DEVELOPMENT AND TESTING OF FEED – A FEEDBACK EXPERT SYSTEM FOR EMS DOCUMENTATION

by

DEVASHISH SAINI

HELMUTH F. ORTHNER, COMMITTEE CHAIR
ETA S. BERNER
TODD B. BROWN
CHARLES J. GODWIN

A THESIS

Submitted to the graduate faculty of The University of Alabama at Birmingham, in partial fulfillment of the requirements for the degree of Master of Science Health Informatics

BIRMINGHAM, ALABAMA

2007
Emergency Medical Services (EMS) are delivered in a chaotic, dynamic, unpredictable, and therefore error-prone environment. Deviations from protocol are common and some may be severe enough to cause harm to the patient. EMS personnel, like all healthcare providers, can benefit from feedback, continued education and medical oversight. A Feedback Expert system for EMS Documentation (FEED) was developed for providing feedback to EMS personnel after the ambulance run is completed and documented electronically. FEED’s knowledge base was derived from Alabama State EMS Protocols and discussions with two EMS faculty members from UAB School of Medicine. FEED is rule-based and uses confidence factors to represent uncertainty. It obtains data input from a Microsoft SQL Server database built according to the specifications of the National Highway Transportation and Safety Administration. The inference engine analyzes this input using the rules in the knowledge base and produces output in the form of a checklist of recommended documentation for the completed run. FEED’s knowledge base and inference engine were developed iteratively using CLIPS (C Language Integrated Production System), an expert system development tool, and the user interface in Visual Basic .NET. FEED was developed, tested and refined over three iterations. Testing included local syntax checking, consistency checking, informal verification, and formal verification and validation using two expert focus groups. One group verified whether FEED met its requirements specifications, and another group validated its output
using real-life patient care reports. The validation focus group revealed a sensitivity of 60% and a specificity of 85% when a cut-off value of 8.5 was chosen for FEED’s validity rating. The review of false positives and false negatives revealed several inaccuracies in the KB, and several potential expert errors. After appropriate corrections and revisions, FEED’s utility could be tested in the field using a larger group of EMS providers, leading to further refinement of all components of FEED. Ultimately, providing feedback to EMS personnel may improve patient care in the field and provide better quality data for EMS research.
ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Helmuth Orthner for his unwavering support and encouragement throughout the thesis project as well as my three years in this program. He gave me opportunities to bring out the best in me, inspiring me to strive for and achieve ever greater goals. Dr. Todd Brown has been a friend, mentor and collaborator I always wished I had. I thank him for believing in me when I was down; and for keeping it real when I was aiming too high. Without Dr Eta Berner’s suggestions and criticism, this project would never have become as polished and comprehensive; I thank her for that. I would also like to thank Charles Godwin for being the expert for this expert system and for patiently suffering through the endless sessions of knowledge acquisition. My thanks also go to the National Institutes of Health/ National Library of Medicine, who funded this project, to Birmingham Fire and Rescue Service for providing the patient care reports, and to Marcie Battles for helping me assemble the building blocks of this project. Lastly, my thanks go out to all my family and friends; who have stood by me over the years and helped me be the person I am today.
TABLE OF CONTENTS

Page

ABSTRACT........................................................................................................................................................................ ii

ACKNOWLEDGMENTS ............................................................................................................................... iv

LIST OF TABLES...................................................................................................................................................... vii

LIST OF FIGURES ...................................................................................................................................................... ix

LIST OF ABBREVIATIONS....................................................................................................................................... x

BACKGROUND AND SIGNIFICANCE................................................................................................................... 1

History and Current State of Emergency Medical Services ................................................................. 1
The Problems in Focus.................................................................................................................................................. 4
Consequences of the Problems............................................................................................................................... 8
Causes of the Problems................................................................................................................................................ 11
Solutions for the Problems................................................................................................................................. 14
Expert Systems – Definition and History........................................................................................................... 17
Expert Systems – Development and Evaluation................................................................................................. 19
Technology in Emergency Medical Services ........................................................................................................ 21
Vision............................................................................................................................................................................ 23
Summary..................................................................................................................................................................... 23

METHODS ........................................................................................................................................................................ 25

Development Model.................................................................................................................................................... 25
Stage I – Planning..................................................................................................................................................... 27
  Feasibility Assessment........................................................................................................................................... 27
  Resource Management......................................................................................................................................... 30
  High-Level Requirements................................................................................................................................. 33
  Preliminary Functional Layout........................................................................................................................... 36
Stage II – Knowledge Definition......................................................................................................................... 38
  Source Identification and Selection..................................................................................................................... 38
  Knowledge Acquisition....................................................................................................................................... 39
  Detailed Functional Layout............................................................................................................................... 40
  Preliminary Control Flow.................................................................................................................................. 45
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Components of the Linear Model of Expert System Development Life Cycle, and Associated Tasks, According to Giarratano and Riley</td>
<td>21</td>
</tr>
<tr>
<td>2</td>
<td>Components of the Modified Linear Model Followed During Development of FEED, and Associated Tasks</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>Iterations of the Spiral Model, and Tasks from Linear Model Followed in Each Iteration during Development of FEED</td>
<td>26</td>
</tr>
<tr>
<td>4</td>
<td>Criteria for Evaluation of Available Expert System Shells</td>
<td>31</td>
</tr>
<tr>
<td>5</td>
<td>Calculation of Sensitivity and Specificity for a New Test T, Through Comparison with a Gold Standard, G</td>
<td>75</td>
</tr>
<tr>
<td>6</td>
<td>Calculation of Sensitivity and Specificity for a New Test T, Through Comparison with a Gold Standard, G, When X is the Cut-off Value</td>
<td>75</td>
</tr>
<tr>
<td>7</td>
<td>Sample Dummy Table for the First ROC Curve, at Cut-off X for FEED’s Validity Rating</td>
<td>76</td>
</tr>
<tr>
<td>8</td>
<td>Sample Dummy Table for the Second ROC Curve, at Cut-off X for FEED’s Validity Rating</td>
<td>77</td>
</tr>
<tr>
<td>9</td>
<td>Participation of Experts in the Verification Focus Group</td>
<td>79</td>
</tr>
<tr>
<td>10</td>
<td>Differences in FEED’s Likelihood Scores (LS) and Seriousness Scores (SS) Before and After the Expert Verification Focus Group</td>
<td>80</td>
</tr>
<tr>
<td>11</td>
<td>Errors Found by the Expert Focus Group, Grouped by Kind of FEED’s Suggestion</td>
<td>82</td>
</tr>
<tr>
<td>12</td>
<td>Errors Found by the Expert Focus Group Grouped by Validity Score</td>
<td>82</td>
</tr>
<tr>
<td>13</td>
<td>Sensitivity and Specificity of FEED at Four Different Cut-off Points, from the First ROC Curve, Comparing Suggestions Labeled as ‘Incorrect’ by FEED to Results from the Expert Focus Group</td>
<td>83</td>
</tr>
</tbody>
</table>
14 Sensitivity and Specificity of FEED at Four Different Cut-off Points, from the Second ROC Curve, Comparing Suggestions Labeled as ‘Incorrect’ or ‘Unknown’ by FEED to Results from the Expert Focus Group ........................................................................................................84

15 Review of False Positive Errors Encountered in the First Comparison, at Cut-off Level of 8.75 for FEED’s Validity Score .........................85

16 Review of False Negative Errors Encountered in the Second Comparison, at Cut-off Level of 8.75 for FEED’s Validity Score .........................86
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Components of FEED at the end of second iteration</td>
<td>41</td>
</tr>
<tr>
<td>2</td>
<td>Components of FEED at the end of the project</td>
<td>44</td>
</tr>
<tr>
<td>3</td>
<td>A typical CLIPS rule with antecedent conditions, which, if satisfied, will ‘fire’ the rule, leading to execution of the consequent commands</td>
<td>51</td>
</tr>
<tr>
<td>4</td>
<td>A screenshot of the login screen of FEED showing available options</td>
<td>59</td>
</tr>
<tr>
<td>5</td>
<td>A screenshot showing the list of cases available to the user who is logged in</td>
<td>59</td>
</tr>
<tr>
<td>6</td>
<td>A screenshot of FEED asking for the value of a particular data element</td>
<td>60</td>
</tr>
<tr>
<td>7</td>
<td>A screenshot of FEED asking if the particular data element had the value specified; in this case, whether the patient had a past history of dialysis</td>
<td>61</td>
</tr>
<tr>
<td>8</td>
<td>A screenshot of FEED showing the feedback provided to the user</td>
<td>62</td>
</tr>
<tr>
<td>9</td>
<td>A sample case report from BFRS</td>
<td>70</td>
</tr>
<tr>
<td>10</td>
<td>First ROC curve, comparing suggestions labeled as ‘incorrect’ by FEED to results from the expert focus group</td>
<td>83</td>
</tr>
<tr>
<td>11</td>
<td>Second ROC curve, comparing suggestions labeled as ‘incorrect’ or ‘unknown’ by FEED to results from the expert focus group</td>
<td>84</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>ACLS</td>
<td>advanced cardiac life support</td>
<td></td>
</tr>
<tr>
<td>ALS</td>
<td>advanced life support</td>
<td></td>
</tr>
<tr>
<td>API</td>
<td>application programming interface</td>
<td></td>
</tr>
<tr>
<td>AUC</td>
<td>area under the curve</td>
<td></td>
</tr>
<tr>
<td>BFRS</td>
<td>Birmingham Fire and Rescue Service</td>
<td></td>
</tr>
<tr>
<td>BLS</td>
<td>basic life support</td>
<td></td>
</tr>
<tr>
<td>BMIST</td>
<td>Battlefield Medical Information Systems Tactical</td>
<td></td>
</tr>
<tr>
<td>CDSS</td>
<td>clinical decision support system</td>
<td></td>
</tr>
<tr>
<td>CE</td>
<td>continuing education</td>
<td></td>
</tr>
<tr>
<td>CLIPS</td>
<td>C Language Integrated Production System</td>
<td></td>
</tr>
<tr>
<td>CPR</td>
<td>cardiopulmonary resuscitation</td>
<td></td>
</tr>
<tr>
<td>CRE</td>
<td>CLIPS rule engine</td>
<td></td>
</tr>
<tr>
<td>dll</td>
<td>dynamic link library</td>
<td></td>
</tr>
<tr>
<td>ECG</td>
<td>electrocardiogram</td>
<td></td>
</tr>
<tr>
<td>ED</td>
<td>emergency department</td>
<td></td>
</tr>
<tr>
<td>EMS</td>
<td>Emergency Medical Services</td>
<td></td>
</tr>
<tr>
<td>EMSIS</td>
<td>EMS Information System</td>
<td></td>
</tr>
<tr>
<td>EMT</td>
<td>emergency medical technician</td>
<td></td>
</tr>
<tr>
<td>ePCR</td>
<td>electronic patient care report</td>
<td></td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>FEED</td>
<td>Feedback Expert system for EMS Documentation</td>
<td></td>
</tr>
<tr>
<td>HTML</td>
<td>hypertext markup language</td>
<td></td>
</tr>
<tr>
<td>IE</td>
<td>inference engine</td>
<td></td>
</tr>
<tr>
<td>IQR</td>
<td>interquartile range</td>
<td></td>
</tr>
<tr>
<td>IV</td>
<td>intravenous</td>
<td></td>
</tr>
<tr>
<td>KB</td>
<td>knowledge base</td>
<td></td>
</tr>
<tr>
<td>LHS</td>
<td>Left Hand Side</td>
<td></td>
</tr>
<tr>
<td>LS</td>
<td>Likelihood Score</td>
<td></td>
</tr>
<tr>
<td>MDS</td>
<td>Minimum Dataset</td>
<td></td>
</tr>
<tr>
<td>MEPARS</td>
<td>Medical Error Prevention and Reporting System</td>
<td></td>
</tr>
<tr>
<td>NASA</td>
<td>National Aeronautical and Space Administration</td>
<td></td>
</tr>
<tr>
<td>NEMSIS</td>
<td>National EMS Information System</td>
<td></td>
</tr>
<tr>
<td>NHTSA</td>
<td>National Highway Traffic Safety Administration</td>
<td></td>
</tr>
<tr>
<td>OAV</td>
<td>Object-Attribute-Value</td>
<td></td>
</tr>
<tr>
<td>OPQRST</td>
<td>onset, provocation, quality, radiation, severity and timing</td>
<td></td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer</td>
<td></td>
</tr>
<tr>
<td>PDF</td>
<td>Portable Document Format</td>
<td></td>
</tr>
<tr>
<td>PEA</td>
<td>Pulseless Electrical Activity</td>
<td></td>
</tr>
<tr>
<td>QA/QI</td>
<td>quality assurance/ quality improvement</td>
<td></td>
</tr>
<tr>
<td>QMR</td>
<td>Quick Medical Reference</td>
<td></td>
</tr>
<tr>
<td>RHS</td>
<td>Right Hand Side</td>
<td></td>
</tr>
<tr>
<td>ROC</td>
<td>Receiver Operator Characteristic</td>
<td></td>
</tr>
<tr>
<td>SDLC</td>
<td>System Development Life Cycle</td>
<td></td>
</tr>
<tr>
<td>Acronym</td>
<td>Full Form</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td></td>
</tr>
<tr>
<td>SQL</td>
<td>Sequential Query Language</td>
<td></td>
</tr>
<tr>
<td>SS</td>
<td>Seriousness Score</td>
<td></td>
</tr>
<tr>
<td>UAB</td>
<td>University of Alabama at Birmingham</td>
<td></td>
</tr>
<tr>
<td>WYSIWYG</td>
<td>What You See Is What You Get</td>
<td></td>
</tr>
<tr>
<td>XML</td>
<td>extensible markup language</td>
<td></td>
</tr>
<tr>
<td>XSD</td>
<td>XML schema definition</td>
<td></td>
</tr>
</tbody>
</table>
BACKGROUND AND SIGNIFICANCE

History and Current State of Emergency Medical Services

Emergency Medical Services (EMS) is defined as an organized system designed to transport sick or injured patients to the hospital. (1) EMS lies at the intersection of health care, public health and public safety, interacting with and carrying out the roles and responsibilities of all three. (2) With the almost universal access to the single emergency number 9-1-1, EMS is often the patients’ point of first contact with the health care system. In the United States, about 16 million patients are transported by EMS to emergency departments (EDs) every year. (3)

The first well-described organized attempt to treat and transport patients occurred under the leadership of Baron Dominique Jean Larry during the Italian campaign of the French Revolution in 1794. (1) After initial failures to emulate the system during the American Civil War, General Jonathan Letterman created the first organized system to transport sick and injured patients in the United States. (4) This system inspired the creation of civilian emergency systems, and the first civilian-run, hospital-based ambulance system was created at the Commercial Hospital of Cincinnati (now Cincinnati General) in 1865. (1) New York was the first city to have a municipality-run ambulance service in 1869. (5) Other cities followed suit, but until the 1960s, these “ambulance” services were usually provided by local funeral homes that used hearses to transport patients to hospitals. (1) In 1910, the American Red Cross began training civilians to provide first aid dur-
ing emergencies. Training of emergency medical technicians (EMTs) began in the 1950s and 1960s, along with the development of modern models of advanced life support (ALS) including defibrillation, airway management and cardiopulmonary resuscitation. (1)

EMS in United States took its present shape in the 1970s after the National Academy of Sciences released a paper titled “Accidental Death and Disability: The Neglected Disease of Modern Society” in 1966. (6) This paper outlined the inadequacies in the pre-hospital and ED care in the United States and proposed 29 recommendations for improvement. The report gained support from surgeons with military experience in World War II and Korea, who maintained that emergency care available to civilians was much worse than that available to the soldiers. (2) Spurred by the report, the federal government created the Department of Transportation that developed the first basic EMS curriculum. (1) California’s Wedworth Townsend Act of 1970 became the first to permit paramedics to act as physician surrogates and provide medical care in the absence of physicians. (7) The EMS Systems Act of 1973 earmarked millions of dollars for the development of EMS systems throughout the country. (8) This provided the impetus for the growth of some 300 regional EMS systems within a few years. (2) Although the intent of the law was development of integrated systems, most EMS systems developed as a patchwork around local needs, characteristics and concerns. (2)

This golden era of EMS effectively came to an end in 1981 with the Omnibus Reconciliation Act. (1) The categorical funding earmarked for EMS gave way to block grants to states for preventive health and other health services, which often got directed to areas other than EMS, leading to budget cuts for most EMS systems. (2) Since then, most of these systems have developed under local or regional authority, each taking shape ac-
cording to the unique needs and characteristics of the jurisdiction. The 1980s also saw the growth of several voluntary national EMS organizations. In 1996, the National Highway Traffic Safety Administration (NHTSA) published the ‘EMS Agenda for the Future,’ (9) and a year later, its implementation guide, (10) envisioning an EMS system that is integrated with the rest of the health care, proactive in providing community health and adequately funded and accessible. In 2007, the Institute of Medicine published its report ‘Emergency Medical Services: At the Crossroads,’ as a part of its ‘Future of Emergency Medicine’ series. (2) This report outlines a vision for the EMS of the 21st century, the challenges faced by EMS today, and recommendations ‘that encompass a wide range of strategic and operational issues, from workforce training, to additional investment in research, to the development of national standards for EMS system performance.’ (2)

EMS in the United States has several unique features. The first is a strong association with fire departments. Almost 90% of first responders and half of all EMS agencies in the US are fire-department based, (2, 11) and about 60% of all fire department calls are for medical aid. (12) Another unique feature is that almost all EMS care is provided by trained EMS personnel, including first responders, EMT-Basic, EMT-Intermediate, and paramedics. This is in contrast to many European countries where a two-tier system provides prehospital care, with physician-manned ambulances forming the second tier. (13, 14) The need for EMS in the US is growing, given the ageing baby-boomer generation, and the ongoing movement to suburbs. (2, 15) The effectiveness of EMS has also grown, with the almost universal coverage of the emergency number 9-1-1, routing of most severely injured patients to specialized trauma centers, advances in resuscitation and emergency medical care, and availability of air ambulance services in rural areas. (2)
The Problems in Focus

This project focused on three inter-related problems in EMS: Errors in patient care, non-compliance with patient care protocols, and erroneous documentation.

Errors made by EMS providers are directly related to patient outcome. Medical errors have received a lot of attention following the publication of the Institute of Medicine report, *To Err is Human*, which reported that preventable medical errors cause at least 44,000 deaths in the United States every year. (16) However, it is difficult to detect, count and measure medical errors, especially in EMS, where it is difficult to observe EMS personnel while they are providing care in the field. Two surveys of EMS personnel that focused on self-reports of medical errors provide some insight. In one survey of 283 EMTs attending a statewide EMS conference, 45% of respondents reported at least one error in the last one year. (17) The other survey of 352 paramedics in San Diego County focused only on medication errors. (18) In this survey, 9% of respondents reported committing a medication error within the last one year. These included dose-related errors (63%), protocol errors (33%), wrong route errors (21%) and wrong medication errors (4%). (18) In another study, 61 events were found from an anonymous event-reporting system and in-depth interviews with 15 paramedics. (19, 20) Of these, 44% were near misses and 56% were adverse events; 54% involved clinical judgment, 21% involved skill performance and 15% were medication errors. Within the first year of its implementation, seven errors were reported to the Medical Error Prevention and Reporting System (MEPARS), a proactive anonymous error reporting system of an EMS agency employing about 200 health care providers. (21)
In a consensus group meeting, a list of common potentially serious errors occurring in EMS was prepared. (22) It included ambulance crashes, missed endotracheal intubations, including unrecognized esophageal-placed endotracheal tubes, incorrect medication, dose, or protocol, and equipment failure. (22) Other EMS errors reported in the literature include incorrect decisions not to transport patients, (23) and poor quality cardiopulmonary resuscitation. (24, 25) However, direct evidence for EMS errors is available only for endotracheal intubation and cardiopulmonary resuscitation (CPR) errors. In two studies, physicians in the ED checked endotracheal tubes placed by paramedics in the field, and found that 5% and 25% of them were misplaced. (26, 27) In another two studies, quality of out-of-hospital CPR was judged based on chest compressions and ventilations recorded by defibrillator devices. One study found that only 29% of cases had adequate CPR quality, (25) and the other reported poor CPR performance on several quality measures when compared to international guidelines. (24) For other kinds of errors, evidence is based on retrospective analysis of patient care reports, where it is difficult to separate erroneous documentation from actual errors or protocol deviations. Several studies based on retrospective analysis of patient care reports have found discrepancies in EMS patient care. Some of them are described below.

Wasserberger et al analyzed 5,944 paramedic runs for three years from January 1981 to January 1984. (28) At that time, all runs included radio consultation with a base physician, who also recorded a run report. The authors compared run reports to standard protocols and to standard medical care, and found an overall compliance rate of 94% to protocols and 97% to standard medical care. Fifty seven percent of the deviations from protocol were actual deviations from standard of care. The most common deviations were
failure to give morphine and prophylactic lidocaine to patients with chest pain, failure to administer naloxone and glucose in patients with altered mental status, failure to apply cervical spine precautions in patients with suspected head trauma, and inappropriate administration of epinephrine before assessment of cardiac rhythm in patients with cardiac arrest. (28)

Salerno et al reviewed 1,246 ALS ambulance runs from seven ambulance services during a two-month period. (29) They found that 16% of the runs had deviations from protocol, of which 55% deviations were minor, 38% were serious, and 7% were very serious. A majority (69%) of these deviations occurred without appropriate communication with medical control. Serious deviations included failure to defibrillate promptly in cases of ventricular fibrillation, slow intravenous fluid rate in cases of shock, incorrect medication administration, and omission of spinal immobilization in a patient with a high likelihood of cervical trauma. (29)

Rittenberger et al analyzed chest pain cases from four different ambulance services. (30) They found that the documentation of OPQRST pain criteria (onset, provocation, quality, radiation, severity and timing) was very variable, ranging from 14% to 65% of cases for different criteria. Other history was very infrequently recorded, except for the history of nausea/vomiting and dyspnea (difficulty breathing). Similarly, specific areas of physical examination were recorded, but others, such as heart sounds and palpation of chest wall, were very infrequently documented. The rate for some interventions such as oxygen, nitroglycerin and intravenous (IV) access attempts were very similar. However, for other interventions, such as administration of aspirin and obtaining a 12-lead electrocardiogram (ECG), one of the ambulance services had much lower rates than others. (30)
In a similar review of EMS cases, McVaney et al found that only 54% of eligible patients with no contraindications to aspirin were given the drug. (31)

Snooks et al conducted two audits of patient care reports from asthma patients in a London ambulance service. (32) Of the patients subsequently diagnosed with asthma in ED, only 58% in the first audit and 75% in the second audit were administered nebulized salbutamol by the ambulance crew. For those that were administered salbutamol, adherence to protocol was judged to be 81%. Some additional patients were given salbutamol, that were found in the ED to not have asthma; however, they did not suffer from any un-toward consequences as a result of this therapy. Documentation of vital signs ranged from 45% for respiratory rate to 95% for level of responsiveness. (32) Moss et al analyzed 76,351 case reports from a centralized database containing EMS records from rural EMS providers. (33) They found that the records were often missing systolic blood pressure (17%), diastolic blood pressure (19%), pulse rate (16%) and respiratory rate (17%). (33)

Selden et al analyzed 2,698 consecutive paramedic run reports and found that of the cases in which the patient was released in the field and not transported, only 65% had adequate documentation regarding appropriateness of the release. (34) Other cases were labeled ‘inappropriate releases’ and these often did not have documentation regarding explanation of risks (51% of inappropriate releases), vital signs (34%), mental status (20%) and lack of impairment (13%). Inappropriate releases were more frequently associated with age 35-54 years, and prehospital diagnosis of no injury, head injury, seizure, minor trauma and ethanol use. (34)
Consequences of the Problems

The three problems discussed above have significant consequences for patient care. Practice of medicine by EMS personnel is considered to be contingent on medical supervision by a physician. (35) In the 1980s, several studies showed that “online” medical supervision (involving direct communication between EMS personnel and physician) was unnecessary in most situations, given that EMS personnel could perform equal or better with the use of protocols and standing orders than with rigorous online medical control. (36, 37) The use of standing orders and protocols was shown to improve outcomes for cardiac arrest patients. (37) Other studies concluded that radio contact with a physician was often unnecessary, and proposed focused use of online medical control. (36, 38-40) One study also reported reduction in on-scene times with standing orders and protocols, (41) though another study observed no difference in trauma cases. (42) Rigorous online medical control, therefore, gave way to “offline” medical control in the form of medical algorithms and protocols. (35)

However, the beneficial effect and advantages of EMS protocols are contingent on compliance by EMS providers. Deviations from protocol can be justifiable due to individual circumstances in particular cases. However, nursing literature indicates that deviations from protocols are common and commonly lead to errors. (43) Even routine procedures such as collection of vital signs may be vitally important. It has been shown in a simulation study of ALS cases that knowledge of vital signs changes the subsequent treatment plan of paramedics. (44) In addition, some of the studies quoted above provide patient outcome data along with protocol compliance rates. Salerno et al reported that 5.5% of patients with protocol deviations suffered from complications, including one
death. (29) In this study, none of the minor deviations, 11% of serious deviations and 21% of very serious deviations led to complications. Selden et al reported one patient complication as a result of inappropriate release. (34) Another study showed that decisions by EMS personnel to release patients in the field without transport are often high-risk and may lead to complications needing hospitalization. (23)

In addition, erroneous or incomplete documentation may hamper care in the ED. Cultural differences between ED staff and EMS personnel, time constraints and high stress levels may lead to loss of clinically important information during patient handoffs. (2) EMS care reports, if adequately completed, can complement the verbal exchange of information during handoffs.

Problems in EMS care documentation also discourage research on the effectiveness of prehospital medications and procedures, as well as research related to system changes like ambulance deployment strategies. (2, 45, 46) Research in EMS is often much more complex than the traditional ‘components’ research in other areas of medicine, and involves complex interrelated questions, diverse data points collected by multiple agencies and disciplines in a complex, uncontrolled environment. (47) Consequently, EMS today suffers from lack of research support for practices and protocols. (2, 10, 48, 49) The overwhelming majority of the published EMS literature is not research reports, but historical articles, editorials, consensus development pieces, biographies, monographs, or guidelines. (50) Between 1985 and 1997, only 54 of the 5,842 publications on EMS were randomized controlled trials. (51) Brice et al found that only 15% of research studies reported for EMS outcomes were randomized clinical trials. (52)
Thus, there is lack of scientific evidence for many of the clinical and system decisions made by EMS personnel and administrators. (9) These decisions are often made by convention or based on research in other medical fields. However, well-established standards in hospital care may not translate very well to EMS as EMS decisions often vary with time and place, and focus on acute management of the patient. (2) Rigorous studies often fail to show benefit of such standards. (51) For example, endotracheal intubation of children with respiratory distress is a widely accepted procedure in inpatient care, whereas research indicates that intubation of these children in the field by EMS personnel may not be beneficial. (2) In addition, there is limited available data to make strategic policy decisions at the regional, state or national level. (2) For example, the efforts to develop a new Medicare schedule for ambulance services ran into difficulties due to lack of available systems data on rural ambulance services. (15) This also limits efforts to outline standard performance indicators at the national level, as well as community performance measurement initiatives. (2) A study on out-of-hospital cardiac arrest reported widely varying survival rates from 2% to 25% in 29 separate EMS service areas. It is, however, unclear whether this difference is an actual difference in outcomes, or due to inconsistencies in data collection. (53) In another similar survey, half of the cities surveyed did not know their survival rates for ventricular fibrillation patients. (54)

In addition, there are consequences specific to the EMS agencies. Negligent medical treatment accounts for up to 77% of malpractice claims against EMS agencies, (55) and up to 35% of the dollars paid out. (56) Inadequate documentation often compounds this problem, as almost all claims are filed several months later, making memory of the case blurry and defense difficult. (57) Also, insurance claims may be denied due to
incomplete or inconsistent documentation. (15) Finally, in the absence of adequate do-
umentation, the agency’s quality assurance initiatives cannot be evaluated. (2)

Causes of the Problems

The causes of these problems are multi-factorial. It has often been stated that
EMS is a very error-prone environment, owing to its very unpredictable nature, usually
high emotional stress, and a compressed time frame. (22, 58) Several studies have found
high stress levels among EMTs. This is not surprising, given the life-and-death situations
and suffering patients frequently encountered by them. (59) These stress levels may be
further elevated by an unsupportive work environment or poor individual coping skills.
(60, 61) This can lead to a lapse of ‘mindfulness’ and impair executive functions neces-
sary for error-free clinical judgment. (62) In addition, long shifts (up to 48 hrs), irregular
working hours and high workload are common in EMS, especially in rural areas. (63)
The resultant fatigue can also lead to errors. (43, 62) In addition, distractions and inter-
ruptions are common, and these have been shown to be related to a higher incidence of
errors in nursing and pharmacy literature. (43, 64) All the above system problems, in ab-
sence of appropriate safeguards, make it more likely for individuals to make errors. (65)

In recent years, there has been a shift from “personal responsibility” to “system
responsibility” in the understanding of errors in health care. (65) However, the impor-
tance of factors specific to the individual cannot be denied. Lack of knowledge of proto-
cols and/or necessary procedural skills is very likely to have an important role in protocol
deviations and errors. Leape et al found that lack of drug knowledge accounted for 29%
of the 334 medication errors that occurred in a 6-month period in a hospital. (66) Simi-
larly, knowledge of CPR guidelines has been shown to affect CPR performance by EMS personnel. (67) Several studies have shown that health care professionals, including EMTs, have variable levels of clinical knowledge and skills. (43) In a survey of EMTs in an urban EMS agency, majority of personnel reported that they were not very confident in the use of EMS protocols, and even less so with the infrequently used protocols. (68) Some paramedics report being stressed during pediatric calls, (69, 70) primarily because of the low frequency with which they encounter such calls—a majority of EMS personnel report seeing less than three pediatric cases per month. (71, 72) In a survey of newly graduated EMTs, less than 40% of the respondents felt that they were ‘very well prepared’ to handle pediatric calls. (73)

Specifically, health care providers commonly make errors in calculation of drug dosages, and the error rates may actually increase with more experience. (74-76) A study of 109 practicing paramedics found poor medication calculation skills: The participants scored an average of 51% on the examination, and only one third of the questions related to intravenous infusions were answered correctly. (77) In addition, EMTs may have problems identifying errors once they have occurred. In one survey, only 68% of events were accurately identified as errors by EMTs. (78) Another survey reported 93% identification rate for severe errors, but rates as low as 36% for mild errors. (17) In certain cases the complexity of the protocol may be an issue. For instance, the quality and rate of chest compressions and ventilations has been shown to be lower in more complex resuscitation protocols. (79)

Many of these problems are magnified for rural EMS agencies, where providers encounter specific cases very infrequently. (2) In fact, in one such system, ‘some person-
nel certified to provide advanced care had never performed certain advanced procedures, such as airway intubation.’ (15) Another study suggested that a rural EMT with defibrillator training would defibrillate only two or three patients in a decade. (80) In 2003, only 9% of rural EMS providers responded to more than 10 calls per week, compared to 81% in urban areas. (2) Also, an estimated three-quarters of the EMS personnel in rural areas are volunteers, compared to only one-third in urban areas, (81, 82) and a higher percentage of those are trained only in basic life support (BLS). (2, 83) The opportunities to maintain and upgrade skills are limited in rural areas, and perhaps more so for volunteers. (15) At the same time, the death rate for children has been found to be higher for rural EMS systems than urban systems. (84, 85) Although other reasons such as scarcity of ALS services and distance to specialized care centers may account for most of the difference, some of it may be due to poor skills of EMS providers.

An additional factor may be the attitude of personnel towards EMS protocols. Research on guideline compliance by physicians indicates that the most common reason for non-compliance is lack of confidence in the guideline. (86) Nurses are known to find ways to work around rules in cases it is impossible to follow the rules. (87) Although the attitudes of EMTs towards protocols have not been specifically studied, low specificity of certain protocols such as the spinal immobilization protocol may contribute to low compliance with the protocols. (88) In addition, negative attitudes towards electronic documentation are frequently reported from EMS. Electronic documentation is often considered an unnecessary burden, especially when it has to be done in addition to paper documentation. (2, 68) It is a common perception that electronic databases at the state or regional level do not add any value for the individual provider, and consequently, ‘it is very
challenging for state agencies to convince local EMS providers, particularly volunteer agencies, to contribute to the state EMS data pool.’ (15) Therefore, if the EMS providers are forced to contribute to the electronic database, the quality of data submitted is likely to be low. (46)

Solutions for the Problems

Given the complex nature of the problem, several remedies have been proposed and tried to reduce errors, improve protocol compliance and increase quality of documentation in EMS. These are discussed below.

Medical supervision has been shown to improve quality of care and patient outcomes in EMS. (89) In 1983, the City of Houston EMS System recruited a full-time medical director who provided one-to-one training in specialized techniques as well as on-site training and feedback to all EMS providers in the agency. Within five years, the survival rate for patients with ventricular fibrillation rose from 0% to 21%, despite reduction in paramedic staff and increase in response times. (89) Accordingly, the need for medical oversight has been articulated in several consensus meetings. (90-92) However, recruitment of physicians to provide medical oversight is difficult and many EMS medical directors are primary care physicians without training in emergency medicine. (2) Less than 25% of EMS agencies have medical directors working more than 20 hrs per week with the agencies. (93) Rural agencies have an even harder time finding medical directors. In some of these agencies, physicians providing medical direction are located as far as 100 miles distant from the providers. (15) Accordingly, medical oversight was
ranked among top five concerns of rural EMS directors in two surveys conducted in 2000 and 2004. (94)

Continuing education (CE) and training, when regularly reinforced, is also effective in skill retention, especially for less-frequently used skills. (95, 96) Nurses who regularly update their knowledge of medications make fewer errors. (43) In a survey of rural EMTs, those that received four or more hours of pediatric CE felt more confident about their ability to treat children, than those that received three hours or less. (72) Similarly, in another survey, EMTs who participated in CE programs experienced had less skill deterioration than non-participants. (97) There is no doubt that CE is important, as initial training is uneven across the United States, (2) and given the large skill attrition rate—within two years of initial training, 50-60% of skills are lost. (97, 98) However, different states have different CE requirements. Although 98% of the agencies in large metropolitan cities require didactic CE credits, clinical CE is required by only 34%, and half of these award CE hours for documented in-field experience. (99) In addition, cost, availability and travel distance have been cited as barriers to obtaining CE, (71) and these may be especially problematic for rural EMS providers. (2) Accordingly, flexible models of CE are being advocated for rural providers, that can ‘be provider-need specific, conducted with varied teaching techniques emphasizing hands-on training and, where appropriate, distance learning.’ (100)

Another conventional method of dealing with errors has been to institute quality assurance/quality improvement (QA/QI) programs, wherein patient care reports are routinely analyzed for identifying deviations from protocol, and the personnel given additional training as needed. (101) QA/QI programs have become very common. In a survey
of EMS agencies in the 200 largest US cities, 82% reported measuring some clinical performance indicators. (93) Commonly measured indicators include cardiac arrest (64%), advanced airway management (57%), trauma management (43%), patient satisfaction (38%) and pain management (22%). (93) QA/QI programs have been shown to improve compliance with protocols and patient outcomes. (102) In a survey of EMS administrators, 96% of agencies that had QA/QI thought it improved patient care. (103) However, this approach has several disadvantages. It can be expensive—an estimate in 1993 showed a cost of $2 per run reviewed for a manual review system, and $4 for a computerized review system. (104) Lack of industry-wide standards makes it difficult to determine agency-specific standards. (2, 103) This approach has also been criticized for creating a “train-and-blame” environment that encourages personnel to hide errors. (21) As a counterpoint, anonymous error reporting systems like MEPARS are being used to identify areas in which system improvements could reduce the likelihood of errors. (21)

Point of care decision support systems, including alerts, reminders, business rules and treatment sets, have been known to increase compliance of health care providers with guidelines and protocols. (105) However, these may not be as valuable for EMS, as most of the documentation by EMS providers is done after the run is complete. Also, although 88% of EMS agencies in the 200 largest cities of the US are entering patient care reports in some form into a computer, only 24% have devices on which care can be documented electronically at patient’s side. (93) Making some aspects of the protocol mandatory has been shown to increase compliance in some cases, (106) and no effect in others. (32)

Practice critiquing systems that provide after-the-fact feedback to the care provider regarding their treatment plan are also effective in increasing compliance with
guidelines and protocols. A systematic review of studies on the effect of feedback on provider behavior showed a modest benefit of feedback (adjusted risk difference ranging from 16% decrease in compliance to 70% increase in compliance; median 5% increase in compliance, inter-quartile range 3% increase to 11% increase in compliance).

The usefulness of feedback systems has several different dimensions. Although not as common as in other health care fields, EMS does involve some application of subjective judgment during patient care, in addition to local practice variations. Critiquing systems acknowledge the limitations of computer systems in dealing with these problems, provide reasonable feedback to the provider, without “telling them what to do” and leave the provider in primary control of patient care. Another advantage is that such a system can be used to deliver focused, practice-based educational material to the provider, structuring complex clinical information around a particular case.

Expert Systems – Definition and History

An expert system or knowledge-based system is defined as “an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution.” There are several dimensions to this definition. Expert systems emulate the reasoning of human experts about the problem domain, rather than the domain itself. Similar to human experts, their expertise is usually limited to a single domain. They solve problems using heuristics and approximations rather than algorithms. Heuristics are rule of thumbs representing knowledge about how to solve problems in the domain. Also, the knowledge in the program (knowledge base, KB) is kept separate from the code that per-
forms the reasoning based on it (inference engine, IE). Other characteristics that define expert systems include the capability to solve real-world problems with adequate speed and reliability, at least at the level of the human expert; and to explain and justify their solutions. (110, 111)

The history of expert systems begins in the 1950s when several general problem solving programs were written, including the General Problem Solver by Newell, Shaw and Simon. Soon LISP programming language was developed to aid in the development of such high-level programs. (109, 111) Clinical decision support systems (CDSSs) like MYCIN, INTERNIST, HELP, DXplain and Iliad were built using expert system technology, and were shown to provide advice comparable to human experts. (112) The first CDSS to use heuristic reasoning was HEME. (113) It assisted physicians in the identification and diagnosis of specific hematological disorders. (114) Another early system provided feedback related to electrolyte imbalance based on rule-based logic. (115)

In 1973, Shortliffe et al introduced MYCIN that used rule-based backward chaining to identify microorganisms causing infections. (116) This expert system started with the identification of a certain microorganism as a goal, and reasoned backwards from there, using appropriate rules to determine what facts fit the microorganism. (109) This was the first CDSS system to incorporate an explanation facility and to separate the knowledge from the program, creating the first reusable expert system shell, EMYCIN. (109) INTERNIST-1 was developed by Miller, Pople and Myers to provide diagnostic support for a wide variety of medical illnesses. (117) It was later further developed as the Quick Medical Reference (QMR). Systems like HELP, developed at the University of Utah, and CARE at the Regenstrief Institute used rule-based logic to analyze hospital data and pro-
vide alerts and reminders to physicians. (113) DXplain developed by Barnett et al, and Iliad developed by Warner et al were able to provide diagnostic support for a large set of illnesses based on signs, symptoms and laboratory findings, and are currently used as educational tools. (112, 113)

Currently, CDSSs are widely used to provide automated interpretation of ECGs, arterial blood gas analysis and pulmonary function tests, and automated differential cell counts. (113) For diagnostic decision support, medical expert system developers have abandoned the ‘Greek oracle’ model that used to assume a passive role for the physician. (113) Instead, focus has shifted to development of systems that aid and support the clinical functions of the physician. The goal now is to tap the innate capabilities of both man and machine, and create systems that improve the efficiency of man and machine working together. Practice-critiquing approach was developed by Miller et al in the 1980s to achieve this synthesis. (108)

**Expert Systems – Development and Evaluation**

Conventional models for software development are considered inadequate for expert system development due to several reasons. (109, 118) Expert systems are usually built around semi-structured problem contexts. The System Development Life Cycle (SDLC) model and the related waterfall model assume that the structure of the problem and the nature of the problem domain are fully understood prior to the beginning of the project. This is often not the case with expert systems. The users usually do not fully understand the problem context, the decision-making processes, their preferences or the environmental conditions, and these keep changing throughout the system development.
This led to the adoption of incremental or spiral models for expert system development, in which the system is developed in increments. First proposed by Boehm, each cycle of the spiral model includes planning, knowledge acquisition, coding and evaluation. Another version of the spiral model was proposed by Lee et al, which includes requirements analysis, requirements verification, knowledge acquisition, setting acceptable level of performance, prototyping, verification and validation in each cycle. The advantages of spiral model are that it allows easy prototyping and easy testing of the functional increments in each cycle.

The linear model of expert system development life cycle has been successfully used in several expert system projects. It describes each of the stages of the spiral model in detail, and outlines the tasks associated with each stage. Each cycle of the spiral model includes some or all the tasks from each stage of the linear model. This model is thus a combination of the linear and spiral approaches.

Evaluation of expert systems is a complex process and particular methods used depend on the stage of development of the system. Mengshoel recommends identifying four different concepts for evaluation of knowledge based systems: Checking, verification, validation and usability testing. Checking implies internal evaluation of the KB. Verification involves testing ‘is the system built right’, and usually involves comparing the product to its specifications. Validation involves testing ‘is it the right system’, and usually includes testing the product against real-world problems. Usability testing involves evaluation of the product during real-world usage by actual users.
Table 1

Components of the Linear Model of Expert System Development Life Cycle, and Associated Tasks, According to Giarratano and Riley. (109)

<table>
<thead>
<tr>
<th>Stage</th>
<th>Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td>Feasibility assessment, resource management, task phasing, schedules, preliminary functional layout, high-level requirements</td>
</tr>
<tr>
<td>Knowledge Definition</td>
<td>Source identification, source importance, source availability, source selection</td>
</tr>
<tr>
<td>Source Identification</td>
<td>Source identification, source importance, source availability, source selection</td>
</tr>
<tr>
<td>and Selection</td>
<td>Source identification, source importance, source availability, source selection</td>
</tr>
<tr>
<td>Acquisition, Analysis</td>
<td>Acquisition strategy, knowledge element identification, knowledge classification system, detailed functional layout, preliminary control flow, preliminary user’s manual, requirements specifications, knowledge baseline</td>
</tr>
<tr>
<td>and Extraction</td>
<td>Acquisition strategy, knowledge element identification, knowledge classification system, detailed functional layout, preliminary control flow, preliminary user’s manual, requirements specifications, knowledge baseline</td>
</tr>
<tr>
<td>Knowledge Design</td>
<td>Knowledge representation, detailed control structure, internal fact structure, preliminary user interface, initial test plan</td>
</tr>
<tr>
<td>Definition</td>
<td>Knowledge representation, detailed control structure, internal fact structure, preliminary user interface, initial test plan</td>
</tr>
<tr>
<td>Detailed Design</td>
<td>Design structure, implementation strategy, detailed user interface, design specifications and report, detailed test plan</td>
</tr>
<tr>
<td>Code and Checkout</td>
<td>Coding, tests, source listings, user manual, installation/operations guide, system description document</td>
</tr>
<tr>
<td>Knowledge Validation</td>
<td>Test procedures, test reports</td>
</tr>
<tr>
<td>Formal Test</td>
<td>Test procedures, test reports</td>
</tr>
<tr>
<td>Test Analysis</td>
<td>Results evaluations, recommendations</td>
</tr>
<tr>
<td>System Evaluation</td>
<td>Results evaluation, recommendations, validation, interim or final report</td>
</tr>
</tbody>
</table>

Technology in Emergency Medical Services

The use of technology in EMS for patient care documentation and decision support is growing. The EMS System Act of 1973 mandated that every EMS system “provide for a standardized patient record keeping system meeting appropriate standards.” (8) However, standards were difficult to define, and those that were, like the EMS Minimum Dataset (MDS), were not very widely used. (124) Electronic data collection in EMS be-
gan in earnest in the late 1980s and early 1990s as an attempt to collect information regarding quality of patient care for quality improvement initiatives. Early efforts included manual entry into computers by data entry specialists and optically scanned ‘lotsa-dots’ forms. (125-128) In 1994, NHTSA proposed a Uniform EMS Dataset consisting of 81 data elements, (129) that has been further refined by the National EMS Information System (NEMSIS) project, (130) and is currently in version 2.2.1. (131) Several efforts towards the development of electronic patient care reports (ePCRs) have been reported, (132-136) and several commercial products are available. (137) In the 200 largest US cities, 88% of agencies report that they are using some kind of electronic documentation. However, only 24% are able to input data throughout the run at the patient’s side. (93) Others have only paper-based documentation, or wait until they reach the hospital or base station to input the data.

The use of CDSS in EMS has been much more limited. Computerized systems have been used to assist audits and quality assurance initiatives, and have been shown to generate more reports than manual review, (127, 128) but are estimated to be more expensive. (104) Field checks associated with computerized data entry systems have been shown to improve compliance with data entry protocols. (126) Other systems, such as Battlefield Medical Information Systems Tactical (BMIST), offer diagnostic and decision support aids and reference material on the same handheld device as the ePCR. (132) One system built alerts and reminders into the ePCR and reported that protocol compliance improved from 50-60% to 80-100%. (133)
Vision

This project envisions an expert system tool to provide feedback to EMS personnel after the ambulance run is completed and documented. It is named ‘Feedback Expert system for EMS Documentation’ (FEED). The input for the system will be obtained from an electronic documentation system. The inference engine will analyze this input using the rules in the knowledge base and produce output in the form of a checklist of expected documentation. This system has potential for use as practice review tool, and can provide incremental, focused education to the EMS personnel. This would improve their knowledge about EMS care protocols and their confidence in the applicability of these protocols. It could also be used for quality assurance by EMS agencies. It could also help EMS personnel identify errors, and give them an option to report those to anonymous error reporting systems in agencies that have them in place. All this has the potential to improve compliance with protocols, reduce errors and improve patient outcomes.

Summary

In summary, there are errors in patient care, deviations from patient care protocols and errors in documentation of patient care in EMS. These can lead to poorer patient outcomes, difficulty in performing research, and litigation against the agency. Apart from system factors that make EMS error prone, individual factors such as knowledge, skills and attitudes of EMS personnel towards protocols are also responsible. Among the several remedies being proposed or used to ameliorate these problems, most have issues with cost, while some are unsuited to the EMS workflow. Practice critiquing systems providing after-the-fact feedback regarding the treatment plan of the care provider have the po-
tential to provide a cheap and effective solution for improving knowledge and attitudes of EMS personnel, thereby improving compliance, reducing errors and ultimately improving patient outcomes.
METHODS

Development Model

The combination of spiral and linear models described above was followed during the development of FEED. The linear model was adapted to match the availability of time and personnel for this project (Table 2). The system went through three cycles of iterative development. Table 3 specifies the tasks performed in each iteration.

The goal of the first iteration was to produce a prototype with very limited functional capabilities. Most of the tasks in the planning stage were performed during this iteration.

Table 2

Components of the Modified Linear Model Followed During Development of FEED, and Associated Tasks

<table>
<thead>
<tr>
<th>Stage</th>
<th>Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td>Feasibility assessment, resource management, high-level requirements, preliminary functional layout</td>
</tr>
<tr>
<td>Knowledge Definition</td>
<td>Source identification, source selection, knowledge acquisition, detailed functional layout, preliminary control flow, requirements specifications</td>
</tr>
<tr>
<td>Knowledge Design</td>
<td>Knowledge representation, detailed control structure, internal fact structure, preliminary user interface, initial test plan</td>
</tr>
<tr>
<td>Code and Checkout</td>
<td>Coding, tests, source listings</td>
</tr>
<tr>
<td>Knowledge Base Evaluation</td>
<td>Test cases</td>
</tr>
<tr>
<td>Test Cases</td>
<td>Local syntax checking, consistency checking</td>
</tr>
<tr>
<td>Checking</td>
<td>Informal verification, expert verification</td>
</tr>
<tr>
<td>Verification</td>
<td>Validation</td>
</tr>
<tr>
<td>Validation</td>
<td></td>
</tr>
</tbody>
</table>
Table 3

Iterations of the Spiral Model, and Tasks from Linear Model Followed in Each Iteration during Development of FEED

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Stage</th>
<th>Tasks Performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>First</td>
<td>Planning</td>
<td>Feasibility assessment, resource management, high-level requirements, preliminary functional layout</td>
</tr>
<tr>
<td></td>
<td>Knowledge Definition</td>
<td>Source identification, source selection, knowledge acquisition, requirements specifications</td>
</tr>
<tr>
<td></td>
<td>Knowledge Design</td>
<td>Knowledge representation, internal fact structure</td>
</tr>
<tr>
<td></td>
<td>Code and Checkout</td>
<td>Coding, tests</td>
</tr>
<tr>
<td></td>
<td>Knowledge Base</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Evaluation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Test Cases</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Checking</td>
<td>Local syntax checking</td>
</tr>
<tr>
<td></td>
<td>Verification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Validation</td>
<td></td>
</tr>
<tr>
<td>Second</td>
<td>Planning</td>
<td>Preliminary functional layout</td>
</tr>
<tr>
<td></td>
<td>Knowledge Definition</td>
<td>Knowledge acquisition, detailed functional layout, preliminary control flow, requirements specifications</td>
</tr>
<tr>
<td></td>
<td>Knowledge Design</td>
<td>Knowledge representation, detailed control structure, internal fact structure, preliminary user interface, initial test plan</td>
</tr>
<tr>
<td></td>
<td>Code and Checkout</td>
<td>Coding, source listings</td>
</tr>
<tr>
<td></td>
<td>Knowledge Base</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Evaluation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Test Cases</td>
<td>Test cases</td>
</tr>
<tr>
<td></td>
<td>Checking</td>
<td>Local syntax checking</td>
</tr>
<tr>
<td></td>
<td>Verification</td>
<td>Informal verification, expert verification</td>
</tr>
<tr>
<td></td>
<td>Validation</td>
<td></td>
</tr>
<tr>
<td>Third</td>
<td>Planning</td>
<td>Preliminary functional layout</td>
</tr>
<tr>
<td></td>
<td>Knowledge Definition</td>
<td>Knowledge acquisition, detailed functional layout, requirements specifications</td>
</tr>
<tr>
<td></td>
<td>Knowledge Design</td>
<td>Preliminary user interface, initial test plan</td>
</tr>
<tr>
<td></td>
<td>Code and Checkout</td>
<td>Coding, source listings</td>
</tr>
<tr>
<td></td>
<td>Knowledge Base</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Evaluation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Test Cases</td>
<td>Test cases</td>
</tr>
<tr>
<td></td>
<td>Checking</td>
<td>Local syntax checking, consistency checking</td>
</tr>
<tr>
<td></td>
<td>Verification</td>
<td>Informal verification</td>
</tr>
<tr>
<td></td>
<td>Validation</td>
<td>Validation</td>
</tr>
</tbody>
</table>
Different options and components were studied and compared. Only a limited amount of knowledge acquisition and representation was done at this stage. Coding took a long time as the new expert system building tools were studied. The testing was limited to informal tests during the coding itself.

The goal of the second iteration was to build a functional system, although the user interface was not developed to the full extent. The design of the system was specified in much more detail, and accordingly, more extensive knowledge acquisition, representation and coding was done. Informal verification was done with test data, and formal verification in an expert focus group.

The goal of the third iteration was to add more components to meet the final requirements specifications. The user interface and knowledge management tools were developed further. Changes made to the KB were limited to those suggested in the verification focus group in the second iteration. The KB was not verified again, but consistency checking and expert validation was performed in this iteration.

Stage I – Planning

Most of the tasks in this stage were performed in the first iteration. Preliminary functional layout was developed in each stage to describe the purpose and high-level functions of the system to be achieved at the end of the stage. (109)

Feasibility Assessment

Extensive literature search was performed to study the importance of the problem. As discussed above, there are problems of protocol deviations, incomplete documentation
and patient errors in EMS, and a feedback system has the potential to provide solutions for them. In addition, criteria were developed (based on Liebowitz, Giarratano and Martin) to judge whether expert system technology was appropriate for the problem domain. (109, 111, 138)

1. **Well-bounded task:** The problem to be solved by the expert system should have a well-bounded domain, to avoid combinatorial explosions and problems with coordinating expertise in different domains. (109, 111) Also, effective problem solving requires an extensive knowledge base, but current expert system technology is limited in the amount of knowledge it can represent, store and manipulate. (138) Therefore, a narrow domain is best suited for expert system technology. In this case, the EMS protocols are fairly restrictive, and the EMTs are limited in the medications they can give the patient and the procedures they can perform. Also, most of the knowledge represented in EMS protocols is procedural and focused on specifying tasks instead of diagnoses, which reduces the number of clinical concepts that need to be represented.

2. **Symbolic processing:** If the problem can be solved using conventional programming, expert systems should not be used to solve it. (109, 111) Expert systems are more suited for unstructured problems that cannot be solved algorithmically. (109) EMS is based on rules and algorithms, but there are plenty of exceptions to make the problem unstructured.

3. **Less than a few weeks to solve:** The problem should not be too complex, increasing the risk of combinatorial explosion. The current state of the art of expert system technology should be able to handle the scope of the problem.
Review of each EMS run sheet takes 5-10 minutes when done manually by a QA/QI expert.

4. **Task performed frequently:** The usefulness of the expert system is maximized when the task is performed frequently. (111) The feedback system can potentially be used by EMS personnel for every EMS run, and the QA officers can use it as frequently as needed.

5. **Significant difference between the best and worst performers:** If there is a large difference of performance between the expert and the novice, expert system will be helpful in bringing the expertise of the expert to the novice. (111) As discussed above, EMTs have wide variations in performance.

6. **Expertise is needed and desired:** There must exist the need and desire for the expertise provided by the expert system. (109, 138) The need for the system has been discussed above. In a survey of paramedic students and practicing EMTs, respondents gave an average rating of 3.9 on a scale of 1-5 to their perceived need of an after-the-fact feedback system. (139)

7. **Test data available:** Availability of test data is very helpful for validation and testing of the expert system. Test data was potentially available from several sources including the Alabama State EMS Information System (EMSIS) database.

8. **General consensus:** The expert system should represent general consensus among experts in the problem domain. The Alabama State EMS Protocols were built as a result of general consensus among medical experts.
9. *Expert participation*: There must be at least one domain expert who is willing to participate and is able to explain the knowledge to the knowledge engineer. Mr. Charles Godwin, NREMT-P, EMS faculty at the University of Alabama at Birmingham (UAB), agreed to provide expert advice during development.

Based on the above criteria, the problem domain was judged to be suitable for expert system development.

*Resource Management*

During this task, the hardware and software requirements for system development were considered and the required components were procured and assembled.

It was decided early on that the system would be built using an expert system shell, instead of programming in a high-level language such as Prolog or LISP. Also a rule-based approach was chosen for use in designing the knowledge base. There were two main reasons for this. (111) First, the existing knowledge base of EMS (the Alabama State EMS patient care protocols) is in the form of procedural rules. A rule-based expert system would be intuitively closer to the existing knowledge. Also, EMS rules are often context independent. For instance, the first step to stop external bleeding is compression, regardless of the site or cause of bleeding.

A set of 29 criteria was established for evaluating commercially available expert system shells. (111) (Table 4) A list of historical and currently available expert system shells was prepared from various sources. (112, 140-142) A total of 66 shells were identified from these sources. Preliminary information about the shells that was available in the above sources was collected in a Microsoft Excel datasheet. In the second stage of
Table 4

Criteria for Evaluation of Available Expert System Shells

<table>
<thead>
<tr>
<th>Component</th>
<th>Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge Base</td>
<td></td>
</tr>
<tr>
<td>Data Dictionary</td>
<td>Ability to import existing NHTSA EMS data dictionary (available in XML format)</td>
</tr>
<tr>
<td>Knowledge representation</td>
<td>Knowledge representation format (Rule-based, frame-based, semantic networks, etc.)</td>
</tr>
<tr>
<td></td>
<td>Mechanism to handle uncertainty (Bayesian approach, belief functions, fuzzy logic, etc.)</td>
</tr>
<tr>
<td></td>
<td>Number of conditions allowed in the IF part of rules (ability to handle complex rules)</td>
</tr>
<tr>
<td></td>
<td>Explanation of rules can be appended to rules (Y/N)</td>
</tr>
<tr>
<td></td>
<td>Free text can be added to rules (Y/N)</td>
</tr>
<tr>
<td>Rule Editor</td>
<td>Rule editor can be accessed without installing the whole development suite (Y/N)</td>
</tr>
<tr>
<td></td>
<td>Rules can be expressed in English (Y/N)</td>
</tr>
<tr>
<td></td>
<td>Ease of using rule editor (on a scale of 1 to 5)</td>
</tr>
<tr>
<td></td>
<td>Hard coding required to enter, edit or update rules? (Y/N)</td>
</tr>
<tr>
<td></td>
<td>Ease of testing of rules (on a scale of 1 to 5)</td>
</tr>
<tr>
<td></td>
<td>Ease of editing/ updating rules (on a scale of 1 to 5)</td>
</tr>
<tr>
<td>Inference Engine</td>
<td>Ability to obtain inputs from electronic documentation tools</td>
</tr>
<tr>
<td></td>
<td>Allow both forward and backward chaining (Y/N)</td>
</tr>
<tr>
<td>User Interface</td>
<td>Ease of learning and using the user interface (on a scale of 1 to 5)</td>
</tr>
<tr>
<td></td>
<td>WYSIWYG editor for development of user interface (Y/N)</td>
</tr>
<tr>
<td>Hardware requirements</td>
<td>Lowest PC configuration required to run final product</td>
</tr>
<tr>
<td></td>
<td>Recommended PC configuration required to run final product</td>
</tr>
<tr>
<td></td>
<td>Final product is able to run on mobile devices (Y/N)</td>
</tr>
<tr>
<td>Other</td>
<td>Ease of learning and using the shell for development (on a scale of 1 to 5)</td>
</tr>
<tr>
<td></td>
<td>The shell is currently supported (Y/N)</td>
</tr>
<tr>
<td></td>
<td>The shell is regularly updated (Y/N)</td>
</tr>
<tr>
<td></td>
<td>Training is available for learning the shell (Y/N)</td>
</tr>
<tr>
<td></td>
<td>Tutorials are available for learning the shell (Y/N)</td>
</tr>
<tr>
<td></td>
<td>The shell has extensive documentation (Y/N)</td>
</tr>
<tr>
<td></td>
<td>Cost of installing the shell for development</td>
</tr>
<tr>
<td></td>
<td>Cost of installing the final product</td>
</tr>
<tr>
<td></td>
<td>Cost of running the final product</td>
</tr>
<tr>
<td></td>
<td>Estimated response time for an analysis requiring firing of 100 rules</td>
</tr>
</tbody>
</table>
review, it was ascertained whether the shell is currently supported and regularly updated. Google search was extensively used in this process. The next stage of review involved visiting the websites of currently active shells, and obtaining more information on knowledge representation, inference engine and cost of development. Free demos and documentation was downloaded if available. Free software was usually easier to evaluate since documentation was freely available.

Finally, thirteen expert system shells were chosen for further review, on the basis of the following important features:

1. Availability of extensive documentation and free software or demo version
2. Currently supported and regularly updated
3. Focus on rules as a basis for knowledge representation
4. Mechanism for handling uncertainty
5. Rules represented graphically or in natural language
6. Extensions available that increase the flexibility of the shell

These thirteen shells were: ACQUIRE, BABYLON, CLIPS, ES, flex, FramerD, Goldworks III, HaleyRules, K-Vision, Mike, OPS/R2, Prolog+Logic Server, and XpertRule Knowledge Builder.

In the last stage of review, C Language Integrated Production System (CLIPS) was chosen as it fits with the above criteria very well. CLIPS is an expert system development tool developed by the National Aeronautical and Space Administration (NASA) and now maintained and updated as an open source project. (143) It represents knowledge in the form of rules and facts, and has additional support for object-oriented representation of knowledge. It is flexible, easy to learn and there are several tools available
that allow integration with other development platforms such as .NET and JAVA. Visual Basic .NET was chosen as the development platform for user interface, as it allows rapid development of interfaces using Windows Forms components. The interface between CLIPS and .NET platform was provided by the CLIPS .NET application programming interface (API), that consists of dynamic link libraries (dlls) built by ProAI in the form of a .NET wrapper for embedding CLIPS code in .NET applications. (144)

These components were installed on a Hewlett-Packard workstation with dual Intel Xeon 3.2GHz processors and 1GB RAM, with Microsoft Windows XP Service Pack 2 as the operating system. Several test programs were built to get familiar with the CLIPS platform and its integration with .NET. In addition, a Microsoft SQL Server was installed to provide a source of data for the system. Scripts available from NEMSIS website were used to build a database according to the specifications of NHTSA’s Uniform EMS Dataset version 2.2.1 (NEMSIS v2.2.1). (145)

High-Level Requirements

The following development guidelines were adapted from Liebowitz to provide high-level requirements for the development of FEED (111):

1. *It must be easy to develop:* The expert system shell or tool must allow easy development of the system, as time and personnel were limited and the knowledge engineer had no prior experience of programming in expert system languages. The chosen tool would have adequate documentation, availability of teaching resources, and would be currently supported by the developers at pre-
sent, so that it would be easy to learn system development using that tool. A suitable tool was found meeting this requirement.

2. *It must have a focus on learning:* One of the main objectives of this tool would be to foster learning. The feedback from FEED would have a non-accusatory tone and there would be positive feedback along with negative feedback. There would be detailed explanations for each rule, and links to allow the user to explore the issue further, if desired.

3. *It must leverage existing technologies and vocabularies:* For rapid development, the system would be developed using existing technologies as far as possible. An expert system tool would be identified to build the knowledge base and inference engine using a high-level language. Existing vocabularies for EMS would be used during the development of the system, primarily NEMSIS v2.2.1. (130)

4. *It should be comprehensive:* The system should be able to provide feedback for a comprehensive set of patient conditions, and be able to consider many types of errors in the analysis. The existing Alabama EMS protocols would form the initial basis for the knowledge base. (146) For the purpose of this project, the system would focus only on cardiac cases. (The functionality for the rest of the cases can be added at a later stage.) Most of this knowledge could be represented in the form of rules, but the system must be able to handle uncertainty in the form of incomplete or unreliable inputs, as well as the uncertainty inherent in medical decision making. A suitable developmental tool would be chosen that meets these requirements.
5. *It must be able to integrate with existing documentation tools:* It would be unreasonable to expect EMS personnel to re-enter the run information into the expert system in addition to their regular documentation. FEED would be compatible with inputs from existing electronic documentation tools, most importantly the extensible markup language (XML) schema definitions (XSDs) according to the NEMSIS specifications. (147)

6. *It must be easy to learn and use:* The expert system should be easy to learn and use since extensive training sessions would not be feasible for rural EMS agencies that would be the main beneficiaries from this project. For this purpose, the interface would be as simple as possible, with ample documentation and help features. In addition, the rule editor would also be easy to learn and use. It would be able to handle rules in English language and require no coding for editing of rules. This would allow continued development of new rules, and customization of rules by EMS administrators to suit the particular education needs of their agencies.

7. *It must have a low cost of installation and maintenance:* The expert system would be able to run on existing hardware in the agencies. This would allow wider acceptance of the tool. Also, it would be able to access the rule editor and edit rules without installing the whole expert system development suite.

8. *It must be able to work in a disconnected environment:* The system must be able to work in an environment where the availability of intranet or internet cannot be taken for granted. The system would be able to install on mobile computing devices, and all components would reside together on each device.
9. *It must be easy to update rules:* The knowledge base must be easy to update. There would be provision for easy checking for conflicts among rules, using ‘what-if’ scenarios, for example.

10. *It must be easy to share rules:* The system would allow easy exporting and importing of rules, to enable sharing of rules among EMS agencies. Also, there would be a rule repository where widely accepted rules would be stored and be available for download.

**Preliminary Functional Layout**

The following vision for the final layout of FEED was developed at an early stage:

FEED would allow EMS personnel, QA personnel, and EMS administrators to obtain feedback on past cases stored in the agency’s patient care documentation database. The completed electronic report of a particular case would be pulled from the database and analyzed using the rules in the knowledge base. The knowledge base itself would be rule-based and primarily based on Alabama State EMS protocols, (146) with inputs from Mr. Charles Godwin (hereafter referred as ‘the Expert’). For incorporating uncertainty in rules, two kinds of scores would be associated for each rule in the knowledge base. The *Likelihood Score (LS)* would represent the probability with which a fact or a rule is believed to hold true by the Expert. The *Seriousness Score (SS)* would represent the seriousness of the worst possible patient outcome if the rule was violated. Each rule would also have extensive explanations, and links for further exploration associated with it.
If the input from the database was deemed to be insufficient for making meaningful conclusions, the user would be asked to enter more data related to the case. Once the analysis was complete, the user would be shown a checklist of expected documentation, on which actual documentation would be checked off. The LS and SS for all rules fired for the particular item in the checklist would be displayed along side. Also, the user would be able to explore further the chain of reasoning and the detailed explanation for each rule. These explanations would have links pointing to resources for further exploration.

In the first iteration, the preliminary functional layout was very limited. Fourteen rules for one particular cardiac protocol (ventricular fibrillation) were acquired with the help of the Expert and added to the knowledge base. The knowledge base included definitions for the fact structure, declarations of initial values of some facts, rules for determining whether particular medications or procedures were appropriate in this case, and rules for determining whether additional input is needed from the user. In this particular protocol, the system asked the user the stage at which the resuscitation protocol was stopped. Additional rules compared the actual documentation with the suggestions, and printed suitable feedback for the user. This prototype was written completely in the CLIPS language. Integration with .NET was not achieved at this stage. Two test cases were made as text files containing relevant data points that could be loaded into the CLIPS engine. All components were loaded into the standard CLIPS engine and the engine was run. The CLIPS engine produced text output in its dialog window representing the feedback for the particular case. This included the suggestion, the likelihood score
and seriousness score associated with the rule (discussed below), and whether the actual
documentation matched the suggestion.

The version of FEED developed by the end of the second iteration was able to
provide feedback on eight cardiac protocols. It allowed the user to login and view a list of
cases according to his/her access level. It then obtained the chosen case from an elec-
tronic database, analyzed it using the rules in the KB, asked the user for additional input
if needed, and provided feedback on an hypertext markup language (HTML) page. It was
able to handle complex interactions between rules. However, at this stage, the feedback
did not include explanations and reasoning behind the suggestions. Also, the rule editor
had limited functionality; it only supported building new rules according to the arguments
specified.

The goal of the third iteration was to meet all guidelines and requirements specifi-
cations not met in earlier versions. Additional functionality was added to FEED and the
code was extensively commented. Errors noted in previous iteration were corrected.
Feedback was enhanced to include explanations and reasoning behind the suggestions. A
database was built to store the feedback and it was made available for future review.
Also, a rule parser was built to check for consistency checking.

Stage II – Knowledge Definition

Source Identification and Selection

As mentioned above, the sources of knowledge for FEED were Alabama State
EMS patient care protocols, (146) and input from the Expert. The protocols have been
developed by the Alabama Department of Public Health’s EMS Division through expert
consensus. Each protocol is based on the primary symptom of the patient, or the impression of the EMS personnel, or in case of certain cardiac protocols, the ECG rhythm; in some cases this is not specifically mentioned in the protocol, for example, in the Cardiac Chest Pain protocol, it is not specified what constitutes ‘cardiac’ chest pain. Within each protocol, there are several action items relating to obtaining history specific to the protocol, looking for specific physical signs, performing procedures and giving medications to the patient. Some protocols have decision forks that determine the subsequent treatment plan. The older versions of the protocol included a flowchart version for each protocol, the latest version (December 2006) does not, however.

Knowledge Acquisition

A preliminary set of rules were built by the knowledge engineer based on the protocols before the knowledge acquisition session with the Expert. These rules were shown to the Expert, who then suggested changes and additions. If there was uncertainty associated with the inference a rule was making, different options were considered: Assigning a low LS to the rule, introducing additional conditions that made the inference more specific, or dropping the rule altogether. The Expert also assigned LS and SS scores to each rule according to his judgment. These are discussed in more detail below. The following questions were found to be useful in eliciting LS: “Given the antecedents of this rule are true, in approximately what percentage of such cases does the rule hold true?” “What conditions could cause EMS personnel to not be able to do as the rule says?” “What could be the exceptions to this rule and how frequently do they occur?” The discussion for each rule was also recorded for use in the explanations associated with the rule. For
some cardiac protocols, additional input was obtained from Stacey McCoy, another EMS faculty who has specific expertise for these protocols: She teaches paramedic classes and advanced cardiac life support (ACLS) courses at UAB. In the third iteration, knowledge acquisition was limited to the improvements suggested in the expert verification in the second iteration. LS and SS of the rules were updated, and additional changes were made as suggested by the experts.

**Detailed Functional Layout**

Figure 1 shows the components that were incorporated into FEED by the end of the second iteration. The following functions were supported:

1. **Login:** On startup, the system asked the user for the username and password. This was compared to the entry for the user in the user database, and the user authenticated if the information provided matched that in the database.

2. **Available Cases:** The system recognized three access levels: EMT, QA/QI personnel and administrator. A user logged in as EMT would have access only to cases entered by him/her. QA/QI personnel and administrator would be able to access all cases. Administrators would have additional access to administrative tools, such as the rule editor.

3. **Documentation Database:** A Microsoft SQL Server database was built to simulate an actual database for EMS patient care reports. This was built using database building scripts available from the NEMSIS technical center. This database was manually filled with test cases using Microsoft SQL Server Management Studio.
Figure 1. Components of FEED at the end of second iteration. Solid lines indicate process flow, dashed lines indicate data flow.
4. **Case Analysis:** After the user chose the case for feedback, the data elements needed by FEED for analysis were loaded from the database. The case was then analyzed using the rules in the KB. If additional user input was deemed necessary, a message window appeared as a popup that asked the user to enter the value of the required data element. The system included controls that checked for and prevented multiple messages for the same data element. Further analysis continued until all necessary data elements had been asked, and all appropriate rules had been fired.

5. **Handling rule interactions:** If the same rule fired more than once, the resultant suggestions were compared and only the one that came about later in the analysis was kept, assuming that this newer suggestion resulted as additional data became available and therefore represented a better picture of the case. Also, if an identical suggestion resulted from more than one different rule, its LS and SS were combined using appropriate calculations (described below in more detail).

6. **Feedback:** Several different kinds of feedback could be given, suggesting that the user should have recorded any value, a particular value, a particular set of values, or a particular range in a given data element, or recorded it a particular number of times or with a particular frequency. In addition, the system would suggest that a particular value or a set of values *not* be recorded in the data element. Each particular feedback was presented to the user along with the LS and SS associated with the feedback.
7. **Matching Feedback with Actual Documentation:** At the end of the second iteration, the system was able to match certain kinds of suggestions with what was actually recorded by the user. This feature was not available for suggestions related to repeated data fields for which recording at a particular frequency or total number of times was suggested.

8. **Rule Editor:** A Windows Form was built that allowed quicker rule-building with less margin of error than manual coding in the CLIPS language. The knowledge engineer specified the antecedents (if conditions) and the consequents (then statements) for the particular rule, and the form generated the rule in CLIPS code, along with the documentation related to the rule.

9. **Vocabulary:** The system used codes specified by NEMSIS v2.2.1 for internal representation of variable names and their values, including KB rules. However, these codes were translated to descriptive names when interacting with the user, for example, when asking for additional input, and when displaying the feedback at the end of the analysis.

Figure 2 shows the components that were present in FEED at the end of the third iteration. The following functions were supported in addition to those already present from the second iteration:

1. **Feedback:** The feedback page was further enhanced to provide explanations for LS and SS along with the scores for each suggestion. The users could also access the reasoning behind the suggestion, including what rules fired to generate the particular suggestion.
Figure 2. Components of FEED at the end of the project. Solid lines indicate process flow, dashed lines indicate data flow.
2. **Output Database**: The feedback generated for each session was stored in a separate database. It was made available for future browsing through the initial login page.

3. **Rule Parser**: A rule parser component was added to the rule editor. It accessed the rules in the KB, and for each rule, looked for other rules with similar antecedents. It helped in the consistency checking of the KB, as described below.

### Preliminary Control Flow

An overview of the process flow and data flows of FEED is provided in Figure 1. This section deals with the control flow specific to the inference engine of FEED. After a case was chosen for analysis, a session of CLIPS rule engine was initiated. The template definitions for facts and class definitions for objects were loaded into this session, followed by the rules in the KB, including both medical rules and rules for program functions. The data for the case was obtained from the database and loaded into the session. The rules were then executed. Among other things, this resulted in generation of objects that represented FEED’s suggestions for the particular case. At all times, the engine kept looking for duplicate objects, and dealt with them appropriately. When execution paused (all executable rules had been executed), the session was examined to ascertain whether the system needed input from the user. If so, the user was asked for input, his/her response was loaded into the session, and the rules were executed again. This was continued until no more input was needed from the user and all executable rules had been executed. Another set of rules was then loaded into the session, which compared the feedback with actual documentation and ascertained whether they matched or not. Each sug-
gestion object was then converted to HTML text and the whole feedback was displayed to the user in an HTML file.

Since the CLIPS engine was not readily visible to the user, its error and warning messages were not directly visible either. A special module allowed the .NET base program to catch these messages from the CLIPS engine through its Input/Output Router functions. These messages were then displayed to the user in a text box in a message window, allowing more efficient debugging of the system.

**Requirements Specifications**

The minimum requirements for the first iteration were largely specified around the demonstration of proof of concept. The prototype at the end of this stage was able to use rules written in CLIPS to analyze case-specific data (also inputted as CLIPS code) and provide suitable feedback related to a limited number of medications and procedures for the ventricular fibrillation protocol.

The following requirement specifications were met by the system at the end of the second iteration:

1. **Data Source:** The system was designed to be able to obtain data input from a database built according to the NHTSA Uniform Pre-hospital EMS Dataset v2.2.1 specifications from NEMSIS (NEMSIS v2.2.1). (131)

2. **Data Elements:** The focus of this project was on providing feedback that is clinically relevant. Therefore, only data elements considered clinically relevant for this project was imported from the database. These included all data elements collected by Alabama State EMS Database in the following sections
of the database: Patient, Situation, Situation/CPR, Assessment and Intervention; and additional data elements as needed by FEED.

3. **Data Vocabulary:** All data elements and their values were referred in FEED using the standard codes specified in NEMSIS v2.2.1. Using the conversion tables provided with the database, these values were changed to their corresponding names for displaying to the user where needed.

4. **Clinical Focus:** The system built as part of this project was limited to adult cardiac protocols. The system was able to provide feedback for cases that have any of the following as the Provider’s Primary Impression: Cardiac arrest, Cardiac rhythm disturbance, and Chest pain/discomfort; and any of these as the Protocol: Asystole, Bradycardia, Cardiac Arrest, Chest Pain/Suspected Cardiac Event, Pulseless Electrical Activity (PEA), Supraventricular Tachycardia, Ventricular Fibrillation/Pulseless V-Tach, and Ventricular Tachycardia. Cardiac protocols were chosen because they offer a good mix of different kinds of rule patterns: Decision to follow the particular protocol specific or vague, number of decision forks, time-dependent or independent, and stand-alone or interacting with other protocols. The knowledge structure supporting these protocols will therefore be able to support other protocols when the knowledge base will be expanded to include them.

5. **Clinical Detail:** There are several ‘action items’ in the protocols that include collection of medical history, signs and symptoms, administering medications and performing procedures. The KB included at least one rule determining the relevance of displaying each action item in each protocol as part of the feed-
back. Of the action items displayed, some were available for matching with
the actual documentation. Others were not, either because they were unavail-
able in structured format in the database, or they were too complex to be in-
cluded in this iteration.

6. **Complexity of inference:** The system was able to use all relevant information
available in structured format in the case documentation. The narrative com-
ponent of the PCR was not used in the analysis.

7. **Additional Input:** Since the information solicited from the user is likely to be
available in the narrative portion of the documentation, it amounts to re-entry
of data from the user’s perspective. It was decided to limit such additional in-
put solicited to 10 data elements for each protocol.

8. **Graceful Exit:** The system was able to “degrade gracefully” at the limits of its
capabilities. If the knowledge base was not capable of handling a particular
case, the system provided no feedback instead of erroneous feedback.

9. **Quality of Code:** Descriptive variable names were used and a consistent style
was used. At this stage, the code was not commented adequately. The knowl-
edge base had each rule described in medical language, in addition to the
standard coding vocabulary used for analysis. The code of the inference en-
gine was designed to be understandable by another programmer, and the
knowledge base by another knowledge engineer.

10. **Quality of Knowledge Base:** The rules in the knowledge base had appropriate
LS and SS (after verification in the expert focus group). The knowledge base
did not have any problems with local syntax.
11. Quality of Feedback: Each LS and SS was associated with a reasonable explanation, outlining the reasons why the score was high or low.

Additional requirements specifications were developed for the third iteration of the project. Those were in addition to the specifications already achieved during the second iteration.

1. Quality of Code: Comments were added throughout the code of the inference engine as well as knowledge base.

2. Quality of Knowledge Base: The rules in the knowledge base had appropriate LS and SS. The knowledge base did not have any problems with local syntax or consistency.

3. Quality of Feedback: Each LS and SS was associated with a reasonable explanation, outlining the reasons why the score was high or low.

Stage III – Knowledge Design

Knowledge Representation and Internal Fact Structure

The KB of FEED was written in CLIPS. It primarily consisted of rules, facts and objects. Rules were used to represent expert knowledge. LS and SS were specified to represent the uncertainty associated with the rule. However, in the first iteration, there were no rules for combining LS and SS for facts produced by different rules. Suggestions, actual documentation and intermediate concepts were represented as facts. Each fact had a slot called ‘var-name’ that represented the name of the data element, and another slot ‘value’ to store its value. In this iteration, suggestion and actual documentation for a particular data element (for example, the procedure ‘defibrillation’) were stored in different
slots of the same fact. This approach was abandoned in later iterations, as it led to problems with repeated rule activations whenever the fact was modified.

For the second and third iterations, a more extensive knowledge representation format was used.

Facts and objects were represented in the Object-Attribute-Value (OAV) format that is commonly used for expert system KBs, (109, 148) where each object had certain attributes or properties, and each attribute could have a certain value. For example, for a blood pressure recording, ‘vitalsign’ could be the object, whose ‘name’ attribute had value ‘SystolicBP’, ‘recording’ attribute had value ‘146’ and ‘datetime’ attribute had value ‘2/4/2007 21:34’. This could be represented in CLIPS as a fact: (vitalsign (name SystolicBP) (recording 146) (datetime 2/4/2007 21:34)), or as an object: (object (is-a VI-TALSIGN) (name SystolicBP) (recording 146) (datetime 2/4/2007 21:34)). FEED had four main kinds of facts and objects: Facts storing information about rules, facts storing values of data elements from actual documentation, objects representing medical concepts, and objects representing suggestions for recommended documentation. Each instance of both the types of objects had certain LS and SS associated with it, which represented, respectively, the likelihood that the particular attribute of the object would have that particular value, and the severity of consequences of not having that particular value. Other minor kinds of facts included those storing lookup values for NEMSIS codes, and those representing the need to ask the user for additional input.

Rules usually had certain OAV triplets in the Left Hand Side (LHS) of the rule, and were ‘activated’ if the objects satisfying the constraints of these OAV triplets existed in the session of the CLIPS rule engine. (Figure 3) Activated rules fired sequentially and exe-
cuted the commands on the Right Hand Side (RHS) of the rule, usually leading to making a new instance of an object representing a suggestion. FEED used the LS of the OAV triplets in LHS of the rule for computing the LS of the OAV triplet in RHS. In addition, each rule itself had LS and SS associated with it, which influenced the calculation of LS and SS of the RHS OAV triplet. There are standard mathematical formulas for these computations that are discussed below. (109, 148)

Figure 3. A typical CLIPS rule with antecedent conditions, which, if satisfied, will ‘fire’ the rule, leading to execution of the consequent commands. Translated to natural language, this rule says, ‘If a cardiac rhythm of supraventricular tachycardia was recorded, and patient is an adult, then it is suggested that adult tachycardia protocol be followed; FEED has assigned a LS of 7 and SS of 8 to this rule.’
While the concept of SS is easy to understand, LS needs more discussion. As described above, LS indicated how much confidence the Expert holds in the rule. It is conceptually equivalent to the ‘Certainty Factor’ used to represent uncertainty in MYCIN. It is made up of two components: For a rule “IF X THEN Y”, the expert has some confidence that Y holds true for every incidence when X holds true; and the expert also has some confidence that Y is not true even if X holds true. LS is an integer from 1 to 10 representing the belief in Y minus the disbelief in Y, if X holds true. (109) Represented mathematically,

$$LS (Y, X) = MB (Y, X) - MD (Y, X)$$

Where:

- LS is the likelihood score for the hypothesis Y due to evidence X
- MB is the measure of increased belief in Y due to X
- MD is the measure of increased disbelief in Y due to X

In the context of this system, LS thus represents the Expert’s opinion of the net number of times Y holds true if X holds true. If X and Y are very strongly related, LS will be very high. On the other hand, if the Expert can envision a number of scenarios in which Y is not true even if X is true, it will lower the LS of the rule. In EMS, these scenarios are most likely to be lack of time, combative or uncooperative patient, and other situations in which data collection is hampered. Also, rules that represent standard of care in EMS are likely to receive high LSs. The final LS, therefore, is the result of a subjective evaluation of the rule by the Expert, considering the possibility of Y being true, given that X is true. For example, for a rule – IF Cardiac Chest Pain Protocol was followed

THEN it is suggested that Nitroglycerin be given to the patient – the Expert considers the
group of cases in which cardiac chest pain protocol is followed, and subjectively assesses the probability of the patient getting nitroglycerin, and assigns a suitable LS (in this case 10/10).

Final LS and SS for suggestions were calculated using rules representing the following considerations (109):

1. For combining the LS scores for different elements (X1 and X2) in the LHS of the rule,
   - If X1 AND X2  LS = lower of LS (Y, X1) and LS (Y, X2)
   - If X1 OR X2  LS = higher of LS (Y, X1) and LS (Y, X2)
   - If NOT X1  LS = - LS (Y, X1)

2. The combined LS of all the elements in LHS is then multiplied with the LS of the rule itself (and divided by 10 to keep the value between 1 and 10) to yield the LS of the object in the RHS of the rule.

3. The SS of the object created in the RHS of the rule is the same as the SS of the rule itself.

4. If a rule fires more than once generating identical objects but with different LS and SS, the more recently created instance is kept and the other is dropped.

5. If more than one rule creates identical objects with different LS and SS, the final LS and SS are determined as follows:
   - If both LS1 and LS2 are positive, CombinedLS = LS1 + LS2 (1 − LS1)
   - If one is negative and other positive, CombinedLS = LS1 + LS2
     \[1 - \min (|LS1|, |LS2|)\]
   - If both LS1 and LS2 are negative, CombinedLS = LS1 + LS2 (1 + LS1)
CombinedSS = SS for the instance with higher LS

After calculating the combined LS and SS, one of the instances is modified to reflect these new numbers, and the other one is dropped.

The above equations for LS are the same as those used for calculation of certainty factors in MYCIN. (109) Those for SS were developed ad hoc for this project.

Rules in the FEED KB were primarily used to determine what action items were recommended by protocols in the particular case being analyzed. These rules can be categorized on several different axes.

1. **Different rules had different goals:** Determining what protocol should have been used; determining recommended action items for particular protocols; determining additional information to be solicited from user; computing intermediate medical concepts; and matching recommended action items with actual documentation.

2. **Different rules had different complexity of reasoning:** Some action items were always recommended, although their Likelihood Scores (LSs) sometimes changed with protocol (e.g., procedure ‘cardiac monitor’); all action items in the particular protocol were recommended, with variable LS which might or might not change with other information (e.g. acquiring patient’s past medical/surgical history in chest pain protocol); and some other action items required calculation of intermediate medical concepts or soliciting more information from user (e.g. giving medication ‘aspirin’ depended on whether patient could swallow and the absence of any contraindications).
3. **Different rules recommended different kinds of object attributes:** Some rules focused on errors of omission, they recommended that particular action items should have been recorded (e.g. procedure ‘intubation’ in cardiac arrest protocol); others focused on contra-indications, recommending that particular action items should *not* have been recorded (e.g. no medication ‘nitroglycerin’ if patient is taking medication for erectile dysfunction); still other rules focused on errors of commission, they recommended that the value of particular action items should have been recorded within a strict range of numbers or values (e.g. dose of oxygen in chest pain protocol restricted to 12-15 L/min); the final group of rules focused on frequency or the total number of times a particular action item should have been recorded during the run (e.g. recording vital signs every 5, 10 or 15 minutes).

In addition, different kinds of data were represented in FEED in different ways because CLIPS treats facts and objects differently. While objects can be directly modified, facts have to be retracted (removed from the rule engine) and asserted again in the modified form. This means that all rules with the modified fact in the LHS of the rule will get activated, and fire again. This may not be desirable in certain conditions. In addition, it is possible to have identical copies of objects, but not facts. Therefore, data that needed to be modified frequently in the running of the system (suggestions, concepts) were represented as objects, and others that were static and could not allow duplicates (actual documentation, information about rules) were represented as facts.
This section describes the control flow of FEED’s inference engine in detail. The program execution thread passed frequently between the base program coded in Visual Basic .NET and the CLIPS rule engine (CRE).

After the user selected a case for review, the value of the primary key of the main table for the case was passed on to the ‘CLIPSLoader’ module. This module created an instance (session) of the CRE and loaded the definitions of fact templates and classes. It then loaded the rules present in the KB, including both medical rules and rules for program functions. The module then created a dataset and filled its tables with corresponding values from the SQL database. It then found data elements in these tables that belonged to the current case, using the primary key, and asserted a fact into the CRE for each of these data elements in this format: (data (varname E17_01) (value 6860)).

As new facts are asserted, and objects created and modified in the CRE, it keeps checking if any of them satisfy any rule’s antecedent conditions. As soon as a rule’s antecedents are satisfied, it is placed on the ‘agenda’ of the engine. The rules on the agenda are fired in sequential order. However, rules with a higher ‘salience’ are fired before rules with lower salience, salience being an integer from -1000 to 1000 specified along with the definition of the rule. The highest salience in FEED KB was for rules for dropping duplicate objects, followed by rules for combining LSs of similar objects (so that only one, the most relevant instance, was left to activate other rules). Other rules had identical, lower salience.

As the rules were executed, new objects were created representing FEED’s suggestions or intermediate medical concepts. If duplicate objects were generated, they were
handled as described above. Additional ‘askvar’ facts were asserted representing FEED’s need for additional input from the user. The execution paused when the agenda became empty, that is, all rules that could be fired were fired. The CLIPSLoader module then passed execution to AskUser module that goes through the facts existing in the CRE session, looking for facts named ‘askvar’. If any such facts exist, a messages window would popup and ask the user for input specific to the askvar. The input from the user was asserted as a new fact into the CRE, and the engine was run again. Once the execution halted again, the CRE was checked again for ‘askvar’ facts and the user prompted again if any existed. This continued until no more ‘askvar’ facts existed and no more activated rules were present on the agenda.

The execution was then passed to the ShowResult module, that loaded another set of rules into the CRE. These rules matched the actual documentation to what FEED suggested. For each suggestion, FEED tried to determine whether the actual documentation matched the particular suggestion, and appended a ‘yes’, ‘no’ or ‘unknown’ to the suggestion accordingly. The ShowResult module then went through the CRE and added a few lines of code to an HTML file for each suggestion object in the CRE. Finally the HTML page was displayed to the user in a web browser. If the user wished to obtain feedback on some other case, a new CRE instance was created and the whole process repeated.

Another component was added to the FEED inference engine to allow access to the internal error and warning messages of the CRE. In initial versions, if the CRE exited due to some reason, no message was passed on to the .NET application, and therefore, no information was available for debugging. In later versions, a special module was added to
the .NET program that caught the CRE messages through an Input/Output Router function. These messages were then displayed in a text box in a popup window and could be used for debugging.

Preliminary User Interface

The user interface of FEED for the second iteration was built with VB .NET Windows Forms components. It started with a login page, showing two options after authentication: Get feedback for the latest case or choose case from a list of available cases. (Figures 4, 5) After choosing the case for review, the user sometimes got a popup window asking for additional input for specific data elements. (Figures 6, 7) At the end of analysis, the user received the complete feedback in an HTML page that opened in a web browser. (Figure 8) For the second iteration, it consisted of the name of the variable suggested, followed by the suggestion, whose language depended on the kind of the particular suggestion. Additionally, the LS and SS for each suggestion were displayed below. Suggestions that matched with the actual documentation were labeled in green, those that did not were labeled in red, and the unknowns were marked blue.

In the third iteration, two major changes occurred with the user interface. The feedback page was enhanced to include explanation and reasoning behind each LS and SS score. Also, feedback stored from previous sessions could be reviewed at a later stage.

Initial Test Plan

The knowledge base evaluation procedures in the second iteration included local syntax checking, informal verification and expert verification. Local syntax checking was
Figure 4. A screenshot of the login screen of FEED showing available options.

Figure 5. A screenshot showing the list of cases available to the user who is logged in.
achieved in a manner similar to the first iteration, utilizing the constraint checking system inbuilt in CLIPS. Both dummy test cases and actual patient care reports were utilized in informal verification of FEED. Towards the end of the iteration, a focus group consisting of six practicing EMTs was convened. They went through the FEED knowledge base and verified each rule for appropriateness of antecedents, LS and SS scores, and the explanations associated with the LS and SS scores. The knowledge base evaluation procedures in the third iteration included consistency checking with the help of a rule parser, and expert validation in a focus group.
As mentioned above, coding for the first iteration was done entirely in the CLIPS language. Two test cases were used to test whether the prototype behaved as expected. Any errors that the system encountered while running were noted, and their cause rectified. Extensive coding was done as part of the second iteration, for all components of FEED. In the third iteration, coding was limited to developing the additional functionality described above.
Case No. 2
FEED suggests that:

✔️ Medication Dosage for Medication Given nitroglycerin have this value: 0.4.
   Likelihood: 9/10 Seriousness: 9/10 (Show more)
   The following rule(s) led to this LS and SS:
   Rule ID: r0058 If cardiac chest pain protocol was followed, and nitroglycerin was given to the patient, then it is suggested that the dose of nitroglycerin be 0.4mg for each tablet or spray.
   Likelihood: 9/10 It is usually possible to deliver nitroglycerine in the required dosage, as each tablet and each spray contains 0.4mg of nitroglycerin.
   Seriousness: 9/10 Accurate dosing of nitroglycerin is important. Inadequate dosing may not allow the patient the full advantage of the vasodilation caused by nitroglycerin. Overdosing (greater than three doses or 1.2mg) is unnecessary and may lead to hypotension.

✔️ Medication Dosage Units for Medication Given nitroglycerin have this value: MG.
   Likelihood: 9/10 Seriousness: 9/10 (Show more)

❓ The value nitroglycerin of the variable Medication Given be collected 1 or 2 or 3 time(s).
   Likelihood: 9/10 Seriousness: 9/10 (Show more)

Figure 8. A screenshot of FEED showing the feedback provided to the user. One suggestion is expanded to show the reasoning and explanations behind the suggestion.

Source Listings

Code for both inference engine and KB was thoroughly commented in the third iteration.
Stage V – Knowledge Base Evaluation

Test Cases

Test cases for evaluation of FEED were compiled from different sources and methods, for use in different kinds of evaluation.

1. *Ad hoc test cases*: These cases were made by the developer, on the fly, as needed to test different components of the system during development. They may not include all the information that is usually present in EMS documentation.

2. *Simulated test cases*: These test cases were prepared by the developer to simulate actual case documentations.

3. *Random-data test cases*: These cases had random data in the fields, for testing whether the system can handle unexpected scenarios.

4. *BFRS cases*: Actual patient reports were obtained (after removing patient and personnel identifiers) from Birmingham Fire and Rescue Service (BFRS). These were manually entered into the local database.

Checking

In this component of testing, the knowledge base was internally checked and the results were used to identify problems that were fixed in the next iteration of the system. This usually occurs in three steps: Local syntax checking, checking for consistency and checking for completeness. (123)
Local Syntax Checking

Testing for local syntax errors allowed identification of programming errors. CLIPS has very robust constraint checking that occurs both on loading the knowledge base, and on running the rule engine. Every time the knowledge base is loaded, CLIPS checks the syntax of the KB, gives an error if it encounters syntactical errors, and fails to load. It also performs what it calls ‘static constraint checking’ to ensure that there are no rules that violate the value constraints of objects and facts, either on LHS or on RHS of rules. If it encounters such an error, it gives an error and fails to load. Also, there is ‘dynamic constraint checking’ similar to static constraint checking but performed while the engine is running and rules are being fired. If errors are encountered, the program is terminated. These features of CLIPS were used to perform local syntax checking every time the knowledge base underwent any major or minor changes. To allow faster checking, the KB files were directly loaded into the standard CLIPS interface and any problems noted and corrected. Since CLIPS failed to load if it encountered errors during loading, any such errors had to be fixed before further testing could occur. Errors encountered during dynamic constraint checking were also fixed as found.

Consistency Checking

This kind of checking ensures that the rules are compatible with each other. A rule parser program was built to analyze the rules in the KB. It found rules with identical or similar LHS and RHS patterns, and presented them to the developer, who then looked for redundancy, conflict and subsumption problems. Any such problems were noted, and if major, corrected in this iteration. Other consistency problems, namely, unnecessary IF
conditions and circularity, were be checked due to time constraints. Checks for unnec-
sary IF conditions are beneficial for increasing the speed and robustness of the system,
but they do not pose a problem in its execution. Circularity causes the system to become
unresponsive, and was not encountered during the whole development process.

Completeness Checking

Completeness checking involves finding the unknown in the system, something
that is missing in the system. (148) There are two components to what is known as ‘com-
pleteness’ of an expert system. A subjective component consists of direct checking of the
knowledge base modules by the knowledge engineer and the expert, exhaustively looking
at each module, and each rule in the module, for self-completeness. (148) This is closer to
verification, and was covered during the expert focus group described below. The objec-
tive component of completeness checking is complex and more suitable at advanced
stages in the development cycle, especially for large projects with multiple knowledge
engineers. Since FEED KB is simple and was developed by a single knowledge engineer,
it did not need objective completeness checking. In larger projects, this step involves in-
direct methods and an algorithmic approach to look for unreferenced attribute values, un-
reachable conclusions, dead-end IF statements and dead-end goals. (123) This kind of
checking has the potential to uncover coding errors like misspelling and can improve the
efficiency and robustness of the KB.
Verification

As mentioned earlier, verification means testing if the system is built right, i.e. according to specifications. The knowledge base of FEED was verified in two steps: Informal verification and expert verification.

Informal Verification

This phase of testing involved verification if the KB was capable of handling the usual case scenarios encountered in the field. Simulated test cases built by the developer were used to test if the system can deal with all the different decision forks in the protocols. In addition, three test cases for each protocol were taken randomly from the set of real-life cases obtained from source 4 above, and used as a ‘training dataset’ for the system. The output of the system for all the above cases was informally checked by the developer for any unexpected firing of rules. If any such problems were encountered, they were noted, and the inference engine and knowledge base checked to look for the cause. Major problems were corrected in the same iteration while minor ones were corrected in the next iteration of system development.

Expert Verification

The objective of this phase was to verify whether FEED’s KB meets the ‘Complexity of Inference’, ‘Quality of Knowledge Base’ and ‘Quality of Feedback’ specifications outlined above, and if not, gather enough information to help it meet the specifications in the next iteration. A focus group of 6 experts was convened, with participants recruited from practicing EMS professionals in Birmingham region. The concept of LS
and SS and their importance to this project was explained to the participants after obtaining written informed consent. This focus group was asked to examine all the rules in FEED’s KB. For each rule, each participant provided his/her own LS and SS ratings, on a scale of 1 to 10. They were asked to consider two questions for each rule: ‘Can the conclusions of this rule be better reached by a different set of LHS arguments?’ and ‘Are reasonable explanations provided for LS and SS of this rule?’

However, many of the rules in FEED exist in pairs, each rule in the pair having the same ‘THEN’ statements, the only difference being whether the particular protocol was actually followed or suggested by FEED. Some others exist in quadruplets, having the same ‘THEN’ conditions for two different protocols. Therefore, the focus group was asked to examine each unique rule (only one out of the pair or quadruplet). Also, only the first 85 rules (out of a total of 350) were sequentially examined. For the rest, only certain specific rules were examined. For the first 85 rules, it was observed that changes to rules with low LS or SS were usually of larger magnitude than for the rules with high LS and SS. Therefore, further discussion was limited to rules for which either LS or SS was less than 8.

For each rule, if the scores of all experts were found to be within 3 points of each other and the original value in the system (the difference between the highest and the lowest score was less than 3), and all experts answered ‘No’ to the first question and ‘Yes’ to the second question, consensus was declared and the rule was not discussed further. In all other cases, discussion was initiated by the moderator, focusing only on the point of contention. For rules with divergent rating, each expert was asked to enumerate their reasons for giving the particular rating to the rule. After each expert has had a
chance to explain, the moderator asked the experts if they wished to change their rating. If the ratings now fell within 3 points of each other, the group proceeded to the next rule. If not, the floor was opened for comments and questions. A maximum of 10 minutes was spent on each rule, at the end of which, the moderator marked the particular score as contentious, and noted the experts’ final scores. For rules where one or more experts answered ‘No’ to the first question, or ‘Yes’ to the second question, those experts were asked to elaborate their reasons for so answering. The moderator then asked other experts to comment, but limited the discussion to a total of 10 minutes for each rule. The proceedings of the focus group were recorded on video.

The video recording of the focus group was transcribed, and the discussion on the two questions was analyzed to assess whether ‘complexity of inference’ and ‘quality of feedback’ specifications were met. Since the specifications aimed for the quality metric of a 100%, it was decided that any answer of ‘Yes’ to the first question and ‘No’ to the second question would be considered to imply that the respective specifications were not met. The discussion was then used to identify specific recommendations for changes in the next iteration of the development of FEED.

Validation

The objective of this phase of testing was to validate FEED by comparing its performance to that of experts in the field. Difficulties usually arise during validation of expert systems in defining a ‘gold standard’ for comparison, since experts usually differ in their opinion. (108) This process becomes even more difficult with practice critiquing systems, since the output here is in reaction to the provider’s plan, and therefore an objec-
tive ‘gold standard’ becomes even more difficult to determine. In addition, the advice provided by a system such as FEED is difficult to categorize in discrete categories of ‘right’ and ‘wrong’, since the system itself asserts through an LS less than 10/10 that it may not be completely confident about those pieces of feedback. The validation methodology outlined below attempted to deal with these problems, and provided a measurable metric of the performance of FEED.

Focus Group

Participants. A focus group was convened consisting of three practicing EMS personnel from EMS agencies in the Birmingham area, who had worked as quality assurance officers.

Cases. A total of 50 cases with ‘Chest Pain’ as the chief complaint were obtained from BFRS. Patient identifiers had been removed from these cases. Each case was a paper copy of a Portable Document Format (PDF) document containing the case details. (See Figure 9 for an example) Twenty two of these cases were randomly chosen from the set. These cases were distinct from the ‘training set’ used for informal verification described above. The cases were further de-identified to remove agency and provider identifiers. After obtaining written informed consent, one paper copy of each case was provided to each participant of the focus group. One case (distinct from these 22 cases) was used as a practice case to help the focus group participants get familiar with the process.
Figure 9. A sample case report from BFRS. Case reports similar to this one were used for the validation of FEED.
Analysis of Cases by Experts. All experts independently analyzed each of the 22 cases, and reported errors found by them. They were asked to look for errors in the following general categories, but focusing on clinically relevant errors:

1. Omission of “required” fields.
2. Administrative errors of omission, representing information that was available but not recorded (e.g. medication recorded but not dose or route).
3. Administrative errors of commission, representing information that was recorded erroneously, without clinical implications (e.g. gender male with pregnancy/childbirth).
4. Clinical errors of omission, representing information that may represent omission of patient care procedures (e.g. spinal immobilization not reported for head trauma patient).
5. Clinical errors of commission, representing information that may represent clinically wrong patient care (e.g. wrong medication dose recorded).

Each expert noted the errors he/she found on a sheet of paper.

Rating of Errors Found. The paper sheets containing the lists of errors found by the experts were collected by the moderator at the end of each case analysis. The moderator then made a single list of all unique errors found by one or more experts in the group. The experts were then asked to rate each error in the final list on a scale of 1 to 10 for validity. They were asked to consider the following factors while providing the rating:

1. Seriousness of error, in terms of potential harm to the patient.
2. Seriousness of error, in terms of loss to EMS agency, or loss of data quality for research.

3. Frequency with which the error is encountered in actual practice.

4. Frequency with which the error is due to external factors not in control of EMS personnel.

5. Whether the error will be perceived as ‘nuisance’ by the EMS personnel using FEED.

For each error, if the validity ratings provided by the experts lay within 3 points of each other (the difference between the highest and the lowest rating was less than 3), no further discussion took place for that error. For the others, discussion was initiated by the moderator and each expert was asked to enumerate their reasons for giving the particular rating to the error. After each expert had a chance to explain, the moderator asked the experts if they wished to change their rating. If the ratings now fell within 3 points of each other, the group proceeded to the next error. If not, the floor was opened for comments and questions. A maximum of 10 minutes was spent on each error, at the end of which, the moderator marked the particular rating as contentious, and noted the experts’ final rating.

*Definition of Gold Standard.* The final ratings given by experts were averaged to get the rating for each error. An arbitrary cut-off of validity level 5 was used to convert the data from the expert panel to dichotomous form: Errors rated more than 5 were considered true errors, and those rated 5 or less were considered non-errors. This final list of errors found by experts was considered the ‘gold standard’ for the purpose of this study.
Data Entry. Each of the above 22 cases was entered into the Microsoft SQL Server database that had been built according to the NEMSIS v2.2.1 standard. The format of data in the forms obtained from BFRS was very different from the NEMSIS standard. The following rules were followed to ensure consistency in data entry, and closeness of entered data to the reality:

1. The general assumption was that the data was being entered through an electronic form that was as similar to the BFRS form as possible, in terms of what fields were filled in a structured format. Also, it was assumed that all fields corresponding to the National Elements (required to be sent to the National EMS database) were filled as structured fields.

2. There were several fields allowing structured collection of data entry in the original forms. These included patient demographics, some elements of patient history, vital signs, medications and procedures. These fields were coded into the corresponding structured fields in the electronic database. Codes and values provided with the NEMSIS v2.2.1 data standard were used as much as possible. If an electronic field marked ‘National Element’ (required to be sent to the National EMS database) was unfilled in the original form, it was coded as -10 (unknown).

3. The structured fields for documenting administered medications in the original form did not include fields for medication dose and units of the dose. If this information was available in the narrative in the original form, it was entered into the structured field, the assumption being that if an electronic form
was built based on the BFRS form, the fields for dose and dose units would be added to allow complete information about the drug to be entered.

4. Since FEED does not attempt to analyze the narrative, it was not entered into the database.

*Analysis of Cases by FEED.* These cases were then analyzed by FEED and an error report generated for each case. The report contained positive as well as negative feedback items. Each feedback item was labeled ‘correct’ if FEED decided that the case data and the suggestion matched each other, ‘incorrect’ if FEED judged that they didn’t match, and ‘unknown’ if FEED was unable to make that decision based on available data.

*Rating of Errors Found.* FEED’s validity rating was calculated for each of these by averaging the LS and SS associated with the error. These validity ratings represented the probability assigned by the system, respectively, of the error being a true error.

*Data Analysis*

*General Considerations.* The comparison of a new test to the gold standard usually involves calculation of sensitivity and specificity. (Table 5) However, this approach assumes that the results for both the test and the gold standard are dichotomous, i.e., it is possible to define cut-off values that separate the positive results from the negative results for both the test and the gold standard. This may not be possible if the cut-off values are not known for certain. For example, the cut-off value of blood glucose level measured by a new test to decide whether the patient is diabetic. In cases like these, when the result
Table 5

Calculation of Sensitivity and Specificity for a New Test T, Through Comparison with a Gold Standard, G

<table>
<thead>
<tr>
<th>Result obtained with the new test</th>
<th>Result obtained with the gold standard</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>G+</td>
</tr>
<tr>
<td>T+</td>
<td>True Positives (a)</td>
</tr>
<tr>
<td>T-</td>
<td>False Negatives (c)</td>
</tr>
</tbody>
</table>

Sensitivity = True Positive Rate = $a/(a+c)$
Specificity = True Negative Rate = $d/(b+d)$

If the test is a continuous range of values, and the cut-off value is not certain, the Receiver Operator Characteristic (ROC) curve can be helpful. For building the ROC curve, dummy tables similar to Table 5 are built for each possible cut-off value of the test, and sensitivity and specificity is calculated for each of these values. (Table 6) The ROC curve is built by built plotting the True Positive rate (Sensitivity) against False Positive rate ($1 - Specificity$) corresponding to each possible cut off value for the test. (150) The area

Table 6

Calculation of Sensitivity and Specificity for a New Test T, Through Comparison with a Gold Standard, G, When X is the Cut-off Value*

<table>
<thead>
<tr>
<th>Result obtained with the new test</th>
<th>Result obtained with the gold standard</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>G+</td>
</tr>
<tr>
<td>T &gt; X</td>
<td>True Positives (a)</td>
</tr>
<tr>
<td>T &lt; X</td>
<td>False Negatives (c)</td>
</tr>
</tbody>
</table>

Sensitivity = True Positive Rate = $a/(a+c)$
$1 - Specificity = False Positive Rate = b/(b+d)$

* Several dummy tables like this will be used for constructing a Receiver Operator Characteristics (ROC) curve, plotting Sensitivity against 1-Specificity.
under this curve represents a quantitative measure of the usefulness of the test. Also, this curve can be used to estimate sensitivity of the test at particular levels of specificity, and vice versa, and can help in deciding the most useful cut-off value for the test. This approach has been used earlier for validation of a Bayesian Expert System. (151)

*Comparisons between Experts and FEED.* Two different ROC curves were built comparing the output of FEED with the results from the expert focus group:

1. *Suggestions marked ‘incorrect’ by FEED vs Expert Focus Group* (Table 7):
   
   For this comparison, a strict definition of ‘error’ was used: Only those suggestions marked ‘incorrect’ by FEED were considered errors, those marked as ‘unknown’ or ‘correct’ were considered non-errors. This comparison provided a reasonable estimate of the maximal specificity that the system could achieve. The specificity figures from this comparison would represent an

Table 7

Sample Dummy Table for the First ROC Curve, at Cut-off X for FEED’s Validity Rating

<table>
<thead>
<tr>
<th>FEED</th>
<th>Expert Focus Group</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Errors found by expert focus group and given validity rating $&gt; 5$ (+)</td>
</tr>
<tr>
<td>Suggestions marked ‘incorrect’ by FEED and given validity rating $&gt; X$ (+)</td>
<td>True Positives (a)</td>
</tr>
<tr>
<td>Suggestions not shown by FEED; or marked ‘correct’ or ‘unknown’; or marked ‘incorrect’ and given validity rating $&lt; X$ (-)</td>
<td>False Negatives (c)</td>
</tr>
</tbody>
</table>

Sensitivity = True Positive Rate = $a/(a+c)$

1 - Specificity = False Positive Rate = $b/(b+d)$
estimate of FEED’s capability to *not* give negative feedback to the user when it was unwarranted. However, since FEED does not analyze the narrative for marking errors as ‘incorrect’ or ‘correct’, it would be expected to be unable to make that judgment for a large number of suggestions (since quite a lot of information resides in the narrative). Therefore, this comparison was likely to underestimate the sensitivity of the system. To offset this problem, another ROC curve was built.

2. *Suggestions marked ‘incorrect’ or ‘unknown’ by FEED vs Expert Focus Group* (Table 8): This comparison assumed a loose definition of ‘error’: Suggestions marked either ‘incorrect’ or ‘unknown’ were considered errors. For the sake of this comparison, it was assumed that the suggestions marked as

<table>
<thead>
<tr>
<th>Table 8</th>
<th>Sample Dummy Table for the Second ROC Curve, at Cut-off X for FEED’s Validity Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>FEED</strong></td>
</tr>
<tr>
<td></td>
<td>Errors found by expert focus group and given validity rating &gt; 5 (+)</td>
</tr>
<tr>
<td>Suggestions marked ‘incorrect’ or ‘unknown’ by FEED and given validity rating &gt; X (+)</td>
<td>True Positives (a)</td>
</tr>
<tr>
<td>Suggestions not shown by FEED; or marked ‘correct’; or marked ‘incorrect’ and given validity rating &lt; X; or marked ‘unknown’ and given validity rating &lt; X (-)</td>
<td>False Negatives (c)</td>
</tr>
</tbody>
</table>

\[
\text{Sensitivity} = \frac{\text{True Positive Rate}}{a+c} = \frac{a}{a+c} \\
1 - \text{Specificity} = \frac{\text{False Positive Rate}}{b+d} = \frac{b}{b+d}
\]
‘unknown’ were ‘potential errors’ that could be resolved by a manual inspection of the narrative. Therefore, this comparison provided a reasonable estimate of the maximal sensitivity the system could achieve. The sensitivity figures from this comparison would represent an estimate of FEED’s capability to identify all errors and potential errors, and show them to the user.

Operational characteristics of FEED were calculated from these curves at four different fixed sensitivity levels appropriate for each curve. Misclassified errors were examined individually (false positives from the first ROC curve, and false negatives from the second ROC curve) and the cause of misclassification identified.
RESULTS

This section describes the results, findings and experiences during the development of FEED.

Checking and Informal Verification

These tests used informal methods to identify errors in the system’s execution and output. Various errors were identified at different stages, and corrected in the same iteration (if major) or in the next iteration (if minor).

Expert Verification

Although six experts participated in the focus group, not all of them were present at all times. Table 9 shows details of their participation.

FEED has a total of 156 unique rules, out of which 76 unique rules were examined in the focus group owing to time constraints. Of these, the first 45 unique rules were

Table 9
Participation of Experts in the Verification Focus Group

<table>
<thead>
<tr>
<th>Number of Experts</th>
<th>Rules Discussed</th>
<th>Number of Unique Rules Discussed</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1 – 2</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>3 – 83</td>
<td>43</td>
</tr>
<tr>
<td>5</td>
<td>84 – 185</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>186 – 350</td>
<td>16</td>
</tr>
</tbody>
</table>
discussed sequentially, and the rest 31 unique rules were discussed only because either their LS or SS was less than 8. A total of about 5 hours was spent, for an average of 6.7 minutes per rule.

The scores for the rules were discussed for 72 of the 76 unique rules. The experts suggested changes in LS for 56 (78%) and changes in SS for 62 (86%) of these rules (Table 10), resulting in median absolute change in LS of 1 (interquartile range (IQR) 0 – 1), and median absolute change in SS of 1 (IQR 0 – 2.75).

The absolute change in LS for rules with initial LS less than 8 (median 3, IQR 2 – 5.75) was different from the change for rules with initial LS of 8 or more (median 1, IQR 0 – 1). Similarly, absolute change in SS for rules with initial SS less than 8 (median 3, IQR 2 – 4.25) was different from the change for rules with initial SS of 8 or more

Table 10

Differences in FEED’s Likelihood Scores (LS) and Seriousness Scores (SS) Before and After the Expert Verification Focus Group

<table>
<thead>
<tr>
<th>Score</th>
<th>Final score minus initial score</th>
<th>Absolute difference between initial and final scores</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Median</td>
<td>IQR</td>
</tr>
<tr>
<td>LS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>All rules</td>
<td>1</td>
<td>0 – 1</td>
</tr>
<tr>
<td>Rules with initial LS &lt; 8</td>
<td>3*</td>
<td>2 – 5.75</td>
</tr>
<tr>
<td>Rules with initial LS ≥ 8</td>
<td>1*</td>
<td>0 – 1</td>
</tr>
<tr>
<td>SS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>All rules</td>
<td>1</td>
<td>0 – 2.75</td>
</tr>
<tr>
<td>Rules with initial SS &lt; 8</td>
<td>3*</td>
<td>2 – 4.25</td>
</tr>
<tr>
<td>Rules with initial SS ≥ 8</td>
<td>1*</td>
<td>0.75 – 1.25</td>
</tr>
</tbody>
</table>

*+ Absolute changes are significantly different in the two rule groups (Mann-Whitney U, p<0.001)
Both these differences were statistically significant. Neither of the changes for rules with initial LS of 8 or more was large enough to be clinically relevant. This justified the decision to limit discussion to the rules with initial LS or SS less than 8.

For 8 rules, the experts had a disagreement in LS scores large enough to require further discussion, and in 10 rules, the disagreement in SS scores was large enough to necessitate further discussion. In all of these rules, consensus was reached after the first round of discussion.

The expert group flagged 8 rules for modification of LHS conditions. In most of these, they suggested that the rule be split into two based on certain variables. Also, they flagged 3 rules for changes to the explanation of LS or SS. Six rules had final LS and SS scores of 1 or 2, indicating the need for their removal from KB, since they would invariably lead to feedback with low validity.

The KB of FEED was found to have problems in the verification, and did not meet the ‘complexity of inference’ and ‘quality of feedback’ requirements. However, enough information was gathered to correct these problems in the next iteration.

**Validation**

A total of 251 errors were found by the focus group, and 769 suggestions were made by FEED (Table 11). The average validity ratings provided by the focus group were fairly dichotomous. All but one lay between 1-2 or 9-10 (Table 12). The arbitrary use of validity level 5, therefore, seems justified. See Figures 10 and 11 for the ROC curves. P values are for the comparison between the area under the curve (AUC) and 0.5.
Table 11

Errors Found by the Expert Focus Group, Grouped by Kind of FEED’s Suggestion

<table>
<thead>
<tr>
<th>FEED</th>
<th>Expert Focus Group</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Errors found by expert focus group and given validity rating &gt; 5</td>
<td>Errors not found by expert focus group; or found and given validity rating ≤ 5</td>
<td></td>
</tr>
<tr>
<td>Not suggested by FEED</td>
<td>49 (24%)</td>
<td>25 (4%)</td>
<td></td>
</tr>
<tr>
<td>Suggested as ‘correct’ by FEED</td>
<td>6 (3%)</td>
<td>167 (26%)</td>
<td></td>
</tr>
<tr>
<td>Suggested as ‘incorrect’ by FEED</td>
<td>71 (35%)</td>
<td>200 (31%)</td>
<td></td>
</tr>
<tr>
<td>Suggested as ‘unknown’ by FEED</td>
<td>79 (39%)</td>
<td>246 (39%)</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>205 (100%)</td>
<td>638 (100%)</td>
<td></td>
</tr>
</tbody>
</table>

Table 12

Errors Found by the Expert Focus Group Grouped by Validity Score

<table>
<thead>
<tr>
<th>Validity Score</th>
<th>Number of errors (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6 (2.4%)</td>
</tr>
<tr>
<td>1.3</td>
<td>1 (0.4%)</td>
</tr>
<tr>
<td>1.7</td>
<td>2 (0.8%)</td>
</tr>
<tr>
<td>4</td>
<td>1 (0.4%)</td>
</tr>
<tr>
<td>9</td>
<td>5 (2.0%)</td>
</tr>
<tr>
<td>9.3</td>
<td>15 (6.0%)</td>
</tr>
<tr>
<td>9.7</td>
<td>21 (8.4%)</td>
</tr>
<tr>
<td>10</td>
<td>200 (80.0%)</td>
</tr>
<tr>
<td>Total</td>
<td>251 (100.0%)</td>
</tr>
</tbody>
</table>

See Tables 13 and 14 for operator characteristics of the ROC curves at different sensitivity and specificity levels. From the curves and tables for the ROC curves, the cut-off level of 8.75 for FEED’s validity score was chosen for further analysis. See Tables 15 and 16 for detailed review of false positive and false negative errors at this cut-off level.
Figure 10. First ROC curve, comparing suggestions labeled as ‘incorrect’ by FEED to results from the expert focus group. (AUC 0.559, p=0.01)

Table 13
Sensitivity and Specificity of FEED at Four Different Cut-off Points, from the First ROC Curve, Comparing Suggestions Labeled as ‘Incorrect’ by FEED to Results from the Expert Focus Group

<table>
<thead>
<tr>
<th>Cut-off for FEED’s validity score</th>
<th>Sensitivity</th>
<th>1 - Specificity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>34.6%</td>
<td>21.0%</td>
<td>79.0%*</td>
</tr>
<tr>
<td>8.75</td>
<td>30.7%</td>
<td>15.5%</td>
<td>84.5%+</td>
</tr>
<tr>
<td>9.25</td>
<td>17.1%</td>
<td>11.6%</td>
<td>88.4%</td>
</tr>
<tr>
<td>9.75</td>
<td>16.1%</td>
<td>8.9%</td>
<td>91.1%**</td>
</tr>
</tbody>
</table>

* Highest observed sensitivity in this comparison. + Highest specificity for that particular sensitivity. ** Highest observed specificity in this comparison.
Figure 11. Second ROC curve, comparing suggestions labeled as ‘incorrect’ or ‘unknown’ by FEED to results from the expert focus group. (AUC 0.642, p<0.001)

Table 14

Sensitivity and Specificity of FEED at Four Different Cut-off Points, from the Second ROC Curve, Comparing Suggestions Labeled as ‘Incorrect’ or ‘Unknown’ by FEED to Results from the Expert Focus Group

<table>
<thead>
<tr>
<th>Cut-off for FEED’s validity score</th>
<th>Sensitivity</th>
<th>1 - Specificity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.75</td>
<td>73.2%*</td>
<td>48.7%</td>
<td>51.3%*</td>
</tr>
<tr>
<td>8.75</td>
<td>59.5%</td>
<td>28.1%</td>
<td>71.9%</td>
</tr>
<tr>
<td>9.25</td>
<td>36.6%</td>
<td>15.7%</td>
<td>84.3%</td>
</tr>
<tr>
<td>9.75</td>
<td>19.5%</td>
<td>12.9%</td>
<td>87.1%**</td>
</tr>
</tbody>
</table>

* Highest observed sensitivity in this comparison. † Highest specificity for that particular sensitivity. ** Highest observed specificity in this comparison.
Table 15

Review of False Positive Errors Encountered in the First Comparison, at Cut-off Level of 8.75 for FEED’s Validity Score

<table>
<thead>
<tr>
<th>Error Category</th>
<th>Count</th>
<th>Unique Count</th>
<th>Error Description</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experts never considered this error; or considered but never listed it as an error</td>
<td>22</td>
<td>1</td>
<td>‘Chest Pain’ not recorded in ‘Protocols Used’ field</td>
<td>22</td>
</tr>
<tr>
<td>Experts reported this error in other cases but missed it in this case</td>
<td>29</td>
<td>7</td>
<td>Procedure ‘cardiac monitor’ not done</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Medication morphine not given</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Symptom ‘difficulty breathing’ not recorded</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Medication nitroglycerin not given</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Medication aspirin not given</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Procedure ‘pulse oximetry’ not done</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Pulse oximetry value not recoded</td>
<td>4</td>
</tr>
<tr>
<td>Invalid error, if information in narrative is considered</td>
<td>28</td>
<td>8</td>
<td>Symptom ‘difficulty breathing’ not recorded</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Medication oxygen not given</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Medication nitroglycerin not given</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Medication aspirin not given</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Units of aspirin dose wrong or not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Procedure ‘cardiac monitor’ not done</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Procedure ‘venous access’ not done</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Procedure ‘pulse oximetry’ not done</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dose of oxygen not within 12-15 L/min</td>
<td>5</td>
</tr>
<tr>
<td>Error based on rules endorsed by verification focus group but not part of Alabama EMS protocols</td>
<td>5</td>
<td>1</td>
<td>Units of aspirin dosage wrong or not recorded</td>
<td>2</td>
</tr>
<tr>
<td>Not a False Positive, miscalculation during analysis</td>
<td>4</td>
<td>2</td>
<td>Units of nitroglycerin dosage wrong or not recorded</td>
<td>2</td>
</tr>
<tr>
<td>Invalid error, inaccuracy in FEED KB</td>
<td>5</td>
<td>2</td>
<td>Dose of aspirin is wrong or not recorded</td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td>93</td>
<td>12</td>
<td>Units of aspirin dosage wrong or not recorded</td>
<td></td>
</tr>
</tbody>
</table>
Table 16

Review of False Negative Errors Encountered in the Second Comparison, at Cut-off Level of 8.75 for FEED’s Validity Score

<table>
<thead>
<tr>
<th>Error Category</th>
<th>Count</th>
<th>Unique Count</th>
<th>Error Description</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error reported by FEED, but given validity score &lt; 8.75</td>
<td>31</td>
<td>4</td>
<td>Pain PQRST details not recorded</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Symptom ‘nausea/vomiting’ not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Examination finding ‘peripheral edema’ not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Procedure ‘12-lead ECG’ not done</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Site of venous access not recorded</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Gauge of IV line not recorded</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Type of IV fluid not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rate of IV fluid not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>History of current illness not recorded</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ECG rhythm interpretation not recorded</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Examination of pupils not recorded</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Response to nitroglycerin not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Past history of similar illness not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Route of aspirin administration not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Examination of extremities not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Reason for not taking to patient’s first choice of hospital not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Reason for leaving hospital not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Glasgow Coma Scale wrong or inconsistent with other findings</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Examination of alertness and orientation not recorded</td>
<td>1</td>
</tr>
<tr>
<td>FEED does not have a rule to report this error</td>
<td>38</td>
<td>14</td>
<td>Pain PQRST details not recorded</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Symptom ‘nausea/vomiting’ not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Examination finding ‘peripheral edema’ not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Procedure ‘12-lead ECG’ not done</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Site of venous access not recorded</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Gauge of IV line not recorded</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Type of IV fluid not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Rate of IV fluid not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>History of current illness not recorded</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ECG rhythm interpretation not recorded</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Examination of pupils not recorded</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Response to nitroglycerin not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Past history of similar illness not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Route of aspirin administration not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Examination of extremities not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Reason for not taking to patient’s first choice of hospital not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Reason for leaving hospital not recorded</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Glasgow Coma Scale wrong or inconsistent with other findings</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Examination of alertness and orientation not recorded</td>
<td>1</td>
</tr>
<tr>
<td>Error Description</td>
<td>FEED Rules</td>
<td>Documentation Error</td>
<td>Inconsistencies found based on information in narrative</td>
<td>Errors based on a protocol not yet included in FEED KB</td>
</tr>
<tr>
<td>----------------------------------------------------------------------------------</td>
<td>------------</td>
<td>---------------------</td>
<td>-------------------------------------------------------</td>
<td>------------------------------------------------------</td>
</tr>
<tr>
<td>FEED has rules to report this error, but needs information that was only available in narrative in this case</td>
<td>8</td>
<td>5</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>FEED rules need to be more complex to report this error</td>
<td>10</td>
<td>3</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Documentation error within narrative</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Inconsistencies found based on information in narrative</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Errors based on a protocol not yet included in FEED KB</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Errors based on clinical knowledge of experts, not in Alabama protocols</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Error missed due to inaccuracy in FEED KB</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Not a False Negative, miscalculation during analysis</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Error missed due to unknown cause</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>107</td>
<td>41</td>
<td>87</td>
<td>12</td>
</tr>
</tbody>
</table>
DISCUSSION

This discussion will focus on issues related to the design of the FEED system and the results of the validation study.

Design of FEED

The development of FEED was facilitated by several factors. The availability of a tool to integrate CLIPS into the Visual Studio .NET environment was very helpful. It allowed easy flow of program control between the ‘activation’-based process flow of CLIPS and the procedural code-based process flow of .NET. In addition, it allowed integration of the powerful knowledge management capabilities of CLIPS with the rich user interface controls available in .NET. The Alabama State EMS patient care protocols were very amenable to transformation into a rule-based knowledge base. Also, ready availability of the Expert was very helpful for clarifications and initial assignments of LS and SS with the rules.

Several problems were faced during the development process. CLIPS has a complex high-level language for coding rules. It is restrictive in terms of what commands and functions are available on the LHS of rules, and what are available on the RHS of rules. In addition, there are some quirks that needed alternate methods of coding to achieve desired results, or in some cases, a redesign of the knowledge structure. For instance, all data, concepts and suggestions in the initial versions of FEED were expressed as facts in
the KB. Some of these facts needed to be modified during the analysis. However, modifying a fact in CLIPS entails retracting (removing) the fact and asserting it again. This would re-activate all the rules looking for the fact, including the rule for modifying the fact. This rule would fire again, leading to another retraction and re-assertion of the fact, getting into an endless cycle. The problem was averted by changing the knowledge representation format from facts to objects for data structures that needed to be modified during the program run (CLIPS objects can be directly modified.)

Another major problem was of inconsistent vocabulary. The NEMSIS v2.2.1 dataset provides a uniform dataset for all pre-hospital data representation needs. However, this standard is not being widely used at present. The Alabama State’s EMSIS database has only recently become compliant with this standard. The earlier version had different field names, different available choices for structured fields, and different coding for these available choices. Also, similar mismatch existed between any electronic data standard and any paper-based patient care report. In addition, all of the above data schemas had different lists of variables that were recommended for collection, and those that were required for collection in a structured format. Furthermore, some of the protocol names, medications and procedures used in the Alabama State EMS protocols were not available for coding in any of the above standards. Several assumptions were made during both development and testing to reconcile these differences. How these assumptions may affect the validity of the system is discussed below.

The development process of FEED from the conception of the idea and initial planning to verification and validation took about 21 months. This was preceded by preliminary studies exploring the need for decision support in EMS. Data from Alabama
State’s EMSIS database was analyzed retrospectively to study the pattern of utilization of online medical control in the state. Two requirements analyses focused on the need for electronic documentation and decision support in EMS. In addition, a review of Alabama State EMS patient care protocols, and a review of literature related to errors and decision support in EMS were performed. The development time of 21 months was equally divided between five major tasks: Looking for an appropriate expert system development tool, learning to efficiently use the tool (CLIPS integrated into .NET using ProAI’s CLIPS wrapper), knowledge acquisition and representation, coding of the inference engine and the user interface, and testing the system.

Verification Focus Group

A focus group seems to be a feasible method of verifying the knowledge base of an expert system. Although the original intention was to limit discussion to only those rules that had contentious feedback, the experts participating in the focus group provided comments for almost all rules, increasing the time needed for the whole review, but at the same time, adding value to the system as their comments were used to enhance the explanations associated with the LS and SS of the rules. An average of 6.7 minutes was spent on each rule, including reading, considering the implications of the rule, providing the scores, and further discussion. A large KB of a thousand rules could be discussed in a day long session of about 10 hours. However, the quality of verification process may deteriorate towards the end of such long sessions, although this was not specifically measured or analyzed for this focus group.
The results of the focus group indicated that the knowledge base did not meet the specifications of ‘quality of knowledge base,’ ‘complexity of inference’ and ‘quality of feedback’ at that stage of development. The LS for 78% and SS for 86% of rules reviewed had to be changed. 26% of rules had LS changes of 3 or more, and 31% of rules had SS changes of 3 or more, which indicates the proportion of changes that are likely to lead to significant changes in the feedback provided by FEED. It is possible that these changes represent the value gained through discussion among several experts. The six experts in the focus group were in consensus with each other (or reached consensus through discussion), and were able to provide sound justifications for the proposed changes. It is also possible that the experts in the focus group came from a similar background, but different from that of the original expert, and the difference represents a difference of expert opinion. This can be avoided in the future by ensuring that experts from different backgrounds are invited to the focus groups, including some with backgrounds similar to the original expert.

The changes proposed for improving the ‘complexity of inference’ and ‘quality of feedback’ were less frequent, and the most common change proposed was to split the particular rule into two based on whether the patient was stable or unstable. Although the KB did not meet specifications, enough information was gathered in this focus group for making changes that would bring the KB closer to the specifications.

Validation Focus Group

The focus group convened for validating FEED also provided a feasible method for the validation process. The ratings provided by the experts to the errors found by them
were fairly dichotomous. This allowed bifurcation of the errors into true errors and non-
errors, thereby facilitating the comparison with FEED’s output. The results from the
comparison were somewhat unexpected and are discussed below.

The first comparison (Tables 7, 13; Figure 10) was performed between sugges-
tions marked ‘incorrect’ by FEED and the results from the expert focus group. At various
cut-off points, the specificity of FEED ranged from 79% to 91%, and the sensitivity from
16% to 35%. The specificity was reasonably high, indicating that FEED unjustifiably la-
beled non-errors as ‘incorrect’ for only 9% to 21% of non-errors.

However, the sensitivity was very low in this analysis, indicating that among the
true errors found by experts, 65% to 84% were not found by FEED or not labeled as ‘in-
correct.’ A large proportion of these false negatives resulted from the way FEED makes a
decision to label suggestions as ‘correct’ or ‘incorrect.’ FEED has access to only the
structured portion of the patient care report, which has limited information. Also, some of
the structured fields may not be required for completing the documentation. Therefore,
FEED is unable to make a definitive judgment regarding the correctness of many sugges-
tions. In such cases FEED labels the suggestion as ‘unknown.’ On the other hand, experts
have access to the narrative, and are able to find errors based on the information in the
narrative. When only suggestions labeled ‘incorrect’ are counted as true errors, the sensi-
tivity of FEED is likely to be underestimated.

The second comparison (Tables 8, 14; Figure 11) was performed between sugges-
tions marked ‘incorrect’ by FEED and the results from the expert focus group, and pro-
vided a better estimate of the sensitivity of the system. A large variability in the sensitiv-
ity was observed for different cut-off values, ranging from 20% to 73%. This indicated
that at best, FEED was able to find 73% of the errors found by experts, although it did not label them all as ‘incorrect,’ many of them were labeled ‘unknown.’

Thus FEED’s maximal specificity was 91% at a cut-off of 9.75 for FEED’s validity score, and the maximal sensitivity was 73% at a cut-off of 6.75. From the tables, 8.75 seemed a reasonable cut-off level for optimal sensitivity and specificity. At this cut-off, the sensitivity was 60% and specificity was 85%. This is similar to the 75% accuracy reported for initial versions of MYCIN. (116) The area under the curve (AUC) for the two ROC curves were significantly different from 0.5 (Figures 10, 11). This indicated that in these comparisons, FEED was better than no feedback; an error found by the experts was likely to be given a higher validity score by FEED, than a non-error.

The review of 97 false positives at the cut-off level of 8.75 revealed several common themes (Table 15). About half of the false positives could be attributed to expert error. None of the cases had filled out the ‘Protocols Used’ field, which resulted in 22 errors. However, this error was never considered by the experts, resulting in 22 false positives. In 29 other instances, FEED reported an error that was usually considered by experts but not in that case. These 51 errors may actually represent an advantage of the expert system over manual review. Since FEED uses automated process to consider all possible errors, it has the potential to uncover errors that could have been overlooked in a manual review. For example, the experts started reporting absence of morphine administration only after the initial five cases were over. After that, it was reported fairly regularly. It is, however, possible that some of these errors actually were false positives. The experts may have considered the error but did not consider it important for the particular
This is unlikely because all the cases were chest pain cases, and the state protocol mandates exactly the same action items for all chest pain cases.

This problem could be avoided in future sessions by having the expert panel review and rate the errors found by FEED, in addition to those found by them. This would ensure that all potential errors are considered by the experts, and those FEED suggestions that are actively judged to be non-errors actually are false positives.

Another common cause of false positives was that the information in the narrative was not available to FEED. If the information in the narrative had been entered in suitable structured fields, FEED would not have generated these 28 errors. This factor has several important implications for the future real-life use of FEED. This problem cannot be solved with more rigorous rules or programming. It represents a current problem in EMS, as entering information in a structured format is not a very high priority with EMS personnel. Encountering this kind of error frequently may lower the acceptability of the system. At the same time, FEED was developed to solve this same problem (incomplete and inconsistent EMS documentation). If users get feedback that they did not enter the patient care report as they should, it is possible that they will do that the next time around.

Another 5 false positives were included in the feedback by FEED based on a rule that is not part of Alabama EMS protocols but was endorsed by the first focus group that verified FEED rules. The protocols do not mention the dose of oxygen administration in the chest pain protocol, but FEED suggests that oxygen be administered at a rate of 12-15 L/min, and labeled the suggestion as ‘incorrect’ if the rate was lower than this range. This was very likely not considered an error by the experts. This could again be confirmed if
the expert group was asked to actively rate FEED’s suggestions for validity. However, if this is an actual false positive, it is unavoidable, as it represents divergent expert opinions.

Another 5 false positives were found to be a result of inaccuracy in FEED’s KB. These can be corrected in future versions of FEED. Also, another 4 errors were due to miscalculation during analysis, and were not found to be false positives on close review.

Thus, only 5 false positives were confirmed to be due to inaccuracy in FEED’s KB. Twenty eight false positives were considered unavoidable. Half of the false positives were judged to be a result of human error by experts.

A review of the 107 false negatives at the cut-off level of 8.75 revealed similar themes (Table 16). A large number of errors in this category are correctable in future versions of FEED, and represented shortcomings of the current version of FEED. In 38 instances, FEED did not have a rule for reporting that particular error, but one could be built. Another 10 false negatives could be avoided by increasing the complexity of the rules already in the KB. FEED only provides feedback based on cardiac protocol, and this resulted in 6 false negatives that would presumably be detected when the KB is expanded to include additional protocols. Two errors were reported by experts on the basis of their clinical judgment, but were not found to be part of Alabama State EMS protocols. One false negative was due to an inaccuracy in FEED KB. The above 57 false negatives are correctable in future versions of FEED. Similar results were reported after validation of the initial versions of MYCIN; in most cases, unacceptable results were because some crucial rule or rules were missing. (116)
Some of these problems may be related to the use of expert opinion as the sole source of knowledge for the KB. Clinical guidelines and protocols are often consensus documents developed by clinical experts, and often contain inexact representation of knowledge. These include use of ‘weasel’ words, sometimes intentionally used to express ambiguity, as well as use of abstract concepts that are difficult to operationalize. (152) However, implementation of these protocols in an expert system necessitates semantic refinement and formal representation in machine-interpretable format, which may be difficult if the levels of ambiguity and abstraction are high. Whenever available, evidence from clinical research and data from medical records or registries allow more discrete evidence-based or data-based abstraction for knowledge representation. Given the scant evidence-base for most EMS protocols, (2, 10, 48, 49) and the non-availability of adequately documented data from registries; (45, 46) clinical guidelines and protocols developed through expert consensus remain the only feasible source of knowledge for EMS systems. As better sources of data and stronger research support become available, they could be considered as better alternatives for building the KB.

Another major cause of false negatives was a high threshold used for the cut-off. Thirty one errors were actually reported by FEED, but considered false negatives because the validity rating given by FEED was less than 8.75. This probably represents differences in expert opinion; as the original validity rating given by experts in the first focus group was lower than 8.75, while that provided by experts in this focus group was between 9-10. It is possible that the experts in this focus group had a different perspective from the other group, as the ratings in this focus group were fairly dichotomous (a large majority between 9-10), while the other group suggested a range of values that was fairly
frequently below 9. As suggested for the verification focus group, this problem can be avoided by ensuring that participating experts belong to diverse backgrounds, including some with backgrounds similar to experts employed in earlier stages of development.

Another 16 false negatives were found to be due to FEED’s inability to consider the information in the narrative. In some cases, FEED could have reported the error, but needed information that was only available in the narrative. Other errors were documentation errors within the narrative, or inconsistencies between the narrative and other information in the case.

The cause for another two false negatives could not be determined in this review. One error was found not to be a false negative on close inspection; a miscalculation had been made during initial analysis.

There exists the potential to further increase the sensitivity and specificity of FEED, especially the sensitivity. Thus, development of an expert system such as FEED needs to be an ongoing process, hand-in-hand with evaluation. Also, this evaluation needs to be multi-faceted, as different kinds of evaluation reveal different errors and sources of errors. Some additional improvement in accuracy may be achieved by changes in the validation methodology. If the expert focus group validated the system’s errors as well, it is very likely that experts would agree that the errors reported by the system were true errors that they overlooked. However, in the end, some problems with accuracy are likely to remain, and some of these may be due to as yet undiscovered problems in FEED’s KB or inference engine. At the same time, some of these inaccuracies will occur because the users did not enter adequate information in a structured format, and the feedback from FEED can potentially induce the user to add them in future case reports.
Although the AUC calculations indicate that FEED is better than no feedback, the current level of accuracy may not be acceptable to end users. The changes suggested above would help improve this accuracy to a level where it can be considered deployable. This expected level of accuracy may be lower than that expected for diagnostic or alert systems. This is because the primary objectives of feedback systems are to change attitudes and to educate users. FEED is designed to provide feedback at the end of the run when the documentation is complete. Therefore, there are no patient care decisions that are made on the basis of this feedback. In addition, at that time, false positive errors may be tolerated better than for real-time alert systems.

Further Development

The current version of FEED is not directly deployable in the field. The problems revealed in the validation need to be corrected. The FEED KB needs to be expanded to include all patient care protocols. Additional content can be added to the explanations, including references to clinical research. The link to the documentation database needs to be made more flexible to accommodate the needs of diverse EMS agencies. In addition, some rules may need to be ‘tweaked’ according to the business rules of each particular agency. Also, more testing needs to be done, including another validation after correcting the above problems, and a usability evaluation by end users.

Thus, after further development and testing as described above, FEED has the potential to be used to provide feedback to the EMS personnel after they complete the documentation of each case. FEED would be deployable in any EMS agency that collects patient care reports electronically and stores them in a database built according to the
NEMSIS v2.2.1 specifications. Ideally, after the EMTs enter the case report electronically, they would ask FEED to provide feedback on the case. They would review the feedback, and explore the explanations and reasoning behind the feedback, including links to best evidence and or refereed research publications. This will educate them about the patient care protocols, expose them to the current state of EMS research, and, hopefully, change their attitudes regarding the need to follow protocols and complete the case documentation. All this has the potential to improve patient outcomes.

Also, in the future FEED’s KB could be modified to generate alerts and reminders for the EMT while the patient care and documentation is going on. However, this will need more rigorous testing and extensive modification as suggested by the results of testing, to further increase the sensitivity and specificity of the system.

We also considered FEED as a tool for the QA/QI officers in EMS agencies. FEED could provide an intelligent scanning service that suggests which PCRs to manually review. This allows the QA/QI officers to find and focus on specific performance measures in need of improvement (e.g., pain management). Currently, the large number of PCR reports makes it difficult to continuously monitor the quality of PCRs.
LIMITATIONS

There are limitations of documentation feedback systems built using expert system technology in general and FEED specifically. Furthermore, the development and evaluation methodology of FEED had additional limitations. These limitations are discussed below.

Feedback systems have been shown to lead to better outcomes when combined with other strategies for improving compliance, especially with educational initiatives. (105, 107) The feedback provided by FEED could be enhanced to include educational components similar to continuing education. Also, error reporting systems are likely to be more successful in revealing system problems, leading to system-wide changes. (21) FEED could complement such a system. Anonymous FEED reports may be disseminated in a manner similar to the error reporting systems, and stimulate discussions about the quality of documentation within the team of EMS providers. Additionally, FEED has the advantage of being able to deliver the educational components in a focused, case-specific manner, with a stronger impact on the user. (108)

Feedback about patient care documentation would not prevent certain EMS errors that are potentially much more serious and are encountered much more frequently than protocol deviations. These include improper technique of performing procedures, and reckless driving leading to ambulance crashes. Another concern may be that a feedback system like FEED provides passive decision support, which can easily be ignored by the
user. This may be remedied if the EMS agency mandates regular use of FEED, or if studying the feedback could be used to earn continuing education credits by the EMTs, or if FEED could provide performance indicators for PCRs that may be tied to promotions, etc.

Expert systems need to be updated as and when the current state of knowledge in the domain changes. Failure to update regularly will lead to loss of confidence in the system and eventual lack of use by the users. This version of FEED does not have the capability for easy update of rules, either manually, or automatic updates from a central rule base. Future versions could include this capability, in addition to a central rule repository allowing sharing of rules among EMS agencies.

Several assumptions were made during the development of FEED that may or may not hold true for a particular agency, limiting the potential applicability of FEED. It is assumed that the agency collects patient care reports in an electronic format based on the NEMSIS v2.2.1 dataset which is the current standard. FEED assumes that data are collected in a structured format as specified by the Alabama State EMSIS database. Also, it is assumed that the EMS personnel themselves enter the case reports electronically, have access to a computer terminal (or portable device), and have the time to look at the feedback between runs.

The KB of FEED was primarily limited to rules based on Alabama State EMS patient care protocols. As discovered during validation, additional rules may be added from the experts’ experience. So far, no effort has been made to actively acquire, represent and validate this additional knowledge. Also, FEED has been exposed to the expertise of only 11 experts so far (two during knowledge acquisition, six during verification and three
It is possible that further exposure would discover more untapped knowledge that could be used to improve the FEED KB. Similarly, the single number of LS may not fully capture the variability existing among the different EMS agencies. Some agencies do not carry nitrous oxide or 12-lead ECG devices at all. Others may have short transport times, meaning that less important action items such as abdominal examination in a chest pain patient may not be frequently performed. Also, FEED does not have rules focusing on the business aspect of EMS (collecting insurance information, for example). This could be added in future versions, potentially increasing FEED’s appeal to EMS agencies.

FEED KB made the majority of its inferences based on the protocol followed in the case, or if that was not documented, it used its assumption about the protocol. This assumption was primarily based on the chief complaint and the primary symptom of the patient, both of which are usually vague and subjective. This may not be specific enough for certain protocols (such as stroke) leading to lower accuracy. Also, in certain cardiac protocols, this assumption was based on the cardiac rhythm recorded in the case. If the cardiac rhythm was not documented, FEED was severely limited in the amount and quality of feedback it provided.

Another limitation is that FEED has not been designed with scalability in mind. For example, when the case data are loaded into FEED, it iterates through several tables in the database multiple times. This may cause performance problems in agencies with a large database. So far, only cardiac protocols have been incorporated into FEED, resulting in about 350 rules. Addition of other protocols may increase this number to about 1500, potentially making the management of the KB cumbersome unless a KB support
system is developed. Also, addition of other protocols may lead to consistency problems, which must be checked and corrected. This may also increase the analysis time for FEED from the current value of 3-4 seconds, to an unacceptable wait time for the user. Giarratano and Riley suggest several methods to enhance the efficiency of the inference engine, including changing the sequence of conditions in the LHS of rules, and a bias towards simpler, more specific rules. (109) These methods could be used to improve FEED in future versions if problems with efficiency are encountered.

The methodology of the two expert focus groups assumed that the participating experts were able to grasp the meaning and intent of the various scores and considered all relevant aspects while providing the score for each rule or error. This may not be true for the initial rules, when the experts were presumably getting familiar with the process, and towards the end, when fatigue may have set in. A known limitation of focus groups is that leaders usually emerge in small focus groups, dominate the conversation and drive the opinion of the whole group. Conversely, those with an introverted personality may try to avoid participating in the discussion and modify their opinion to match that of the group. Thus, the results from the expert focus groups may not fairly represent the opinions of all the members of the group. This problem may be avoided by following a consistent structured format for the focus group, giving equal time and opportunity to each participant for voicing his/her opinion.

Also, based on the changes seen in the initial 45 unique rules discussed in the verification focus group, it was assumed that rules with both LS and SS of 8 or higher would not need major changes. Consequently, only half of all the unique rules in the KB were reviewed by the experts. This assumption may not be true and the rules in the later
part of the KB may be different from the initial ones in this regard. If this is so, unverified rules may lead to erroneous feedback. In future sessions, it is suggested that ample time be allocated to allow examination of all rules in the KB, and the discussion strictly limited to contentious rules for conserving time.

The verification focus group revealed several deficiencies in the KB, which were corrected in the third iteration, but the KB was not formally verified again using another focus group. This may have been important since the verification focus group suggested several changes to the KB, including major changes (3 points or more) in LS and SS for about 30% of the rules examined. A reiteration of the verification process would have indicated whether the KB reached a stable configuration or not. It is suggested that the KB should be verified in every iteration until a stable, verified stage is reached, and thereafter whenever there are major changes to the KB.

The validation focus group also had limitations. Only chest pain cases were analyzed and therefore, only this aspect of FEED KB was validated. The suggestions and errors in different cases were almost identical, representing a narrow problem domain. The sensitivity and specificity of FEED is likely to be lower than seen in the focus group if it is validated using cases representing all cardiac protocols.

FEED remains untested as yet with regard to usability of the interface, usefulness of the feedback, and effectiveness in changing behaviors. No end-user feedback has been obtained so far. Further testing may reveal problems that need to be corrected before the system becomes deployable.
CONCLUSIONS

The current state of expert system technology is advanced enough to build an automated system for providing after-the-fact feedback to EMS personnel on their patient care documentation. FEED is such a system. It was iteratively developed using CLIPS, Visual Studio .NET and a CLIPS wrapper for .NET. Existing patient care protocols from Alabama State and advice from EMS experts allowed knowledge representation in a rule-based format. FEED was tested on various dimensions. Local syntax checking, consistency checking and verification of KB revealed several problems that were corrected during development. Validation of FEED’s output by an expert focus group revealed a sensitivity of 60% and a specificity of 85% at a particular cut-off value for the validity rating provided by FEED. The review of false positives and false negatives revealed several inaccuracies in the KB, and several potential expert errors. This level of accuracy is reasonable for an initial version of a feedback expert system. After suitable corrections and additional development and testing, FEED could be deployed in EMS agencies for further evaluation of the utility of its feedback to EMS personnel.
LIST OF REFERENCES


131. NHTSA uniform pre-hospital EMS dataset version 2.2.1: Final documentation and data dictionary. 2006 [cited 2007 Mar 29]. Available from:

http://www.nemsis.org/media/pdf/NEMSIS_Data_Dictionary_v2.2.1.pdf


APPENDIX

INSTITUTIONAL REVIEW BOARD APPROVAL
Protection of Human Subjects
Assurance Identification/IRB Certification/Declaration of Exemption
(Common Rule)

Policy: Research activities involving human subjects may not be conducted or supported by the Department and Agency adopting the Common Rule 45 CFR 46, June 18, 1991, unless the activities are exempt from or approved in accordance with the Common Rule. Sec. 101(b) of the Common Rule for exemptions, Institutions submitting applications or proposals for support must submit certification of appropriate Institutional Review Board (IRB) review and approval to the Department or Agency in accordance with the Common Rule.

1. Request Type
   [ ] ORIGINAL
   [ ] CONTINUATION
   [ ] EXEMPTION
   [ ] OTHER:

2. Type of Mechanism
   [ ] GRANT
   [ ] CONTRACT
   [ ] FELLOWSHIP
   [ ] COOPERATIVE AGREEMENT
   [ ] OTHER:

3. Name of Federal Department or Agency and, if known, Application or Proposal Identification No.

   [ ] Name of Principal Investigator, Program Director, Fellow, or Other
   [ ] NAME, DEVASHISH

4. Title of Application or Activity
   Verification and Validation of a Feedback Expert System for EMS Documentation

5. Name of Assurance/IRB Review
   [ ] This Assurance, on file with Department of Health and Human Services, covers this activity.
   Assurance Identification No., expiration date, IRB Registration No., IRB Registration No.

   [ ] This Assurance, on file with (agency/dept.), covers this activity. Assurance No., expiration date, IRB Registration/Identification No. (if applicable)

   [ ] No assurance has been filed for this institution. This institution declares that it will provide an Assurance and Certification of IRB review and approval upon request.

6. Assurance Status of this Project (Respond to one of the following)
   [ ] This activity has been reviewed and approved by the IRB in accordance with the Common Rule and any other governing regulations.
   Date of IRB meeting or Expected Review on date,

   [ ] This activity contains multiple projects, some of which have not been reviewed. The IRB has granted approval on condition that all projects covered by the Common Rule will be reviewed and approved before they are initiated and that appropriate further certification will be submitted.

7. Certification of IRB Review (Respond to one of the following IF you have an Assurance on file)
   [ ] This activity has been reviewed and approved by the IRB in accordance with the Common Rule and any other governing regulations.
   Date of IRB meeting or Expected Review on date,

   [ ] This activity contains multiple projects, some of which have not been reviewed. The IRB has granted approval on condition that all projects covered by the Common Rule will be reviewed and approved before they are initiated and that appropriate further certification will be submitted.

8. Comments
   Please note: UAB IRB Protocol Number is X061114002
   Protocol subject to Annual continuing review.
   Title
   Verification and Validation of a Feedback Expert System for EMS Documentation

IRB Approval Issued: 12/8/06

9. The official signing below certifies that the information provided above is correct and that, as required, future reviews will be performed until study closure and certification will be provided.

10. Name and Address of Institution
    University of Alabama at Birmingham
    701-20th Street South
    Birmingham, AL 35294

11. Phone No. (with area code) (205) 334-5789
12. Fax No. (with area code) (205) 334-1301
13. Email: smoore@uab.edu

14. Name of Official
    Marilyn Doss, M.A.

15. Title
    Vice Chair, IRB

16. Signature

17. Date

Sponsored by NIH

Public reporting burden for this collection of information is estimated to average less than an hour per response. An agency may not conduct or sponsor, a collection of information unless it displays a currently valid OMB control number. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to the OS Reports Clearance Officer, Room 501 200 Independence Avenue, SW, Washington, DC 20201. Do not return the completed form to this address.