunto limitado de adaptações, para um sistema que com a sua contínua utilização conseguirá aprender e criar novas adaptações ao contexto, mais adequadas ao perfil do utilizador. Esta “inteligên-

cia” do sistema só será possível com recurso a algoritmos de aprendizagem automática.

Na secção 2 deste artigo apresentaremos uma revisão do estado da arte, sendo a secção 3 dedicada à descrição da arquitectura do CE-e Adaptativo. Finalmente na secção 4 apresentamos a conclusão e propomos direcções para o trabalho a efectuar no futuro.

2. O ESTADO DA ARTE

Hoje em dia vivemos rodeados de sistemas computacionais (e.g. *desktops*, *laptops*, PDAs, entre outros). Vivemos numa época em que todos os aspectos da nossa vida são influenciados ou determinados por decisões ou acções efectuadas por sistemas computacionais.

Este apoio só é possível se o sistema conseguir determinar ou reconhecer o contexto em que tanto a máquina como o seu utilizador estão enquadrados [Blum2005].

Muitos investigadores têm referido a importância e os benefícios que um conhecimento do contexto pode trazer à usabilidade dos sistemas informáticos [Bradley2005], podendo desta forma aumentar e enriquecer a Interação Pessoa-computador.

2.1 Context-awareness

A principal ideia por detrás da *context-awareness*¹, resulta do pressuposto que os computadores podem “sentir” e reagir com base nas alterações do ambiente envolvente. Para que esta reacção seja possível os sistemas computacionais terão de possuir um conjunto de informação proveniente de estímulos e informação anterior que em conjunto com regras pré-estabelecidas, permitirão ao sistema reagir com uma consequência lógica e útil à situação do momento [Schilit1994].

Um sistema *context-aware* pode e deve tentar fazer suposições sobre a envolvente do utilizador. Em [Dey2001] o contexto é definido como “qualquer informação que possa ser utilizada para caracterizar uma qualquer situação de uma entidade”.

Inicialmente o contexto foi apenas entendido como sendo a localização, mas nos últimos anos foi proposto que a localização não fosse entendida como o contexto da situação, mas como um dos componentes do contexto [Dey2001]. Foi ainda proposto que o contexto (adaptação) pudesse ser utilizado para a construção de interfaces adaptativas e inteligentes, nas quais a interacção seja a mais implícita possível.

Muitos dos estudos e tentativas de adaptação ao contexto têm passado pela utilização de dispositivos de localização por satélite. Estes sistemas apenas utilizam um pequeno componente de um relativamente grande universo de possíveis adaptações ao contexto. Um sistema que apenas utiliza a localização para se adaptar ao contexto, não pode ser considerado um sistema *context-aware*, devendo

¹ Conhecimento ou consciência do contexto envolvente.

ser considerado um sistema *location-aware* [Barnard2006].

As aplicações com maior divulgação neste campo são os dispositivos de navegação por GPS utilizados nos veículos automóveis e os dispositivos do tipo guia turístico, normalmente dispositivos do tipo PDA ou *smartphone*. Estes sistemas efectuem uma adaptação dos conteúdos que exibem em função do local onde se encontram.

O *Context-awareness* é maioritariamente utilizado, para além dos sistemas de localização por satélite, em sistemas de computação ubíqua². Neste tipo de sistemas torna-se possível a utilização de sensores e outros dispositivos que captam com precisão o contexto, podendo assim, por exemplo, propor e efectuar alterações de luminosidade ou da abertura das janelas em edifícios, de forma a minimizar os consumos energéticos.

2.2 Paradigmas sobre a Adaptabilidade e Previsibilidade

Os primeiros sistemas computacionais tinham como objectivo a resolução de problemas de índole científica ou comercial. Nos dias de hoje todos nós vivemos rodeados de sistemas informáticos, uns mais visíveis, outros mais ubíquos, sendo a interacção cada vez mais intensa, obrigando por isso a cada vez mais acções por parte do utilizador.

Ainda nos primeiros sistemas computacionais, a interface era algo sem grande importância no seu desenvolvimento, sendo esta uma mera ferramenta para fazer a ligação entre a máquina e o utilizador, mas proporcionalmente ao aumento da complexidade dos sistemas, também as interfaces tiveram de evoluir, tornando-se por vezes altamente complexas ou demasiadamente complicadas, ou recheadas de um sem número de *widgets*, que não favorecem a interacção nem favorecem o entendimento das funcionalidades que o sistema e a respectiva interface têm para oferecer.

2.2.1 Interfaces Adaptativas e Preditivas

”Uma interface adaptativa é um sistema computacional interactivo que melhora a sua capacidade de interacção com o utilizador, com base em experiências parciais de interacção com esse mesmo utilizador” [Langley1997].

Dois dos problemas que mais afectam o desempenho dos utilizadores de sistemas computacionais são a grande quantidade de informação que as interfaces possam apresentar e o facto de se terem de tomar decisões em curtos espaços de tempo [Norcio1989].

Apesar de nos últimos anos terem sido dados passos significativos no que toca à Interacção Pessoa-computador, muitos sistemas computacionais ainda apresentam interfaces de utilizador, complicadas e desajustadas. Esta situação resulta do facto de em muitos destes sistemas a interface não ser tida como parte do sistema, sendo vista

como um mero pacote de *software* do sistema cuja única função é interligar o sistema e o utilizador. Assim, muitas das interfaces não integram no seu desenho princípios de desenho centrado no utilizador, nem estão suficientemente integradas com o sistema, não proporcionando assim uma boa interacção do utilizador com a máquina. Desta desadequação resulta uma não optimização da experiência e da performance do utilizador.

Para ultrapassar este problema foram criadas metodologias de desenho de interfaces, nas quais estas são desenhadas à medida do utilizador ou então o desenho tenta ser o mais adequado possível a todos os eventuais utilizadores do sistema, mesmo que para atingir o grau de adequação óptimo, se tenha de configurar, em utilização alguns aspectos da interface.

Mas o facto de adoptarmos uma metodologia de desenho centrada no utilizador, não nos garante que a interface possa ser igualmente usável por todos os utilizadores, nomeadamente se esses utilizadores forem portadores de algum tipo de deficiência. Uma possível resposta para este problema poderá ser a construção de interfaces adaptativas.

A ideia base em que assentam as interfaces adaptativas é muito simples: *”Em vez de ser o utilizador a adaptar-se ao sistema, adapta-se este ao utilizador.”*

Apesar de assentar num pressuposto conhecido, os problemas inerentes são em grande número e complexidade. Uma interface adaptativa tem de se basear em modelos cognitivos e no ambiente circundante [Norcio1989]. Estes modelos tentam explicar o nível de perícia e experiência do utilizador, através de parâmetros do tipo: (i) comandos efectuados, (ii) taxas de erro, (iii) velocidade de digitação, entre outros.

Outro grande problema reside no diálogo, ou seja, o diálogo entre a interface e o utilizador deve ser o mais específico possível para o utilizador em causa. Finalmente, a interface tem obrigatoriamente de ser pensada como um componente do sistema e não como uma mera peça que interliga a camada aplicacional e o utilizador.

Uma interface adaptativa pode efectuar as adaptações de duas formas:

- A primeira hipótese de adaptação é uma adaptação relativa ou balizada, ou seja, é efectuada uma adaptação limitada deixando nas mãos dos utilizadores a readaptação da interface, no caso da adaptação efectuada pela mesma não ser satisfatória;
- A segunda hipótese é deixar totalmente nas mãos da interface a responsabilidade pela adaptação.

A segunda hipótese é por muitos considerada como a verdadeira adaptabilidade, sendo no entanto muito difícil de atingir. Uma adaptação totalmente dinâmica apenas conseguirá ser atingida à custa de algoritmos de aprendizagem automática (*machine learning*).

O conceito por trás da aprendizagem automática pode ser por si só considerado como adaptativo, porque os algo-

² Modelo de interacção pessoa-computador em que a computação está integrada nos objectos do dia-a-dia.

ritmos deste tipo baseiam-se em informação anterior para tomarem decisões sobre dados actuais.

Contudo uma interface adaptativa também tem os seus problemas e entre outros poderemos referir o facto de o utilizador não conseguir criar um modelo mental do sistema, se a aparência do sistema sofrer muitas e frequentes alterações. Outro potencial problema poderá ser a perda de controlo que o utilizador poderá sentir. Por último e talvez o maior problema das interfaces adaptativas acaba por ser o custo e a complexidade da implementação.

2.2.2 Interfaces Atentas e Sugestivas

Outro paradigma sobre a adaptabilidade e previsibilidade, deriva da atenção e da sugestão. Neste caso as interfaces passam a estar atentas e automaticamente sugerem acções aos utilizadores.

Uma interface atenta estabelece automaticamente prioridades e de acordo com as mesmas apresenta ao utilizador a informação ou as opções mais adequadas ao contexto e ao momento de utilização do sistema, para que os recursos, quer do utilizador quer do sistema sejam optimizados, não sobrecarregando assim nenhum dos intervenientes na interacção [Vertegaal2002].

Tal como nas interfaces adaptativas, as interfaces atentas baseiam as suas decisões num conjunto de informações e modelos sobre decisões anteriormente efectuadas pelo utilizador, estados anteriores e estados futuros.

No entanto as interfaces atentas podem-se subdividir em vários subtipos, de uma forma geral esta divisão é efectuada de acordo com a capacidade que as interfaces têm de monitorizar o estado de interacção e atenção do utilizador para com o sistema e a interface.

Nos sistemas cuja interacção é efectuada através de janelas, a atenção do utilizador é captada e comunicada ao sistema de uma forma **explícita** através da organização das janelas, por parte do utilizador de uma forma manual. Por outro lado sistemas que possuam mecanismos sensíveis, tais como sistemas de *eye tracking* para seguirem as acções do utilizador e perceberem onde a atenção do utilizador está centrada, são classificados como sistemas **implícitos**. Uma interface atenta tenderá a utilizar na sua estrutura uma combinação destes dois tipos de interfaces.

Por outro lado temos as interfaces sugestivas, que apenas fornecem ao utilizador pistas sobre a próxima possível acção do utilizador, através do realce de certos componentes da interface [Igarashi2001].

Um utilizador ao interagir com uma interface sugestiva, apenas tem de escolher uma das opções em destaque ou pode ignorar todas as sugestões caso nenhuma corresponda à opção desejada. Por norma as sugestões são geradas por um sistema normalmente apelidado de motor de sugestões, que constantemente observa o estado do sistema, gerando então um conjunto de sugestões que correspondam aos padrões mais próximos do estado actual. Uma interface sugestiva pode ser vista como sendo um sistema de interacção por gestos, ou seja, ao invés de

atualizar o sistema automaticamente, ao ser descoberto um padrão idêntico, a interface apresenta um conjunto de sugestões e pede ao utilizador que escolha uma destas.

Uma interface sugestiva é uma extensão às interfaces preditivas e está a ser encarada por muitos investigadores como a interface do futuro, para a grande maioria dos sistemas, nomeadamente os sistemas WIMP (*Windows, Icons, Menus and Pointing Devices*).

3. ARQUITECTURA DO CE-E ADAPTATIVO

A interface do protótipo funcional do CE-e foi desenhada de acordo com uma metodologia de desenho para sistemas interactivos com o focus no utilizador, sendo por isso bastante simples e intuitivo [Alexandre2007]. No entanto como o CE-e pretende fornecer o máximo de ajuda possível em tarefas de *notetaking* e de organização da informação, pretende-se que a interface possa aliviar, o máximo possível, o utilizador de tarefas rotineiras e repetitivas, o que poderá ser particularmente útil para utilizadores com graves problemas motores.

Assim, com base na experiência própria, na opinião de técnicos e de outros peritos, optamos por uma interface do tipo sugestivo, que apresentará ao utilizador a hipótese mais provável, de acordo com os padrões que a base de conhecimento apresenta.

3.1 A Base de Conhecimento

Tal como foi referido anteriormente, para sistema poder prever ou sugerir a acção mais provável para um determinado contexto, tem de se socorrer de conhecimento passado e modelos de interacção. Para a construção desta base de conhecimento, recorre-se frequentemente a sistemas que registam as acções do utilizador – *logging*.

Há já algum tempo que investigadores que trabalham em sistemas de Comunicação Aumentativa e Alternativa (CAA), tentam medir o impacto dos avanços tecnológicos no acto de comunicar com recurso a este tipo de sistemas. Assim, de forma a facilitar a análise dos dados recolhidos, com base em sistemas de *logging*, foi definido um formato padrão que permite efectuar a análise aos dados de uma forma sistemática, o *Universal Logging Format* [Lesher2000].

A estrutura do ficheiro de logs proposta em [Lesher2000], apresenta três partes distintas: (i) Cabeçalho que especifica o conteúdo e o formato dos registos; (ii) Corpo que é composto por *n* linhas, cada uma representativa dos dados relativos a um *log*; (iii) Secção de análise, sendo esta opcional, onde se podem colocar algumas estatísticas dos *logs*.

cid	name	type	notnull	diff_value	pk
0	ID	INTEGER	0		1
1	date	NVARCHAR(40)	0		0
2	weekDay	NVARCHAR(20)	0		0
3	time	NVARCHAR(50)	0		0
4	form	NVARCHAR(50)	0		0
5	discipline	NVARCHAR(50)	0		0

Figura 2 – Estrutura dos logs do CE-e


```

1 @relation CE-e_laura_2010-03-24
2
3 @attribute Day (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday)
4 @attribute Hour numeric
5 @attribute Form (InsertEditNote, Form1, FormAgenda, booksForm, BooksForm, ScheduleViewForm, NotesForm)
6 @attribute Discipline ("Apoio ao Estudo ", "Inglês ", "Estudo do Meio ", "Matemática ", "Lingua Portuguesa ", "")
7 @attribute Event ("formatar texto alinhamento centro", "formatar texto cor", "formatar texto highlight", "formatar texto tamanho fonte",
8 "formatar texto sublinhado", "formatar texto bold", "novo evento", "navegar vista configuracoes", "editar nota texto",
9 "cancelar escolha nota texto", "configurar manual disciplina", "apagar nota texto", "editar hiperligacao", "escolha nota texto",
10 "formatar texto italico", "abrir manual disciplina", "download ficheiro", "inserir hiperligacao", "iniciar nova aula",
11 "abrir manual exercicios", "terminar aplicacao", "iniciar aplicacao", "navegar vista livros", "abrir caderno",
12 "escolha impressao objecto", "cancelar escolha impressao objecto", "impressao nota-link-img", "navegar aula anterior",
13 "navegar form agenda", "navegar aula posterior", "inserir nota texto")
14
15 @data
16 "Thursday",100331,"FormAgenda","", "navegar vista livros"
17 "Thursday",100334,"BooksForm","", "abrir caderno"
18 "Thursday",100347,"NotesForm","Estudo do Meio ", "navegar vista livros"
19 "Thursday",100350,"BooksForm","", "abrir caderno"
20 "Thursday",100509,"NotesForm","Lingua Portuguesa ", "iniciar nova aula"
21 "Thursday",100639,"NotesForm","Lingua Portuguesa ", "inserir hiperligacao"
22 "Thursday",100852,"FormAgenda","", "navegar vista livros"
23 "Thursday",100855,"BooksForm","", "abrir caderno"
24 "Thursday",100910,"FormAgenda","", "navegar vista livros"
25 "Thursday",100912,"BooksForm","", "abrir caderno"

```

Figura 5 – Aspecto parcial do ficheiro submetido à ferramenta Weka

base de conhecimento e apresentando de seguida a hipótese que mais se adequa ao contexto em causa e à componente temporal do momento.

Os primeiros testes efectuados, com recurso a uma árvore de decisão C4.5 [Quilan1993] (Figuras 4 e 5), demonstraram que poderiam ser atingidos resultados com uma percentagem mínima de acertos na ordem dos 50%, valor que justifica que se desenvolva um esforço de implementação, dado que o sistema possui, neste momento 40 classes. No entanto destas 40 opções que o sistema possui, dependendo do ecrã onde o utilizador se encontra, apenas são apresentadas entre 3 a 17 opções, o que em termos médios nos dará a cada momento 7 opções possíveis, sendo que 50% de 7 dará aproximadamente 4. Uma percentagem de acertos de 4 em 7 é um resultado bastante encorajador.

Outros algoritmos de classificação foram testados em busca de melhores resultados ou de resultados que confirmassem os obtidos com recurso a árvores de decisão, nomeadamente os Modelos "Escondidos" de Markov (do Inglês *Hidden Markov Models*) [Rabiner1989]. À semelhança das árvores de decisão, os modelos "escondidos" de Markov, são particularmente adequados para conjuntos discretos de dados. Os resultados obtidos com este algoritmo foram igualmente encorajadores, encontrando-se ao nível dos obtidos pela árvore de decisão C4.5. Ao contrário das tradicionais redes de Markov, em que o estado seguinte depende apenas do estado actual do sistema, os modelos "escondidos" de Markov tentam inferir sobre qual a sequência mais lógica de acontecimentos ou estados que produziram um determinado estado ou objectivo, dado que em muitas situações os sistemas não conseguem determinar certos estados - estados "escondidos". Este algoritmo acabou por ser o escolhido, devido à sua mais rápida implementação no sistema, através de classes no código existente do CE-e.

O motor de inferência do sistema lidará com um conjunto pequeno de atributos, como se pode ver na figura 4 o motor terá apenas quatro atributos que necessita de analisar, dado que o quinto atributo é o atributo classe, que servirá para treinar o classificador. No entanto este classificador terá de lidar com um sistema multi-classe, sendo

as classes em grande número. A matriz de confusão construída pela ferramenta Weka (ver figura 3) demonstra de uma forma bastante clara o número e dimensão das classes, sendo este um problema para o classificador, dado que a árvore resultante possuirá um grande número de nós e folhas, aumentando assim a probabilidade de erro na sugestão da hipótese mais provável. No entanto os 52% de acertos conseguidos garantem que pelo menos uma em cada duas sugestões estará correcta reduzindo assim a carga sobre o utilizador, na realização de tarefas que acrescentam valor ao trabalho do utilizador no sistema.

3.3 A codificação

A codificação do motor de inferência desenrola-se em torno de dois algoritmos, uma árvore C5, que deriva da árvore C4.5 de [Quilan1993] e, um algoritmo baseado em modelos "escondidos" de Markov. Apesar de o primeiro ter apresentado melhores resultados nos testes realizados na ferramenta *Weka*, a codificação do sistema avançou com recurso ao segundo algoritmo. Esta decisão foi motivada pela urgência de possuímos um protótipo pronto antes do início do ano escolar, para que se pudesse comprovar a real utilidade da Interface Adaptativa do Caderno Escolar electrónico.

Apesar do motor de inferência deste primeiro protótipo da Interface Adaptativa do Caderno Escolar electrónico, ter sido implementado com recurso ao algoritmo baseado em modelos "escondidos" de Markov, temos como objectivo final a implementação do sistema com recurso ao algoritmo baseado na árvore de decisão C4.5/C5, dado que esta é mais rápida e ligeiramente mais eficaz na procura de padrões e na classificação em classes.

Ao nível da codificação, estas implementações seguem metodologias diferentes, enquanto que a árvore de decisão será implementada através da concepção de uma DLL baseada em código disponibilizado em *open source* por Quilan [Quilan1993], a segunda seguiu uma implementação por classes integradas no código fonte do CE-e.

Assim, foi possível implementar e testar o sistema em busca de erros ou problemas de implementação, em tem-

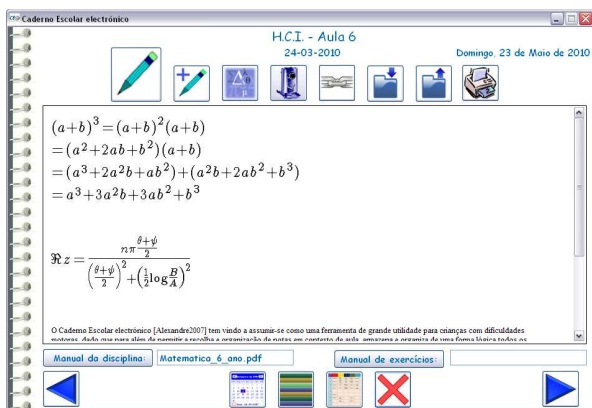


Figura 6 – Protótipo do sistema com o motor de inferência a sugerir a opção mais provável

po útil, antes do início do ano escolar no ensino básico, onde o sistema será efectivamente colocado à prova.

Após a prova de conceito ter sido realizada, a DLL baseada na árvore de decisão de [Quilan1993], continuará a ser desenvolvida. Este esforço justifica-se pelo facto de uma DLL ser independente do sistema e portátil, permitindo assim a sua reutilização noutros sistemas, para além de ser mais fácil a construção de novas versões do CE-e, com e sem interface inteligente, sem que esta situação origine a construção de duas versões, incompatíveis.

Com a introdução do motor de inferência no protótipo funcional do CE-e, o sistema teve de sofrer algumas alterações, não visíveis, que apenas se reflectiram no seu código fonte, para que os vários componentes da sua interface possam ser alterados dinamicamente por indicação do motor de inferência da interface.

4. CONCLUSÃO E TRABALHO FUTURO

Neste artigo propusemos e apresentamos uma interface adaptativa para uma ferramenta que pretende ser uma alternativa aos tradicionais cadernos escolares, para estudantes com necessidades especiais, em particular aqueles que apenas possuem problemas físicos, que lhes restringe a mobilidade e a habilidade para manipular os tradicionais mecanismos de *input*.

Actualmente dispomos de um protótipo do CE-e com a interface adaptativa, em fase final de desenvolvimento. Este protótipo inicial apenas será utilizado para testes e prova de conceito, que estão a decorrer.

A interface faz sugestões ao utilizador com recurso à análise de uma base de conhecimento, que contem informação sobre acções anteriormente efectuadas no sistema. A análise é efectuada com base em técnicas de aprendizagem automática (problema de classificação multi-classe), através de algoritmos baseados em *Redes de Markov*, numa primeira fase e em árvores de decisão na altura da sua implementação final.

Com vista à construção de uma base de conhecimento que conseguisse responder às questões colocadas durante a fase de análise e decisão, o mecanismo de registo de *logs* passou por uma base de dados relacional, com recurso à

tecnologia SQLite, dado que a interrogação de bases de dados em SQL é uma prática bastante eficiente. De igual forma o registo de logs num sistema deste é bastante simples e eficiente, não apresentando qualquer tipo de sobrecarga, quer para o caderno electrónico, como para o sistema operativo.

Como se pretende que o sistema vá aprendendo ao longo do tempo, o ficheiro de *logs* poderá crescer indefinidamente tornando a sua análise pelo motor de inferência da interface do CE-e, muito lenta. Para ultrapassar este potencial problema, o ficheiro *logs* poderá ver o seu crescimento limitado. Esta limitação no crescimento, poderá ser regulada com recurso ao tempo, sendo os *logs* com datas posteriores a trinta dias (efectivamente registados no mesmo) eliminados. Se verificarmos que este período é demasiado curto, ou longo demais, esta regra será reajustada.

No entanto, a hipótese de existir uma limitação temporal dos *logs*, pode ainda sofrer alterações dado que estamos a analisar qual a opção que trará maiores probabilidades de acertos. Se por um lado o sistema apenas conhecer os últimos trinta dias de utilização, terá um conhecimento temporalmente limitado, mas esse conhecimento poderá incidir sobre as opções que o utilizador tem utilizado com maior frequência. Por outro lado e, partindo da hipótese que os *logs* podem crescer indefinidamente, o sistema terá uma visão global de todas as acções e escolhas efectuadas pelo utilizador, o que poderá trazer maiores probabilidades de acerto. Esta decisão só poderá ser tomada, após um período de utilização e testes de duração não inferior a seis meses, que demonstrará qual o melhor tipo de base de conhecimento, “com esquecimento” ou “sem esquecimento”

Uma eventual solução para a recolha contínua de logs, poderá passar pelo seu processamento em *offline*, o que levará à construção de um índice de acções, à semelhança do que se passa num sistema de recuperação de informação em bases de texto [Manning2008].

Outro objectivo, inerente a esta proposta, é dotar o sistema de um conjunto de regras fixas, que em caso de não correspondência entre a mesma e o resultado obtido pelo motor de inferência, se sobreporá à inferência do mesmo. Como exemplo deste tipo de regras temos o horário escolar que poderá fornecer informações sobre qual a tarefa mais provável a executar no sistema, de acordo com a componente tempo fornecida pelo sistema operativo.

O mecanismo de sugestão coloca em realce a opção mais provável. Este realce está aliado a uma tecla de atalho, o que evita um movimento de rato para selecção da opção.

O CE-e pretende ser uma alternativa aos tradicionais cadernos escolares. Se a introdução de uma interface adaptativa provar ser uma mais-valia, os estudantes com dificuldades físico-motoras, terão ao seu dispor uma ferramenta que os ajudará ainda mais na organização de todas as tarefas escolares que envolvam escrita e na organização de todos os materiais escolares.

Nesta altura o CE-e possui uma interface adaptativa que consegue prever qual a acção mais provável para um determinado contexto, o que veio reforçar o principal objectivo do CE-e: Ajudar e aliviar os estudantes com necessidades especiais de tarefas rotineiras, que não acrescentam mais-valias ao trabalho que têm de efectuar, de forma a ajudar na sua integração no ensino regular.

Por último gostaríamos de agradecer a preciosa colaboração do corpo docente do Instituto Politécnico de Beja afecto ao Laboratório de Sistemas de Informação e Interactividade (LabSI²), pelos contributos e ideias dadas para este projecto.

5. REFERÊNCIAS

- [Alexandre2007] Alexandre, L., Garcia, L., Bruno, L.. Development of an electronic scholar notebook for students with special needs. DSAI2007, Vila Real, Portugal. 2007.
- [Alpaydin2004] Alpaydin, E. Introduction to Machine Learning. MIT Press. 2004.
- [Barnard2006] Barnard, L., Yi, J., Jacko, J., Sears, A. Capturing the effects of context on human performance in mobile computing systems. 2006.
- [Blum2005] Blum, M.L. Real-time context recognition. Master's thesis, Department of Information Technology and Electrical Engineering, Swiss Federal Institute of Technology Zurich (ETH), 2005.
- [Boyle2001] Boyle, J., Weishaar, M. The effects of strategic notetaking on the recall and comprehension of lecture information for high school students with learning disabilities. Learning Disabilities Research & Practice, 16(3), pages 133–141, 2001.
- [Bradley2005] Bradley, N., Dunlop, M. Toward a multidisciplinary model of context to support context-aware computing. Human Compute-Interaction, 20:403 – 446, 2005.
- [Dey2001] Dey, A. Understanding and using context. Personal Ubiquitous Computing, 5:1:4 – 7, 2001.
- [Dix2004] Dix, A., Finlay, J., Abowd, G., Beale, R. Human Computer Interaction, 3rd Edition. Prentice Hall, 2004.
- [Garcia2003] Garcia, L. Conceção, implementação e teste de um sistema de apoio à comunicação aumentativa e alternativa para o português europeu. Tese de Mestrado. Universidade Técnica de Lisboa, Instituto Superior Técnico, 2003.
- [Garner1995] Garner, S.R. Weka: The Waikato environment for knowledge analysis. In Proc New Zealand Computer Science Research Students Conference, pages 57-64, University of Waikato, Hamilton, New Zealand, 1995.
- [Gonçalves2003] Gonçalves, M., Panchanathan, G., Ravindranathan, U., Krowne, A., Fox, E. JCDL'03 Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries. 2003.
- [Igarashi2001] Igarashi, T., Hughes, J. A suggestive interface for 3D drawing. Proceedings of the 14th annual ACM symposium on User interface software and technology. Pages: 173 – 181. Orlando, Florida. 2001.
- [Langley1997] Langley, P. Machine Learning for Adaptive User Interfaces. Proceedings of the 21st Annual German Conference on Artificial Intelligence: Advances in Artificial Intelligence. Pages: 53 – 62. 1997.
- [Lesh2000] Lesh, G., Moulton, B., Rinkus, G., Higginbotham, G. A Universal Logging Format for Augmentative Communication. Enkidu Research, Inc. Lockport, NY. 2000.
- [Maes1994] Maes, P. Agents that reduce work and information overload. Communications of the ACM 37(7), 1994.
- [Manning2008] Manning, C.D., Raghavan, P., Schütze, H. An Introduction to Information Retrieval. Cambridge University Press. 2008.
- [Martinez2006] Frias-Martinez, E., Chen, S., Liu, X. Survey of Data Mining Approaches to User Modeling for Adaptive Hypermedia. IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews, Volume 36, No. 6. 2006.
- [Norcio1989] Norcio, A., Stanley, J. Adaptive human-computer interfaces: A literature survey and perspective. IEEE Transactions on Systems, Man and Cybernetics, 19, No.2:399 – 408, 1989.
- [Quilan1993] Quinlan, R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA. 1993.
- [Rabiner1989] Rabiner, Lawrence R. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE 77. 1989.
- [Rodrigues2007] Rodrigues, N. Desenvolvimento de mecanismo de interacção preditiva para aumentar o desempenho de tarefas. Projecto de Licenciatura. Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Beja, 2007.
- [Schilit1994] Schilit, B., Adams, N., Want, R. Context-aware computing applications. IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'94), 1:89 – 101, 1994.
- [Schilit1994-2] Schilit, B., Theimer, M. Disseminating active map information to mobile hosts. IEEE Network, pages 22 – 32, 1994.
- [Teitelman1984] Teitelman, W. A tour through cedar. Proceedings of the 7th international conference on Software engineering. Páginas: 181 – 195. Orlando, Florida, United States. 1984.
- [Vertegaal2002] Vertegaal, R. Designing Attentive Interfaces. ETRA'02. New Orleans - Louisiana, USA - 2002