CHAPTER 3

AI TECHNOLOGY FOR LANGUAGE LEARNING

There are two extreme views about a computer's ability to be intelligent. One is that intelligence is a rule governed activity and that if we can figure out all the complex rules that contribute to intelligence, it is possible to create a computer as exactly intelligent as humans. A contrary view is that intelligence is a God given gift unique to humans which cannot be defined, described or duplicated. The truth in fact lies somewhere in between. Some intelligent activities are rule governed. We can try to make the machine initiate this activity. There are other areas of intelligence which are probably not rule governed, or they are rule-governed but cannot be described or explained. If we can make a machine simulate even small parts of intelligence then it can be said to exhibit Artificial Intelligence.

There are many definitions of Artificial Intelligence. The following two definitions are relevant to the study. "Artificial Intelligence is a discipline concerned with the building of computer programs that perform tasks requiring intelligence when done by humans"1, is a definition from the point of view of a computer programmer. A definition from the psychological point of view is, "Artificial Intelligence is an approach to understanding behaviour based on the assumption that intelligence can best be analysed by trying to reproduce it"2.

2. Ibid.
Artificial Intelligence (AI) is an interdisciplinary study involving cognitive psychology and computers. Garnham points out that as the reproduction of human intelligence is carried out through simulation by the computer it is actually a computer science discipline. We can arrive at a compromise without hurting the sentiments of the psychologists who believe Artificial Intelligence is their discipline and the computer scientists who think Artificial Intelligence is more computer oriented, by arriving at the conclusion that Artificial Intelligence is the study of intelligent human behaviour and that its goal is to produce machines that can simulate human intelligence.

For the purpose of the present study we can define Artificial Intelligence as the computer based solution of complex problems through the application of processes that are analogous to the human reasoning process. The theory behind Artificial Intelligence research is concerned with developing computer systems that simulate human behaviour. It involves intelligence normally associated with human intelligence. It concentrates on problem solving in the abstract.

"Artificial Intelligence is not the study of computers, but of intelligence in thought and action. Computers are its tools, because its theories are expressed as computer programs that enable machines to do things that would require intelligence if done by people".

There are two stands with regard to Artificial Intelligence - strong AI and weak AI. The strong AI position is taken by its founders McCarthy, Simon and Newell. They equate the intelligence of people with computer programs. As they see intelligence as symbol manipulation they believe that

intelligence "can take place as well in an electronic machine as in the human mind". Marvin Minsky goes to the extent of proclaiming that "computers will become so intelligent that people will just have to rely upon being looked after as pets".

John Searle and Hubert Dreyfus take a weak AI position. Searle asks us to imagine a situation where a person locked inside a room engages in a dialogue with Chinese speaking people outside the room by exchanging printed Chinese symbols to and fro. Unless the English speaking person inside the room is given a book with all the rules of the Chinese language written in it in English, he cannot communicate with the Chinese outside though he may appear to be doing so. The status of the computer is analogous to the man in the room who communicates without understanding. So, however intelligent a system may appear to be to the computer programmer, it does not really understand what it is doing. Dreyfus also questions the idea of formalising reality. The myth that the mind is an entity that can be separated from the things around it is carried on erroneously to the area of A.I. according to Dreyfus. This study takes a weak AI stand.

A more acceptable doctrine for determining the intelligence of a system is the Turing Test. While communicating if we are unable to say whether the communicator at the other end is a human being or a computer then that system is said to have passed the Turing test.

5. Ibid, p.80.
No system till today has passed the Turing test. And no system ever will because human emotions like perception and intuition cannot be replicated by any machine.

Considering the maxims of strong AI, weak AI and the Turing Test can we come to the conclusion that non Artificial Intelligence systems are unintelligent? Or can we say that a program written in PROLOG or LISP is intelligent and that a program written in BASIC or FORTRAN is not intelligent. Why then is Artificial Intelligence superior and more sophisticated than a simple program in BASIC. A study of the characteristics of an Artificial Intelligence system beginning with its history detailed later in the chapter provides a convincing answer to this question.

3.1 INTELLIGENT CALL

The use of Artificial Intelligence in developing CALL programs led to an offshoot of CALL namely Intelligent CALL. Before trying to understand what Intelligent CALL is, it is important to note the factors that determined the need for transition from CALL to Intelligent CALL. The following factors determine the transitory part of the last few decades.

1. The euphoria over CALL had died a natural death. Novelty had waned with time which was natural. Studies conducted in this connection revealed that the group of students taught using CALL methodology performed better than those taught through the conventional method. The decisive factor might have been greater self-motivation of the CALL students owing to enthusiasm for novelty.
2. New ideas for CALL material had burned out. Experiments with drill and practice were anything but novel. Research to improve reading and writing using CALL were commendable in their efforts. But it did not give much freedom to the user. Especially in the field of writing, curtailing the freedom of the user was a disadvantage. If the CALL program provided freedom to the user, it became difficult for the computer to monitor the situation.

3. Further, there was a general lack of ingenuity in programming CALL material. Attention was diverted to improving teaching methodology.

4. Teachers' attitudes were negative to CALL. They have been questioning its pedagogical value.

5. The computer technology has undergone a sea change. Keeping in tune with the sophisticated technology, it is necessary to graduate from CALL programs in BASIC to CALL programs in PROLOG.

In intelligent CALL, there has been an earnest attempt to build a system that knows the subject it is handling. The instructions should not be predetermined. The computer should be truly interactive and enable the learner to have a free natural dialogue with it, which means the computer should be able to understand and respond in natural language.

The significance of the need for Intelligent CALL is clear from the declaration of Brian Farrington that the development of CALL is at the end of one chapter and the start of another. He adds that we are moving into
an entirely new phase, the most distinctive feature of which is sure to be the appearance on the scene of Intelligent Tutoring Systems for language learning, Intelligent CALL. Brian Farrington defines Intelligent CALL as

"CALL materials written by computer scientists instead of being written by language teachers or linguists or that it just means CALL materials with programs in PROLOG or LISP instead of BASIC.... intelligent CALL' can be fairly applied to any exercise in which an attempt is made to get the system to process language in a way that approximates or appears to approximate, to that used by human beings".

This study gives the following definition of Intelligent CALL (ICALL).

An Intelligent CALL program is a CALL program which employs AI technology to perform a task which when performed by a human being requires intelligence. Such a program is said to be intelligent to that extent.

This working definition of Intelligent CALL (ICALL) for the purposes of this study does not specify the lower limit or the upper limit of intelligence so as to accommodate the simplest of intelligent activities within their realm of CALL programming and at the same time aim at preparing CALL material that can simulate the most complex of intelligent behaviour


7. Ibid.
for language learning activities. But an Intelligent CALL program need not necessarily be written in Prolog or LISP as claimed by Farrington. A program in wordstar can also be an Intelligent CALL program if it is able to perform intelligent indexing. Because of the inherent abilities of AI languages like LISP and PROLOG to exhibit intelligence, ICALL programs are generally programmed in AI languages. A study of AI beginning with its history is necessary for a better understanding of the term "intelligent" in Intelligent CALL.

3.2 HISTORY OF AI

Active research in producing machines which think began in 1940. Immediately after World War II, a very thin dividing line existed between AI research and work on electronic stored-program computers. The concept of thinking machines was being developed by Alan Turing who was also simultaneously working on the Manchester Mark I computer and the idea of how to judge the intelligence of a machine.

Towards the end of the second world war, scientists in Britain and America were attempting to create an electronic device that could perform complicated numerical computations according to the instructions given in stored programs. There were no two opinions about the numerous applications of such a machine, the computer. But the capabilities of this computer was limited to numerical operations.

Turing, convinced with the success of formal logic, was among the first to suggest that the applications of the computer should be extended to logical operations (i.e., instead of just '+', '-', 'x', '/' etc., 'and', 'or', 'not'

etc. should also be included). He believed that these general operators could be used to assemble the more specialized numerical operators needed for arithmetic calculations. Programs that included arithmetic and logical operators would have the dual advantage of executing numerical problems and symbolic problems. But the builders of computer systems did not realize the significance of logical operators until the eighties.

As scientists explored the possibility of using computers for non-numerical operations, psychologists were trying to simulate human behaviour through computer software. Eventually the endeavours of the two schools merged giving rise to the birth of an interdisciplinary field that involved symbolic processing and human problem solving - Artificial Intelligence.

The term Artificial Intelligence was coined by John McCarthy, during the Dartmouth Conference held in the summer of 1956. The conference was jointly organised by John McCarthy and Marvin Minsky. By this time, Herbert Simon and Allen Newell had implemented the Logic Theorist program. McCarthy, Minsky, Simon and Newell are considered the co-founders of Artificial Intelligence.

Simon and Newell founded the A.I. Laboratory at Carnegie Mellon University. McCarthy and Minsky founded the A.I. Laboratory at M.I.T. McCarthy later moved from MIT to Stanford University and founded an A.I. laboratory there. At about the same time, the department of Machine Intelligence was founded by Donald Michie at the Edinburgh University. These four universities were the premier research universities in the area

of A.I. The term A.I. was controversial for some years till it eventually came to be accepted.

Even before the Dartmouth conference, there had been many investigations on the nature of intelligence. But it was directed towards understanding intelligence as manifested by people. It was only after the advent of the computer that research in the direction of simulating intelligent behaviour by machines began.

The six major areas of AI are\textsuperscript{10}

1. Knowledge Based Systems
2. Natural Language Understanding
3. Pattern Recognition
4. Intelligent Computer-assisted learning
5. Speech recognition
6. Models of human cognition

The six areas of Artificial Intelligence is illustrated in the form of a diagram in Fig. 3.1.

Early AI research involved the building of systems that understood natural language. HEARSAY was the first speech-understanding system. It was developed at Carnegie Mellon University in the late 1960s. It was one of the early attempts in Artificial Intelligence's longstanding goal to develop computer systems that could interact with users through natural language.
One possibility of achieving the computer's natural dialogue system was to limit the input to typed dialogue information, specifying the vocabulary and restricting the sentence structure so that the user could know to what level he could interact with the computer. INTELLECT was one such natural language system. It used typed-entry and a limited vocabulary to answer data base enquiries.

Attempts to understand spoken input were made by determining the speech wave pattern created when the user spoke into the microphone. This threw up the further problem of designing a system that could recognize human speech sounds. HEARSAY was the first system to make a breakthrough in dealing with spoken input. It did not just deal with sentences but connected speech as well.

HEARSAY's handling of connected speech can be explained by considering the difference between words spoken in isolation and words uttered as connected speech. For example, let us examine three isolated words:

1. bread
2. butter
3. and

Consider the connected speech bread n butter. It is uttered in one breath. There is no pause between the words. Sounds are merged and phonemes are left out. It is difficult for a machine to understand connected speech which is so easy for humans.

Studies have shown that in normal conversation humans understand not just by recognizing speech sounds but by anticipating
meaning depending upon the context. Human intuition plays a decisive role here. Also, the speaker's and listener's inherent "mini-grammar" helps them understand the meaning of sentences.

HEARSAY handled database queries by accepting speech input in the form of waves. The output consisted of hypotheses based on the input. It then guessed what the speaker had said. Different independent knowledge bases helped analyze the speech input. Each knowledge source acted as the contributor to the blackboard - the common memory. There was a master control program which decided its next course of action based on the latest information in the blackboard.

HEARSAY was thus a system made of several subsystems:\(^\text{11}\)

(i) rule-based sub-system to analyze sound signals
(ii) sub-system checking into the blackboard to see if the sound signals are recognizable phones.
(iii) sub-system analyzing groups of phones and syllables merged into words based on rules built into it.
(iv) sub-system analyzing sentences on the blackboard seeing if they match at least approximately with the data base queries that will be initiated whenever an approximate pattern is found.

The HEARSAY II project was completed in 1975. It achieved the remarkable feat of handling a vocabulary of about 1,000 words with limited

grammar. It was successful in interpreting the user's queries and responding to them. It proved that it was possible for a computer to enter into a natural dialogue with a user.

HEARSAY also won the conquest for symbolic, heuristic programming over other conventional methods when it came to dealing with meaning. That multiple sources of knowledge contributed to the analysis and solution finding paved the way for greater research to take over.

ELIZA is another natural language program in LISP created by Weizenbaum for psychotherapy. Though its application in psychotherapy is limited as Weizenbaum himself admits, it is a remarkable program that interacts with the user in natural language. It operates on the principles of semantic matching.

When the user enters into dialogue with the machine, ELIZA picks up the key words and matches it with the phrases that have been built into it. There are many loopholes in the program. For eg., it has stock phrases with which it replies and slightly alters the skeleton phrase keyed in by the user without understanding a word of what it is doing. All said and done, the effort is to be appreciated because it is an earnest beginning in AI's natural language understanding systems.

3.3 AI IN EDUCATION
3.3.1 Conventional Educational Computing Before AI

When Computer Assisted Learning first made its mark, drill-and-practice programs on the lines of Programmed Instruction were popular. The instructions given in the program formed the stimulus and the user provided the response by choosing the correct answer. If the answer was
correct something like "congratulations, go ahead" appeared on the computer screen.

Criticism has been severe on the inanity of such drill and practice programs. It was tantamount to turning the clock back to an earlier era in education. The emphasis was misplaced. Rote learning rather than understanding took precedence in priority.

The need for focus on the learner's difficulties, especially when the learner types in an incorrect response and lack of provision in the computer program to be more encouraging and sympathetic towards the learner caught the teacher's and programmer's attention. In this context, Hartley talks about providing learning options that can enhance learning. "Only by allowing the student himself to make choices, to justify them and see their effects, will he learn about the process of making educational decisions; only in this way will he become self-evaluative and learn how to learn."\(^\text{12}\)

The computer technology in spite of its limitations could not be blamed for the mediocrity of CAL software. What then went wrong? The scientists were aware of what the computer could do. The teacher was aware of what to teach. The learner was clear in his mind about his needs. But there was no single person who could understand the computer scientist, the teacher and the learner and put the ideas of all of them into a computer program. The need of the hour was collaboration. Artificial Intelligence is a collaborative study demanding the knowledge of psychology and computing which could be put to use to overcome the deficiencies of CALL.

3.3.2 PSYCHOLOGY and A.I in EDUCATION

As stated in the history of A.I research, educational psychology laid the foundation on which the edifice of AI was built. AI was based on the precepts of Cognitive Science.

As psychology grew away from Behaviourism blaming it for being narrowly conceived, Cognitive Science, concerned with mental processes in human learning and memory came into its own. Cognition refers to the processes through which information coming from the senses is transformed, reduced, elaborated, recovered, and used. Cognitive processes are important in perception, attention, thinking, problem solving and memory. These aspects involving educational psychology are intimately related to the learning processes through AI.

According to Richard Ennals, "educational psychology and cognitive science...should be given primacy in the educational context over computer science considerations. In return, it provides a medium in which theories can recognizably be formulated and formalized, constructively compared and subjected to practical tests of efficacy"¹³. As a result of the collaboration between cognitive psychology and computer studies, the works in psychology of Piaget, Bruner, Gagne, Paski and Scott have left an indelible mark on the building of AI systems. Their work was a valuable contribution to the starting point for AI in education. Later Artificial Intelligence research in education saw the emergence of Intelligent Tutoring Systems as a potential teaching aid.

3.3.3 Intelligent Tutoring Systems (ITS)

Intelligent Tutoring Systems may be an answer to the language teacher's grouse against the ubiquitous drill and practice CALL software. Intelligent Tutoring Systems were considered 'intelligent' because they were able to respond in a manner more interactive than CALL. Scientists working on intelligent systems are trying to simulate the activities of the human brain. Though it is impossible to recreate the thinking and decision making ability of the brain on a microchip, tutoring systems based on the principle of Artificial Intelligence have undoubtedly made a breakthrough in 'understanding' natural language, however limited it may be.

Intelligent systems have been used for spelling analysis. When a word is misspelt it can be detected immediately by the spelling analyser and the type of error involved in misspelling classified.

ITS have also been successfully used to create software in syntax and morphology. By using intelligent systems, learning became exploratory. The learner was able to see for himself how the program worked at any stage of its working. It improved his understanding of the program and the target lesson.

Higgins enumerates five factors for an education system to be characterized as 'intelligent'\textsuperscript{14}.

1. representation of the subject knowledge.

O'Shea and Self remark that, "If we want to build computer-assisted learning programs to answer

unanticipated questions and to individualise teaching we must try to make the necessary knowledge available to the computer”\(^\text{15}\).

2. model of the learner
   i.e., an account of strategies that learners may adopt in solving problems together with a means of storing the history of how the current learner has dealt with the problems so far presented.

3. a means of self-adjustment; the machine too has to be able to learn from self-experience.

4. a channel by which the machine can explain its own decisions and procedures in a way which corresponds to human ways of thinking.

5. a language understanding system, so that the machine can make sense of the learner’s inputs in natural language.

If all the above mentioned features are present in a teaching system we can call it an intelligent teaching system. ITS share the same drawbacks of the microcomputer despite their intelligent behaviour.

- they have no knowledge of their own. The knowledge they claim to possess is more artificial than intelligent or real
- their performance is incomparable to the interaction between the teacher and the learner

inspite of understanding the input in natural language they cannot understand the learner's feelings.

though they attempt to imitate the human brain there is no emotional involvement.

Therefore, their intelligence is very restricted and narrow in its applications.

The architecture of an ITS is prescribed by Anderson et al as having a four part model - a Tutor interface capable of rich interaction with the learners, a Tutoring knowledge system that knows how to teach Bug Catalogue, an Ideal student model that knows who to teach and a system that knows what it is teaching. Each of Anderson's models are split into twin components by Yazdani so that

- Domain knowledge + Inference Engine = Tutor Interface
- Bug Catalogue + User Model = Tutoring knowledge system
- Tutoring Skills + Planner = Student Model
- Explanation Patterns + Student Tutor Interface = Teaching System

The left hand column indicates Yazdani's models and they have been equated with Anderson's model of an ITS given on the right hand side column to establish that the two can be equated.


Thus, tutor interface should have specific domain knowledge and an inference system. The knowledge system for tutoring in turn must have a bug catalogue and a user model. The student model should have the skills required for tutoring and a planner of the tutoring method. The teaching system on the whole should be able to explain any query from the student at any stage of the program and more importantly should have an efficient interface between the student and the tutor.

To summarise, an ITS should have a domain specific knowledge base, an inference engine, a bug catalogue, tutoring abilities that should essentially include facilities for explanation and a natural language user interface.

The success of ITS depends on the nature of knowledge representation. We can have AI programs with different types and levels of learning strategies, explanations, tutorials and remedials in the domain knowledge. A decision making procedure can be incorporated which can help the student choose the style and type of learning according to his knowledge. This kind of decision making by the learner gives more freedom to the learner and more scope for learning. For example, in TUTOR, an Intelligent Computer Assisted Instruction system developed in the UK by Logica in 1984, the student can pick a lesson from the syllabus and choose the type of training he wants:

- novice
- skilled
- self-test
- examination
In Novice, tasks are followed by explanations with remedial feedback. The system assumes that the learner has undergone novice training in the case of skilled training and provides tasks and remedials. The student handling TUTOR for the first time can use self-test to assess himself. To a student willing to examine himself after undergoing TUTOR's training, the examination training can give exams and the results can be stored.

It is not clear how intelligent we want ICALL programs to be. We are quite satisfied with the slow and moderate progress intelligent systems are making. Even now in 1995, when we say 'intelligent' system we don't fantasise a system that can do and understand everything. Neither are we sure about the level of intelligence that ICALL programs should exhibit.

In this context, semi-intelligent systems may be a stop-gap solution on our way to achieving the utopian state of completely intelligent systems. A semi-intelligent program is a "conventional dumb' CALL program which employs limited, or not so limited, AI techniques to increase the range of language it can process, give it more power and make it more flexible, but which does not attempt to model the competence of a natural speaker".\footnote{V.J., Cook, and D. Fass, 'Natural Language Processing by Computer and Language Teaching', System, 1986, 14,2,pp 163-70.}

Examples of semi-intelligent systems are syntactic parsers, cloze exercises. LITTRE is a semi-intelligent tutorial program for translation. It does not use a parser. It is an Expert System which requires an exhaustive knowledge base because it works sans the parsing technique. The semantic range of processing is wide and admirably sensitive to contextual and stylistic aspects of the language. LITTRE checks the learner's input with its
data base. The data base is capable of syntactic analysis at various levels. LITTRE abides by the precepts of Kemmis' revelatory learning\textsuperscript{19}.

LITTRE is limited in its intelligence as it requires laborious man hours to prepare the knowledge base. It is a very time consuming project as all the errors have to be classified and analysed and the possible responses to be anticipated by the programmer.

We do not need a fully-automatic-intelligent system for teaching language as fully-automatic intelligent system may not give much scope to make mistakes and improve upon it, something fundamental to the learning process. By being refined and 'finished' in its approach it may tend to be hollow.

In a semi-intelligent system there can be constant revision and updation of the knowledge base. This is an advantage to the programmer who has failed to anticipate certain things or discovered new information from testing the software with the learners.

It is necessary to acquaint ourselves with the features of AI before embarking upon the development of ICALL programs.

### 3.4 CHARACTERISTICS OF A.I.

A true AI system has the following features:

1. representation
2. decoding

\textsuperscript{19} See 'Learning Paradigms' in Chapter 1 of this thesis, p 4.
3. inference
4. control of combinatorial explosion
5. indexing
6. prediction and recovery
7. dynamic modification
8. generalisation
9. curiosity
10. creativity

**Representation:** Representation is knowledge representation which involves presenting knowledge as information and as relationship between these bits of information.

**Decoding:** In the AI context, the ability of a program to understand all the implications of the stated problem that would help the program to arrive at a solution faster.

**Inference:** The ability to make inferences or arrive at new information based on available facts.

**Control of Combinatorial Explosion:** The problem with searching everything is that their time complexities grow exponentially with problem size. This is called combinatorial explosion. Combinatorial explosion can be controlled by adopting heuristic search and pruning the problem size and thereby making the search more efficient.

**Prediction and Recovery:** Many programs have been developed that are able to learn a single concept from training instances. Some important tasks in this case are:

- **Classification**: the system is presented with new unknowns and is asked to classify them as positive or negative instances of a concept.
| Prediction | if the training instances are successive elements of a sequence, the system is asked to predict future elements in a sequence. |
| Recovery | to identify missing components / elements. |
| Data compression | the system is given all possible instances (the full instance space) and is asked to find a concept that compactly describes them. |

**Dynamic modification:** Change in the content of rules or facts as and when the program is executed.

**Generalization:** The ability of a program to successively "see" the problem in bigger picture. Problem reformulation in order to capture the global picture, of which the stated problem is a part.

**Curiosity:** The ability of a program, in AI context, to deduct fresh facts and rules from the given ones, without external requirements, either as a continuous process or occasionally.

**Creativity:** A creative act may be defined as one that is viewed as both valuable and novel and one that, in addition, reflects well on the cognitive abilities of the actor (whether human, animal or machine). It is not necessary that an AI system should have all of the above mentioned features. But it is interesting to note that all the above mentioned features can be exploited for the purpose of teaching of reading. The above mentioned components of AI can be applied to various tasks that AI can perform.

The task domains of Artificial Intelligence as described by Rich & Knightly are listed below.  

Mundane Tasks

Perception
- vision
- speech

Natural Language
- Understanding
- Generation
- Translation

Common-sense reasoning
Robot control

Formal Tasks

Games
- Chess
- Backgammon
- Checkers
- Cro

Mathematics
- Geometry
- Logic
- Integral Calculus
- Proving Properties of Programs

Expert Tasks

Engineering
- Design
- Fault finding
- Manufacturing planning

Scientific analysis
Medical diagnosis
Financial analysis

Teaching and learning task are obvious omissions in the list.
3.4.1 Knowledge Based Systems (KBS)

KBS is one of the most important areas of AI. The knowledge in Knowledge Based Systems of AI plays two roles:\(^2\)

1. It may define the search space and the criteria for determining a solution to a problem. We call this knowledge essential knowledge.

2. It may improve the efficiency of a reasoning procedure by informing that procedure of the best places to look for a solution. We call that knowledge heuristic knowledge.

A good Knowledge Based System should possess four principles. They have been summarised as follows:

1. Representational Adequacy is the ability to represent all of the kinds of knowledge that are needed in that domain.

2. Inferential Adequacy is the ability to manipulate the representational structures in such a way as to derive new structures corresponding to new knowledge inferred from old.

3. Inferential Efficiency is the ability to incorporate into the knowledge structure additional information that can be used to focus the attention of the inference mechanisms in the most promising directions and

---

4. Acquisitional Efficiency is the ability to acquire new information easily. The simplest case involves direct insertion by a person, of new knowledge into the database. Ideally, the program itself would be able to control knowledge acquisition\textsuperscript{22}.

The differences between conventional programs and Knowledge Based Systems is essentially a difference in their knowledge. This difference in knowledge has been tabulated for the purposes of this study Table 3.1.

**Table 3.1**

Differences between Conventional Programs and Knowledge Based Systems

<table>
<thead>
<tr>
<th>Conventional Programs</th>
<th>Knowledge Based Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. representation and use of data</td>
<td>representation and use of knowledge</td>
</tr>
<tr>
<td>2. algorithmic process</td>
<td>heuristic process</td>
</tr>
<tr>
<td>3. repetitive process</td>
<td>inferential process</td>
</tr>
<tr>
<td>4. effective manipulation of numbers (date)</td>
<td>effective manipulation of symbols</td>
</tr>
<tr>
<td>5. cannot handle incomplete information</td>
<td>can handle incomplete and uncertain information</td>
</tr>
<tr>
<td>6. cannot offer explanation on how solutions are arrived at</td>
<td>can offer explanations, help and suggestion</td>
</tr>
</tbody>
</table>

\textsuperscript{22} Ibid, p.109.
Despite the intelligence exhibited by the knowledge in KBS it has certain limitations to be remedied.

1. The domains of Knowledge Based System is usually very narrow. This impedes the process of generalisation.
2. When KBS are employed for a specific task no attempt is made to globalise the intelligence. The specific task may be just one member and the intelligence should try to solve a class of such problems.

It is necessary to collect human expertise from various fields viz., linguistics, pedagogy, cognitive psychology, computing and computer programming to make a good ICALL program. Culling knowledge from various experts and storing them in a common system is another area of AI called Expert Systems.

Research related to AI techniques in the area of languages to support symbolic reasoning led to the development of Expert Systems in the middle 1960s. These Expert Systems solved complicated problems which otherwise involved extensive human expertise. Expert Systems simulate human reasoning process by applying specific knowledge and inferences.

Knowledge Based Expert Systems (KBES) employed AI techniques to analyze problems and make decisions. Prof. Edward Feigenbaum of Stanford University has defined an Expert Systems 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. Knowledge necessary to perform at such a level, plus the
inference procedures used can be thought of as a model of the expertise of the best practitioners of the field.²³

Knowledge based Expert Systems relied on 'facts' and 'heuristics'. 'Facts' were universally accepted information by experts in a given field. Heuristics were inferences or judgements based on facts that were characteristic of expert level decision making in that field. The facts constituted the knowledge base and the heuristics constituted the Inference Engine. The efficiency of an Expert System depended on the exhaustiveness of its knowledge base. The architecture of a Knowledge Based Expert System is given in Fig.3.2

![Architecture of a Knowledge Based Expert System](image)

Fig. 3.2
Architecture of a Knowledge Based Expert System

Thus, a KBES has the knowledge base and inference engine as its primary unit. The secondary units comprise knowledge acquisition, explanation and a natural language user interface. The knowledge engineer contributes to the knowledge acquisition subsystem and the learner (or user) derives learning from the natural language user interface through explanations.

The requirements of an ideal Expert Systems are as follows:

1. Knowledge in the domain of interest should be extensive
2. Capability for applying search techniques
3. Ability for heuristic analysis
4. Capacity to infer new knowledge from existing knowledge
5. Ability for symbolic processing
6. Ability to explain the reasoning behind the inferences

Graham Davies and John Higgins opine that Expert Systems for CALL are difficult to design and stress the need for the hardware to be more powerful. Their explanation is that "language teachers have a clear picture of the way students learn a new language and can apply this knowledge' to designing the system. The judgement and branching which are inherent in most expert systems are based on the computer's ability to diagnose the precise nature, frequency and patterns of the students errors. But why students make certain types of mistakes is not always clear even to the most experienced language teacher. Some errors can be analysed, such as simple spelling mistakes, wrong agreements or mother tongue
interference, but others are the result of blind guesses and may indicate that the student has not mastered the correct forms\textsuperscript{24}.

This takes us back to the argument that as all human behaviour cannot be described or explained, designing ICALL programs can prove to be difficult. But all innovations involve difficulty and overcoming this difficulty will require a basic understanding of how language learning takes place. To make a beginning, even following an experienced teacher's intuitions may be sufficient to build a sound rule-based explanation facility in the ICALL program.

Expert Systems can thus act as tutors as in the case of ITS. When as Expert System takes on the role of a pupil, it might be a more effective method as it requires the learner to teach the machine and there is no better way of learning than by teaching.

User modeling for learning is based on knowledge engineering and knowledge acquisition. Knowledge engineering provides for an iterative analysis of the problem. The techniques of AI help in implementing the solution step by step in a prototyping manner. This serves for the adequate modelling of expertise and human inference capabilities called knowledge acquisition\textsuperscript{25}.


The processing of natural language by computers is another area of AI which has a bearing on future ICALL research. The scope of AI in Natural Language Processing (NLP) can be broadly classified in the following manner.

1. Information and documentation is one area of NLP involving the relationships between words and their concepts as represented by computers while building thesaurus, automatic indexing etc.

2. Machine translation is another area that studies the syntactic and semantic relationships between words and sentences. It gives a contrastive study of different languages.

3. NLP can also be used to build models of dialogues just like in natural languages taking into consideration the discourse and mental models of the speaker and the listener and then interpreting the dialogues.

Though it sounds ambitious, computers are being developed to handle natural languages - understand, speak and listen to the natural language of humans. Realizing this ambition will require the computer to know the speaker's intention, mental state etc. It is encouraging to note that machine translation has been fairly successful as it involves appreciable amount of language analysis technology even though it must be admitted that machine translation will never ever be able to match human intelligence in understanding and processing natural language this is because when AI is faced with the problem of handling natural language, it simultaneously involves an understanding of the mental mechanism associated with it. The computer's ability to imitate the behaviour of the human brain is very much limited as it is impossible to quantify mental tendencies.
3.4.2 Software and Hardware Background
3.4.2.1 AI Languages

AI languages differ from other programming languages in their composition and use. AI programmers like to think that AI programs are as dynamic as non-AI programs are static. AI languages are based on knowledge and inference, abstraction and incremental development and enjoy the status of 'experimentation' still. The concept of incremental development gives ample opportunity to modify and expand the system.

AI languages are of four types\textsuperscript{26}.

1. procedural
2. declarative
3. object-oriented and
4. functional

**Procedural Languages** e.g., LISP, LOGO and POP-II requires that the steps of computing be specified by the programmer.

**Declarative Languages** logic programming or rule based programming or declarative programming. In a declarative language, the programmer does not specify the steps of computing. The facts and rules for making inferences have to be stated explicitly. It is left to the computer to carry out the task of checking the facts and answering the queries of the user based on the rules. It is significant to note that knowledge is represented as assertions of truth. It is appropriately called logic programming because there is an equivalence between facts and inferences.

and between rules and theorems. PROLOG is the most important declarative language.

Object-oriented Languages involve objects or computational entities which receive and send messages to other objects to perform actions. It is used for graphic items.

Functional Languages simply carry out functions with no other tasks involved. For example if they are asked to add up two numbers A and B, they simply add and give the result without assigning its value to a variable. This is a beneficial method of proving the correctness of the program as it involves no side effects. PROLOG and LISP can be used as functional languages.

3.4.2.2 About Prolog

PROLOG stands for PROgramming in LOGic. It is a 'theorem-proving', logic-oriented' language. It was devised by Alain Colmerauer in the early 1970s at the University of Marseilles, France. Prolog is a powerful language for manipulating a database of propositions which means commercial data processing is a computer application area of significance. With increase in the efficiency of computers and decrease in cost, more and more applications of computers came to be realized. The inevitable shift from machine-oriented to human-oriented applications banked on logical reasoning for its success. And the programming language which was built on the foundation of logic programming is PROLOG. Prolog is becoming increasingly popular in this age of Artificial Intelligence.

Prolog is a declarative language ie. given the facts and rules it can make deductions. As opposed to procedural languages, there is no necessity
to give step by step description of the problem. A description of facts and rules is sufficient. Prolog has its own method to solve the problem. The declarative approach of Prolog is an advantage to overcome errors in procedures such as loops used in BASIC and PASCAL.

The problem description in Prolog is so well structured that it can be used as a specification tool’. PROLOG is based on first order logic and is very handy for manipulating symbols.

3.4.2.3 Advantages of Prolog

The following are the advantages of prolog.

1. It is based on mathematical logic
2. It has built in unification, back tracking and backward chaining inference mechanism
3. It supports declarative knowledge representation and incremental program development
4. It is available in a variety of hardware platforms and operating systems including MS-DOS, Windows, OS/2 and UNIX
5. It supports interface with C’ and C++’ and with other established relational database systems (such as Oracle, SQL, Paradox etc).
6. It supports extensive graphics facility
7. It is amenable to parallel processing
8. It supports fuzzy logic implementation
9. It has remarkable flexibility that allows quick prototyping
PROLOG scores over LISP on a few accounts:

1. The syntax of LISP is cumbersome. It involves too many branches. PROLOG is less unwieldy. Even a novice can learn to handle it easily.
2. PROLOG has a built-in inference mechanism based on matching. This is an invaluable advantage in areas of natural language processing, decision making in management etc.
3. It is easy to make additions at every stage of programming in Prolog. This is a highly valuable asset for teaching using ICALL programs written in prolog.
4. PROLOG is very elegant and perfectly suited to process natural language.

This does not mean that LISP can be written off as an AI programming language. As this writer’s study involved natural language for learning, TURBO PROLOG Version 2.0 was chosen as the programming language for the preparation of ICALL material because of the advantage of Prolog mentioned above and because Turbo Prolog was available to this researcher.

Prolog is most suitable for natural language processing. AI programming languages are symbolic processors but PROLOG is unique is not adapting a step by step algorithmic code for problem solving. PROLOG like natural language is a rule based system. Facts and rules describe the interrelationships between objects and concepts in PROLOG. Facts are truth assertions. PROLOG solves problems based on these facts and rules. Natural language does not demand algorithmic or arithmetic processing even if a certain amount of quantification may be necessary. "Prolog is a highly flexible and elegant language, one which is perfectly adapted...to
process natural language\textsuperscript{27}. In order to use a programming language to understand and process natural language it is necessary to know the differences between programming language and natural language. It is given in Table 3.2 below.

**Table 3.2**

<table>
<thead>
<tr>
<th>PROGRAMMING LANGUAGE</th>
<th>NATURAL LANGUAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. No ambiguity</td>
<td>Ambiguous</td>
</tr>
<tr>
<td>2. No redundancy</td>
<td>Redundant</td>
</tr>
<tr>
<td>3. Interpretation is</td>
<td>Interpretation dependent on various factors like situation, speaker etc.</td>
</tr>
<tr>
<td>consistent. It is the same for any two persons</td>
<td></td>
</tr>
<tr>
<td>4. Syntax and semantics clearly defined</td>
<td>Impossible to define semantics and syntax clearly</td>
</tr>
<tr>
<td>5. Same result and effects under any circumstances</td>
<td>Inconsistent result and effects, dependent on circumstances</td>
</tr>
</tbody>
</table>

The table indicating the contrast between the two languages reveals that using the technically correct programming language to describe the ambiguous, redundant and inconsistent natural language might be an advantage.

3.4.2.4 Problems of AI Programming

There are problems regarding the methodology of AI programming. The significant difference between AI programming methodology and general programming methodology is in the attitude of the researcher. It is typical of an AI researcher to write a program to understand the problem better. Emphasis is laid on designing, modifying and evaluating programs at the cost of accuracy.

An AI researcher does not necessarily sacrifice efficiency at the altar of research but sometimes constraints of computation and time have a telling effect on the mediocrity of AI programs. A program cannot be accepted as ‘intelligent’ if it takes more than the desired amount of time. The complexity of the AI algorithm is partly to blame for the inefficiency.

Robert Hawley, in his book *AI Programming Environments* points out another major difficulty: "The major difficulty in AI programming is that the combined demands of acceptable efficiency and sophisticated behaviour force AI programmers to write extremely complex programs which are difficult to understand, much less extend or debug. A very important technique for controlling the complexity is to build systems in layers of abstraction. A major difficulty with exploiting the ‘layers of abstraction’ programming technique is that each level is interpreted by the one below and the system becomes too slow."28

3.4.2.5 **Hardware Requirement**

The earliest AI programs ran on IBM computers. Since the 1960s DEC machines were used which later became VAXs. Now it is possible to install AI on microcomputers. But microcomputers are not suitable for writing and running large-scale programs.

For AI, today the IBM PC 486 or Pentium having 32 MB memory running on MS DOS and Windows (to access adequate memory) is adequate.

The choice of the software and hardware is important for the effective and successful application of an ICALL program. It is important to be sure about the availability, suitability and mutual compatibility of the hardware and software resources being used.

This overview of the AI scene giving a brief survey of AI systems, languages and its application is a stepping stone towards a study of the development and methodology of AI related to language learning. A study in their relatedness follows in the next chapter.