CHAPTER 4

A DSS for Neonatal Prevalent Disease Diagnosis & Management using Expert System Technology†

4.1. Introduction

A child is the potential of a country, especially, the neonates constitutes a large population group, and also vulnerable or special risk group. The risk is related with growth, development, disease pattern and survival. Thus by improving health status of neonates, proper diagnosis of the neonatal diseases is essential. Scarcity of the neonate experts in rural and remote areas causes the higher mortality rate of neonates [1]. To alleviate the lack of human proficiency and assisting the existing experts for their decision making capacity, an expert system for prevalent disease diagnosis and management would be very useful.

Decision Support Systems for disease diagnosis have become most essential component of now days. It is seen that utilization of these kinds of systems are not very satisfactory in every respect. There is still a tendency to avoid taking help of such kinds of decision support system service by the domain related personnel. This leads to less improvement of medical care to the sufferer.

Despite of having applications of knowledge base and rule base kinds of technology for the development of decision support system, many other techniques have also been used rigorously now. The techniques like Artificial Intelligence, Data Mining, Rough Set Computing, Artificial Neural Network, MLP, Genetic Algorithm and many others algorithms have been using efficiently in diagnosis system development. For an example for finding the patterns in the neonatal knowledge base, neural networks may be the very useful tool.

Among the various systems in the field of diagnostic expert system our system is also working efficiently diagnosing neonatal disease and also giving valuable suggestions in terms of management. The system generates the list of diseases by looking at each sign and symptom, and then match with lab findings to determine how strongly each of the physical findings is associated with the disease under consideration of the neonatal disease.

This chapter presents a rule-based object oriented expert system for prevalent disease diagnosis and management for neonates. It shows good performance as apparent from its performance evaluation.

A child is the future of a nation and every nation should take proper care for each and every child. An estimated two-thirds of childhood deaths occur in infancy, and, in turn, two-thirds of infant deaths occur in the first month of life. In addition to these four million neonatal deaths, primarily due to serious bacterial infections, birth asphyxia, and complications of pre-maturity and intrauterine growth restriction, an estimated 3.9 million pregnancies end in stillbirth [2]. These deaths, at least half, could have been avoided if proper interventions were taken up. Research suggests that integration of maternal and neonatal health care services is the key to the health benefits for mothers and their newborns. This is particularly happens if a given intervention has potential for direct health benefits to mothers and newborns both. Majority of the neonates in developing countries are born and has been given care in rural homes. The available information is mostly hospital based [3]. Neonates not only constitute a large population group, but also vulnerable or special risk group. The risk is related with growth, development, disease pattern and survival. From the commonly accepted indices, it is evident that mortality rates in this age group are higher than adult population especially in developing countries.

Thus by improving health status of neonates, we contribute to the health of the general population. These considerations have led to the formulation of special health services for children all over the world. The study one chapter 3 shows that the incidence of the diseases namely, Neonatal Septicemia, HIE, Metabolic Disturbances, Neonatal jaundice etc. are quite high in the North eastern parts of India specially in Tarai region [4]. If neonates suffer from these diseases and proper management is not available in proper time the mortality and morbidity will increase considerably. To overcome the problems mentioned above proper treatment planning is required. And for proper treatment planning the basic requirement is Neonatologists. But the pediatric expert distribution in rural region is not at all satisfactory [5]. To mitigate the scarcity of the domain expertise, an expert system for prevalent disease diagnosis and management for neonates may help considerably.

Expert System technology has been applied in many medical disciplines for new born babies whom we call neonates [6-12]. To the best of our knowledge, no such comprehensive work has been reported for neonates for such prevalent disease diagnosis and management.
4.2. A Probable Architecture of Expert System

4.2.1. Expert System

According to Britannica Concise Encyclopedia, “An expert system is a computer-based system which may act or respond like a human expert in a particular field. Expert systems are built on knowledge gathered from human experts, analogous to a database but containing rules that may be applied to solving a specific problem. An interface allows the user to specify symptoms and to clarify a problem by responding to questions posed by the system. Software tools are exists for helping designers build a special-purpose expert system with minimal effort. An outgrowth of work in artificial intelligence, expert systems show promise for an ever-widening range of applications. There are now widely used expert systems in the fields of medicine, personnel screening, and education.”

In another views of McGraw-Hill Science & Technology Encyclopedia, “Expert system is the methods and techniques for constructing human-machine systems with specialized problem-solving expertise. The search of this area of artificial intelligence research has emphasized the knowledge that underlies human expertise and has simultaneously decreased the apparent significance of domain-independent problem-solving theory. In fact, new principles, tools, and techniques have emerged that form the basis of knowledge engineering.”

Specifically, Expert system is one of the branches of the Artificial Intelligence. It is an AI program having expertise knowledge level about a particular domain and knows how to use its knowledge to respond properly and more accurately. Here “Domain” is the area or within a range which the task is being performed. Ideally the expert systems process is a substitute process of a human expert. A person without having domain knowledge or having less domain knowledge can solve a problem with the use of expert system. The source of knowledge may come from a human expert or from books, magazines and internet. Expert system sometime called knowledge base system as the in expert system knowledge is having ultimate importance.

4.2.2. Components of Expert System

An Expert system or the knowledge-base system basically contains the following contents and figure 4.2 depicts the Expert System Architecture.

- Knowledge Base:

  The knowledge base consists of specific knowledge about some specific domain. Knowledge base includes both implicit knowledge and explicit knowledge. Much of the knowledge in the knowledge base is not stated explicitly, but inferred by the inference engine from explicit statements in the
knowledge base. This makes knowledge bases more efficient data storage and gives them the power to exhaustively represent all the knowledge implied by explicit statements of knowledge. Knowledge bases can have many different types of knowledge. Gathering the knowledge for the knowledge-base is called knowledge acquisition.

- **Knowledge-Acquisition Interface**

Knowledge-acquisition is the process of expressing knowledge in the knowledge-base. Knowledge engineer uses the knowledge accusation interface to interact with expert and the procedure of gathering the information for incorporating into knowledge-base. It also assists experts in conveying their knowledge in a form which is suitable for reasoning by the computer and even for the knowledge engineer. Knowledge accusation interface mostly capable of data entry, editing, creating rules, checking the syntax, debugging the errors and also validating the inputs. The interface is giving tremendous benefits to the expert system developers by means of generating specific user interface and on screen helps.

- **Inference Engine**

Inference engine is basically a program which infers new facts from known facts using inference rules. The inference engine uses general rules of inference to reason from the knowledge base and draw conclusions which are not explicitly stated but can be inferred from the knowledge base. Symbolic and mathematical both types of reasoning can be done through inference engines. Inference strategies may be varied from different engines because of varsatality in knowledge representation.

There are basically three stages of inference process which is carried out recursively; they are as follows [24]:

- **i) Match**
- **ii) Select**
- **iii) Execute**
In the *Match* stage of inference, working memory contents are compared to the facts and with the rules in knowledgebase. If the match is found then the equivalent rules are placed in conflict set. When all the matches are found, one of them is selected for the execution. The *Selection* procedure is depending upon the number of conjuncts is there in the left. Actually the smallest rule number is selected for execution. The selected rule is then *Executed* with the right hand side rule is then ready for action process. Figure 4.1 shows the Cycle of System Inference.

- **User / Client Interface**

Unlike knowledge accusation interface, user interface also is a part of the program which communicates with the user. Users of the system frequently review a system by its interface. Functionality is not always verified or judged by the user. For avoiding unnecessary errors, user interface is one of the important parts of designing the program. Poor user interface design is the reason why so many software systems are never used. The user interface can be judged by how well it reproduces the kind of interaction one might expect between a human expert and someone consulting that expert. Most users of business systems interact with these systems through graphical user interfaces. The user interface is the part of the program that interacts with the user. It prompts the user for information required to solve a problem, displays conclusions, and explains its reasoning. Thus by designing easy user interface decision making process acts faster.

- **Translator (Rule Base)**

Translator basically used for translating the acquisitioned knowledge and make them prepare for generating rules. In fact, the process frames the rule base.
Converting the information into knowledge is the vital task for an expert system cell. Translator may of various categories depending upon rule specific transformation. For the interpretation process Abstract Data Type (AST) is used. Implementation of this AST makes interpretation easy and efficient by the Inference Engine. This can be used for generating complex and useful rules because of its expressiveness nature. A rule AST is converted into an equivalent form suitable for storage in the Knowledge-base.

- **Object / Class (Rule Base)**

  Expert System Cell may of different paradigm. In an object oriented approach the cell can generate specific Object and Classes using different categories. Parameters involved in producing class and thereby creating instances of the classes, which is called object.

### 4.2.3. Expert System Shell: Level 5 Object

The Expert System Shell is a component integrating for creating, editing, and executing rules on any problem-independent environment. Afterwards applying inference procedure, make the knowledge-base based on the rules. For developing Expert systems a variety of tools and languages can be applied. Examples of such languages are PROLOG and LISP. Developing the expert system using this kind of languages needs specialized programmer or developer and also it takes longer time to build any knowledge-base. Comparatively, Expert System Cell has a tendency to take less time. Because of having good developmental tools and easy interfaces for development comparatively less expertise person may also successfully use this.

For our expert system development and research purpose we use an Expert System Cell named Level 5 Object. It has capabilities like display and rule editors, debugging aids, and graphical and imaging capabilities or interfaces. LEVEL 5 Object is a product of Microsoft Windows (release 3.0 or higher) by the Information Builders Inc, USA. It has an integrated array of powerful tools like GUI development, Forms and Display builders and also has got capability to chain more than one knowledge base together. Besides this the Level 5 Object is having following facilities:

- Values editor to determine the status of object and attributes.
- Object, display, rule editors.
- Tool for monitoring the session.
- A database interface.
- Windows editor.
- An agenda editor (for specifying expert system goals).
Knowledge tree editor and navigator.
Import/export facilities.
Rule Talk Editor.
Reasoning Editor.
Run and Execute facility.

As the name suggests, Level 5 Object, is the development tool based on an object-oriented design and implementation. Object oriented concept is now an emerging trend in developing expert system now a day. Features related to Object-oriented structure in Level 5 Object Cell are as follows:

- Objects can inherit the properties of other objects.
- Object attributes can have attachment that characterize their use and definition (e.g., search method, initial values, displays, queries, "when changed" procedures).

Figure 4.2. Knowledge Base Architecture.
User displays can be developed that contain objects of many different types and can allow the user to select, modify, or determine their value and status.

Systems that are object oriented can be easily modified because it is easier to determine where and how knowledge is represented.

Interfaces to different forms of knowledge are available.

There is huge difference between writing a program in OOPs concept and in conventional programming process. OOPs concept actually increase the productivity of the developer or the programmer.

One of the major advantages of using Level 5 Object was the ability to use different inference procedures. Both forward chaining and backward chaining inference can be possible here.

4.2.4. **Forward-Chaining Inference**

The forward-chaining inference engine starts with known facts or data and infers new facts about the situation based on the information contained in the application. The process continues until no further conclusions can be deduced from the initial data. Because the process is triggered by initial events, it is also referred to as data-driven or event-driven reasoning. It involves the use of "when changed" or "when needed" procedures and demons.

Procedures and demons are algorithmic statements that direct the computer to process the knowledge in a previously specified manner. The use of "when changed" procedures involves a change in the value of an attribute which is triggered by the user selecting a response to a question and then clicking the mouse on the "enter" push button. The push button is attached to an attribute that is linked to a "when changed" procedure that contains a set of IF-THEN-ELSE procedural statements used to evaluate the user's response. Another use of forward chaining involves the attachment of displays to a push button. This in effect "advances" the execution of the expert system in a predetermined logical manner.

4.2.5. **Backward-Chaining Inference**

The backward-chaining inference engine starts with a desired goal or objective and proceeds backwards along a chain of reasoning in an attempt to gather the information needed to verify the goal. In LEVELS OBJECT, a chain of reasoning can
The goal's search order defines the specific combination of these items and determines how and in what order the backward-chaining inference engine will attempt to verify the goal. It involves the execution of rules in search of a goal. The goal in this case is a conclusion or recommendation. The user supplies information about the structure or procedure and the system attempts to find a suitable goal.

This chapter presents a rule-based, object-oriented expert system for prevalent disease diagnosis and management of neonates. In section 4.3, the common causes of neonatal deaths have been pointed out. Even this has already been discussed in chapter 2. Section 4.4 explains the knowledge engineering process. Section 4.5 presents the system design and implementation. Section 4.6 describes the evaluation and performance of the expert system. In the last section that is section 4.7, we present our conclusion of the study.

4.3. Common Causes of Neonatal Deaths

In previous chapter, we have discussed about the common causes of neonatal death in global and national respect as well. Here again the causes that have been discussed in the research study. The neonatal health problems are mentioned in Table 4.1. The probable health problems of a sick neonate are summarized below.

Table 4.1. Neonatal Health Problems.

<table>
<thead>
<tr>
<th>Neonatal Health Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Birth asphyxia, Neonatal Sepsis, Hypoxic-Ischemic Encephalopathy (HIE), Preterm, Low birth weight, Delayed breastfeeding, Problems in breastfeeding, Diarrhea, Hemorrhage, Conjunctivitis, Skin Infection, Abnormal Jaundice, Me conium Aspiration, Hyaline Membrane Disease (HMD), Pneumonia, Upper Respiratory Infection (URI), Hypothermia, Umbilical Sepsis, Tetanus, Convulsive Disorder, Unexplained fever, Failure to Gain Weight.</td>
</tr>
</tbody>
</table>

4.4. Knowledge Engineering

Knowledge Engineering is the part of expert system development. Basically, Knowledge Engineering refers to a process of building, maintain and developing of
knowledge based expert system [13]. Integrating the domain knowledge in computer system for solving complex problem is the main task of Knowledge Engineering. To engineer the knowledge a high quality of human expertise is absolutely required [14].

4.4.1 Process of Knowledge Engineering

There are three processes that basically used in knowledge engineering process for the development of any expert system. Fig. 4.1 Shows the KE activities or processes:

![Knowledge Engineering Process](image)

The process of Knowledge Engineering is broadly categorized in three types:

1. **Acquisition of Knowledge**
2. **Analysis & Modeling of Knowledge**
3. **Verification of Knowledge**

4.4.2. Acquisition of Knowledge

Accruing expert knowledge is a crucial part of knowledge engineering. It is a subsystem that helps the expert for building up the knowledge bases. To develop an actual expert system and to have the highest performance, knowledge acquisition plays one of the most vital roles during the development of the system. This process actually gathers information which is relevant about the domain from domain experts primarily. Here the system’s knowledge had been acquired through three main sources namely, (a) Medical Experts, (b) Published literature, (c) Real observation at the hospitals and (d) Text Books [16, 21].

Three Experts in Pediatrics, each of them having 12-33 years of experience were the first source of the main knowledge. The main experts are from Dept. of Pediatrics, North Bengal Medical College and Hospital and others from said department and other hospitals were consulted through structured interviews. To record the knowledge extracted from the domain experts, forms were prepared in consultation with them. The experts were asked to give their judgments for different sets of possible real observation and cases. In this phase, it was decided to involve multiple experts in the
knowledge acquisition process. Thereafter, knowledge was gathered from various research publications including books and workshop reports published by Indian Pediatrics, some other journals and text books [16, 21].

For real observed cases, frequent visits were conducted at neonatal unit of Dept of Pediatric Medicine of North Bengal Medical College & Hospitals, Siliguri, West Bengal, India. All real life physical observations were recorded and incorporated in the knowledge base of the system.

4.4.3. Analysis & Modeling of Knowledge

The very next process after knowledge accusation is analyzing and modeling of the knowledgebase. The unstructured knowledge as acquired from the above three sources of knowledge was then structured by the knowledge engineer. The knowledge was then represented in an Object Oriented form for later implementation. An Object Oriented Approach to KR (Knowledge Representation) schemes is more structured than other well known schemes and improves consistency, understandability, maintainability and modifiability of the knowledge base [16]. The knowledge in the system is stored in as group of objects. Each group is represented by a class with its attributes. A class defines the general properties of structure of a group of objects. Attributes describe the object’s important characteristics. The knowledge library class serves as a database. Various pictures were collected from North Bengal Medical College during field surveys and from experts. Both the experts and medical persons are accustomed with various medical terms like “APGAR”, “HIE” etc. which have been incorporated as attribute values.

We have applied an approach called “Select and Modify” [17]. As per the needs of the system the model is selected and modify whenever there is a need. This process is again subdivided into four activities. They are:

i. Selection of an interpretation model according to a set of selection criteria.

ii. Evaluation of interpretational model and investigate it for checking suitability for the application.

iii. Modification of interpretational model for making suitable applicability. and lastly,

iv. Domain knowledge accusation for modified interpretation model.

4.4.4. Verification of Knowledge

Knowledge verification is the phase whereby making quality assertion of the acquired knowledge. There are two points of concern: Establishment of review procedure and Conflicts Resolving for Multiple Expert procedure.
Establishment of Review Procedure

There are three phases of establishing the review procedure. They are knowledge elicitation, knowledge analysis and modeling, and implementation.

- At the elicitation stage, reviewing is conducted by letting the domain expert’s opinion on the results of the knowledge elicitation sessions.
- At the analysis and modeling stage, the domain experts review the filled forms describing the domain knowledge. The knowledge engineer performs this activity by walking through them with the presence of the domain experts.
- Lastly at the implementation stage reviewing is conducted by letting the domain experts review any early prototype and finalize the knowledgebase for implementation.

Conflict Resolution for Multiple Experts

Because when two experts give different knowledge for the same thing, then trying to resolve this conflict yields more reliable knowledge, hopefully, agreed upon by both of them. Thus multiple experts’ conflict resolution is considered as a way of verifying the acquired knowledge. If no consensus is reached the expert who is recognized to be more specialized in the area of disagreement is considered for knowledge management.

4.5. System Design and Implementation

The system is designed to aid the decision-making process for identification of common diseases in neonates. It also selects the controls measure taken/ management for confirmation of disease. A system flow diagram of the system is shown on figure 4.4. There are basically three phases in the expert system. During the first phase, preliminary identification of the sick neonate in the field can be done on the basis of neonatal information observed in the field. This identification is related with complain/ sign-symptoms of the sick baby. This identification is further confirmed with the knowledge related to characteristics such as Birth Weight, APGAR Score, Gestational Age etc. The system uses domain knowledge in this phase. The system asks for the inputs related to the sick neonate history and sign & symptoms for which he/she is suffering. Firstly the system asks for history related with mother and child, like Mothers age, LMP, DOD ( for calculating term/preterm/post term baby), Parity(no of children), Mothers Blood Group, History of hereditary diseases, weight of the baby( to find out LBW / VLBW/ ELBW / NORMAL or finding out weather the bay is AGA(appropriate for gestational age/ SGA(small for gestational age )/LGA(large for gestational age) etc.
After that it asks for the primary signs such as lethargy, refusal to suck poor cry, poor weight gain, hypothermia, sclerema, excessive jaundice, bleeding, GI disorder, seizure, sluggish neonatal reflex etc. The production of rules of inference are used a good level of accuracy is achieved in the resulting identification. The forward chaining of reasoning is performed with the rule based knowledge.

For inferences in the system “IF {I} AND {C} THEN {D} CF” rules have been used, where {I} is the information of the sick baby, {C} is complaint or symptoms and {D} is the differential diagnosis. Here CF is the certainty factor which often call confidence factor associated with the rule. The confidence factor CF attached to the most of the rule represents actual confidence of that rule. The value assigned to CF ranges from 0 to 100. This knowledge is taken from expert doctors after few interactions. Multiple experts in the pediatric domain actually have suggested the confidence factor (CF) at the time of knowledge acquisition.

The format chosen for the definition of rules allows flexibility in structuring the knowledge [18, 20]. A predecessor of any rule may be a composite of number of clause connected through logical operation OR and AND. In addition to the “if-then” structures, common to all rule-based systems, knowledge representation supports a new type of concept –the criterion – that enables a more compact and natural way of expressing rules of the form: “if at least \( n \) conditions are satisfied then…”[22].

Knowledgebase structure of the system having classes, instances, attributes and their types and other details have been attached in Appendix A.
Phase - I

Input

Information about the sick baby
(I) = (I₁, I₂, ..., Iₙ)

Complain/Sign-Symptom
(C) = (C₁, C₂, ..., Cₙ)

Inference Engine

Primary Output

Primary Diagnosis

Phase - II

CLINICAL INVESTIGATION

For Diagnosis Disease Conformation

Inference Engine

FINAL OUTPUT

Final Diagnosis

Phase - III

INPUT/RECEIVE FIELD STATUS

{F} = (F₁, F₂, F₃, ..., Fₙ)

Inference Engine

OUTPUT

Control Management

Figure 4.4. System Flow Diagram.
4.6. Evaluation of the System

The system has been analyzed for 68 real field cases in different neonatal disease category. The beginning of using the expert system causes no additional clinical work. Only minor changes in working practice required for the auxiliaries, who normally perform the sampling and for noting the mother’s name and hospital identification number before analysis.

The system has been designed for the simplification in the use. Users’ quick acceptance of the system is also endorsed with this. The multiple sticky labels for patient notes eliminated the time consuming and error prone process of transcription, and this helped significantly with user acceptability. For trail run, some randomized cases taken from the set of real field case used for evaluation. The study has found that 90% accuracy or certainty of the system. This has been verified by the expert pediatricians. Two case studies are presented in Appendix B as representative examples.

4.7. Conclusion

This chapter presents an integrated environment for the development of knowledge based decision support systems applied to diagnose and manage the common neonatal diseases. There are few researches also used this kinds of system but in various other domains [7,12,23]. This system provides graphical support for every phase of the development cycle, from knowledge acquisition and addition to reasoning explanation and knowledge-base validation. It is an interactive Object Oriented Rule-Based Expert System developed to meet the needs of such places where is scarcity of the medical expert for diagnose of neonatal disease and give proper clinical control management, especially in hospital, and primary health centres in remote villages.

Appropriate Graphical User Interface components incorporated in the system a facility to select and deselect multiple options from the easy to use menu. It needs almost no training for its user. Easy and commonly used linguistic variable used in inputs and output which makes this system more effective. Even less-trained person seeking expert system guidance for proper neonatal disease diagnosis and treatment plan practices also gets help out of using this system. The extensive system testing and validation process in conjunction with the overall structured project methodology, has greatly contributed to the expert system reliability and user-acceptance. It proves itself as a useful and beneficial tool and supporting hand for the domain experts.
References


