# Cost/Benefit Based Adaptive Dialog: Case Study Using Empirical Medical Practice Norms and Intelligent Split Menus

Jim Warren
*School of Computer & Information Science, University of South Australia, warren@cs.unisa.edu.au*

## Abstract

The notion of an adaptive user interface, one that accommodates user needs based on knowledge of the task at hand, is compelling but difficult to make practical. This paper examines models of the utility (as balancing of cost and benefit) in the initiation of task-specific dialog based on conditional probability of user goals in context. Illustrations in this paper are based on an empirical model of General Practice (GP) medicine as derived from a large database of GP/patient encounters. Application is explored with respect to generation of dynamic pick-lists (intelligent split menus) of diagnoses in the context of patient reason for encounter (RFE); medication warning dialogs are also briefly addressed. Simulated data entry demonstrates the effectiveness of various adaptation options. Findings include significantly superior accuracy per item of varying-length cost/benefit based pick-lists as compared to fixed length pick-lists. We conclude with discussion of the power of empirical cost/benefit based adaptation to build user engagement and with future directions toward application in clinical practice information systems.

## 1. Introduction

Fischer [1] suggests that human-computer communication has explicit and implicit channels. While the explicit channel is based on bit-mapped display, mouse, voice and the like, the implicit channel is based on shared knowledge structures, including knowledge of the problem domain. "Intelligent behavior builds on large amounts of knowledge about specific domains. This knowledge constrains the number of possible actions and describes reasonable goals and operations" [1]. Brennan notes that computer systems "treat all user utterances as if they were non-sequiturs" [2]. Situation-specific adaptation of direct manipulation (DM) structures (e.g., menus) based on problem domain knowledge structures is an avenue through which the human-computer communications channel can widen based on system expectation of contextually-reasonable user utterances.

The present work is motivated by the problem of data entry in health information systems, where a broad range of real health benefits are achievable *iff* accurate and timely clinical data input can be achieved. Special-purpose decision support systems (DSSs) can provide clinical benefits such as reduction of adverse events (drug reactions, infections, etc.) and errors that lead to adverse events [3-4]. To be effective, however, guidelines that support clinical decisions need to be integrated into the clinician's routine workflow [5]; guideline advice has to be patient-specific and provided at the time and place of consultation [6-7]. Achieving integration of DSSs and electronic medical records (EMRs) in clinical workflow presupposes that reasonably complete and accurate clinical data is being entered into on-line systems. Moreover, if clinicians are to receive feedback on their decision at the time of decisionmaking, then they (or hovering attendants) must be entering clinical data at the time and place of patient care. While a variety of scenarios of nurse support and automated data monitoring are feasible in the hospital setting, in General Practice medicine there is no present realistic alternative to the General Practitioners (GPs) themselves entering at least some information (notably diagnosis and treatment) if they are to benefit from a DSS.

In this paper we explore cost/benefit based user interface adaptation that relies on probability estimates for user data entry actions based on empirical data. We relate the specific problem of applying *intelligent split menus* (see next section) for GP data entry to the far more general concept of utility thresholds for user interface adaptation/action in mixed-initiative user interfaces, as described by Horvitz [8]. Ideally, this paper helps to further unified thinking about DM, intelligent agents and knowledge-based systems. At the minimum, it illustrates a technique with significant benefits for clinical data entry efficiency. The paper utilises a fairly straightforward method of Bayesian estimation based on a large database of uniform records (one might say a 'machine learning' or 'data mining' approach; herein it is termed estimation of *empirical practice norms*). However, the results should generalise to any method of describing domain knowledge that can yield probability estimates for alternatives.



0-7695-0969-X/01 $10.00 © 2001 IEEE

Authorized licensed use limited to: University of South Australia. Downloaded on June 3, 2009 at 23:53 from IEEE Xplore. Restrictions apply.

## 2. Background and Related Work

One application of problem-domain knowledge is to formulate *hotlists*. The idea of the hotlist is to enhance human-computer communication by presenting the $k$ most likely options (goals, actions, operations) for easy selection by the user. For instance, a conditional probability estimate based on a user profile and user query analysis yields the "Help Wizard's Best Guesses for Help" in the Lumiere/Excel system [9]; these sorts of "guesses" are presented as a hotlist by the Microsoft Office Assistant. For data entry improvement, the hotlist can be of one field of a record in terms of another; for instance, in this paper we focus on hotlists of diagnoses in light of the patient's RFEs (reasons for encounter – symptoms, complaints, checkups, etc.). Such hotlisting could be particularly useful in a pen-based computing environment where typing is relatively awkward and thus we wish to maximise the ability to specify items from long lists with point-and-click methods.

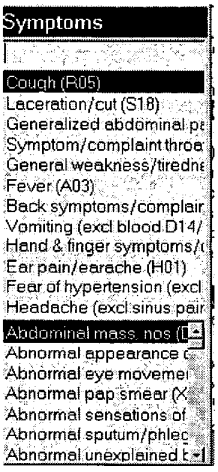



Figure 1. Split menu presentation with 12-item hotlist at top and scrolling alphabetic list underneath (from Mediface demonstrator; available for free download from http://www.cis.unisa.edu.au/~cisjrw/).

A useful presentation of hotlists is in a split menu (figure 1). *Split menus* [10] group a selection of frequently used items in a top section of the menu and then give the remaining items in a bottom section. Sears and Shneiderman [11] find that split menus can be faster and preferred by users as compared to conventional (alphabetic) or fully frequency sorted menus. However, their consideration of split menus focussed on ones where the hotlist remains constant across the duration of the usability trial. Herein we refer to a split menu as *intelligent* if its top section alters its values to suit factors of the current context (a particular moment in entering a patient record). One commonplace intelligent split menu is the font pull-down in Microsoft Word, which has a set of likely font selections above a complete alphabetic list of all available fonts.

Canfield [12] developed intelligent split menus for clinical data entry in echocardiography. Statistical associations between data fields (*anatomy* and *prediction* slots in a frame-based representation) were developed automatically by analysis of existing health care records. Canfield showed (by simulation) a two- to five-times reduction in data entry effort by using anatomy to create a hot list of predictions. The present author and others have examined reimplementation and a number of refinements of Canfield's model to extend intelligent split menus into GP data entry (e.g., [13-15]), which is much less constrained than echocardiography. While success in narrow modeled-defined terms, such as reduction of mouse-clicks [12] is a positive finding, this fails to provide a framework for comparison of the value of knowledge-based adaptation to static user interface designs or of intelligent split menus to other adaptive options (e.g., anthropomorphic agents).

Norcio and Stanley present an overview of adaptive user interfaces wherein they identify the knowledge resources of an adaptive user interface to include knowledge of the user, the task/domain, the interaction and the system [16]. Moreover, they (borrowing from [17]) identify chief drawbacks of adaptive interfaces: (a) increasing difficulty for the user to develop a coherent mental model of the system, (b) a feeling of loss of control on the part of the user, and (c) additional complexity and cost in implementing such a system. The intelligent split menu seems to minimise the first two drawbacks in that the area of adaptation is constrained to the hotlist within an otherwise familiar menu structure (have you ever felt noticeable disorientation or loss of control by way of the upper portion of the Word font menu?).

Horvitz [8] describes principles of *mixed-initiative user interfaces* that blend DM and interface agency, a particularly useful aspect of which is a model of the expected utility of an action, A, initiated by the user interface agent given evidence, E:

$$eu(A \mid E) = p(G \mid E)u(A \mid G) + p(\neg G \mid E)u(A \mid \neg G) \quad (1)$$

where $p(G \mid E)$ is the probability that the user has a goal, G, given evidence of user intent, E; $u(A \mid G)$ is the utility of the user interface agent taking action, A, to achieve G given that G is in fact a goal of the user; and $u(A \mid \neg G)$ is the (lower) utility of the agent taking action A when G is not the user's goal. Horvitz refines this model to include an intermediate agent action, D, where the agent engages the user in dialog to clarify user goals. Such an action D has higher utility (lower cost) than immediately taking action A to affect the goal G when the user did not intend that goal, $u(D|\neg G) > u(A|\neg G)$. In



Authorized licensed use limited to: University of South Australia. Downloaded on June 3, 2009 at 23:53 from IEEE Xplore. Restrictions apply.

clinical practice, we consider the cost of A|¬G to be prohibitively high (low utility). This owes to the undesirability of having a machine "practice medicine" (due to the unsuitability of the machine as a bearer of clinical responsibility) and the known negative reaction of clinicians to systems that present themselves as a 'Greek Oracle' [18].

We apply Horvitz's utility concept (in terms of cost and benefit) to place adaptive actions of DM interfaces in a unified framework with thinking about interface agent actions. Horvitz's paper is already close to doing this, but the emphasis seems to be on "the effective integration of automated services with[in] direct manipulation" rather than the unification *of* DM and agency approaches. Moreover, illustrations in [8] tends to run toward anthropomorphic genies that seem far to the Maes side of the Shneiderman/Maes debate [19] on interface agents versus DM interfaces. In the present paper, the chief dialog action under consideration is whether or not to display an *i*th item in an intelligent split menu based on the cost/benefit assessed along the lines of equation (1).

Hotlists can be generated in any direction among the fields of a record (drugs on diagnoses, RFEs on other concurrent RFEs, etc.) to suit the user. In [14] we describe analysis of General Practice data based on conditional probability on all interactions of the RFE, diagnosis and drug axes and present a demonstration prototype.

## 3. Method

### 3.1 The Data

To develop a model of General Practice medicine, we analyse a data set intended for this purpose, the Australian Morbidity and Therapeutic Survey (AMTS), describing 113,000 General Practice encounters in Australia during 1990-1991 [20] (accessed from the Social Science Data Archives of the Australian National University, Canberra, http://ssda.anu.edu.au/, April 2000). AMTS employs a coarse-grained coding system, WONCA's International Classification of Primary Care (ICPC) [21], which is arguably well suited to the level of detail that is useful to the GP. We examine three dimensions of the data: (a) RFE, (b) diagnosis (Dx), and (c) treatment (in fact we look just at drug prescriptions [scripts], coded at the level of generic drug name, and not at procedural treatments, such as removing excessive ear wax). Each record provides up to 3 RFEs, up to 4 Dxs, and up to four drug scripts for *each* Dx.

In the data set there are 168,504 RFEs, 167,005 diagnoses and 112,337 drug scripts in the 113,469 encounters. 1059 distinct RFE codes, 988 distinct diagnosis codes and 882 distinct generic drug names are employed. The most commonly treated diagnosis is Uncomplicated Hypertension (ICPC code K86), being treated in 10,377 encounters with 123 distinct generic drugs. Figure 2 graphs the cumulative frequency of drug treatments for the 20 most frequent drugs applied to each of the three most commonly treated diagnoses. As can be seen, a relative handful of drugs account for most treatment of any given diagnosis, however, each disease has its own unique character. For instance, there are relatively few common treatments for asthma (10 drugs accounting for almost 90% of scripts), as compared to hypertension and URI (upper respiratory infection – the common head cold). The "head" (first several values) of the URI distribution consists of antibiotics (a practice more actively discouraged now than it was 10 years ago) and headache remedies. The very substantial "tail" of the distribution (20% of the scripts outside of the 20 most popular drugs) is indicative of the fact that there is no cure for the common cold, but a wide variety of treatments of symptoms. The spread of hypertension treatments owes to the fact that there are several tiers of drug therapy available depending on severity of high blood pressure and individual characteristics.

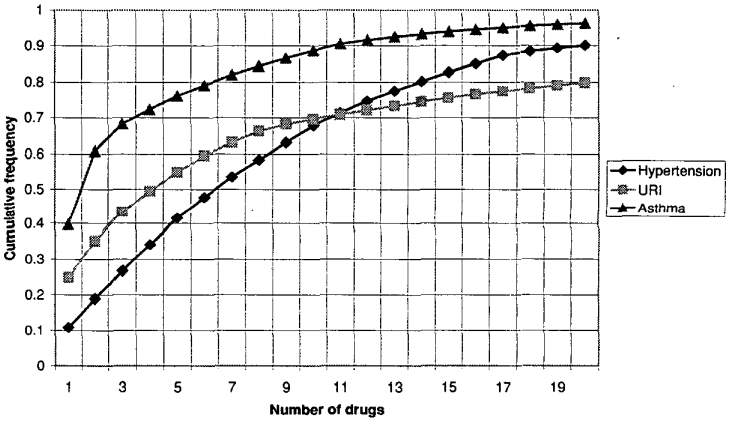



Figure 2. Cumulative frequency distribution of scripts for the 3 most commonly treated diagnoses.

A useful tool in looking at the relationship of two fields of an encounter (e.g., to look at the interaction of specific RFEs and diagnoses) is to create interaction tables as per figure 3. In figure 4 the interaction of sore throat with acute otitis media (earache), uncomplicated hypertension, URI and herpes is illustrated.

Several useful statistics can be defined in terms of the cells of the interaction table:

1. *Conditional probability*, $p(dx \mid RFE) = a / (a + c)$; (2)
2. *Sensitivity* $= a / (a + b)$; (3)
3. *Specificity* $= d / (c + d)$. (4)



Authorized licensed use limited to: University of South Australia. Downloaded on June 3, 2009 at 23:53 from IEEE Xplore. Restrictions apply.

High sensitivity and specificity are useful characteristics of a clinical test – one wants to order a test that is likely to show a "positive" result if the disease is present (sensitive) and that is likely to have a negative result if the disease is absent (specific). However, for complaints that are volunteered by a patient, sensitivity and conditional probability may be more useful than specificity.

### 3.2 Models for Hotlist Formulation

The most straightforward model for hotlist formulation is to order diagnoses in descending order of conditional probability in light of the patient's RFE. The $k$ most likely diagnoses are presented as the hotlist, where $k$ is whatever hotlist length presents the most desirable balance of *hit rate* (frequency with which desired diagnoses actually appear on the hot list) and number of items to present to the user (longer lists take more time for the user to scan and consume more screen real estate). A further complication arises in that patients often have more than one RFE per encounter. We use a crude approximation:

$$P(dx \mid RFE_1 \,\&\, RFE_2 \,\&\, \ldots\, RFE_n) = \max(P(dx \mid RFE_1), P(dx \mid RFE_2), \ldots P(dx \mid RFE_n)) \quad (5)$$

This approximation is actually quite logical if the multiple RFEs are disparate in nature (e.g., cough and routine cardiac checkup).

| Diagnosis | Complaint: Yes | Complaint: No | |
|---|---|---|---|
| **Yes** | a | b | a+b |
| **No** | c | d | c+d |
| | a+c | b+d | Total |

Figure 3. Template of an interaction table with observed frequencies a, b, c and d.

The author's previous work [14] examined hotlist length $k$ as a constant, however, a more refined approach is to list howsoever many items are of value. A number of factors can be identified as contributing to the cost of listing the $i$th item in a hostlist:

1. The "dismay cost" of presenting item $i$ (high if item $i$ is not particularly appropriate to the case at hand);
2. The "clutter cost" of presenting an $i$th item (any item, even if not explicitly scanned by the user, will take up screen realestate and add to the level of visual disruption brought on by presentation of the hotlist);
3. The "processing time" (visual scanning and cognition) to test if item $i$ matches the user's goal.

Similar factors contribute to the benefit of listing item $i$:

1. The "reassurance value" of presenting item $i$ (high if item $i$ is appropriate to the case at hand and hence the user is reassured that the system is "following" the likely nature of the case);
2. The "time savings" *iff* item $i$ matches the user's goal and can hence be selected by a simple menu "tick" rather than being selected by some further method (e.g., searching a long alphabetic list, using typing, or voice input).

In the first instance the "processing time" and "time savings" factors are the most tangible. Adapting equation 1 and using equation 5 as our estimate of the probability that $dx_n$ is in the set of the user's current goals, we have the rule:

Place $dx_n$ in hotlist
iff *Benefit* × $p(dx_n \mid RFEs)$ -
*Cost* × $(1 - p(dx_n \mid RFEs)) > 0$ (6)

The hotlist length then varies from case to case and depending on the values of cost and benefit, with the rational hotlist being the set of dxs satisfying equation 6 sorted in descending order of weighted Benefit minus

Acute otitis media

| | Sore throat: Yes | Sore throat: No | |
|---|---|---|---|
| **Yes** | 107 | 2181 | 2288 |
| **No** | 5284 | 104638 | 109922 |
| | 5391 | 106819 | 112210 |

URI

| | Sore throat: Yes | Sore throat: No | |
|---|---|---|---|
| **Yes** | 2875 | 5118 | 7993 |
| **No** | 2516 | 101701 | 104217 |
| | 5391 | 106819 | 112210 |

Hypertension

| | Sore throat: Yes | Sore throat: No | |
|---|---|---|---|
| **Yes** | 101 | 10167 | 10268 |
| **No** | 5290 | 96652 | 101942 |
| | 5391 | 106819 | 112210 |

Herpes

| | Sore throat: Yes | Sore throat: No | |
|---|---|---|---|
| **Yes** | 15 | 166 | 181 |
| **No** | 5376 | 106653 | 112029 |
| | 5391 | 106819 | 112210 |

Figure 4. Interaction tables based on a 99% sample of the AMTS data set.



Authorized licensed use limited to: University of South Australia. Downloaded on June 3, 2009 at 23:53 from IEEE Xplore. Restrictions apply.

weighted Cost. In this context, $p(dx_i \mid RFEs)$ is in truth an imperfect criterion for the probability of item $i$ being part of the user's goal set, $p(dx_i \in G)$, since we should consider that the $i$th item will only be viewed if the first $i$-1 items have not covered all elements of G (i.e., the user's still looking for another diagnosis). Significant correlations among diagnoses in the same case (common "comorbidities") would alter the estimated value of listing an item. However, at present we are ignoring the effect of such correlations.

It is possible to construct fairly accurate models to predict human performance in tasks such as specifying items from a menu using the Model Human Processor (MHP) [22]. Applying MHP as per Eberts [23] we find that reviewing a menu item to find a match takes time:

$$t_{review} = (\tau_p + 2 \times \tau_c + \tau_m) \quad (7)$$

Where $\tau_p$ is time to perceive an item, $\tau_c$ is cognitive processing (in this case two cycles, one to match and one to decide a course of action) and $\tau_m$ is motor time to look to the next item. Using "Middle Man" (typical person) values for the timing constants, the review time becomes:

$$t_{review} = (100ms + 2 \times 70ms + 70ms) = 310ms$$

Thus 310ms is a defensible estimate of *Cost*. An appropriate value of *Benefit* is harder to derive analytically since it will depend on the precise mix of scanning, scrolling and typing completion (notwithstanding other technologies, such as voice) that the user may employ should the hotlist fail. We will explore the effect of a range of *Benefit* values with *Cost* fixed at 310ms (the ratio being the only important factor in our cost/benefit model as defined). Benefit values on the order of several seconds seem reasonable in light of the full diagnosis list being 988 items and diagnoses having less intuitive alphabetical order than, say, generic drug names (e.g., "uncomplicated hypertension" sorts under "u" not "h").

As a further refinement to our model we consider the role of sensitivity. Sensitivity itself is not a good ordering criteria for hotlists. Consider from figure 4 that sore throat has a modest sensitivity for herpes at 9% -- but hotlisting herpes on sore throat would be very strange in light of the overall rarity of herpes. However, sensitivity may have a role as a gatekeeper for hotlists. In particular, hypertension has such a high overall frequency that it is apt to appear in the hotlist for almost any complaint when sorted by conditional probability. Listing an item of low sensitivity for the case at hand could incur a high "dismay cost" in that it could undermine the user's confidence in the system's model.

The experiments reported in the next section explore various levels of *sensitivity cutoff*; i.e., hotlist entries must exceed a certain sensitivity level if they are to be included, thus removing coincidental entries from the hotlist. This use of sensitivity helps to compensate for the fact that the AMTS data set presents up to three RFEs and up to four diagnoses for each encounter with no explicit causal linkage, merely the fact that they all co-occur in the same encounter.

### 3.3 Hotlist Simulation Method

To demonstrate and evaluate our hotlisting model we simulate data entry on encounters held back from training. A random selection of 1% of the AMTS encounter records (1145 encounters) are held back from training to form a testing set. For the balance of encounters, we count the co-occurrence of RFEs and diagnoses to instantiate cross-tabulations (as per figures 3 and 4) for each RFE code with each diagnosis code. From these cross-

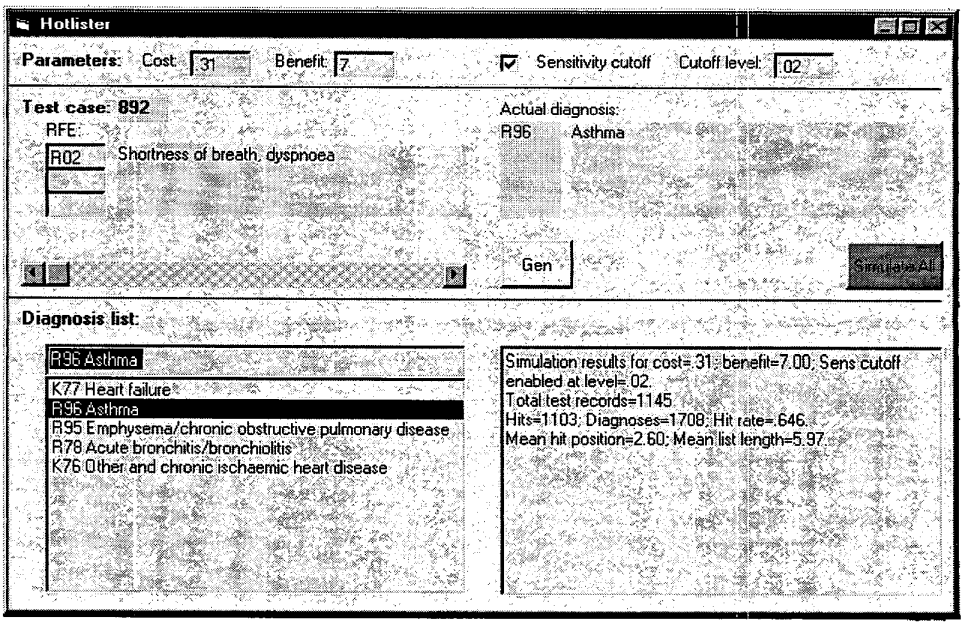



Figure 5. Hot list explorer application.



Authorized licensed use limited to: University of South Australia. Downloaded on June 3, 2009 at 23:53 from IEEE Xplore. Restrictions apply.

tabulations we compute a list of diagnoses with conditional probability (Eqs 2 and 5) and sensitivity (Eq 3, also maximised over RFEs) for the given set of RFEs in a test encounter. A hotlist is then presented for a test case based on the parameters of the *Cost* (fixed at 0.31s as per Eq 7), *Benefit* and *sensitivity cutoff* where all diagnoses satisfying Eq. 6 and the sensitivity cutoff are included in the hotlist in order of descending conditional probability.

We have developed a hotlist explorer program (figure 5) to produce simulated hotlists for test cases with interactive input of the parameters. Pressing the "Simulate All" button generates the hotlists for all 1145 test cases and reports the summary statistics of *hit rate* (frequency with which the actual test-case diagnoses appeared on the hotlists), *mean hit position* (when a diagnosis appeared on the list, the average position at which it was found) and *mean hotlist length* (over the 1145 lists).

## 4. Results and Discussion

Table 1 reports the results of simulations over a range of parameter values including several values of fixed hotlist length $k$ (where the observed mean list length is slightly less than $k$ due to some RFEs having less than $k$ diagnoses observed to co-occur with them in the training data).

The results show that the cost/benefit method for including items in the hotlist gives consistently superior performance on hit rate per item listed over the use of a fixed-length hotlist. For instance, the mean difference in hit rates comparing the cost/benefit experiments at benefit = 5, 7, 10 and 12 to the four hit rates observed for constant-length list is significantly greater than zero (P(t) = 0.047, one-tailed). The use of a sensitivity cutoff gives no quantitative advantage, although, as long as the cutoff is lowered along with the cost/benefit ratio for longer hotlists, there is no evident degradation in hit rate performance per item hotlisted. Thus, the observable advantage, if any, of sensitivity cutoff is in the presentation of a more relevant-seeming list, which may lower the "dismay cost" discussed earlier.

The hotlists are useful in terms of diagnosis selection efficiency if the time to select a diagnosis by other user interface methods is over 2 or 3 seconds. This can be shown algebraically by extension of Eq 7 to the review of a whole hotlist. For instance, the mean time investment for the user to scan the hotlists at cost/benefit ratio 0.31:7 with no sensitivity cutoff is:

Table 1. Hotlisting results for simulated data entry of 1145 general practice encounters with 1708 total diagnoses (*Cost* held at 0.31 for analyses using cost/benefit methods).

| method | k | benefit | sens. cutoff | # of hits | hit rate | mean hit pos | mean list len |
|---|---|---|---|---|---|---|---|
| Constant length list | 5 | N/A | N/A | 1009 | 0.591 | 1.99 | 4.99 |
| | 8 | N/A | N/A | 1130 | 0.662 | 2.51 | 7.98 |
| | 12 | N/A | N/A | 1235 | 0.723 | 3.17 | 11.96 |
| | 15 | N/A | N/A | 1278 | 0.748 | 3.53 | 14.93 |
| Cost/benefit | N/A | 3 | none | 924 | 0.541 | 1.91 | 3.38 |
| | " | 5 | none | 1081 | 0.633 | 2.46 | 5.4 |
| | " | 7 | none | 1167 | 0.683 | 2.96 | 7.72 |
| | " | 10 | none | 1251 | 0.732 | 3.54 | 11.46 |
| | " | 12 | none | 1288 | 0.754 | 3.92 | 13.71 |
| Cost/benefit with sensitivity cutoff | " | 3 | 0.02 | 894 | 0.523 | 1.81 | 2.96 |
| | " | 5 | 0.02 | 1033 | 0.605 | 2.27 | 4.39 |
| | " | 7 | 0.02 | 1103 | 0.646 | 2.6 | 5.97 |
| | " | 10 | 0.02 | 1167 | 0.683 | 2.99 | 8.45 |
| | " | 12 | 0.02 | 1194 | 0.699 | 3.2 | 9.75 |
| | " | 15 | 0.02 | 1224 | 0.717 | 3.48 | 11.53 |
| | " | 18 | 0.02 | 1249 | 0.731 | 3.81 | 13.59 |
| | " | 10 | 0.01 | 1210 | 0.708 | 3.21 | 9.42 |
| | " | 12 | 0.01 | 1240 | 0.726 | 3.48 | 11.02 |
| | " | 15 | 0.01 | 1275 | 0.746 | 3.84 | 13.37 |
| | " | 10 | 0.005 | 1232 | 0.721 | 3.35 | 10.18 |
| | " | 12 | 0.005 | 1267 | 0.742 | 3.67 | 12.02 |
| | " | 15 | 0.005 | 1304 | 0.763 | 4.06 | 14.86 |
| | " | 10 | 0.0025 | 1244 | 0.728 | 3.47 | 10.68 |
| | " | 12 | 0.0025 | 1280 | 0.749 | 3.82 | 12.67 |
| | " | 15 | 0.0025 | 1321 | 0.773 | 4.25 | 15.78 |



Authorized licensed use limited to: University of South Australia. Downloaded on June 3, 2009 at 23:53 from IEEE Xplore. Restrictions apply.

$t_{invest}$ = 310ms x (0.683 x 2.96 + (1 – 0.683) x 7.72)
= 1385ms

This investment, resulting in the identification of the desired "target" diagnosis 68.3% of the time and resulting in a resort to conventional selection or specification the other 31.7% of the time, pays off if:

$t_{conventional} > 310ms \times 2.96 + [7.72 \times 310ms \times (1 - 0.683)] / 0.683 > 2028ms$

where $t_{conventional}$ is the time to identify the desired diagnosis by other means (discounting the final mouse-click or other action to actually select the item in both $t_{invest}$ and $t_{conventional}$). Unless the user has access to reliable voice technology (appropriate for summarising notes after a consultation, but possibly a poor fit for in-consultation decision support as compared to a discreet pen-based system), the 2 second lower bound places the hotlists firmly in the realm of positive utility.

## 5. Further Areas of Application

Referring back to figure 2 we see substantial "tails" on the empirical cumulative frequency distributions of drugs for a given diagnosis. Is all this variation warranted? Or does it represent a variation in quality of clinical practice? Or perhaps variation in quality of the clinical coding (i.e., the reality may be OK, but the way in which the ICPC scheme was used to create the records shows variation)? While hotlisting uses the most frequent items in the distribution, perhaps there is some user interface applicability for the least frequent items as well.

There appears to be some justification in giving negative/challenging feedback in association with the "tail" distribution selections (say, the contextually least frequent 2%, substantiated by GP opinion in [24]). A passive form of such feedback could be colour coding in conventional pick-lists (such as the bottom portion of a split menu) providing dimmer colouring of tail and never-seen drugs in comparison to the contextually common ones. Keeping in mind that these tail treatments are by definition only 2% of cases, it may be appropriate to engage the user in a clarification dialog (e.g., "Abnormal treatment: Belomethasone Inhaler appears as a treatment for Uncomplicated Hypertension in only 3 of 10,377 cases (0.03%)"). Such a dialog may prompt the user to reconsider their course of action. Alternatively, valuable medico-legal documentation may result if the user chooses to proceed, but enters an explanation.

The recent report to the Health Ministers by the National Electronic Health Records Taskforce (http://www.health.gov.au/healthonline/ehr_rep.htm) places preventable deaths in Australia due to adverse events in care at over 12,000 per annum and the cost (excluding deaths) at A$3 billion per annum. While it is difficult from the current data to attribute a fraction of this cost to GP behaviour (in fact, most cost owes to hospital events), one can imagine estimating the dollar value of a warning dialog in terms of cost of an adverse event and its probability of preventing said event. This benefit estimate could then be balanced against the cost estimate (e.g., GP time at $100/hr and 3.6s to review an alert yields a cost of 10 cents).

## 6. Conclusions and Further Directions

In this paper we have investigated the use of empirical norms, as learned from a large database of General Practice records, as a basis for feedback to make on-line data entry more engaging and relevant to clinicians. Conditional probability and sensitivity assessments can provide improved data entry efficiency in the form of "hotlists" (dynamic pick-lists) of relevant selections for the case at hand. Moreover, the results can be viewed in the more general context of illustrating utility based dialog actions. A wide range of dynamic user interface behaviours – e.g. adding an element to a menu, popping up a warning dialog box, or invoking an anthropomorphic interface agent – can be viewed in the unified framework of empirical estimation of utility. Such an estimation is derived from potentially tangible quantities: (a) the cost of the interface's embedded model guessing wrong, (b) the benefit of guessing right, and (c) the conditional probability of a guess in fact matching the user's goals.

It is important to be mindful that empirical norms are not static, universal or necessarily good. GPs assessing hypertension records in the AMTS data set (from 1990-91) pointed out that it was hard to recall what was acceptable practice 10 years ago in that so much had changed. To be useful in practice, the norms must be contextualised to at least the national level and recent years. Integration with expert knowledge is vital. The best place for the knowledge acquisition to begin is with the hotlists of common situations, such as the treatment of URI (head cold). Attaching expert negative feedback flags to common (esp. hotlisted) treatment errors (e.g., antibiotics to make patients with URI go away happy [25]) would be a method for quick dissemination of education to contravene large-scale bad practice habits.

We conclude by emphasising the potential for on-line systems based on empirical norms to "bootstrap" into better systems. On-line feedback should promote "cleaner" data sets by: (a) highlighting errors and quality deficiencies in coding with unhelpful hotlists, as well as empirical or rule-based warnings that practice appears inappropriate (e.g., contraindicated by clinical evidence); (b) flagging correctly-coded-but-inappropriate practices as violations of rules and/or norms. Moreover, as users find the systems relevant and efficient, increased usage will lead to larger, more current and more localised data sets, permitting construction of better-fitting normative models.



Authorized licensed use limited to: University of South Australia. Downloaded on June 3, 2009 at 23:53 from IEEE Xplore. Restrictions apply.

## Acknowledgments

Thanks are owed to Patrick Bolton and Oliver Frank, General Practitioner, for their help in interpreting clinical data. Thanks are owed to Debbie Richards, Sophie Cockcroft and Richi Nayak for acting as a sounding board for the first author's fuzzy ideas. Finally, thanks are owed to Debra Warren, ultimate SQL whiz.

## References

[1] G. Fischer (1989). Human-computer interaction software: Lessons learned, challenges ahead, *IEEE Software*, January, 44-52.

[2] S. Brennen (1990) Conversation as direct manipulation: an inconoclstic view. In, *The Art of Human-Computer Interaction Design* (Ed; BK Laurel), Lawrene Erlbaum, Hillsdle, NJ, pp. 339-352.

[3] R. Raschke, B. Gollihare, T. Wunderlich, J. Guidry, A. Leibowitz, J. Peirce, L. Lemelson, M. Heisler, C. Susong, (1998). A computer alert system to prevent injury from adverse drug events, *JAMA* **280**, 1317-1320.

[4] R. Scott Evans, et al. (1998). A computer-assisted management program for anitbiotics and other antiinfective agents, *NEJM* **338**(4), 232-238.

[5] J. Grimshaw and I. Russel, Effect of clinical guidelines on medical practice: a systematic review of rigorous evaluations, *Lancet* **324** (1993) 1317-1322.

[6] R. Shiffman, Representation of clinical practice guidelines in conventional and augmented decision tables, *J Amer Med Inform Assoc* **4**(1997) 382-393.

[7] R.D. Zielstorff, Online practice guidelines, *J Amer Med Inform Assoc* **5** (1998) 227-236.

[8] E. Horvitz (1999) Principles of mixed-initiative user interfaces. In, *Proc. ACM SIGCHI Conf. on Human Factors in Computing*. ACM Press, pp. 159-166.

[9] E. Horvitz, J. Breese, D. Heckerman, D. Hovel and K. Rommelse (1998) the Lumiere project: Bayesian user modeling for interring the goals and needs of softwaree users. In, *Proc. Fourteenth Conf. on Uncertainty in Artificial Intelligence.*

[10] Sears A. *Layout appropriateness: guiding user interface design with simple task descriptions* [dissertation]. College Park (MD): Computer Science Dept., Univ. of Maryland; 1993.

[11] Sears A, Shneiderman B. Split menus: effectively using selection frequency to organize menus. *ACM Trans Computer-Human Interaction* 1994; 1 (1):27-51.

[12] Canfield G. Priming intelligent split menus with text corpora for computerized patient record data-entry. *Int J Bio-Medical Comput* 1995;39:263-73.

[13] Spenceley, S. & Warren, J. (1998) The intelligent interface for on-line electronic medical records with temporal data mining, *Proc. Thirty-First Hawaii International Conf. Systems Sciences*, IEEE Computer Society Press, January.

[14] Warren, J. & Bolton, P. (1999) Intelligent split menus for data entry: a simulation study in general practice medicine, *J Amer Med Inform Assoc, Symposium Supplement*, November 1999, pp. 450-454.

[15] George, S. & Warren, J. (2000) Statistical modelling of general practice medicine for computer assisted data entry in electronic medical records, *International Journal of Medical Informatics*, Vol. 57/2-3, pp. 77-89, July.

[16] Norcio, A. & Stanley, J. (1989). Adaptive human-computer interfaces: a literature survey and perspective, *IEEE Trans. Systems, Man, and Cybernetcis*, 19(2), March, 399-408.

[17] S. Greenburg & I. Witten (1985) Adaptive interfaces – a question of viability, *Behavior and Inform. Tech.*, 4(1), 31-45.

[18] R.A. Miller and F.E. Maserie (1990) "The demise of the 'Greek oracle' model for medical diagnostic systems," *Methods of Information in Medicine*, Vol. 29, pp. 1-2.

[19] P. Maes & B. Shneiderman (1997) Direct manipulation vs. interface agents, *Interactions* IV(6), ACM Press.

[20] Bridges-Webb C, Britt H, Miles D, Neary S, Charles J, Traynor V. Morbidity and treatment in general practice in Australia 1990-1991. *Med J Aust* 1992 October 19;suppl.

[21] Juillet, M.J. (1997) ICPC Tabular List, Université Libre de Bruxelles. URL: http://www.ulb.ac.be/esp/wicc/table-en.html (last updated 22 Sept. 98; accessed 29 May 2000).

[22] Card S, Moran T, Newal A. The psychology of human computer interaction. Hillsdale (NJ): Erlbaum; 1983.

[23] Eberts, R.E., *User Interface Design*, Prentice-Hall, 1994.

[24] Warren, J. & Bolton, P., Empirical Norms as a Lever for On-line Support of General Practice, submitted to *Proc. 34th Hawaii International Conf. System Sciences*, Maui, January 2001.

[25] Bolton, P., Usher, H., Mira, M., Prior, G. & Harding, L. (1996). Description of and results from a computerised data collection in primary care. In, (McGuiness, B. & Leeder, T., Eds.) *Proc 4th Nat'l Health Informatics Conf.* Health Informatics Society of Australia, Melbourne.



Authorized licensed use limited to: University of South Australia. Downloaded on June 3, 2009 at 23:53 from IEEE Xplore. Restrictions apply.