MMI 406: Decision Support Systems and Health Care

Berner, E. S. (ed.). (2007). *Clinical decision support systems: Theory and practice* (2nd ed. or later). New York, New York: Springer, Health Informatics Series.

Contents

Ch1: Overview of Clinical Decision Support Systems*

Eta S. Berner and Tonya J. La Lande

Clinical decision support systems (CDSS) are computer systems designed to impact clinician decision making about individual patients at the point in time that these decisions are made.

computer-based physician order entry (CPOE) systems, coupled with CDSS, have been proposed as a key element of systems' approaches to improving patient safety

Types of Clinical Decision Support Systems

Recently, sophisticated data mining approaches have been proposed for similar retrospective analyses of both administrative and clinical data6

such retrospective analyses are not usually considered to be CDSS.

CDSS differ among themselves in the *timing* at which they provide support (before, during, or after the clinical decision is made) and how active or passive the support is, that is, whether the CDSS actively provides alerts or passively responds to physician input or patient-specific information.

Finally, CDSS vary in how easy they are for busy clinicians to access.

CDSS also differ in whether the information provided is general or specialty-based

Another categorization scheme for CDSS is whether they are knowledge- based systems, or nonknowledge-based systems that employ machine learning and other statistical pattern recognition approaches

Knowledge-Based Clinical Decision Support Systems

Many of today's knowledge-based CDSS arose out of earlier expert systems research, where the aim was to build a computer program that could simulate human thinking.9,10

The intent of these CDSS was no longer to simulate an expert's decision making, but to assist the clinician in his or her own decision making.

The system was expected to provide information for the user, rather than to come up with "the answer

The user was expected to filter that information and to discard erroneous or useless information.

The user was expected to be active and to interact with the system, rather than just be a passive recipient
of the output

There are three parts to most CDSS.These parts are the knowledge base, the inference or reasoning engine, and a mechanism to communicate with the user.13

- the knowledge base consists of compiled information that is often, but not always, in the form of if–then rules
- The second part of the CDSS is called the inference engine or reasoning mechanism, which contains the formulas for combining the rules or associations in the knowledge base with actual patient data
- Finally, there has to be a communication mechanism, a way of getting the patient data into the system and getting the output of the system to the user who will make the actual decision.

In most of the CDSS incorporated into electronic medical records (EMR) systems, the data are already in electronic form and come from the computer-based patient record, where they were originally entered by the clinician, or may have come from laboratory, pharmacy, or other systems. Output to the clinician may come in the form of a recommendation or alert at the time of order entry, or, if the alert was triggered after the initial order was entered, systems of email and wireless notification have been employed.

CDSS have been developed to assist with a variety of decisions

1. support for laboratory test ordering.
2. Diagnostic decision support systems have been developed to provide a suggested list of potential diagnoses to the users

These systems generally do not generate only a single diagnosis, but usually generate a set of diagnoses based on the available information.

3. Other systems can provide support for medication orders

the knowledge base consists of compiled information that is often, but not always, in the form of if–then rules.

Other types of knowledge bases might include probabilistic associations of signs and symptoms with diagnoses, or known drug–drug or drug–food interactions

The decision support system’s knowledge base contains information about diseases and their signs and symptoms.

The knowledge base might contain values for therapeutic and toxic blood concentrations of the medication and rules on what to do when a toxic level of the medication is reached the knowledge base might include a drug database

knowledge base would contain local protocols or nationally accepted treatment guidelines

Nonknowledge-Based Clinical Decision Support Systems

some of the nonknowledge- based CDSS use a form of artificial intelligence called machine learning, which allows the computer to learn from past experiences and/or to recognize patterns in the clinical data.19 Artificial neural networks and genetic algorithms are two types of nonknowledge-based systems

Artificial Neural Networks

Artificial neural networks (ANN) simulate human thinking and learn from examples.
1

An ANN contains 3 layers, which include the input layer, output layer, and hidden layer.19The input layer is the data receiver and the output layer communicates the results, while the hidden layer processes the incoming data and determines the results.19

the ANN analyzes the patterns in the patient data, to derive the associations between the patient’s signs and symptoms and a diagnosis.

Many of the knowledge based CDSS cover a wide range of diseases

Neural networks often focus on a more narrow range of signs and symptoms,

21 The system will study this information, make guesses for the correct output, compare the guesses to the given results, find patterns that match the input to the correct output, and adjust the weights of the connections between the neurodes accordingly, in order, to produce the correct results.21 This iterative process is known as training the artificial network.21

the data including a variety of signs and symptoms from large numbers of patients who are known to either have or not have a myocardial infarction can be used to train the neural network. Once the network is trained, i.e., once the weighted associations of signs and symptoms with the diagnosis are determined, the system can be used on new cases to determine if the patient has a myocardial infarction.

Advantages include eliminating the need to program IF–THEN rules and eliminating the need for direct input from experts.19 ANNs can also process incomplete data by inferring what the data should be and can improve every time they are used because of their dynamic nature.19 ANNs also do not require a large database to make predictions about outcomes, but the more comprehensive the training data set is, the more accurate the ANN is likely to be.2

disadvantages.The training process involved can be time consuming.19 ANNs follow a statistical pattern recognition approach to derive their formulas for weighting and combining data. The resulting formulas and weights are often not easily interpretable, and the system cannot explain or justify why it uses certain data the way it does, which can make the reliability and accountability of these systems a concern.19

Genetic Algorithms Another nonknowledge-based method used to create CDSS is a genetic algorithm (GA).

“GAs ‘reproduce’ themselves in various recombinations in an effort to find a new recombinant that is better adapted than its predecessors. (page 239).”19 In other words, without any domain-specific knowledge, components of random sets of solutions to a problem are evaluated, the best ones are kept and are then recombined and mutated to form the next set of possible solutions to be evaluated, and this continues until the proper solution is discovered.

many physicians are hesitant to use these CDSS in their practice because the reasoning behind them is not transparent

Effectiveness of Clinical Decision Support Systems

Clinical decision support systems have been shown to improve both patient outcomes, as well as the cost of care.

The systems can minimize errors by alerting the physician to potentially dangerous drug interactions, and the diagnostic programs have also been shown to improve physician diagnoses.33–36 The reminder and alerting programs can potentially minimize problem severity and prevent complications. They can warn of early adverse drug events that have an impact on both cost and quality of care.4,29,37,38,39

the strongest impact on reducing medication errors have been done at institutions with very sophisticated, internally developed systems, and where use of an EMR, CPOE, and CDSS are a routine and accepted part of the work environment

there was little change in health outcomes after the system was implemented. Further examination showed that, although the guideline was triggered appropriately, clinicians did not go past the first page and essentially did not use it.18 Another study found that clinicians did not follow the guideline advice because they did not agree with it.

reminder systems and alerts usually work, but systems that challenge the physicians’ judgment, or require them to change their care plans, are much more difficult to implement

four factors were the main correlates of successful CDSS implementation.
The factors were:
1. providing alerts/reminders automatically as part of the workflow;
2. providing the suggestions at a time and location where the decisions
were being made;
3. providing actionable recommendations; and
4. computerizing the entire process.

Integration into both the culture and the process of care is going to be necessary for these systems to be optimally used.

There are several reasons why implementation of CDSS is challenging. Some of the problems include issues of how the data are entered. Other issues include the development and maintenance of the knowledge base and issues around the vocabulary and user interface. Finally, since these systems may represent a change in the usual way patient care is conducted, there is a question of what will motivate their use, which also relates to how the systems are evaluated.

Implementation Challenges

The first issue concerns data entry, or how the data will actually get into the system. Some systems require the user to query the systems and/or enter patient data manually. Not only is this "double data entry" disruptive to the patient care process, it is also time consuming, and, especially in the ambulatory setting, time is scarce. It is even more time consuming if the system is not mobile and/or requires a lengthy logon. Much of this disruption can be mitigated by integrating the CDSS with the hospital information system and EMR.

A related question is who should enter the data in a stand-alone system or even in the integrated hospital systems

a decision support system or computer-based patient record or some other system with a controlled vocabulary, that they realize either the system cannot understand what they are trying to say or, worse yet, that it uses the same words for totally different concepts or different words for the same concept

Future Uses of Clinical Decision Support Systems

it is likely that increased commercialization will continue.
CDSS for non-clinican users such as patients are likely to grow

There is increasing interest in clinical computing and, as handheld and mobile computing become more widely adopted, better integration into the process of care may be easier54

The issue of the use of information technology in general, and clinical decision support systems in particular, to improve patient safety, has received a great deal of attention recently

It has been suggested that as CDSS and other advanced computer systems become more available, the Hooper case may not only provide legal precedent for liability for failure to use available technology

while the use of CDSS may lower a hospital's risk of medical errors, healthcare systems may incur new risks if the systems either cause harm or are not implemented properly

Guidelines for Selecting and Implementing Clinical Decision Support Systems

Assuring That Users Understand the Limitations

The vendors of these systems have an obligation to learn from the developers, and to inform the clinicians using the CDSS of its strengths and limitations.

Assuring That the Knowledge Is From Reputable Sources

Users of CDSS need to know the source of the knowledge if they purchase a knowledge-based system

Assuring That the System Is Appropriate for the Local Site

Vendors need to alert the client about idiosyncrasies that are either built into the system or need to be added by the user

The survey asked vendors whether they provided a knowledge base/medical logic model to their customers. If they answered yes, they were asked what the knowledge source was, if the knowledge base was updated, how often the knowledge base was updated, and if there was an additional charge for the updates. If they answered no to providing a knowledge base, they were asked if they provided templates for the user to develop rules, if there was an additional charge for these templates, how much effort was involved for the customer to build these rules, and whether they provided mechanisms to obtain/buy rules from somewhere else, and if there was a charge.

that the decision support system provided is really just an expert system shell and that local clinicians need to provide the "knowledge" that determines the rules.

but local clinicians must still review the logic in shared rules to assure that they are appropriate for the local situation.

Using in-house clinicians to determine the rules in the CDSS can assure its applicability to the local environment, but that means extensive development and testing must be done locally to assure the CDSS operates appropriately. Often a considerable amount of physician time is needed.

Assuring That Users Are Properly Trained

the vendor should also inform the client how much technical support and/or clinician training is needed for physicians to use the system appropriately and/or understand the systems' recommendations.

Thus, vendors of CDSS need to be clear about what expertise is assumed in using the system, and those who implement the systems need to assure that only the appropriate users are allowed to respond to the CDSS advice

Will users lose their ability to determine when it is appropriate to override the CDSS? This “de-skilling” concern

Monitoring Proper Utilization of the Installed Clinical Decision Support Systems
They must be calibrated to alert the user often enough to prevent serious errors, but not so frequently that they will be ignored eventually

Assuring the Knowledge Base Is Monitored and Maintained
monitoring would need to be ongoing to ensure the knowledge does not get out of date
Conclusion

we must continue to remember that the role of the computer should be to enhance and support the human who is ultimately responsible for the clinical decisions

---

Ch 4: Design and Implementation Issues

Factors that play a role but are not critical include inadequate computers and peripheral devices, difficulty some people have working with computers, systems not user-friendly, physicians’ high regard for their own capabilities, and fear of computer competition, as well as the limited nature of the programs. In our estimation, the critical impediment to the development of decision programs useful in medicine lies in the impossibility of developing an adequate database and an effective set of decision rules

Another major design issue is the lack of integration into standard information systems,15 the field of medicine, unlike organic chemistry, lacks a general theoretical model. In particular, medical diagnosis is fraught with uncertainty.

five “deficiencies” of expert systems technology

1. Lack of “deep” (causal) knowledge of the domain (i.e., systems do not understand physiology);
2. Lack of robustness and flexibility. Systems, when faced with a problem not contained in their knowledge bases, cannot (1) solve the problem, (2) recognize their inability to solve the problem, nor (3) develop a strategy for doing so;
3. Inability to provide deep explanations;
4. Difficulties in verification;
5. Inability of systems to learn from experience

The inability to reason with specialized data types (e.g., temporal, spatial), is another obvious shortcoming of many CDSS

Technical Design Issues

Adding Structure to Medical Knowledge

The goal of knowledge representation is to provide intelligent systems with information about a specific domain in a form that can be processed efficiently

If we were to decide later that predicting the ultimate effect of a drug on a clinical state is the desired output, temporal and physiologic information must be somehow represented in our knowledge base

Knowledge Representation Formats

Most knowledge representation schemes fall into one of four categories: logic, procedural, graph/network, or structured

database management systems will undoubtedly play a significant role in this arena as more clinical information systems use this format for data storage

Logic-Based Knowledge Representation

Propositional logic was the first representational format widely used for artificial intelligence research

Now MCV appears as a “predicate” which provides information concerning the relationship of the “objects” it acts on (increased and pernicious anemia, as in the first example). In this form, questions can be asked of the type of MCV (*x*, iron deficiency), which may be read as “what is the value of the MCV in iron deficiency anemia?” This new flexibility, the ability to add predicates to a knowledge base and then to use those predicates to answer questions, provided a significant boost to the use of logic as a basis for expert system design.

Logic-based representations are declarative in nature in that they consist of true or false statements, and all questions are resolved through standard logic-inferencing mechanisms

In a logic-based system, the diagnosis of anemia associated with “increased” MCV would be made by looking through all the “MCV” logic predicates and finding those that have “increased” as an object. All matching predicates would then be returned

Procedural Knowledge Representations

Procedural formats, on the other hand, provide more explicit information about how the knowledge base is to be used to answer a question, it is not simply a “look up” of known facts

```
IF MCV is increased
THEN conclude pernicious anemia
IF MCV is decreased
THEN conclude iron deficiency anemia
```

These process statements are provided in the form of rules. Rule-based systems are prototypical procedural representations and have been the dominant format for medical expert systems

Networks

Networks are specialized structures consisting of nodes (representing facts, events, objects, processes, etc.) and arcs which link the nodes

Even more significant is the capacity of networks to capture causal, temporal, and other hard-to-model knowledge quite readily. Decision trees23 and artificial neural networks24 are other types of network representation schemes which have recently come into favor with CDSS designers.

Data Representation

Structural representations emphasize the "packaging" of knowledge into well defined pieces with higher levels of organization

Frames are complex data structures which contain information about the concept being described along with procedural information detailing how the frame may change over time

Database management systems (DBMS) offer another structured format for knowledge representation. There are two types of databases which are frequently found in clinical settings—relational and object-oriented

Relational databases are based on a record structure in which each record has a number of fields

However, a column cannot hold more complex data structures, for example, another record

Object-oriented database management systems (OODBMS) permit greater expressiveness by permitting the storage of data types that cannot be handled by relational, table-based systems

Structured query language (SQL) may be used to "ask" questions of a database, however, SQL does not support the creation of inferences, i.e., the ability to draw inferences from the data.

A major drawback to using a database as a knowledge base is the lack of a specific knowledge processing mechanism for these systems

adding inferencing capabilities is not a trivial task.

Special Data Types

Providing support for medical decisions presents unique problems to system designers because of the size of the problem domain. Adding to the situation is the need to provide knowledge about dynamic states. This requires not only facts about the objects themselves (diseases, tests, drugs, and so on), but also information concerning how these things might change over time.

The need for causal, temporal, and spatial knowledge is a major challenge for system designers.

no widely accepted format for representing the passage of time, three dimensional anatomical relationships, nor physiological information

The effective handling of temporal knowledge has been an important artificial intelligence (AI) research area from the beginning

The passage of time, of which humans have an innate understanding, is not so easily represented in digital format. Basic temporal concepts such as distinguishing between future and past events, time dependency (i.e., did event X occur three minutes, three days or three years before event Y), and concurrency (while X is occurring, Y usually happens) are essential if CDSS are to reason about prognosis, outcomes, toxicities, and so on.

Events are represented as intervals. Intervals may have attached parameter value which can be numerical (primitive) or qualitative (abstract). There are three types of abstracted intervals: state, gradient, and rate.

Default Knowledge

There are a number of important problems in knowledge representation and knowledge base design that are independent of format—inconsistency, degree of expressiveness, and incompleteness are ready examples

The most interesting, and without doubt, one of the most difficult to solve, is that of default or common sense knowledge

Reasoning

The failure of techniques used in the design of limited domain systems to “scale up” to more general systems is a major driving force behind research in medical artificial intelligence and, by extension, CDSS.

The ability to reason from “first principles” and to understand the effects of time on disease processes are considered essential to building robust systems that have more human-like capabilities

issues such as the computational burden of large numbers of calculations in networks, handling conflicting rules in knowledge bases, gracefully handling uncertainty and ignorance, and methodologies for acquiring new knowledge are sufficiently formidable so as to attract the attention of researchers

Rule-Based and Early Bayesian Systems

The most basic inference mechanism utilized in medical diagnostic systems is propositional logic, locality.” If there is a statement of the form “if a, then b” and “a” is known to be true, then we can conclude that “b” is true regardless of whatever else is known to be true.

Locality is very useful in logic systems where all facts are either completely true or completely false

three reasons why most systems based on propositional logic are unworkable for medical diagnosis.

- Laziness, in this instance, describes the reluctance of system designers to do the work necessary to “list a complete set of antecedents or consequents needed to ensure an exceptionless rule, and it is too hard to use the enormous rules that result
- Theoretical ignorance is simply an acknowledgment that there is no theory of medicine to guide modeling of the domain.
- Last, practical ignorance is a statement of the fact that, for any particular patient, even if we knew all the applicable rules, we would rarely have access to all the required information

The developers of MYCIN addressed the issue of uncertainty by pioneering the use of “certainty factors”—numerical estimates of the confidence in a particular fact

Certainty factors can take on values from −1 (indicating certainty that a condition is not true) to 1 (that it is true). Zero indicates that little is known about a particular fact.

IF chest pain is present
THEN conclude MI 0.65 (certainty factor of 0.65)

IF chest pain is present
THEN conclude esophageal reflux 0.4 (certainty factor of 0.4).

In order to arrive at the correct diagnosis, some mechanism must be in place to adjudicate between the two rules or the knowledge base

designers must ensure that the two rules will never conflict

Leeds abdominal pain system was based on simple Bayesian computation.

However, early Bayesian systems had their own problems. The most significant was the number of probability estimates required to make the system workable. In addition, each new piece of evidence required recalculation of all pertinent probability estimates, resulting in a burdensome number of computations.A final requirement of early Bayesian systems was “conditional independence” (an assumption that all relationships between evidence and hypothesis are independent).

Causal Reasoning

Causal reasoning, simply defined, is the use of deep domain knowledge (i.e., pathophysiology, anatomy) to assist in the decision-making process

potential benefits— describing the evolution of diseases over time, reasoning about interactions among diseases, and the ability to understand specific mechanisms

CASNET2 was the first medical expert system based upon causal precepts. Designed to assist in the diagnosis of glaucoma, CASNET's knowledge is represented in the system as a network of pathophysiologic states. A particularly interesting feature of CASNET is the hierarchical organization of its knowledge base. At the lowest level are patient signs, symptoms, and tests. The middle layer consists of pathophysiologic states such as corneal edema and elevated intraocular pressure. The highest knowledge level is composed of disease categories—open angle glaucoma, secondary glaucoma, and so on. Connections between the layers represent direct causal relationships, allowing diseases at the highest level to be viewed as aggregations of patient findings and pathophysiologic states. Reasoning is carried out by navigating a path from findings to disease, testing pathway nodes by calculating a likelihood value for each, then following the highest likelihood pathway.

Causal links are implemented in the knowledge base in the form of "may be caused by" relationships, serving to constrain the number of nodes evaluated during the diagnostic process

The lack of knowledge concerning the actual mechanism for a number of diseases remains a major impediment to the creation of causal systems

inclusion of causal knowledge in these systems.

Another design issue for causal systems is level of detail.

Perhaps the ultimate design issue is that of "understanding."

A final matter is that of temporal representation in causal networks.

Probabilistic Reasoning

maintain huge probability tables (joint probability distributions)

assuring conditional independence of findings.

A solution to both problems was advanced by the findings of a number of researchers in the form of belief networks.14,

A belief network is a directed acyclic graph (the arrows point in one direction and there are no circular paths) consisting of nodes that contain conditional probability data. Nodes may be thought of as "parent" and "child," with parent nodes connected to child nodes by one-way arrows. Conditional probability tables at each node reflect the effect of all the parent nodes on the child node.

one of the problems of early Bayesian systems was conditional independence. This criticism is addressed by network designers via the use of causal relationships when creating networks and by use of a catch-all probability estimate in the form of a "noise parameter." If we say the probability of MI is 0.7 given finding X, and 0.2 given finding Y, and if 1.0 represents certainty, then 0.3 represents the noise for X and 0.8 for Y. Conditional independence values are not exact, but the use of noisiness permits usable systems to be built.3

Decision-Theoretic Reasoning

A somewhat newer approach to medical expert systems design is the use of decision theory in the reasoning process.41,42 Decision theory is based upon the concept of utility—the value to the decision maker of a particular outcome

a pure probabilistic system would offer as its conclusion the diagnosis with the highest probability (for the sake of argument assume that this is reflux). In a decision-theoretic system, the cost to the patient of suggesting reflux when the correct diagnosis is MI would be calculated before offering a final conclusion. Thus utility serves to "remind" the system of the "cost" of an incorrect diagnosis or suggested action. A

A significant problem with decision-theoretic systems is that of determining how the utilities included in a system will be determined

Possibilistic Reasoning

another decision-making dilemma—imprecision in the expression of a finding or factor

membership may be possible to some extent in a number of sets. Thus a 35-year-old woman would have partial membership in the old set (say 0.3) as well as in the young set (0.7).

Accounting for Ignorance

The Dempster–Shafer theory proposes that probability estimates be qualified by using a "belief function" that computes one's belief in a particular proposition

Common Sense Reasoning

Common sense reasoning, at its most basic level, is about making assumptions. This is an indispensable capability that we use constantly. Common sense (default) reasoning allows objects to be grouped into recognizable classes that can be mentally manipulated based on common traits

New facts concerning objects currently represented in the knowledge base must be reconciled with those already present.

Case-Based Reasoning

Case-based knowledge bases have two distinct parts: the case itself and an index that aids efficient context-based retrieval. Case-based systems acquire knowledge by solving problems. Cases are stored knowledge that reflect past experience in solving problems. Each case has three components—

problem/situation description, solution, and outcome. The problem/situation-description describes the past situation or problem that was solved. It includes the goals of the reasoner as the problem was being solved, as well as information about the problem environment. The solution component contains information regarding how the problem was solved.

The result of applying the solution, whether the attempt succeeded or failed, and why, are stored in the outcome component. Access to cases is controlled by an index

The key to solving problems in case-based systems is matching the current problem to past experience

advantages: (1) they are better at solving problems with open-ended, poorly defined concepts; (2) they arrive at solutions faster; (3) they are better at solving problems where no good algorithm is available; and (4) cases may serve as explanations

In a large knowledge base, retrieval efficiency is an important determinant of performance. Therefore, indexing is a key research area.

Knowledge Acquisition

Knowledge Engineering

Knowledge engineering is the process of building a knowledge base. A knowledge engineer is a professional with an understanding of issues in knowledge representation, tool selection, artificial intelligence languages, and software design. A knowledge engineer works with a “domain expert” to obtain the necessary data to build a knowledge base (knowledge acquisition)

Domain experts can be very poor at describing what they do or how they approach a problem. The knowledge engineer often has to learn the domain in order to identify major unifying concepts. Next, the actual problem to be solved must be agreed upon, and, finally, knowledge representation formats and a reasoning mechanism must be chosen. Errors made at any step can result in significant delays and frustrations. Once completed, maintenance becomes a serious problem which can worsen with turnover of the development team.

Managing Knowledge Using Ontologies

more sophisticated decision support systems will require access to not only facts, but also key concepts that underlie the domain

The field dedicated to creating these higher-level conceptual maps is ontological engineering, and the resulting knowledge construct is an ontology

three basic purposes: communication, computational inference and reuse, and knowledge management52, Communication is enhanced by the requirement that all terms and concepts be standardized and clearly defined, providing for a common reference point for all ontology users and builders

In order to function as a viable resource for an organization or any group of users, “ontological commitment” (all parties who use the ontology agree to abide by its concepts, terms, and relationships) is a must

Ontology Organization

Ontologies seek to represent the world (or at least a specific domain) in an organized manner.

Objects, events, and processes that occur in the real world are captured in a standardized manner based upon fundamental concepts taken from the domain. The highest level (most abstract) of concept organization is referred to as an “upper ontology.”20 The upper ontology provides the basis for all concepts that will eventually make-up the final operational ontology.

Concepts are rendered in computable form as categories (templates/ classes) that have specific properties.

Upper ontology categories are abstract entities; as such, they do not refer to a specific person or plant. Rather, the most fundamental properties of each are defined at this level (with special attention given to those properties that separate one category from another). A specific person or plant can then be represented as an object that inherits (“is-a” relationship) properties from all the categories from which it was derived.

Inheritance that occurs via hierarchies may be single (derived from a single upper ontology concept) or multiple (derived from more than one high-level concept).

A second mechanism for enriching the representational capability of ontologies is that of associations. Associations permit links to be formed between classes and define allowable interactions between the two

For example, a “uses for food” association might be defined between the “Humans and Plants: Edible Classes.” Events (birth) and processes (movement) can be represented using this conceptual organization as well.The key is that all those who use the ontology must agree to the exact meanings and proper usage of all concepts, terms, and objects

Example

UMLS Semantic Network

The UMLS is an ongoing project of the National Library of Medicine as part of its efforts to improve access to reference resources.The UMLS acts as a link between a number of disparate vocabularies and coding systems (e.g., 0 ICD-9, CPT, SNOMED) and the Medical Subject Headings (MeSH) coding system. Its major components are the Metathesaurus, which integrates terms and concepts from over 60 code/vocabulary systems, and the Semantic Network, which provides a high-level conceptual framework for categorizing terms found in the Metathesaurus

GALEN

GALEN (Generalized Architecture for Languages, Encyclopedias, and Nomenclatures) is designed to act as a terminology for use in clinical systems58 GALEN is managed by OpenGALEN, a

SNOMED CT

SNOMED CT61 is a terminology based on SNOMED RT62 and Read Codes Version 3.63 SNOMED is a comprehensive clinical terminology encompassing 344,000 concepts and close to 1.3 million semantic relationships.

Ontology Issues in Decision Support

The Frame Problem

Complex decision support activities, such as multistep guidelines with steps that occur over time, have to contend with the problem of changing data.

The "frame of reference" problem,20 keeping track of what should and should not change after some key event, is a major challenge.

There are two problems: keeping track of how the patient has changed and what has remained the same. Humans quite naturally reason out these "what-if" scenarios; computers require large knowledge bases with complex rule sets to mimic this type of reasoning

Lack of Standards

key issue in building any ontology is term selection. Naming things is very important, and sharing an ontology within a domain requires that all users share a common terminology. SNOMED CT appears to be the most promising candidate for health care.

Also, one cannot mistake agreement on terms for agreement on definitions

Capturing Clinical Detail—Codes, Classifications, Nomenclatures, Vocabularies

The capture of detailed, analyzable clinical information in electronic form is a well recognized problem in the field of medical informatics.72,7

Though a number of terminology sets exist, none has been found that satisfies the wide range of data requirements in the broad domain of health care. Adding to the problem is the lack of a standard definition for the various terminology sets classifications as “. . . an ordered system of concepts within a domain, with implicit or explicit ordering principles

The most common classification in use in the United States is the International Classification of Diseases (ICD).

An obvious problem with the ICD system is the lack of clinical detail. Systems designed for monitoring epidemiologic trends lack the granularity needed for clinical care.82,83,84 A decision support system that relied on ICD codes for suggesting interventions would be reduced to making only the most general advice concerning possible actions due to the paucity of specific clinical information about the patient.

The Current Procedural Terminology (CPT),85 published by the American Medical Association since 1966, was designed to help with reimbursement for medical services

A recent addition to the library of available code sets is the Logical Observation Identifiers Names and Codes (LOINC) set.86,87,88 LOINC provides a formal multi-axial code set for laboratory results and clinical observations

LOINC provides a significant amount of clinical detail and is much more readily suited for clinical decision support tasks than either the ICD or CPT.

A nomenclature is defined as a system of words used in a particular discipline. A key distinction between nomenclatures and codes/classifications systems is that the former permits one to form new concepts by combining terms, whereas the latter do not

SNOMED CT is the latest iteration of the SNOMED nomenclature

“Clinical terminology” and “controlled clinical vocabulary” are somewhat more difficult to define

controlled clinical vocabularies are that they are highly granular term sets arranged in well defined hierarchies that are designed to support the capture of all aspects of clinical care

Vocabulary Utilization Issues

three types of clinically relevant terminologies: application, user interfaces, and reference. Reference terminologies are defined by them as “. . . a set of concepts and relationships that provides a common reference point for comparison and aggregation of data about the entire health-care process, recorded by multiple different individuals, systems, or institutions.” Interface terminologies are those that are encountered by the user during data entry.

Data Entry Concerns

entering a complete history and physical examination for a typical primary care encounter in this manner would quickly become quite tedious. Solving the problem seems to require a trade-off between efficiency and expressiveness.

The ability to compose complex concepts from the juxtaposition of two or more atomic concepts, referred to as a “compositionality,” greatly increases the expressive power of any vocabulary but brings with it the problem of how to prevent the creation of nonsense concepts

In practice, it would require that either the applications have extensive knowledge of medical concepts and language or that additional technologies must be used to provide the required functionality.

Terminology servers95,96 aid in the deployment of vocabularies by providing services to application developers such as lexical matching, word completion, terms composition, and other key functions

An additional interface issue is that of local terminologies. In health care there are often many ways to say the same thing. Permitting users the freedom to use familiar terms and phrases is a reasonable system design goal. However, providing this flexibility in the presence of a controlled terminology can be tricky.

There is also the possibility that, by creating local term sets, some level of “semantic drift” will occur between sites using the same reference terminology, defeating much of the value of having a standard reference terminology.

Storage and Retrevial of Encoded Data

Compositionality creates a second design problem for system designers at the database level

When no atomic concept is available, then complex concepts must be created using collections of atomic concepts (postcoordination

Even in situations in which the same codes are used, there is no mandatory order in which they must be sequenced

Unless a central authority dictates all terms for all concepts, it will be quite easy (and very likely) that different organizations will create incompatible postcoordinated terms for the same clinical concept, thereby reducing the portability and pooling of data for analysis—one of the main justifications for having a standard clinical terminology.

Human–Computer Interaction

CDSS must take clinicians’ work habits into account.

Systems must be available at the point of care and should be easy to use if they are to be considered clinical tools.

stand-alone systems which require significant data entry will not be used on any regular basis. Finally, the singleproblem focus of many systems means that they will be needed only on rare occasions, at which time it may not be worth the trouble to locate and use them

Problem Knowledge Couplers (PKCs), as advanced by Weed,99 represent a rather unique approach to the use of diagnostic/therapeutic decision support.

In fact, they are designed so that even nonmedical personnel can enter the patient's data, although the physician must still interpret the output. PKCs consist of a knowledge base of diagnoses, findings, and management options. Each individual coupler addresses a single presenting problem.The couplers permit controlled input of findings and guide the clinician in the process of diagnosis and management.

other problems systems designers need to avoid.

The first is a preoccupation with computer artifacts.

This preoccupation can hamper the quest for the best problem-solving process and techniques for solving the particular problem.

Next, they argue that systems designers may fail to use appropriate models for solving problems and may fail to communicate clearly the design issues to potential users.

The final problem mentioned by Heathfield and Wyatt is that designers sometimes focus on system development and ignore organizational issues.

Organizational attitudes and support play a critical role in the development and implementation of any technology. The multidisciplinary nature of CDSS development makes this process even more vulnerable to problems of changing personnel, funding, administrative buy-in, and shifting organizational goals.

Fortunately, many of the criticisms of CDSS are readily addressed by the growing use of electronic health record systems (EHRS) and computerbased physician order entry (CPOE) systems. EHRS solve many of the CDSS problems related to work flow, data entry, and types of decision support provided.100,101 They provide a standard interface for users and a data model for CDSS designers. Access to laboratory, pharmacy, and other standard data is a key feature of EHRS, and allows CDSS designers to shift the focus from data entry to data access and user interaction. Alerts and reminder systems have proven to be effective for preventive care, error reduction, and patient safety.102,103 However, more sophisticated decision support functions, such as automation of complex guidelines, will require advances in EHRS design However, more sophisticated decision support functions, such as automation of complex guidelines, will require advances in EHRS design. Automated guidelines that contain multiple steps and act over multiple patient encounters must address issues related to the frame problem and must have access to well structured conceptual knowledge about patients and disease states. Thus, a robust ontology and controlled vocabulary are minimum requirements.

if there is no ability to alter the type (warn for only severe interactions) or frequency of advice (only for new medicines, not refills) users may become reluctant to use the system. Similarly, when complex automated guidelines become available, the challenge to system designers will be not only how to provide advice seamlessly, but also how to quietly recede into the background and allow the clinician to proceed with the task at hand

In fact, we may find that the ultimate issue in human–computer interaction is not one of functionality, but sovereignty. That is, once the challenges of workflow, responsiveness, and ease of use are addressed, there will remain the problem of humans who do not wish to take advice from a machine.

Conclusion

One thing is certain: should all the problems associated with the design and implementation of CDSS ultimately be solved, to gain wide acceptance they must provide decision support without violating two of the most fundamental social and intellectual features of the practice of medicine. First, they must not intrude on the sanctity of the patient-physician relationship. Second, they must do nothing to remove or alter the quiet satisfaction derived from knowing that one has made a difference

Ch8 Clinical Decision Support at Intermountain Healthcare

The Electronic Health Record (EHR) is the primary driver for the growing use of computerized decision tools. The growth in use and sophistication of the EHR has provided a backdrop against which clinical decision support systems (CDSS) appear as a logical consequence

Contributors to the science of applying computer systems to clinical practice include the several sites where hospital-based, medical decision support has been implemented and studied.Among the leaders in these efforts have been groups at the Regenstrief Institute in Indianapolis,8 Columbia-Presbyterian Medical Center in New York,9 Beth Israel Hospital in Boston,10 and the HELP System at the LDS Hospital in Salt Lake City.11 Successful efforts to incorporate decision support into order entry systems at the Brigham and Women's Hospital in Boston12 and Vanderbilt University Medical Center in Nashville13 are helping to define the direction that healthcare computing will follow in the future.

Most CDSS in hospitals depend on simple algorithms to inform and remind users of important clinical data or of medical facts, which may change the decisions they have made or will make. Examples of these include decisionsupport tools that critique medication orders, and the system for identifying life-threatening laboratory results

constituents of an environment appropriate for the creation of robust, enterprise CDSS

A key example of these new decision support models is the CDSS associated with Computer-based Physician Order Entry (CPOE). CPOE differs dramatically from the classical decision support environments. These were generally constructed around a vision of the physician's workflow that differed little from the behaviors supported by a wholly paper medical record.

The overall setting for the CDSS examples described here is the HELP Hospital Information System (HIS). This system is a culmination of more than 25 years of development and testing.11 It currently operates on high availability hardware supplied by the HP NonStop Enterprise Division. Software components of the HELP system have also been installed in many of the 20 hospitals operated by Intermountain Healthcare (IHC). At the LDS Hospital, IHC's central, tertiary care facility, the information system communicates with users and developers through approximately 2,000 terminals and more than 200 printers. The system is interfaced with a variety of other computer systems, including a billing system, a laboratory system, a medical records system, a digital radiology system, and a collection of local area networks (LANs) used by a variety of departments for local research and departmental management functions.

The HELP System consists of an integrated clinical database, a framebased medical decision support system, programs to support hospital and departmental clinical and administrative functions, and the software tools needed to maintain and expand these components. The integrated clinical database contains a variety of patient data (Table 8.1) kept online during the patient's stay to allow review by health-care professionals at terminals throughout the hospital. These terminals allow the entry of pertinent clinical data into the HELP system by personnel who are involved in patient care. In addition, automated systems capture clinical information directly from monitors and other instruments in the hospitals' ICUs

The HELP System contains a decision support subsystem based on a modular representation of medical decision logic in frames.14 These modules are used to: (1) define the data used in making the target medical decision; and (2) encode the logic that converts the raw data into the proposed decision

The history of decision support in the HELP System extends more than 25 years into the past.This classic hospital information system includes two types of CDSS systems. The first type focuses on narrowly circumscribed medical conditions. The logic is typically simple and the data requirements modest.

The second type of CDSS is much less common.This type of tool attempts to discriminate among a group of important *diagnostic* entities using raw medical data. Diagnostic systems often attempt the challenging task of managing large degrees of uncertainty using pattern matching algorithms

two elements of medical decision support applications are critical to their success.These are: (1) the mechanism by which the systems acquire the data used in their decision algorithms; and (2) the interface through which they interact with clinicians to report their results.

different models of computerized assistance may be needed for different types of clinical problems

different categorizations of decision support

1. Processes which respond to clinical data by issuing an alert;
2. Programs activated in response to recorded decisions to alter care (typically new orders); these applications work by critiquing the decision and proposing alternative suggestions as appropriate;
3. Applications that respond to a request by the decision maker by suggesting a set of diagnostic or therapeutic maneuvers fitted to the patient's needs;
4. Retrospective quality assurance applications where clinical data are abstracted from patient records and summary decisions about the quality of care are made and fed back to caregivers.

Alerting Systems

Alerting processes are programs that function continuously, monitoring select clinical data as it is stored in the patient's electronic record.They are designed to test specific types of data against predefined criteria. If the data meet the criteria, these systems alert medical personnel. The timing and character of the messages vary with the alerting goals.

A typical example is a subsystem implemented within the HELP System that monitors common laboratory results and detects and alerts for potentially life-threatening abnormalities in the data acquired.This type of application is notable for the simplicity of its decision logic as well as for the magnitude of its potential impact. The HELP System captures results from the clinical laboratory through an interface to a dedicated laboratory information system (LIS).The results are collected and returned to the HELP System for storage in the clinical record as soon as they are collected and validated in the LIS

abnormalities in laboratory results, especially those that are unexpected, may not receive the timely attention they deserve. a Computerized Laboratory Alerting System (CLAS) designed to bring potentially life-threatening conditions to the attention of caregivers.21–24

A flashing yellow light was installed in the division, and whenever an alert was generated for a patient in that division, the light was activated. It continued to flash until the alert was reviewed and acknowledged on a computer terminal. The second approach was less intrusive to the nursing staff. Whenever anyone accessed the program used to review a patient's laboratory results, any unacknowledged alerts for that patient were immediately displayed along with the data that had triggered them.

This type of decision support intervention is becoming increasingly common as hospital information systems evolve.26 In the inpatient environment where the severity of illness is steadily increasing, the possiblility of better alerting has the potential to improve quality of patient care.

A recent alerting application designed to work in the outpatient setting is among the first to take advantage of a new, enterprise CDSS infrastructure. This application automates a part of the Chronic Anticoagulation Clinic's (CAC) anticoagulation protocol.This clinic manages patients that are taking anticoagulation drugs (principally Coumadin) for extended periods of time. The objective is to maintain each patient's International Normalized Ratio (INR) within a range specified for the patient. A key component is arule-based system that monitors coagulation studies for compliance with these goals and presents alerts to the clinical user through a computerized in-box. Alerts for dangerously altered INRs are also sent to the clinic nurse practitioner's pager so that immediate action can be taken

Critiquing Systems

critiquing processes begin functioning when an order for a medical intervention is entered into the information system. Such methods typically respond by evaluating an order and either pointing out disparities between the order and an internal definition of proper care or by proposing an alternative therapeutic approach.

Embedded in the blood-ordering program is a critiquing tool designed to ascertain the reason for every transfusion and to compare the reason against strict criteria. The approach used provides information specific to the type of transfusion planned.

If the guidelines are met, the order is logged and the blood bank and nursing division are informed electronically and via computer printout. If the criteria are not met, the user is presented with a message stating the applicable criteria and relevant patient data. The physician or nurse may optionally decide to place or cancel the order. If the order is made, he or she is required to enter the reasons for the decision to override the system

The criteria used are the result of a consensus effort by the LDS Hospital medical staff. The criteria were developed using primarily published guidelines but with some adaptations for local conditions (altitude of 4,500 feet). The criteria have undergone several modifications based on experience as well as new definitions of standards for these therapies

The program relies heavily on the integrated clinical database in the HELP System. It accesses data from: (1) the admitting department; (2) the clinical laboratory; (3) surgical scheduling; (4) the blood bank; and (5) the orders entered by nurses and physicians.

The process used by the blood-ordering program is different than that used in the alerting application in that it involves a dialogue with the user. As a result, the critique can provide a series of informational responses designed to assure that the user is fully aware of the status of the patient as well as of accepted guidelines governing blood product usage

Suggestion Systems

The third category of computer applications designed to support medical decision making is potentially the most interactive.This group of processes is designed to react to requests (either direct or implied) for assistance. These processes respond by making concrete suggestions concerning which actions should be taken next.

the program conducts an interactive session with the user during which a suggestion concerning a specific therapeutic decision is sought.The system then reviews relevant data, including data that has been requested from the user, and formulates a suggestion for an intervention based on the medical knowledge stored in its knowledge base.

they decided to standardize care by strict adherence to predetermined treatment protocols. At first, they developed a set of paper protocols. As the protocols became more complex, it became clear that they would be difficult to follow manually. Therefore, it was decided to computerize them. The result was a set of computerized rules that were designed to direct, in detail, the management of patients in both the test and control branches of a study of extracorporeal CO2 removal (ECCO2R).31–33 While the rules were designed initially for this research, they were soon made general enough that they could be used in the management of other patients requiring ventilator support.

The protocols were created by a group of physicians, nurses, respiratory therapists, and specialists in medical informatics. The initial study period was to be 18 months. Subsequent development concentrated on first eliminating errors in protocol logic, second on extending the scope of these tools, and finally on reworking behavioral patterns in the intensive care setting so that the protocols could be effectively implemented.

As a consequence, development and study of these protocols has continued. Figure 8.1 summarizes the results of their use in 111 LDS Hospital patients, and compares these results to those of two other groups Massachusetts General Hospital (MGH) and a group in Europe (the European Collaborative Study) interested in the problem of treating ARDS. It is becoming increasingly clear that the standardization of complex ventilator care decisions possible with computers has a pronounced benefit for patients

Diagnostic Decision Support in the Help System

Diagnostic decision support systems (DDSS) differ from the CDSS described above.Typical decision support systems can draw attention to specific data elements and/or derive therapeutic suggestions from these elements

Computerized diagnostic decisions are generally involved with different goals, interfaces, and decision algorithms than the applications previously described

Proven Diagnostic Applications

the goal of an ADE detection system is to determine the existence of a drug reaction from the patient data collected during the routine documentation of patient care

An ADE recognition subsystem has been implemented in the HELP system.35–36 This ADE subsystem continuously monitors patients for the occurrence of an ADE. The system does so by inspecting the patient data entered at the bedside for signs of rash, changes in respiratory rate, heart rate, hearing, mental status, seizure, anaphylaxis, diarrhea, and fever. In addition, data from the clinical lab, the pharmacy, and the medication charting applications are analyzed to determine possible ADEs.

The system evaluates all of the patients in the hospital and generates a daily computer report indicating which patients have possible ADEs. A clinical pharmacist then follows up on these patients and completes the evaluation using a verification program

HW4: An additional effect of this program appears to be a reduction in the number of severe ADEs seen. During the year beginning in January of 1990, 41 ADEs occurred. In this time frame, physicians were notified of verified ADEs only if they were classified as severe or life threatening. In two subsequent periods (the year of 1991 and the year of 1992) early notification of physicians was practiced for all severities of ADE. Numbers of severe ADEs decreased to 12 and 15 during the follow-up time periods (*p* ⍰ ⍰ 0.001).

In an effort to understand the impact of the drug reactions that were the target of this application, the costs of ADEs were examined. In studies that used the computer tools described above, investigators found that length of hospital stay for patients with ADEs was increased by 1.91 days and that costs resulting from the increased stay were $2,262. The increased risk of death among patients experiencing ADEs was 1.88 times.38 Thus, the cost savings and impact on quality of care in reducing ADEs was substantial.

Nosocomial Infections

Another application in use at the LDS Hospital is designed to recognize nosocomial, or hospital acquired infections

The computerized surveillance system used in LDS Hospital relies on data from a variety of sources to diagnose nosocomial infections. Information from the microbiology laboratory, nurse charting, the chemistry laboratory, the admitting office, surgery, pharmacy, radiology, and respiratory therapy are used. Once each day, a report is produced detailing the computer's findings. This report can be used to follow up on the patients for whom there is evidence of nosocomial infection

HW4: For the group of 155 patients, the computer's sensitivity was 90% with a false positive rate of 23%, while the infection control practitioners demonstrated a sensitivity of 76% and a false positive rate of 19%.When the hours required to use each approach were estimated, the computer-based approach was more than twice as efficient as the entirely manual technique.

In an effort to extend the process of managing hospital-acquired infections, an extension to the infection control system was developed. The goal of the enhancement was to predict which patients were likely to contract a nosocomial infection in the hospital in the future.The tool is based on different decision algorithms

Antibiotic Assistant

The "antibiotic assistant" application provides three basic services. First, it assembles relevant data for the physicians so they can determine whether a specific patient is infected and what sorts of interventions might be appropriate. Information such as the most recent temperature, renal function, and allergies are presented. Second, the system suggests a course of therapy appropriate to that patient's condition. Finally, the program allows the clinician to review hospital experience with infections for the past six months and the past five years

The result of this monthly analysis is an assessment of the likelihood of each pathogen for every combination of the patient-related variables. For example, once the analysis is complete, the percentage of hospital-acquired bacteremias due to *Escherichia coli* in male patients age 50 or less who are on the cardiovascular service will be stored in the program's knowledge base. The analytic programs also evaluate susceptibility data to determine which antibiotics are likely to cover the most probable pathogens for each combination of patient variables

This probabilistic knowledge is then filtered through a set of rules created by infectious disease experts.

The resulting knowledge base is used by the antibiotic assistant program to make presumptive diagnoses of infectious organisms and to suggest treatments appropriate to these organisms

Assisting Data Collection

Efforts to direct data collection in the HELP system have concentrated on the patient history.The

Three techniques for collecting the history were explored. The first was a simple branching questionnaire.This approach takes full advantage of the hierarchical relationship between more and less specific questions.

The second technique has been called decision-driven data acquisition (DDA). With this technique, a frame-based, Bayesian expert system analyzes all data available at any point in the patient interview. The individual disease frames determine which additional information is needed to evaluate the likelihood of the particular disease. Each frame proposes one or more questions. From this list, a supervisory program selects a group of five questions, which are then presented to the patient. The system passes through this cycle multiple times until criteria are met indicating that no additional data are needed

A third approach has also been tested. It is similar to the DDA method except that it was adapted for use in a setting where the patient was not present at a computer terminal. The approach begins when a paper questionnaire containing screening questions is presented to a patient. Staff members enter the answers into the computer, and the patient's data are compared to the diagnostic frames. The questions are scored in a filtering process, and then from 0 to 40 additional questions are printed for the patient to answer. After the patient answers these additional questions, the answers are entered into the computer and the process is completed.

Infrastructure for an Enterprise Clinical Diagnostic Support Systems

This infrastructure is comprised of five main modules: *data-drive*, *time-drive*, *rule node*, *dispatch node*, and *configuration manager*.

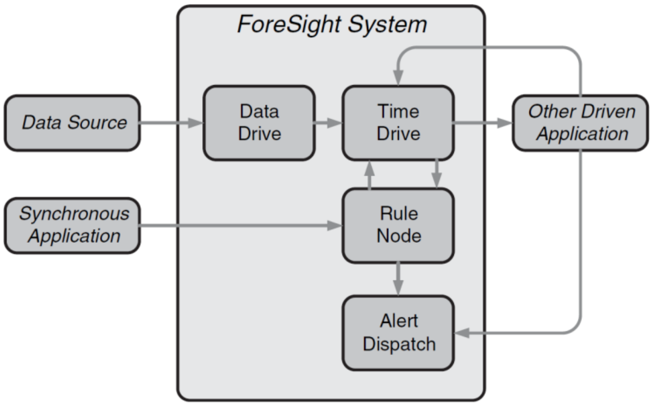


FIGURE 8.3. CDSS infrastructure.

Data-drive is the module responsible for activating the rules whenever any clinical data are stored in the database (new, updated, or logically deleted).Whenever data are stored in the clinical data repository, a copy is forwarded to the *data-drive* module. The data instances are filtered using a configuration file that identifies data for which decision rules exist. Only those that match continue to be processed. They are transformed into a standard data representation and sent to the *time-drive* module.This allows a temporal offset between the receipt of the data and the execution of the rules.

Data that arrive in the *time-drive* module can be held there for a predetermined amount of time before they are delivered to the *rule node*. The objective is to be able to activate the rules at certain times of the day, or after a certain period of time. The holding time can be from seconds to years. In most cases, data that come from the *data-drive* typically have no waiting time, and are immediately delivered to the *rule node*.

The *rule node* was designed to allow wide choice in the methods used for processing the data. It can run different inference engines, allowing different representations of knowledge. We have tested with rule in pure java code as well as logic-executed in third-party inference engines. The *rule node* receives the data and verifies which rules or protocols should be executed. Besides *data-drive* and *time-drive*, the rule node can also be activated synchronously, i.e., directly by an application. If the activating application were an interactive application, it would be able to activate a needed rule set directly and receive in reply the computed decisions. These could then be presented to the waiting user.

If additional data are necessary to execute triggered rules, the data are retrieved from the database and converted into the same common data model.

After the rules are executed, the conclusions that are generated are sent to the *dispatch node*. The *dispatch node* is responsible for saving the conclusions to the EHR and delivering them to a destination specified by the user or the rule developer. Currently, the *dispatch node* can send the rules’ conclusion (e.g., alerts, critiques, suggestion, etc.) to pagers, cell phones, email, and to an electronic “in-box” specific to each user

A *configuration manager* controls the functioning of the all the modules. It is Web-based and allows the system to be managed and configured from any browser. Modules can be configured without having to deactivate the system. The configuration manager also permits the monitoring of salient system functions including error states and performance.

Intermountain Healthcare's Clinical Knowledge Management Infrastructure

new component-based clinical information system

The new definition has grown to embrace systems that access collections of more general advice while still respecting the context provided by a selected patient's data and the applications invoked by the user

The new definition applies to a variety of informational interventions including: (1) tools that reach across the Internet to query commercial and public collections of medical advice, to bring back references appropriate to the medical context surrounding the query; (2) systems that query local collections of problem-specific clinical guidelines to provide contextspecific advice on medical care.This advice seeks to promote decisions that are consistent with IHC's care standards; and (3) collections of orders designed to provide a context-specific starting point for clinicians using IHC's new Computer-based Physician Order Entry (CPOE) system

This view is embodied in a comprehensive clinical knowledge management (CKM) strategy, The focus, instead of being on examples, is on the processes of coordinating development and exploitation of computerized medical knowledge and tools to support these processes.

Infrastructure Overview

The strategy for managing the largely descriptive knowledge represented is based on coordinated initiatives that identify and disseminate clinical best practices to help reduce clinical variability and improve disease management processes and outcomes. These initiatives, known as "*Clinical Programs*," 58 are developed by interdisciplinary teams supported by specialized workgroups. Development teams and workgroups are recruited from practicing clinicians who provide both domain knowledge and local or regional representation. A senior physician, recognized as a system-wide domain expert, is commonly the leader of these teams

Content development priorities are established by guidance councils, taking into account the most prevalent and/or variable diagnostic conditions and clinical work processes, complemented by key patient safety processes

Tools to Manage Clinical Knowledge

A complete software infrastructure to support the clinical knowledge management strategy just described has also been developed. The software infrastructure aims at supporting distributed and collaborative processes for authoring, reviewing, and deployment of knowledge content. During the authoring and review phases, all the knowledge content is stored and organized by a *knowledge repository* (KR).

The KR is the cornerstone of the clinical knowledge management software infrastructure. The KR has been implemented using a flexible database model and can be used to store multiple categories of knowledge content, ranging from unstructured narrative text to well structured documents and executable logic modules. Each KR record is considered a *knowledge document* that is preferably represented in XML, but many of the most common *multipurpose internet mail extensions* (MIME) formats are also supported.

Every knowledge document is associated with a *header* XML document that is used to store detailed document *metadata*. The *header* is used to implement the KR's version control mechanism, providing a detailed record of all the changes and enhancements made to any given knowledge document. In terms of searching and retrieving knowledge documents from the KR, a set of specialized services has been created, leveraging existing XML document transformation and presentation standards.60 The KR currently provides services to find, retrieve, and/or manipulate the knowledge documents according to the needs of various client applications

Authoring and review processes for KR documents are supported by two web-based applications: the *Knowledge Authoring Tool*" (KAT), and the *Knowledge Review Online* (KRO). KAT is an authoring environment that allows clinical experts to create knowledge documents using XML as the underlying representation formalism.61,62The authoring environment generates XML instances using data entry templates created from document models expressed in XML Schema.62 The templates are used to guide and enforce the underlying structure of each knowledge document, implementing a variety of data types that can be used to create simple narrative documents, as well as richly tagged structured documents

The main function of KRO is to support an open and distributed review process, where practicing clinicians, i.e., end-users of the knowledge documents, have the opportunity to provide direct feedback to the document authors. The implementation of KRO exposes all the KR knowledge documents to nearly all IHC clinicians through IHC's intranet. Whenever a review is submitted, the author is promptly notified by e-mail. Reviews are also stored in the KR and can be accessed by any other KRO user. Also through KRO, clinicians can subscribe to e-mail alerts that keep them informed about updates and modifications to the documents they have selected. The functions available in KRO are designed to be exposed as simple Web services, enabling users to submit a review or to subscribe to an e-mail alert from within the clinical applications that they routinely use to take care of patients (CPOE, Results Review, etc.).

Application of the Clinical Knowledge Management Infrastructure to Computer-based Physician Order Entry

The CPOE implementation strategy is based on context specific *order sets* as a key factor to encourage physicians' acceptance of the new system. The development of these order sets utilizes the CKM infrastructure described above, with the underlying assumption that order sets are, in fact, intervention tools to promote the implementation of clinical care processes that embody best practices and evidence-based guidelines and protocols

Currently, the editorial process for the creation and maintenance of order sets is initiated and controlled exclusively by the lead author. Development teams or workgroups are responsible for nominating the lead authors. Using KAT, the author can create an order set by simply filling the template that has been designed specifically for order sets.61 Once the authoring phase is completed, the author can publish the order set, so others can review its content and analyze its appropriateness

As indicated above, the review phase is supported by KRO.Within KRO, every comment and suggestion regarding an order set is instantaneously made available to the author and to the other reviewers. If suggestions made by reviewers require modifications to the order set, the author can make those modifications using KAT and promptly publish a new version of the order set. The authoring and review cycle can be repeated several times, until the content of the order set is considered adequate for clinical use. The approval for clinical use results from the consensus of the group that nominated the lead author. Once the order set is approved, the author is responsible for activating it. The activation is obtained by just changing the status of the order set to "active." At this point, the order set is automatically made available to the CPOE system.

Once order sets are made available to the CPOE system, clinicians begin to use them during the ordering process. In reality, the activation of a brand new order set for clinical use marks the beginning of a secondary review cycle, where authors start receiving feedback from the actual users of the order sets. During this secondary review cycle, the authors are again responsible for analyzing and adopting, or not, the modifications suggested by the users

HW4 The solution implemented by IHC is based on a collaborative knowledge management approach, where knowledge experts retain the authority to create and modify most of the knowledge content necessary for the CPOE system.

Ch9: Clinical Decision Support Within the Regenstrief Medical Record System

Architecture of the Regenstrief Medical Record System

In this section, we offer a brief overview of the RMRS architecture, including the knowledge base, inference engines, and user interfaces used for decision support.

Knowledgebase

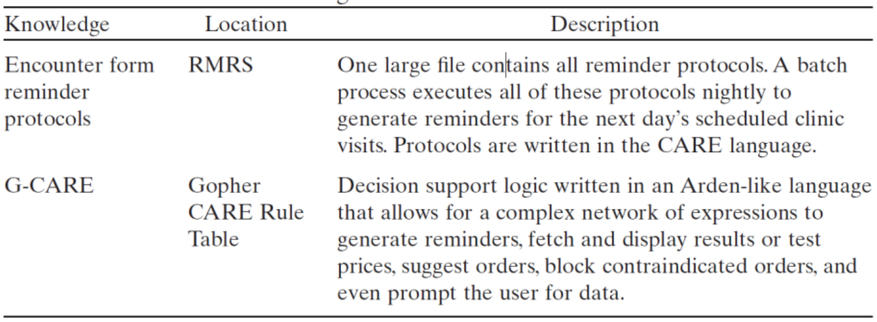
The knowledge within RMRS is spread throughout multiple, integrated resources. The electronic medical record (EMR) exists on a central system that is tied to all of the ancillary systems and data sources.39 The PC-based Medical Gopher order-entry system runs on a local area network (LAN) connected to, yet distinct from, the central repository

decision-support resources are stored on the Gopher's LAN for performance purposes (see Table 9.1).

Inference Engines

There are two main "engines" running within the RMRS to deliver decision support (see Table 9.2).

TABLE 9.2. RMRS inference engines.



Knowledge	Location	Description
Encounter form reminder protocols	RMRS	One large file contains all reminder protocols. A batch process executes all of these protocols nightly to generate reminders for the next day's scheduled clinic visits. Protocols are written in the CARE language.
G-CARE	Gopher CARE Rule Table	Decision support logic written in an Arden-like language that allows for a complex network of expressions to generate reminders, fetch and display results or test prices, suggest orders, block contraindicated orders, and even prompt the user for data.

CARE rules running on a server generate reminders for visit notes,14 while a PC-based extension, G-CARE, drives the immediate response of suggested orders, corollary orders, blocking orders, and dynamic menus during the ordering process.23The notification of drug-drug interactions and patient allergies is performed by specialized codes within the CPOE system.

Communicating with the User

Users receive decision support from the RMRS in numerous ways (see Table 9.3). Generally speaking, providers receive decision support and interact with the system through two primary methods: reports (e.g., encounter forms) and the CPOE system. The RMRS prints out diagnosis lists, patient-based reminders, and dynamic data entry prompts on the encounter form for each visit. It also prints out a single-page clinical abstract summarizing recent lab results, visits, and medications.40 In both inpatient and outpatient settings, all physicians interact with our CPOE system when writing orders or notes. All clinics have workstations in the doctor charting room, however, several have computers in the exam room as well. Reminders produced by the Medical Gopher's inference engine can be presented to an ordering provider in several formats:

1. Textual information, for example, an informational message relevant to the current order.
2. Annotated orders that can be accepted or rejected with a keystroke or mouse click.
3. A request for information, such as the patient's body weight or height.
4. An insertion of patient-specific values or a calculation result into a test display or menu item.

Timing is an important challenge in the successful use of reminders in decision support.Therefore, we have designed the CPOE system to trigger reminders in various contexts:

1. Upon selecting an orderable item: these reminders are usually designed to either discourage or redirect the provider to an alternative, and less expensive or safer order.We call these “blocking rules.”
2. Following the completion of an order: these are suggestions for actions that might be required *because* of the new order, for example, in response to an order for intravenous gentamicin, to suggest that gentamicin levels be followed.
3. After entering a problem into the patient’s problem list or declaring the reason for a visit: medical problems can trigger requests for more information or trigger problem-specific suggestions

Lessons from More Than a Quarter Century of Experience with Computerized Decision Support

• Computerized reminders . . .
• . . . can change clinical behavior;11
• . . . can reduce errors and improve adherence to practice guidelines;3,30
• . . . do not necessarily provoke providers to review the associated literature;12
• . . . have a strong and persistent effect on patient care;14
• . . . can promote preventive medicine in both the outpatient14 and inpatient29 settings;
• . . . have a greater effect than delayed feedback for enhancing preventive care;16 and
• . . . can increase discussion and completion of advance directives.31
• Presenting prior test results can reduce unnecessary testing.24
• Offering providers predictions of abnormal results can reduce testing.26
• Displaying the charges for diagnostic tests significantly reduces the number and cost of tests ordered, especially for patients with scheduled visits.25 This effect does not persist if charges are no longer displayed.
• Reminders for flu shots can generate better patient outcomes.20

• Requiring physicians to respond to computer-generated reminders improves their compliance with preventive care protocols;18 however, promoting preventive care through computerized reminders presents further challenges in the inpatient setting.28
• CPOE and CDSS can significantly reduce patient charges and hospital costs.27
• CPOE-based CDSS can be attained with little or no time added to the patient care process.4

Experience-Based Lessons
• Start with the assumption “the user is always right” because computer systems often lack fine details and reminder rules cannot anticipate every situation.
• Users should be able to override nearly every decision.

• Workflow is paramount!
• Keep it simple.
• Workflow is one of the most critical aspects of delivering excellent, efficient patient care. Decision support often introduces new steps (whether it is a new piece of paper to be reviewed or an alert within CPOE that must be navigated). Implementers of decision support must be cognizant of the impact on workflow

Whenever possible, favor provider-oriented workflow.
• Avoid punishing the user with additional obstacles when simple rewording or changing a default value will do

- When possible, allow for free text

- Response-times should be sub-second, i.e., “blazingly fast.”

- Don’t overwhelm the users.

- Never underestimate the power of user feedback—seek it out!

- Up-time is critical.
- To effectively incorporate decision support into workflow, providers must be able to depend on the system.

Ch2: Mathematical Foundations of Decision Support Systems

What we usually mean by a clinical decision support system (CDSS) is a program that supports a reasoning task carried out behind the scenes and based on clinical data

These programs employ numerical and logical techniques to convert clinical input into the kind of information that a physician might use in performing a diagnostic reasoning task

FINDINGS is a subset of JONES-CRITERIA-MAJOR, or, in set terminology: FINDINGS ⊆□ JONES-CRITERIA-MAJOR

The *cardinality* or *size* of a set is simply the number of elements in the set.

|FINDINGS| □ □ 3

intersection of this set and JONES-CRITERIA-MAJOR is written: CLINICAL-FINDINGS ∩□ JONES-CRITERIA-MAJOR

The union of two sets is the set of all elements that belong to either set.

the set resulting from the union of our patient's findings and
the Jones major criteria is written:

CLINICAL-FINDINGS ∪□ JONES-CRITERIA-MAJOR

A cover of a set is a set of subsets in which each element of the covered set appears at least once as a member of one of the sets in the cover set.

Boolean Logic

A syllogism is a form of deductive reasoning consisting of a major premise, a minor premise, and a conclusion. The premises are combined, using rules of predicate logic, into a conclusion

Major Premise: Low-CO2 ⇒□ OVERVENTILATED
Minor Premise: Over-ventilated ⇒□ HIGH-RATE-ALARM
Conclusion: Low-CO2 ⇒□ HIGH-RATE-ALARM

The syllogism above is an example of rule chaining, where two rules are chained together to form a new conclusion. Specifically, the simple system outlined above is a *forward-chaining deduction system*, because the system starts with *if* statements and moves to a *then* statement

A *backward- chaining deduction system* does this—it starts with the "then" end of a set of rules and works backwards to answer questions based on its rule set. In the flu shot example, a backward-chaining system would start with the "Does this patient need a flu shot" question and immediately learn thatthe diagnosis of asthma would cause this rule to be satisfied. The system might then ask the user or query a clinical database about the presence of this diagnosis.

it would be helpful if the response from the system were something like “the high rate alarm should probably be sounded.” Such a system would then need to be able to handle probabilities, as well as certainties, which most CDSS do. MYCIN, for example, reports its conclusions in terms of their likelihood.

Probability

If we then want to know what the probability is, of finding a patient in our specified population with both diseases, we simply multiply the two probabilities (0.6 and 0.3) to get 0.18, or 18%.

If we wanted to know how many patients in the above example had diabetes *or* hypertension (remember: this would also include those with both diseases in the usual mathematical sense of *or*), we would compute: Pr(diabetes OR hypertension)   Pr(diabetes)   Pr(hypertension) – Pr(diabetes AND hypertension).

A conditional probability is the probability of an event (or the probability of the truth of a statement) *given the occurrence of another event* (or the truth of another statement).

Bayes’ Rule

$$\Pr(\text{hepatitis} \mid \text{jaundice}) = \frac{\Pr(\text{hepatitis}) \times \Pr(\text{jaundice} \mid \text{hepatitis})}{\Pr(\text{jaundice})}$$

Pr(A AND B)   Pr(A)   Pr(B | A)

Pr(B AND A)   Pr(B)   Pr(A | B)

$$\Pr(\mathrm{A} \mid \mathrm{B}) = \frac{\Pr(\mathrm{A}) \times \Pr(\mathrm{B} \mid \mathrm{A})}{\Pr(\mathrm{B})}$$

By using numerical estimates of the probability of diseases, findings, and conditional probabilities, Bayes’ rule can help make medical decisions.

Only after making the simplifying assumption that most disease findings are independent of one another8 do many CDSS use Bayesian approaches. One such system, Iliad,9 profitably employed this assumption.

Informal Logic

Even if we create a reasoning system that follows all the rules of logic and probability, it would be difficult to come up with all the numbers that must be assigned to each event in even a small clinical database. Many successful CDSS have circumvented this difficulty by employing informal rules of logic to accomplish the reasoning task, without creating an intractable datagathering task.

In the early development of one of the most famous CDSS, MYCIN,4,10 the creators of the system developed their own logic system (heuristic) that made intuitive sense. This system employed “certainty factors” which ranged from −1 (false) to ⍰ 1 (true). A certainty factor of 0ndicated no belief in either direction in the statement’s veracity. In combining several statements with the AND function into a single combined statement in MYCIN, one simply takes the minimum certainty factor of all the statements as the certainty factor of the combined statement

The General Model of Knowledge-Based Decision Support Systems

There are similarities between physician and CDSS reasoning, although a CDSS might arrive at a similar conclusion to a physician without employing the same model of reasoning. Physicians do use some probabilistic information when they make decisions

However, physicians use this information in informal ways; in other words, they do not actually use numbers in formulas to make diagnostic decisions.l3–15 Another feature of real-life clinical decision making is that physicians do not require complete information to make a decision. Most doctors are comfortable making decisions based on incomplete or contradictory information

CDSS rely on well defined numerical techniques to do their reasoning, and they do require sufficient information to complete their formulae

CDSS are not well suited to situations in which hard data are unknown.

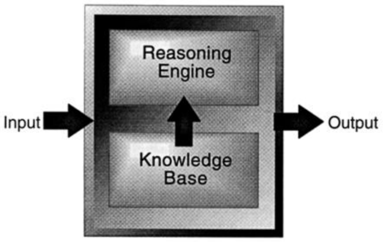


FIGURE 2.1. A general model of a clinical diagnostic decision support system.

The user supplies input appropriate to the system (i.e., terms from the system’s controlled vocabulary to represent clinical data), and the system supplies output (e.g., a differential diagnosis).The reasoning engine applies formal or informal rules of logic to the input and often relies on additional facts encoded in the system’s knowledge base. The knowledge base is the compilation of the relationships between all of the diseases in the system and their associated manifestations (e.g., signs, symptoms, laboratory and radiographic tests). Maintaining the knowledge base in such systems is the most significant bottleneck in the maintenance of such systems, since the knowledge base needs to be expanded and updated as medical knowledge grows

Input

The manner in which clinical information is entered into the CDSS (user interface) varies from system to system, but most diagnostic systems require the user to select terms from its specialized, controlled vocabulary.

Comprehension of natural language has been an elusive goal in the development of CDSS

In a CDSS, it is common for the user's input to be restricted to a finite set of terms and modifiers. How well the system works in a given clinical situation may depend on how well the system's vocabulary matches the terms the clinician uses.

usually there are pertinent temporal factors related to symptoms that are difficult to express in a controlled vocabulary. For example, "sudden onset, 20 minutes ago, of chest pain radiating to the left arm" has a very different meaning from "five-year history of continuous chest pain radiating to the left arm

While CDSS often include a vocabulary of severity and location modifiers, temporal modifiers are more difficult to build into a system, since minute changes in the timing of onset andduration can make a big difference in the conclusion the system reaches

One solution to the problem of temporal modeling in CDSS is to use an explicit model of time, in which the user is asked to specify intervals and points in time, along with temporal relationships between events (e.g., event A occurred before event B), in order to drive a temporal reasoning process within the CDSS

A simpler approach is to model time implicitly. In implicit time,17 temporal information is built into the data input elements of the CDSS; no special temporal reasoning procedures are required. For example, one input item could be "history of recent exposure to strep." By joining the concept "history of" with the concept of a particular bacterial pathogen, one successfully abstracts the temporal nature of this finding,

Inference Engine

The inference engine is the portion of the CDSS that combines the input and other data according to some logical scheme for output.

One such scheme for an inference engine is the Bayesian network.

A Bayesian network is a way to put Bayes' rule to work by laying out graphically which events influence the likelihood of occurrence of other events.

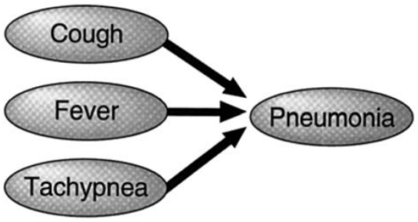


FIGURE 2.2. A Bayesian network for the diagnosis of pneumonia.

Conceptually, this network simply states that the diagnosis of pneumonia is supported by the presence of three symptoms. The strength of association—that is, how strongly pneumonia is suggested by each of the three symptoms—varies with each symptom–disease pairing.

By “activating” all three nodes (cough, fever, and tachypnea) the probability of pneumonia is maximized. Of course, each of these three nodes might be tied to other disease states in the knowledge base (like lung cancer or upper respiratory infection

Production rule systems are another method of programming an inference engineThe rules of predicate logic dictate the functioning of such an engine as it combines statements to form new conclusions

An appealing solution to the problem of constructing inference engines in a clinical setting is to develop a cognitive model of actual clinical reasoning. In other words, one could study the reasoning that a physician uses and attempt to create a computerized version of that cognitive task.

Workers in the field of artificial intelligence, in modeling human cognition, have developed the notion of “frames” or schemes, as a reasonablecognitive model. A frame consists of a set of “slots” into which fit details of a particular kind of information

One important aspect of inference engines is their independence from their knowledge base

Knowledge Base

For CDSS to work, they must possess some form of medical knowledge. Obviously, the method of encoding this knowledge must match the inference engine design

a CDSS based on a Bayesian network must contain probabilities—prior, conditional, and posterior—of diseases and findings.A big obstacle to building such a knowledge base is that many relevant probabilities are not known

Once one creates a knowledge base and populates it with some amount of data, the next task is to create a way to maintain it.

many CDSS become too expensive to maintain. The knowledge-acquisition bottleneck23 has been recognized as a problem in CDSS research.

Output

The output of CDSS is usually in the form of a list of possibilities, ranked in some order of probability

It is the less likely diagnosis that one might fail to consider that interests physicians in CDSS

Nonknowledge-Based Systems

Neural Networks

There are systems that can learn from examples

Neuron bodies ("nodes") are connected to one another by axons and dendrites ("links"). Nodes may be turned on or off, just as a biological neuron can be in an activated or inactivated state. Activation of a node causes activation of a signal on a link. The effect of that signal depends on the weight assigned to that link. In most learning neural networks, some nodes are input nodes and some are output nodes. In the CDSS context, the input nodes would be findings and the output nodes would be possible diseases

By presenting to a neural network many thousands of cases of sore throat (where the outcome is known), the neural network would "learn," for example, that the presence of cough decreases the likelihood of strep, and the height of fever increases this likelihood

The appealing feature of neural networksis the ability of the system to learn over time

A neural network changes its behavior based on previous patterns.

In a domain where the relationship between findings and diseases might change, likeinfectious disease surveillance, this changing behavior can be desirable

Another desirable feature of neural networks is the lack of necessity to understand complex relationships between input variables; the network learns these relationships as it changes the links between its nodes. This is the principal difference between neural networks and Bayesian networks.

With neural networks, the links are established as the network is developed, often on the basis of a learning process, without regard to pathophysiologic facts. A disadvantage of neural networks, however, is that unlike the other systems discussed, the "rules" that the network uses do not follow a particular logic and are not explicitly understandable.

Genetic Algorithms

Genetic algorithms take a similar approach.To use a genetic algorithm, the problem to be solved must have many components (e.g., a complex cancer treatment protocol with multiple drugs, radiation therapy, and so on). By selecting components randomly, a population of possible solutions is created.The fittest of these solutions (the one with the best outcome) is selected, and this subpopulation undergoes rearrangement, producing another generation of solutions. By iteratively extracting the best solutions, an optimal solution can be reached

The main challenge in using genetic algorithms is in creating the criteria by which fitness is defined

Ch3: Data Mining and Clinical Decision Support Systems

Introduction

Data mining is a process of pattern and relationship discovery within large sets of data. The context encompasses several fields, including pattern recognition, statistics, computer science, and database management

the main goal of data mining is to convert data into meaningful information. More specifically, one major primary goal of data mining is to discover new patterns for the users. The discovery of new patterns can serve two purposes: description and prediction. The former focuses on finding patterns and presenting them to users in an interpretable and understandable form. Prediction involves identifying variables or fields in the database and using them to predict future values or behavior of some entities

Data Mining and Statistical Pattern Recognition

Pattern recognition is a field within the area of data miningIt is the science that seeks analytical models with the ability to describe or classify data/measurements. The objective is to infer from a collection of data/ measurements mechanisms to facilitate decision-making processes

One approach to pattern recognition is called statistical pattern recognition.

Statistical pattern recognition implies the use of a statistical approach to the modeling of measurements or data.5 Briefly, each pattern is represented by a set of features or variables related to an object. The goal is to select features that enable the objects to be classified into one or more groups or classes

Data Mining and Clinical Decision Support Systems

A typical decision support system consists of five components: the data management, the model management, the knowledge engine, the user interface, and the user(s).7

One of the major differences between decision support systems employing data mining tools and those that employ rulebased expert systems rests in the knowledge engine. In the decision support systems that utilize rule-based expert systems, the inference engine must be supplied with the facts and the rules associated with them thatare often expressed in sets of “if–then” rules

In this sense, the decision support system requires a vast amount of a prioriknowledge on the part of the decision maker in order to provide the right answers to well formed questions

On the contrary, the decision support systems employing data mining tools do not require a priori knowledge on the part of the decision maker. Instead, the system is designed to find new and unsuspected patterns and relationships in a given set of data; the system then applies this newly discovered knowledge to a new set of data. This is most useful when a priori knowledge is limited or nonexistent

Many successful clinical decision support systems using rule-based expert systems have been developed for very specialized areas in health care

However, such systems can be challenging to maintain due to the fact that they often contain several thousand rules or more. In addition, these “if–then” rule systems have difficulty dealing with uncertainty

Supervised Versus Unsupervised Learning

Supervised Learning

Supervised learning, also called directed data mining, assumes that the user knows ahead of time what the classes are and that there are examples of each class available. (Figure 3.1A) This knowledge is transferred to the system through a process called training. The data set used in this process is called the training sample.The training sample is composed of dependent or target variables, and independent variables or input. The system is adjusted based on the training sample and the error signal (the difference between the desired response and the actual response of the system). In other words, a supervised learning system can be viewed as an operation that attempts to reduce the discrepancy between the expected and observed values as the training process progresses. With enough examples in the training data, the discrepancy will be minimized and the pattern recognition will be more accurate.

The goal of this approach is to establish a relationship or predictive model between the dependent and independent variables. Predictive modeling falls into the category of supervised learning because one variable is designated at the target that will be explained as a function of other variables. Predictive models are often built to predict the future values or behavior of an object or entity

The nature of the target/dependent variabledetermines the type of model: a model is called a classification model if the target variable is discrete; and a regression model if the target variable is continuous.

A Priori Probability

In supervised learning, the frequency distribution, or a priori probability, of the classes of a certain training set (or a sample taken from the general population) may be quite different from that of the general population to which the classifier is intended to be applied. In other words, the training set/sample may not represent the general population

Therefore, it is necessary to adjust the output of a classifier with respect to the new condition to ensure the optimal performance of the classifier.16

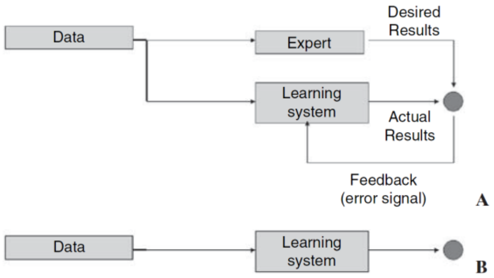


FIGURE 3.1. A, Supervised learning. B, Unsupervised learning.

Unsupervised Learning

In unsupervised or undirected learning, the system is presented with a set of data but no information is available as to how to group the data into more meaningful classes (Figure 3.1B). Based on perceived similarities that the learning system detects within the data set, the system develops classes or clusters until a set of definable patterns begins to emerge. There are no target variables; all variables are treated the same way without the distinction between dependent and independent variables

Classifiers for Supervised Learning

In supervised learning, classification refers to the mapping of data items into one of the predefined classes

one of the critical tasks is to create a classification model, known as a classifier, which will predict the class of some entities or patterns based on the values of the input attributes

Choosing the right classifier is a critical step in the pattern recognition process.

Some of the more widely used and well known techniques that are used in data mining include decision trees, logistic regression, neural networks, and nearest neighbor approach

Decision Trees

The use of decision trees is perhaps the easiest to understand and the most widely used method that falls into the category of supervised learning. Figure 3.2 is the graphical representation of a simple decision tree using two attributes.A typical decision tree system adopts a top-down strategy in searching for a solution. It consists of nodes where predictor attributes are tested. At each node, the algorithm examines all attributes and all values of each attribute with respect to determining the attribute and a value of the attribute that will “best” separate the data into more homogeneous subgroups with respect to the target variable. In other words, each node is a classification question and the branches of the tree are partitions of the data set into different classes. This process repeats itself in a recursive, iterative manner until no further separation of the data is feasible or a single classification can be applied to each member of the derived subgroups. Therefore, the terminal nodes at the end of the branches of the decision tree represent the different classes

Logistic Regression

Logistic regression is used to model data in which the target or dependent variable is binary, i.e., the dependent variable can take the value 1 with a probability of success *p*, or the value 0 with the probability of failure 1 − *p*. The main objective is to develop a regression type model relating the binary variable to the independent variables. As such it is a form of supervised learning. It can also be used to examine the variation in the dependent variable that can be explained by the independent variables, to rank the independent variables based on their relative importance in predicting the target variable, and to determine the interaction effects among independent variables. Rather than predicting the values of the dependent variable, logistic regression estimates the probability that a dependent variable will have a given value

The function relating the probabilities to the independent variables is not a linear function and is represented by the following equation:

$$p(y) = 1/\{1 + e^{(-a-bx)}\}$$

where *p*(*y*) is the probability that *y*, the dependent variable, occurs based on *x*, the value of an attribute/independent variable, *a* is the constant, and *b* is the coefficient of the independent variable.

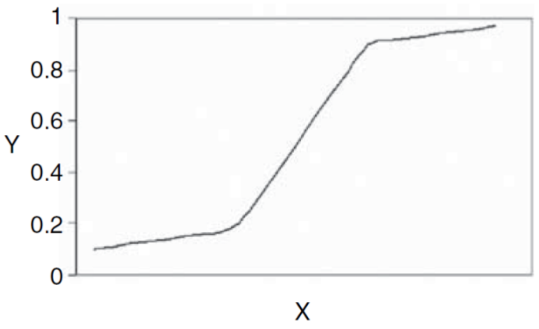


FIGURE 3.3. Logistic regression model.

Neural Networks

The original development of the neural network programs was inspired by the way the brain recognizes patterns.A neural network is composed of a large number of processors known as neurons (after the brain cells that perform a similar function) that have a small amount of local memory and are connected unidirectionally (Figure 3.4). Each neuron can have more than one input and operates only on the inputs it receives via the connections. Like some of the data mining tools, neural networks can be supervised or unsupervised. In supervised neural networks, examples in the form of the training data are provided to the network one at a timeFor each example, the network generates an output that is compared with the actual value as a form of feedback. Once the output of the neural network is the same as the actual value, no further training is required. If the output differs from the actual value, the network adjusts those parameters that contributed to the incorrect output. Once adjustment is made, another example is presented to the network and the whole process is repeated.The process terminates when all parameters are stabilized.The size and representativeness of the training data is obviously very important, since a neural network could work fine on the training set, but not generalize to a broader sample

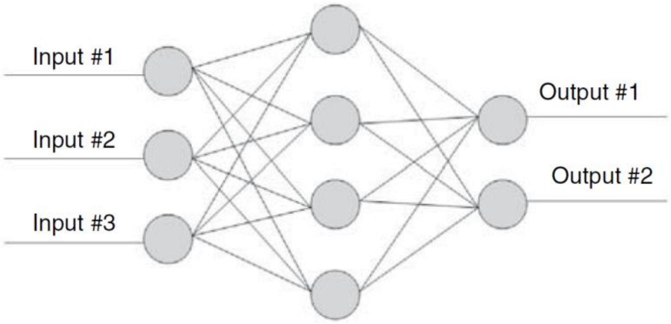


FIGURE 3.4. Neural network.

Nearest Neighbor Classifier

When a system uses the nearest neighbor (NN) classification, each attribute is assigned a dimension to form a multidimensional space.A training set of objects, whose classes are known, are analyzed for each attribute; each object is then plotted within the multidimensional space based on the values of all attributes. New objects, whose classes are yet to be determined, are then classified according to a simple rule; each new object is analyzed for the same set of attributes and is then plotted within the multidimensional space based on the value of each attribute. The new object is assigned to the same class of its closest neighbor based on appropriate metric/measurements. In other words, the NN rule assumes that observations which are the closest together (based on some form of measurement) belong to the same category (Figure 3.5). The NN rule is often used in situations where the user has no knowledge of the distribution of the categories

One extension of this approach is the k-nearest neighbor approach (k- NN). Instead of comparing to a single nearest prototype, one can take into account k-neighboring points when classifying a data point, if the number of preclassified points is large. For each new pattern, the class is assigned by finding the most prominent class among the k-nearest data points in thetraining set. (Figure 3.5) This approach works very well in cases where a class does not form a single coherent group but is a collection of more than one separate group

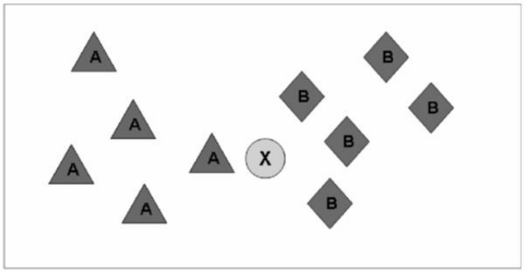


FIGURE 3.5. Nearest neighbor (NN) classifier. There are two classes: A (triangles) and B (diamonds). The circle represents the unknown sample, X. For the NN rule, the nearest neighbor of X comes from class A, so it would be labeled class A. Using the k-NN rule with k = 4, three of the nearest neighbors of sample X come from class B, so it would be labeled as B.

Evaluation of Classifiers

ROC Graphs

We can plot the true positive rate on the Y axis and the false positive rate on the X axis; a receiver operating characteristic (ROC) graph results (Figure 3.6). The true positive rate (also known as sensitivity) is obtained by dividing the number of true positives by the sum of true positives and false negatives. The false positive rate is obtained by dividing the number of false positives divided by the sum of true negatives and false positives; the false positive rate can also be expressed as “1 minus specificity,” where specificity is equal to true negatives divided by the sum of true negatives and false positives.The ROC graph is a two-dimensional graph that depicts the trade-offs between benefits (detecting cancer correctly, or true positive) and costs (false alarm or false positive). Each classifier generates a pair of true positive and false positive rates, which corresponds to a point on the ROC graph.The point (0, 1) represents perfect classification, i.e., 100% true positive rate and 0% false positive rate. One classifier is considered superior to another if it has a higher true positive rate and a lower false positive rate, corresponding to a more “northwest” location relative to the other on the ROC graphIn general, the false alarm rates go up as one attempts to increase the true positive rate. Classifiers with points on the southwest corner of an ROC graph are more “conservative” since they make positive predictions only with strong evidence; therefore there is a low true positive rate, but also few false positive errors. On the other hand, classifiers on thenortheast corner of an ROC graph are more “liberal” since they make positive prediction with weak evidence; therefore they have high true positive rates, but also high false positive rates

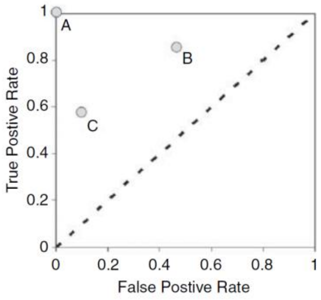


FIGURE 3.6. Receiver operating characteristic (ROC) curve. Point A represents perfect performance. The performance of C is more conservative than B.

Some classifiers, such as neural networks, yield a numeric value which can be in the form of a numeric score or probability that represents the likelihood an object belongs to a certain class.These classifiers can be converted into discrete, binary (yes versus no) classifiers by setting a threshold, i.e., if the output score is above the threshold, the classifier produces a "Yes, else a No". By choosing a different threshold, a different point in the ROC graph is produced. As a result, varying the thresholds will produce a curve in the ROC graph for a particular classifier. Given an ROC curve, one can select the threshold corresponding to a particular point on the ROC that produces the desired binary classifier with the best true positive rate (correctly diagnosed cancer) within the constraints of an acceptable false positive rate (false alarm). This is chosen based on the relative costs of the two types of errors: missing a diagnosis of cancer (type I error) versus creating a false alarm (type II error).

The area under the ROC curve (AUC) provides a single statistic (the CStatistic) for comparing classifiers. It measures the accuracy of the classifiers. Consider the situation in which a classifier attempts to separate patients into two groups; those with disease and those without. One can randomly pick a patient from the disease group and one from the nondisease group and apply the classifier on both. The area under the curve represents the percentage of randomly drawn pairs where the classifier correctly classifies the two patients in the random pair. The value of AUC ranges from 0.5 to 1. A classifier with an AUC of 0.5 would be a poor classifier, roughly equivalent to flipping a coin to decide the class membership. A classifier with an AUC close to 1 results in better classification of entities to classes. For example, in Example 3.6, the resulting trained neural network model yielded a normalized area under the ROC curve of 0.95.

Kolmogorov-Smirnov Test

The Kolmogorov-Smirnov test, or KS test, is used to determine whether the distributions of two samples differ from each other orwhether the distribution of a sample differs from that of the general population. The KS test provides what is called the D-statistic for comparison of classifiers.

Unsupervised Learning

Cluster Analysis

Unsupervised classification refers to situations where the goal is to classify a diverse collection of unlabeled data into different groups based on different features in a data set. Unsupervised classification, also known as cluster analysis or clustering, is a general term to describe methodologies that are designed to find natural groupings or clusters based on measured or perceived similarities among the items in the clusters using a multidimensional data set (

they are based on two popular approaches: hierarchical clustering and nonhierarchical clustering. The former, which is the most frequently used technique, organizes data in a nested sequence of groups that can be displayed in a tree-like structure, or dendrogram.

There are several problems that are associated with clustering. One problem is that data can be grouped into clusters with different shapes and sizes. Another problem is the resolution or granularity, i.e., fine versuscoarse, with which the data are viewed

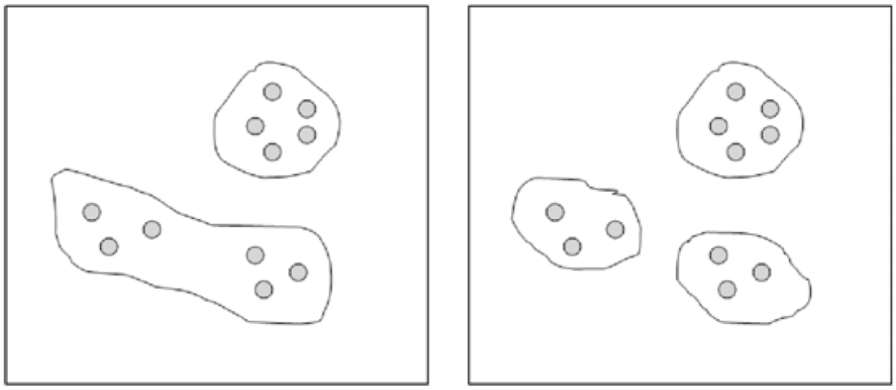


FIGURE 3.7. Cluster analysis. Two clusters of data (left); three clusters (right) using the same set of data.

the selection of an appropriate measure of similarity to define clusters is a major challenge in cluster analysis

Gene Expression Data Analysis

One of the applications of cluster analysis in medicine is the analysis of gene expression

Common research questions often fall under the following categories: class discovery, class prediction, and gene identification. Class prediction refers to the classification of samples based on certain behaviors or properties such as response to therapy, whereas gene identification involves the discovery of genes that are differentially expressed among different disease groups

Class discovery refers to the discovery of previously unknown categories or subtypes based on some similarity measure calculated from the gene expression data. Cluster analysis is often the method of choice in accomplishing this task, because samples are clustered into groups based on the similarity of their gene expressions without utilizing any knowledge of any predefined classification schemes such as known histological tumor classification

Other Techniques

Genetic Algorithms

Based on the same idea of “survival of the fittest,” a genetic algorithm initially tries to solve a given problem with random solutions. These solutions are often referred to as the genomes, or a collection of genes. The gene represents the smallest unit of information for the construction of possible solutions. The next step is to evaluate or quantify the fitness of all the available genomes or solutions based on a fitness function.The latter returns a value of goodness or fitness so that a particular genome or solution may be ranked against all other genomes or solutions. Those solutions with better fit are ranked higher among others and are allowed to “breed.” Once the initial evaluation is completed, the genetic algorithms examine new solutions by letting all the current solutions “evolve” through mutual exchange of “genetic materials” among solutions to improve the genomes and/or mutation (i.e., randomly changing the genetic materials) to “create” new solutions. The new solutions are then evaluated using the same fitness functions to determine which solutions are good and which are not and need to be eliminated. Thus the process repeats itself until an “optimal” solution is attained.

One major advantage is that a genetic algorithm almost always guarantees finding some reasonable solution to problems, particularly those that we have no idea how to solve. Further, the final solution is often superior to the initial collection of possible solutions. Another benefit is that genetic algorithms tend to arrive at a solution much faster than other optimization techniques. Also, the strength of the genetic algorithm does not depend upon complex algorithms but rather on relatively simple concepts

Despite the power of genetic algorithms, however, some parameters, such as the size of the solution population, the rate of mutation and crossover, and the selection methods and criteria, can significantly affect their performance

Biological Computing

the idea of biological computing actually involves the use of living organisms or their components, e.g., DNA strands, to perform computing operations. The benefits include the ability to hold enormous amounts of information, the capability of massive parallel processing, self-assembly, self-healing, selfadaptation, and energy efficiency.

Ch11 Decision Support for Patients

Holly B. Jimison, Paul P. Sher, and Jennifer J.B. Jimison

The new field of consumer health informatics deals with “developing and evaluating methods and applications to integrate consumer needs and preferences into information management systems in clinical practice, education, and research.”1 This technology, both hardware and software, is part of a growing trend toward empowering consumers to take a more active role in their own health care and to provide the necessary information to enhance their decision making

Role of Consumer Health Informatics in Patient Care

Research studies have shown that access to health information can enable patients to be more active participants in the treatment process, leading to better medical outcomes

the process of sharing information enhances the doctor-patient relationship

Involvement in one's medical care also involves the concepts of patient empowerment and self-efficacy.

Empowerment and Self-Efficacy

empowerment can be thought of as the process that enables people to "own" their own lives and have control over their destiny

Studies demonstrate that patients who feel "in control" in a medical situation have better outcomes than those who feel "powerless.

Similarly, self-efficacy is a patient's level of confidence that he or she can perform a specific task or health behavior in the future

Several clinical studies have shown self-efficacy to be the variable most predictive of improvements in patients' functional status.1

The feeling of empowerment and selfefficacy can be enhanced, for instance, by online support groups where patients are able to connect, communicate, and engage in problem solving with others who have similar medical problems

Incorporating Patient Preferences

For patients to be adequately informed to make decisions regarding their medical care, it is important that they obtain information about the quality of life associated with the possible medical outcomes of these decisions.

Information on patient preferences is important for tailoring information to patients and for providing decision support

In addition to differences in preferences for health outcomes, patients differ in the degree to which they choose to be involved in decision making.

The Computer as a Health Information Medium

These studies tend to show that video and slides are educationally more effective than books and audiotapes. Computer approaches have the additional advantages of interactivity, providing feedback in the learning process and the ability to tailor information to the individual patient

Health Information and Decision Support Systems for Patients

General Health References

Oftentimes, content creators and publishers, such as A.D.A.M.38 and Healthwise39 will license their content to several other clients that, in turn, deliver Web material for patients. These clients may be health insurance companies, health portals, such as WebMD40

Web sites that present a full complement of health information will also typically organize information by age

Finally, online medical dictionaries, disease-specific discussion boards, and “ask-an-expert” services are also often found as components of health portal sites

Drug Information

Information about prescriptions is most commonly obtained from content providers and publishers that specialize in just that feature.

Examples of searchable drug databases for patient use are RxList,43 DrugInfoNet,44 and RxMed

Additionally, some of the health portals offer a Web tool that specifically checks for drugdrug interactions for a particular patient’s set of prescriptions

Diagnostic Decision Support

Some of the health portals that offer general reference and drug information also offer interactive tools to assist patients in health assessment, symptom management, and limited diagnostic information

Health risk appraisals usually take the form of a questionnaire with questions on family history and health behaviors. After completion, patients receive a tailored printout with a summary of results that may help them prioritize their health goals.

These types of assessments allow thepatient to know when to pursue diagnostic advice from a health care professional and when to seek treatment

Many of the health portals also offer calculator style tools to help patients manage their health

Occasionally, the health Web sites will offer diagnostic aids for patients. However, there has been some reluctance to offer advice that is overly specific. The usual approach on the health sites that offer symptom-based diagnosis is to assess a symptom or two and then present a list of possible causes, with links to further reading

This goal of this approach is not to provide the patient with a diagnosis or specific recommendation, but to prepare the patient to be an informed participant in making the treatment decision during the next visit to the clinician

The Foundation for Informed Medical Decision Making (FIMDM) has taken the decision assistance even further. They focus on treatment decisions where patient preferences on health outcomes are important.48 They use video (in some cases interactive) to convey to patients what it might be like to live with possible future outcomes.

the American Medical Association’s Family Medical GuideTM is a program consisting of seven modules: (1) diseases, disorders, and conditions; (2) an atlas of the body; (3) symptoms and self-diagnosis; (4) your healthy body; (5) injuries and emergencies; (6) diagnostic imaging techniques; and (7) caring for the sick

The diagnostic symptom charts are flow diagrams in which each question is read to the user by the computer. A “yes” or “no” answer to the question directs the user through the flow diagram to the next question, leading to a patient-specific recommendation

The Home Medical Advisor ProTM CD-ROM software has a symptom analysis program for single symptoms, and a symptom complex analysis program for multiple symptoms

The two patient software packages, Medical HouseCallTM and Pediatric HouseCallTM, have sophisticated diagnostic features. The algorithms were derived from a diagnostic and treatment expert system for clinicians, known as IliadTM, developed in Utah by a team from Applied Medical Informatics, and from the Department of Medical Informatics and the University of Utah.4

In adapting the Iliad knowledge base for home use with patients, the developers eliminated the physical exam findings and lab tests. The vocabulary was translated into “consumer language.”

Helping Patients Judge the Quality of Health Information

Because the quality of health information is so critical for consumers, several organizations have created guidelines for judging thequality of information on the Web for consumers.50–52 Some of the criteria included in all of these guidelines are topical relevance, currency of the information, accuracy, and authoritativeness or objectivity

The relevance of a site is context-specific and depends on the particular question an individual consumer has in mind. To find appropriate materials, sites must be clearly organized and/or have intelligent search functions

The final aspect of relevance to an individual has to do with whether the material is action oriented,

Currency or the timeliness of information is an important consideration. It is often difficult to have a generalized policy on how often health materials need to be updated. However, most professional sites ensure at least quarterly review of all materials.

the Health on the Net (HON) Foundation52 has promoted an ethical code of conduct and a set of standards for Web site developers to ensure the reliability of medical and health information available on the World Wide Web. Consumer health sites that display an HON certificate signify that they are in compliance with the HON code of conduct and standards.

Patient Access to Decision Support Systems

A lack of reading ability is a functional barrier affecting use of computer systems

Most studies on the comprehension of health education handouts, typically show that only half of the patients are able to comprehend written health materials

Health materials should be written at least three grade levels lower than the average educational level of the target population

Organization and clarity need to be considered in creating educational materials.62 Computers with multimedia capabilities can correct some of these problems by conveying information through video, audio and graphics that would normally be presented as written text. These systems can also be adapted for multiple foreign languages

Most developers have not invested the time to develop systems that are culturally and linguistically relevant to diverse populations

Finally, the question of who will pay for the access and use of technologies for consumer health information is still an unresolved issue

The Future of Decision Support Systems for Patients

The criteria for evaluating computer-based decision support systems for patients are similar to the criteria for physician systems, namely accuracy and effectiveness

Careful needs assessment before system development, usability testing during development, controlled clinical trials, and studies of use and outcomes in natural settings are all critical to our understanding of how to best provide health information and decision assistance to patients.

Ch5: Diagnostic Decision Support Systems*

Randolph A. Miller and Antoine Geissbuhler

Definitions of Diagnosis

In order to understand the history of clinical diagnostic decision support systems and envision their future roles, it is important to define clinicaldiagnosis and computer-assisted clinical diagnosis. A simple definition of diagnosis is:2 *the placing of an interpretive, higher level label on a set of raw, more primitive observations* [Definition 1].

A more involved definition of diagnosis, specific for clinical diagnosis, is:2 *a mapping from a patient's data (normal and abnormal history, physical examination, and laboratory data) to a nosology of disease states* [Definition 2].

A more accurate definition is found in the Random House Collegiate Dictionary. There, diagnosis is defined as:3 *"the process of determining by examination the nature and circumstances of a diseased condition"* [Definition 3].

three components of "comprehensive information needs:" (1) currently satisfied information needs (information recognized as relevant to a question and already known to the clinician); (2) consciously recognized information needs (information recognized by the clinician as important to know to solve the problem, but which is not known by the clinician); and (3) unrecognized information needs (information that is important for the clinician to know to solve a problem at hand, but is not recognized as being important by the clinician).

The challenge is to quickly and efficiently reconcile one body of information with the other.1,4 DDSS can potentially facilitate that reconciliation. A DDSS can be defined as: *a computer-based algorithm that assists a clinician with one or more component steps of the diagnostic process* [Definition 4]

The utility of making specific diagnoses lies in the selection of effective therapies, making accurate prognoses, and providing detailed explanations

Human Diagnostic Reasoning
Diagnostic reasoning involves diverse cognitive activities, including information gathering, pattern recognition, problem solving, decision making, judgment under uncertainty, and empathy

Most models of diagnostic reasoning include the following elements: the activation of working hypotheses; the testing of these hypotheses; the acquisition and interpretation of additional information; and confirming, rejecting, or adding of new hypotheses as information is gathered over time

In clinical diagnosis, early hypothesis generation helps to constrain reasoning to "high yield" areas, and permits the use of heuristic methods to further elucidate a solution.20 Studies have shown that most clinicians employ the hypothetico-deductive method after early hypothesis generation.
1

three abstract categories of human diagnostic reasoning strategies: probabilistic, causal, and deterministic

Probabilistic modelssuch as Brunswik's lens model25 and Bayesian26,27 approaches, as well as decision analysis28,29 define statistical associations between clinical variables and use mathematical models to compute optimal decisions

Models of causal (pathophysiological) reasoning, such as those developed by Feinstein32,33 in the 1970s, establish cause-and-effect relations between clinical variables within anatomic, physiologic, biochemical, and ultimately, genetics-based representations of the reality

Causal, pathophysiological reasoning uses shared, global, patient-independent knowledge33 and provides an efficient means of verifying and explaining diagnostic hypotheses

In deterministic models, production rules, i.e., specifying appropriate actions in response to certain conditions, are used to represent the basic building blocks of human problem-solving. Such if–then rules representing compiled knowledge can be expressed in the form of branching-logic flowcharts and clinical algorithms for nonexperts to follow. However, production rules do not deal effectively with uncertainty,34 which is a disadvantage in clinical practice, where uncertainty is a common feature

Historical Survey of Diagnostic Decision Support Systems

The majority of important concepts related to current DDSS were developed and presented in the literature prior to 1976. In a comprehensive 1979 review of reasoning strategies employed by early DDSS, Shortliffe, Buchanan, and Feigenbaum identified the following classes of DDSS: clinical algorithms, clinical databanks that include analytical functions, mathematical pathophysiological models, pattern recognition systems, Bayesian statistical systems, decision-analytical systems, and symbolic reasoning (sometimes called "expert" systems)

Logical systems, based on "discriminating questions" to distinguish among mutually exclusive alternatives, have played an important role since the pioneering work by Bleich and his colleagues38 on acid base and electrolyte disorders. To this day, such systems are applicable to narrow domains, especially those where it is fairly certain that only one disorder is present.

Bayes' rule is applicable to larger domains

A large number of groups have subsequently developed, implemented, and refined Bayesian methods for diagnostic decision making, and a wave of enthusiasm surrounds current work on Bayesian belief networks for clinical diagnosis. 1 Probabilistic systems have played, and will continue to play, an important role in DDSS development

alternative is heuristic reasoning, reasoning based on empirical rules of thumb

The HEME program for diagnosis of hematological disorders was one of the earliest systems to employ heuristics and also one of the first systems to use, in effect, criteria tables for diagnosis of disease states.

MYCIN used backward chaining through its rule base to collect information to identify the organism(s) causing bacteremia or meningitis in patients

Systems based on fuzzy set theory and Bayesian belief networks were developed to overcome limitations of heuristic and simple Bayesian models.1

Neural networks represent an entirely new approach to medical diagnosis, although the weights learned by simple one-layer networks may be analogous or identical to Bayesian probabilities

Developing, Implementing, Evaluating, and Maintaining Diagnostic Decision Support Systems

There are a number of problems that have limited the ultimate success of DDSS to date. These include: difficulties with domain selection and knowledge-base construction and maintenance; problems with the diagnostic algorithms and user interfaces; the problem of system evolution, including evaluation, testing, and quality control; issues related to machine interfaces and clinical vocabularies; and legal and ethical issues

Clinical Domain Selection

DDSS domain selection is often problematic. Substantial clinical domains must be chosen in order to avoid creating “toy” systems. However, construction of knowledge bases, to support substantial DDSS, can require dozens of person-years of effort in broad domains such as general internal medicine.

Different problems affect DDSS with narrow domains. One problem is garnering an adequate audience

Knowledge-Base Construction and Maintenance

Knowledge base maintenance is critical to the clinical validity of a DDSS.1 Yet it is hard to judge when new clinical knowledge becomes an established “fact.”

Knowledge-base construction must be a scientifically reproducible process that can be accomplished by qualified individuals at any site.67 Knowledge-base construction should be clinically grounded, based on “absolute” clinical knowledge whenever possible

the long-term value and viability of a system depends on the quality, accuracy, and timeliness of its knowledge base

Even initially successful DDSS cannot survive unless the medical knowledge bases supporting them are kept current

Diagnostic Decision Support Systems—Diagnostic Algorithms and User Interfaces

Just as computer-based implementation of many complex algorithms involves making trade-offs between space (memory) and time (CPU cycles), development of real-world diagnostic systems involves a constant balancing of theory (model complexity) and practicality (ability to construct and maintain adequate medical databases or knowledge bases, and ability to create systems which respond to users’ needs in an acceptably short time interval).

DDSS must be designed to permit users to apply individual tools to assist with the sequence of steps in the diagnostic process in the sequence that the user prefers at the time,not in an arbitrary sequence selected by the DDSS algorithm.

Untilthere is a standard, integrated environment and user interface that allows smooth transition among dedicated applications, DDSS are not likely to be used heavily

Diagnostic Decision Support Systems Testing, Evaluation, and Quality Control

It is extremely important during system development to conduct informal “formative” type evaluations. As a part of this process, new cases must be analyzed with the DDSS on a regular (e.g., weekly) basis. After each failure of the DDSS to make a “correct” diagnosis, careful analysis of both the system’s knowledge base and diagnostic algorithms must be carried out.

Formal evaluations of DDSS should take into account the following four perspectives: (1) appropriate evaluation design; (2) specification of criteria for determining DDSS efficacy in the evaluation; (3) evaluation of the boundaries or limitations of the DDSS; and (4) identification of potential reasons for “lack of system effect.”64 Each of these issues is discussed below

Appropriate Evaluation Design Evaluation plans should be appropriate for the information needs being addressed, the level of system maturity, and users’ intended form of DDSS usage (or specific system function evaluated).62,64

In 1994, Berner and colleagues evaluated the ability of several systems to generate first-pass differential diagnoses from a fixed set of input findings. 70 These findings were not generated by everyday clinical users, but from written case summaries of real patient data

Specification of Criteria for Determining Diagnostic Decision Support Systems Efficacy in the Evaluation

Evaluations must determine if the criteria for “successful” system performance are similar to what clinical practitioners would require during actual practice

Criteria for the establishment of a “gold standard” diagnosis should be stated prospectively, before beginning data collection

Evaluation of the Boundaries or Limitations of the Diagnostic Decision Support Systems

A system may fail when presented with cases outside its knowledge-base domain, but if an evaluation uses only cases from within that domain, this failure may never be identified.The limits of a system’s knowledge base are a concern because patients do not accurately triage themselves to present to the most appropriate specialists

Identification of Potential Reasons for “Lack of System Effect”

A model of all of the possible influences on the evaluation outcomes would include DDSS-related factors (knowledge-base inadequacies, inadequate synonyms within vocabularies, faulty algorithms, etc.), user-related factors (lack of training or experience with the system, failure to use or understand certain system functions, lack of medical knowledge or clinical expertise, etc.) and external variables (lack of available gold standards, failure of patients or clinicians to follow-up during study period).

Diagnostic Decision Support Systems Interface and Vocabulary Issues

it is common wisdom that DDSS are most likely to succeed if they can be integrated into a clinical environment so that patient data capture is already performed by automated laboratory and/or hospital information systems.

By integrating DDSS into healthcare provider results reporting and order entry systems, the usual computer-free workflow processes of the clinician can be replacedwith an environment conducive to accomplishing a number of computerassisted clinical tasks,making it more likely that a DDSS will be used.

In order to facilitate data exchange among local and remote programs, it is mandatory to have a lexicon or interlingua which facilitates accurate and reliable transfer of information among systems that have different internal vocabularies (data dictionaries). The United States National Library of Medicine Unified Medical Language System (UMLS) project, which started in 1987 and continues through the present time, represents one such effort

Legal and Ethical Issues

The complexity of these issues makes it very difficult to formulate appropriate regulatory policy.

The Future of Diagnostic Decision Support Systems

One byproduct of the success of these systems is that users may be less vigilant in questioning system accuracy

Ch6 Ethical and Legal Issues in Decision Support

Kenneth W. Goodman

Another way of putting this is to say that computers cannot, either in principle or at least for the foreseeable future, supplant human decision makers. This observation entails ethical obligations, namely that computers ought not to be relied on to do what humans do best, and that a “computer diagnosis” cannot, as a matter of course or policy, be allowed to trump a human decision or diagnosis

Three core areas of ethical concern have emerged in discussions of computer systems that are used to remind, consult, or advise clinicians: (1) care standards; (2) appropriate use and users; and (3) professional relationships

Care Standards

The overarching question may be put thus: does the new technology improve patient care? If the answer is affirmative, we may suppose we have met an important responsibility. If the answer is negative, it seems clear we should not use the new technology

The problem is, we often do not know how to answer the question. That is, we are sometimes unsure whether care will be improved by the use of new technologies

Standards evolve in the health professions because they plot the kinds of actions that are most successful in achieving certain ends.To fail to adhere to a standard is thus to increase the risk of error, at least in a mature science

Sometimes there are good reasons to violate a standard.This demonstrates how some clinical progress is possible: if everyone in all cases stuck to a rigid standard, there would be no internal evidence to support modifications of the standard

In the case of computer-assisted diagnoses, the challenge is perhaps best put in the form of a question: does use of a decision support system increase the risk of error?

This means that we are pressed to answer an ethical question (is it acceptable to use a decision support system?) in a context of scientific uncertainty (how accurate is the system?). Many challenges in contemporary bioethics share this feature, namely, that moral uncertainty parallels scientific or clinical ignorance

What we generally want in such cases is a way to stimulate the appropriate use of new technologies without increasing patient risk. One approach to doing this is given the nearly oxymoronic term “progressive caution.” The idea is this: “Medical informatics is, happily, here to stay, but users and society have extensive responsibilities to ensure that we use our tools appropriately.This might cause us to move more deliberately or slowly than some would like. Ethically speaking, that is just too bad.”10 Such a stance attempts the ethical optimization of decision-support use and development by encouraging expansion of the field, but with appropriate levels of scrutiny, ovesight, and, indeed, caution.

Appropriate Use and Users

One way to abuse a tool is to use it for purposes for which it is not intended. Another is to use a tool without adequate training. A third way is to use a tool incorrectly (carelessly, sloppily, etc.) independently of other shortcomings

A medical computer system may be used inappropriately if, for instance, it was designed for educational purposes but relied on for clinical decision support; or developed for modest decision support (identifying a number of differential diagnoses) but used in such a way as to cause a practitioner to abandon a diagnosis arrived at by sound clinical methods.

Identifying qualifications and providing training must be key components of any movement to expand the use of decision support software. Ethical concerns arise when we are unsure of the appropriate or adequate qualifications and levels of training.

What this means is that the novice might not know when the system is in error or producing flawed output, when it is operating on insufficient information, when it is being used in a domain for which it was not designed, and so on.

There are several reasons we must also focus ethical attention on the use of decision support software by computationally naive health professionals. Such professionals might not use such software to good effect (either by over- or underestimating its abilities), might not be using it properly, or, like the novice, might not know when the system is being used in inappropriate contexts

Such fears can be addressed by requirements that users of CDSS have appropriate qualifications and be adequately trained in the use of the systems.

that the use of diagnostic software cannot, in the long run, advance ethically without a better sense of where to establish guideposts for qualifications and training

A further ethical concern about appropriate use and users emerges from the potential to deploy decision support systems in contexts of practice evaluation, quality assessment, reimbursement for professional services, and the like. One can imagine an insurance company or managed care organization using decision support to evaluate, or even challenge, clinical decisions.

Professional Relationships

That a computer might be used to (help) render a diagnosis causes us to run the risk of what we will call the "computational fallacy." This is the view that what comes out of a computer is somehow more valid, accurate, or reliable than human output. Providers and patients who take such a view introduce a potentially erosive, if not destructive, element into shared decision-making contexts.

In any case, it is inappropriate to use computer data or inferences to trump hesitant patients, or bully them into agreeing with a health professional

A related issue is likely to arise with increased frequency as patients gain access to decision support software and use it to make demands on physicians, or at least to challenge or second-guess them. The difficulties raised by these demands and challenges will multiply as these systems improve

a physician should respond to such use by making clear that computers are not surrogates for health professionals and that the practice of medicine or nursing entails far more than statistical induction from signs, symptoms, and lab values.

Legal and Regulatory Issues

Computers and software raise conceptually fascinating and important practical questions about responsibility and liability. Further, the question of whether a decision-support system is a medical device needing governmental regulation is a source of tension and debate

Liability and Decision Support

The overriding legal issue related to computational decision support is liability for use, misuse, or even lack of use of a computer to make or assist in rendering medical decisions

The negligence standard applies to services, and strict liability applies to goods or products, although negligence can sometimes also apply to goods, as in cases of negligent product design.There is no consensus about whether decision-support systems are services or products, in part because these systems have properties that resemble both services and products

If a patient is injured by a defective system, it remains to be determined who used the system (the physician? the patient?) and whether it was misused. Also, it can be exquisitely difficult to identify the defect in a computer program,15 as well as to answer the important question as to whether a physician could have intervened and prevented the application of mistaken advice.

Neither is there a clear standard of care for use of decision-support software by clinicians. Physicians or nurses might someday be found negligent either for accepting a mistaken computer diagnosis or, having erred in diagnosis themselves, for failing to have used a decision-support system that might have proved corrective.