A Computer-Based Tool for Generation of Progress Notes

Keith E. Campbell, M.D.,[1] Karen Wieckert, M.S.,[2]
Lawrence M. Fagan, M.D. Ph.D.,[1] and Mark A. Musen, M.D. Ph.D.[1]

[1]Section on Medical Informatics, Stanford University, Stanford, CA 94305-5479 USA
[2]Institute for Research on Learning, 2550 Hanover Street, Palo Alto, CA 94304-1115 USA

IVORY, a computer-based tool that uses clinical findings as the basic unit for composing progress notes, generates progress notes more efficiently than does a character-based word processor. IVORY's clinical findings are contained within a structured vocabulary that we developed to support generation of both prose progress notes and SNOMED III codes. Observational studies of physician participation in the development of IVORY's structured vocabulary have helped us to identify areas where changes are required before IVORY will be acceptable for routine clinical use.

1. COMPUTER-BASED GENERATION OF PROGRESS NOTES

Integration of computer-based decision-support tools into the routine workflow of the physician is a major challenge facing medical informatics today. Whether physicians see patients in hospitals or in clinics, they spend substantial time reviewing previous progress notes and creating new notes to describe patient encounters and treatment plans. A computer-based tool that makes this process more efficient would be welcome.

In this paper, we describe the design rationale and development process for IVORY, a tool that helps physicians to generate progress notes efficiently (Figure 1). This tool also captures coded clinical data of the type required to facilitate real-time decision support, outcomes research, and quality assurance.

2. DESIGN CONSIDERATIONS

In designing IVORY, we relied on accepted concepts from the problem-oriented structure of progress notes championed by Weed during the 1960s [1].

The PROMIS system [2] was one of the earliest automated tools for physicians' entry of progress notes using Weed's problem-oriented structure. In 1980, a formal evaluation revealed that physicians would not use the PROMIS system because the interface was awkward, the organization of the generated progress notes was inflexible, and the interaction with the computer was time consuming compared to that with a manual record. Because of these problems, the PROMIS system was removed from the Medical Center of Vermont.

When PROMIS was developed in the 1970s, software designers did not understand the effects that implementation of their systems would have in the workplace. Since that time, developers have been incorporating new methods into the design process to enhance user acceptance and, thereby, the success of software systems. These methods include user-centered design, information-flow studies, cognitive psychology, and sociological analysis of the workplace [3,4]. In addition, computer technology has evolved to allow users to interact with computers with less effort than was previously possible.

Two separate considerations are key in the overall design of a system. The first is identification of clinical requirements that make a system acceptable to physicians. The second is user participation in the design process itself. These In Section 3, we describe the rationale for IVORY's clinical requirements and the vocabulary structure that we developed to meet those requirements. In Section 4, we describe user participation in the design process itself. This combination of a rigorous analysis of clinical requirements and a participatory design process make it possible for us to develop computer-based decision-support tools that integrate smoothly into the clinical environment.

3. CLINICAL REQUIREMENTS

Safran [5] showed, in a study at Beth Israel Hospital, that physicians readily enter data directly into a computing system when they have appropriate tools and believe that the system will help them to care for their patients. Although this experience is encouraging, the data entered by physicians consisted mainly of problem lists and medications, and lacked a structured representation of patients' subjective and objective findings. These data provide sufficient clinical information for limited decision support, but additional support would be possible if the entire progress note were entered and captured symbolically.

Recent efforts have created such applications [6, 7], a particularly extensive project being the PEN&PAD system at the University of Manchester [8]. This system allows physicians to enter findings and modifiers using a logically developed, structured vocabulary; it generates a progress note by simply stringing together entered findings and modifiers into a fixed format. However, this format may not be appropriate for progress notes.

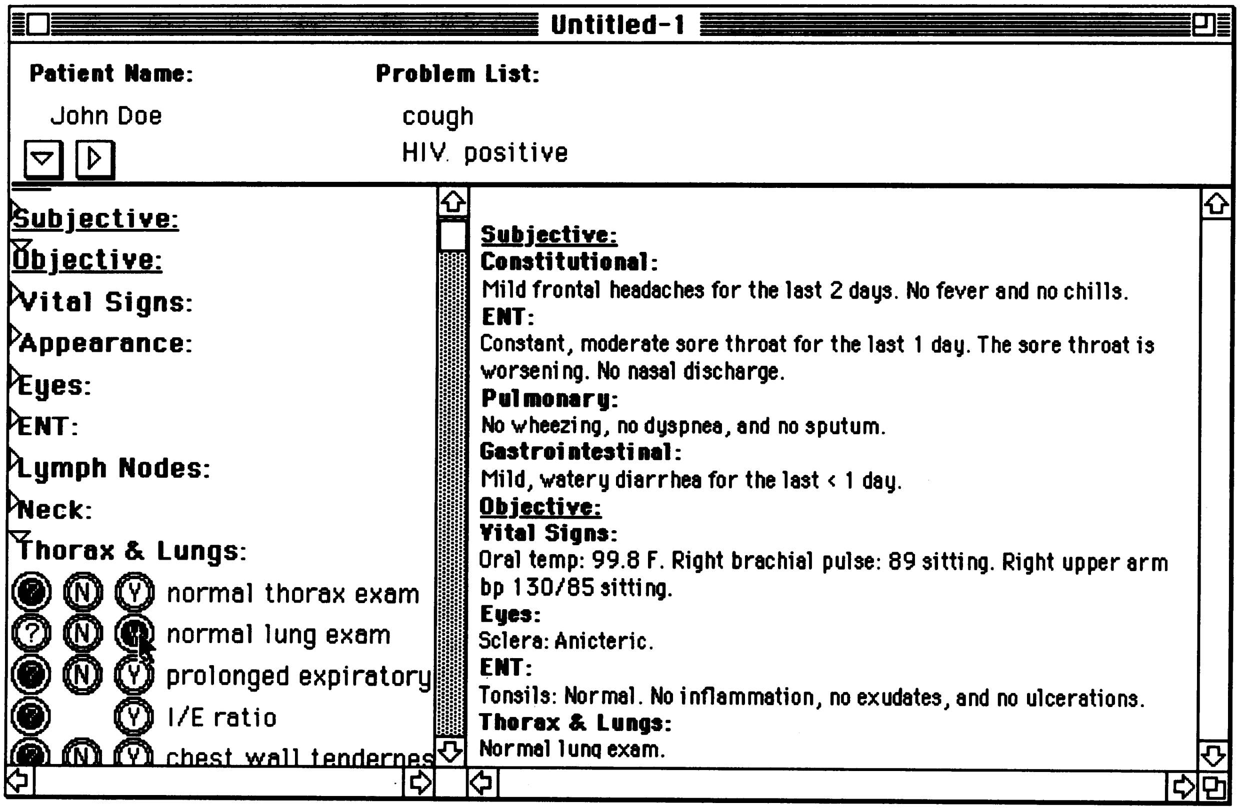

Figure 1. Main screen from IVORY, showing an IVORY-generated progress note.

Other researchers' experience with patient-history-taking systems indicate that PEN&PAD's simple format may not be the best. Quaak has shown that generating fluent prose and structuring its presentation properly is important in conveying patient information to the health-care provider [9].

In summary, to be acceptable to physicians, progress note entry must be efficient. The progress note also must convey accurate clinical information structured to make the information easy to locate.

3.1 Structure of Progress Notes

The structure of a progress note has significant influence on the interpretation of the clinical information contained within the note. The most important rule in composing a progress note is to present the information in a logical, consistent manner. Physicians typically do so by grouping the progress note into four sections; the result is called the SOAP format:

- *Subjective*: Record of patient's reported symptoms
- *Objective*: Record of physician's observations about the patient
- *Assessment*: Physician's opinion relating subjective and objective information to the progression or regression of a particular disease or problem
- *Plan*: Description of treatments or further diagnostic tests to be ordered for the patient

Weed [1] advocated that a separate SOAP note be recorded for each patient problem. Many physicians, however, found that this method fragmented the recording of historical and observational data about a patient; they therefore chose to modify the SOAP note by grouping all subjective and all objective information together. We have adopted this modified version of the SOAP note as the basis for the progress notes generated by IVORY. IVORY generates prose progress notes by combining findings (e.g., cough) with modifiers (e.g., severity and/or onset).

3.2 Generation of Fluent Prose

As mentioned in Section 3, generation of fluent prose and proper structuring of the presentation influence the usability of different reports. We relied on Kukich's work [10] to guide the design of our text-generation scheme. Kukich analyzed factors responsible for fluent written text and for common fluency defects found in computer-generated text. Two common defects were term repetition and lack of appropriate clause combination. The IVORY text-generation scheme combines related findings into single sentences, and, within a sentence, prevents the repetition of terms.

IVORY's patient-description vocabulary has three fields for each individual finding, the values of which are used by the text generator to combine findings into clauses. These fields are used in conjunction with a finding's presence or absence to control the combination process. This scheme for combining findings allows the creation of combined clauses such as "3-day history of cough and sore throat, and a 1-day history of ear pain. No fever, no chills, and no diarrhea." In this example, we prevented repetition of temporal modifiers by combining "cough" and "sore throat" into a single phrase. IVORY combined the positive findings to create a single sentence, and also combined the negative findings to create a single sentence.

Some defects that Kukich noted—such as excessively long sentences, term repetition, and overuse of syntactic forms—can be prevented with simple algorithms. Other defects—such as poor detail filtering—are not easily prevented and would require that we embed extensive medical knowledge within the tool. This work would be unrealistic and is unnecessary—the progress-note generation tool should empower the physician by simplifying progress-note creation, not by making her knowledge superfluous. We want the physician to participate interactively in determining the content and composition of the progress note.

Within the progress note as a whole, we use a simple organization that makes the note easy to read and that conveys important points quickly. Organizing the note into sections by organ system, and presenting organ systems in a head-to-toe ordering (as physicians do in their hand-written notes), allows quick location of specific information and comparison of individual symptoms across multiple progress notes.

Within an organ-system subsection, we impose a second level of structure. Because we present positive findings first in chronological order, the reader can quickly determine the patient's symptoms and also can see how they have evolved over time. By presenting the pertinent negative findings next, we mimic the rhetoric of typical hand-written progress notes. By organizing the entire progress note with this structure, we make specific information easy to find. This structure is apparent in the progress note that is shown in Figure 1.

3.3 Structure of Findings

As mentioned in Section 3.2, each finding in the controlled vocabulary has fields used by the text generator to specify how related findings should be combined. In Table 1, we present a prototypical entry for the finding "icteric sclera." Each finding in the vocabulary has eight fields: three values to control clause combination (as discussed in Section 3.2), two names used by the sentence generator to generate create text, two entries of SNOMED III codes, and one unique identifier.

We chose the finding "icteric sclera" because it illustrates how we use the two names of a finding to generate phrases. If this finding were positive, IVORY could generate the phrase "mildly icteric sclera," using the positive name. If the finding were negative, using the same name for the finding would generate the phrase "no icteric sclera." Although this sentence conveys the proper idea, allowing the use of a different name for negative findings gives us more flexibility in our text-generation scheme. By using the negative name for the finding, we can generate the phrase "anicteric sclera," which is a phrase that many clinicians prefer.

Another example of this feature is associated with the finding "appetite change." We have encoded the negative name of this finding as "appetite change" and the positive name as "appetite" with a required modifier of "increased" or "decreased." By using these names, we can generate the phrases "decreased appetite" or "increased appetite," which we find more natural than the phrases "decreased appetite change" or "increased appetite change." Similarly, we can generate the phrase "no appetite change" if the finding is negative.

The two SNOMED III code entries are used to generate canonical representations of clinical findings for use by real-time decision-support systems or by other applications needing encoded clinical data. These canonical representations employ the conceptual-graph formalisms that we have described previously [11].

3.4 Structure of Modifiers

In addition to the findings in IVORY's structured vocabulary, we have a set of modifiers that can combine with findings in predefined ways. Ivory uses several independent axes of modifiers to describe a finding further (e.g., onset, severity, trend, frequency, and aggravating factors). Modifiers are stored in tables with unique identifiers, with a textual descriptor of the modifier for use in text generation, and with a canonical representation of SNOMED III codes. Table 2 shows entries for the modifiers "mild," "moderate," and "severe."

Table 1. Vocabulary entry for "icteric sclera."

Field Name	Field Value
Finding ID	7300
Name positive	icteric sclera
SNOMED•CG pos	[M-57610: #]->(G-C006)-> [T-AA110]†
Name negative	anicteric sclera
SNOMED•CG neg	(G-0009)->[M-57610: #]-> (G-C006)->[T-AA110]§
Chain positive	yes
Chain negative	no
Generation template	5

† Translation of SNOMED codes: [icterus: #]->(in)->[sclera]

§ (not)->[icterus: #]->(in)->[sclera]

Table 2. Modifier table entries for "mild," "moderate," and "severe."

Mod ID	Modifier	SNOMED•CG Concept
34	mild	[G-A001]
35	moderate	[G-A002]
36	severe	[G-A003]

Table 3. Constraint-table entries that define possible relations of "icteric sclera" to modifiers in Table 2.

Finding ID	Modifier ID	SNOMED•CG Relation
7300	34	->(G-C230)->‡
7300	35	->(G-C230)->
7300	36	->(G-C230)->

‡ ->(has severity)->. Combination of "icteric sclera" with the modifier "moderate" using this relation will create the following conceptual graph:

[M-57610:#]-
->(G-C230)->[GA002]
->(G-C006)->[T-AA110]

Each finding has constraints on the possible modifiers with which it can combine. We have implemented constraints by using a relational table that identifies all legal combinations. In addition, this table defines the relationship between the finding and the modifier, again using the conceptual-graph formalisms applied to SNOMED III codes. The entries that allow relations of "icteric sclera" and the modifiers "mild," "moderate," and "severe" are presented in Table 3.

4. USER PARTICIPATION IN DESIGN AND TESTING

Section 3. has highlighted technical aspects of our design. In this section, we describe our efforts to involve clinical personnel in the design process. This involvement is essential for creation of systems that integrate smoothly into the clinical environment.

Initially, we developed IVORY as a standalone Macintosh application that explored different methods for entering data and for generating prose for progress notes. Shortly after we completed this application, IVORY was incorporated into a decision-support system called T-HELPER [12], which provides clinicians with real-time decision support for treatment of HIV-positive patients. Incorporating IVORY into T-HELPER required our group to explore issues relating to clinician participation in the development of an adequate vocabulary and in the modification of IVORY's interface to meet clinicians' needs and expectations.

Experimental use of IVORY by physicians in our clinic began in December 1992. Prior to the system's clinical deployment, we collected 30 hours of audio- and videotape documenting physician behavior during our development and testing activities. Since deployment, we have collected approximately 20 hours of observational data about the system's clinical use. This video, audio, and observational record of design and testing by clinicians has served as an important resource for us as we redesign IVORY and T-HELPER. We discuss a specific example of our design process with clinicians in Sections 4.1 and 4.2.

4.1 Paper, Pencil, and Practice

JS, a post-doctoral fellow in infectious diseases who worked with our project, has done extensive work to help us refine the IVORY vocabulary, beginning with a paper listing of an initial lexicon described previously [13]. This vocabulary contained subjective and objective findings, grouped by organ systems. We asked JS to list the common medical problems of HIV-positive patients and the findings that IVORY's vocabulary requires to create progress notes for those problems. She then examined a paper listing of our starting vocabulary and indicated which existing findings were needed and which new findings should be added.

This examination required her to reflect on the clinical encounter itself, on past experiences with specific patients, and on her infectious-disease training. She had to imagine the encounter, the elicitation of the patient's history, and the physical examination. She considered findings that suggested a particular problem, but also findings that differentiated that problem from other problems.

Next, JS began to work with other clinician progress notes to acquire other viewpoints about what findings IVORY's vocabulary might contain. Based on her recommendations, we made numerous changes to the vocabulary to make it more acceptable to other clinicians. These changes involved either the addition of new findings or the addition of findings synonymous with existing findings. For example, one clinician did not like the finding "reddened skin" and wanted "erythema" added to IVORY's vocabulary.

4.2 Experimental Use in the Clinic:

We have observed the clinical use of IVORY by three physicians, and our limited data suggest that progress notes generated by IVORY are acceptable. These clinicians agreed to use IVORY-generated progress notes in their patient charts. Annotations to these notes usually are phrases not captured by IVORY's vocabulary. These annotations are usually short—no more than 2 or 3 phrases for a one-half-page progress note. When one of the physicians creates a progress note with IVORY for a specific patient, he frequently suggests new findings to be added to the system. We anticipate that the number of these additions will decrease with time.

5. DISCUSSION

As a progress-note-generating tool, IVORY combines computer-based text generation with provider entry of primary data. This interactive text-generation tool differs from traditional text-generation methods, where text is generated solely from a set of input data. IVORY incorporates a simple text-generation scheme to make progress-note creation more efficient, but relies on physician participation to ensure fluency and appropriateness of output. Specifically, IVORY relies on the physician to provide appropriate detail filtering and to prevent unnecessary repetitions.

Integration of IVORY into the T-HELPER system has required extensive work. Physician participation in the design process for IVORY required our collaborators to reflect on their present practice, which is not a simple process. We had to educate them about our design and goals and to structure our interactions to enable them to reflect on their practice. Designers must be able to dynamically reconfigure strategies for interaction with user participants. Our vocabulary development would have been easier had we been able to anticipate the types of tools and interactions necessary for vocabulary change and validation. Because successful anticipation of interaction strategies comes from analysis of empirical data, it must be part of the early development process.

Our work with IVORY and T-HELPER has resulted in the creation of a functional medical-record system that allows physician data-entry and the possibility of real-time decision support. Early clinical deployment of this system provides us the opportunity to study questions regarding user acceptance. In particular, we are interested in assessing how well a system designed with the participation of an individual practitioner can generalize to other practitioners. Thus, we will study the adequacy of the vocabulary, the acceptance of computer-generated progress notes, and their respective evolution over time.

Acknowledgments

This work was supported in part by the National Library of Medicine under grants LM05305 and LM07033, by the Agency for Health Care Policy and Research under grant HS06330, and by the National Science Foundation under grant IRI-9022293. We thank Jeanette Sison for participating in our project. We also thank Lynne Hollander and Lyn Dupré for their editorial assistance.

REFERENCES

[1] Weed LL. *Medical Records, Medical Education, and Patient Care*. Cleveland, OH: Case Western Reserve, 1969.

[2] Fischer PJ, Stratmann WC, Lundsgarrde HP, and Steele DJ. User reaction to PROMIS: Issues related to acceptability of medical innovations. In *Annu Symp Comput Appl Med Care,* pp. 1722–1730, Washington, DC, 1980.

[3] Winograd T and Flores F. *Understanding Computers and Cognition.* Norwad, NJ: Ablex Publishing Corporation, 1986.

[4] Norman DA, and Draper SW. *User Centered System Design: New Perspectives on Human-Computer Interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates, 1986.

[5] Safran C, Rury C, Rind DM, and Taylot WC. A computer-based outpatient medical record for a teaching hospital. *MD Computing* 8(5):291–298, 1991.

[6] Naeymi-Rad F, Almeida F, and Trace D. IMR-Entry (Intelligent medical record-entry). In *Annu Symp Comput Appl Med Care,* pp. 783-784. Baltimore, MD, 1992.

[7] Bell DS, Greenes RA, and Doubilet P. Form-based clinical input from a structured vocabulary: Initial application in ultrasound reporting. In *Annu Symp Comput Appl Med Care,* pp. 789-790. Baltimore MD, 1992.

[8] Rector AL, Nowlan WA, and Kay S. Foundations for an electronic medical record. *Meth Inform Med* 30:179–186, 1991.

[9] Quaak MJ, Westerman RF, Schouten JA, Hasman A, and van Bemmel JH. Appraisal of computerized medical histories: Comparisons between computerized and conventional records. *Comp Biomed Res* 19:551-564, 1986.

[10] Kukich K. Fluency in natural language reports. In McDonald DD and Bolc L. (eds), *Natural Language Generation Systems,* pp. 280–311. New York: Springer-Verlag, 1988.

[11] Campbell KE and Musen MA. Representation of clinical data using SNOMED III and conceptual graphs. In *Annu Symp Comput Appl Med Care,* pp. 354–358. Baltimore MD, 1992.

[12] Musen MA, Carlson RW, Fagan LM, Deresinski SC, and Shortliffe EH. T-HELPER: Automated supported for community-based clinical esearch. In *Annu Symp Comput Appl Med Care:* pp. 719–723. Baltimore MD, 1992.

[13] Campbell KE and Musen MA. Creation of a systematic domain for medical care: The need for a comprehensive patient-description vocabulary. *MEDINFO 92,* pp. 1437–1442. Geneva, Switzerland, 1992.