Introduction to Knowledge Based Agents

Swatimali@somaiya.edu

Swati Mali

Knowledge Based Agents

'If a machine is expected to be infallible, it cannot also be intelligent."

--Alan Turing

After studying this chapter students are supposed to

- Understand the significance and reasons for studying knowledge-based agents
- Identify limitations of problem-solving agents
- Discuss limitations of logic and knowledge-based agents
- Explain basic concepts related to knowledge-based agents such as inference, entailment, knowledge representation languages, generic knowledge-based agent , etc
- Understand basic inference rules and inference methods like forward chaining, backward chaining and resolution
- Understand the properties of inference algorithms

This chapter introduces the concept of knowledge based agents, significance of knowledge based agents, knowledge representation, inference rules and the reasoning algorithms.

The intelligent agents typically start with their work with learning the mundane tasks of perception and natural language skills. Further they need to develop their skills in common sense reasoning. This knowledge handling and reasoning comes very naturally to human beings. As the AI is defined as making machines do things which currently people are doing better, it has to be incorporated in intelligent systems as well. A human brain contains vast knowledge of various entities which helps in achieving choice of successful actions. Though implementation of the same amount of knowledge and specification of general purpose rules isn't an easy business. Natural language and interpretation of the statements is quite complicated. The natural languages are quite ambiguous and the same statement may have multiple meanings based on the context and tone of utterance. The AI division of knowledge representation and reasoning addresses this problem systematically and enable the intelligent agents to perform the reasoning process.

Aristotle (384–322 BC) is known to have been a pioneer to propose solutions on this problem. He introduced the concept of formal logic and reasoning. He suggested the analytical reasoning process and labelled his laws of thoughts as **syllogisms.** The approach was highly based on knowledge and inference processes to answer the queries scientifically. Though with some limitations to logic and laws of thoughts approach, Aristotle's work is very valuable in the domain of knowledge representation and reasoning field. The knowledge based agents are massively dependent on logic.

Reasons for studying knowledge based agents-

The problem solving agents aren't the complete solution to intelligent systems. They face limitations in generating the correct output due to limited and fixed knowledge, their inability to use knowledge about hidden aspects of the task environment, weakness in handling exploration or unknown environments, limitations in processing information expressed in natural language, lack of flexibility, etc. Let's address these issues one after the other.

Limited and fixed knowledge:

The knowledge about any domain is vast, constantly changing and is hard to characterize. on the contrary, the problem solving agents work with restricted and fixed knowledge. The table driven and simple reflex agents outputs are not complete and correct due this limited set of knowledge entities. The goal based and utility based architectures though, with the knowledge of how the task environment evolves, can give better results than their predecessors. However, it is impossible for the agent designer to add every minute detail in the model description and hence those solutions are not perfect. A meeting scheduler may schedule meetings in a row but may fail to take into consideration the travelling time between two meeting points. Even if it is programmed to do so, it requires extra knowledge to estimate commute time during different seasons, different hours of the day and on special days like *Anant Chaturdashi or Gokul Ashtami*.

The inability to use knowledge about hidden aspects of the task environment:

Knowledge and reasoning play a very critical role while working with partially observable and dynamic environments. The knowledge based agent puts together domain specific knowledge and current state information before selecting the appropriate action. 'A dead person remains dead for all times thereafter' could be the hidden knowledge used to answer if the person dead five years back is still dead. These domain specific pieces of knowledge are acquired through experience, books and other resources and are crucial in answering queries.

Weakness in handling exploration or unknown environments

The agent doesn't have entire knowledge about the partially observable environments. The knowledge is received at every step, some beliefs get strengthened, while some get contradicted. The general knowledge combined with the current state, helps the knowledge based agents in taking decision. Problem solving agents cannot handle such dynamicity of the knowledge, rather their knowledge is fixed.

Limitations in processing information expressed in natural language

Natural language has baggage of ambiguity, sarcasm and having context oriented meanings of the word. The NLP addresses intention of the speaker as the hidden state which plays critical role in understanding meaning of the statement.

The bank manager ate pasta salad with fork.

Here, the bank refers to the financial institute and not the river bank, the two-word set meaning for pasta salad isn't the same for baby food and the proposition 'with' changes treatment given to the food as the word after the same changes. For example with friends, with sautéed veggies and with fork have altogether different meanings. The knowledge based agents can process this information by considering syntax and semantics of the language.

Lack of flexibility

Knowledge based agents can easily accept new knowledge and new goals. Learning process is addition of new knowledge and adopting the same while giving the very next result. The learning phenomenon is rather complicated in terms of problem solving agents.

Limitations of Logic and knowledge based agents

The 19th century logicians proved that all kinds of things in the world can be presented using logic and notations. However, it lacks in two aspects.

Lack in informal language representation

The **formal logic** is characterized by syntax definition and truth values. The syntax definition has argument form and semantics. The formal logic basically has a premise and one or more conclusions based on if the premise is true or false.

The **informal logic** uses arguments in context of dialogues. It may have a premise but the conclusions might be hidden or absent.
A formal statement that has parts of speech entities are easier to represent. However, a statement that doesn't follow the same, is difficult to represent using formal knowledge representation method.

Example formal logic statement- The maid stole the jewellery or she milked the cow.
Conclusion- Either of the half part in given statement is true.

Example informal logic- A statement by a person who isn't in favour of leaving maid alone near safe at home-

"The maid's intentions seem to rob us off. She seems to know everything around the safe".

Hidden conclusion- To not to leave the maid alone near safe or to terminate her employment, which isn't directly conveyed in the statement.

The rhetorical statements are also counted as informal logic.

Uncertainty

The real life events do not have very fixed outcome. Every event has some or other probability of occurrence associated with it. This uncertainty of the event taking place or not, cannot be expressed using logic. It can only be dealt by some approach based on probability theory.

Example- Certain knowledge.
Fact- If it rains, the road gets wet.
Given- it's raining.
Conclusion- road is wet.
Example- Uncertain knowledge
Fact- Smoking or exposure to pollution causes cancer.
Given-

i. There is a Person1 who is a smoker living in unpolluted environment.
ii. There is a Person2 who is a passive smoker living in polluted environment.

For the above both the statements, one cannot conclude if Person1 and Person2 are suffering from cancer. The answer can be computed in terms of probability provided that one has enough data about both the cases with a better sample size. Even after providing good amount of sample data set, one cannot ignore the possibility of exception to the rule. i.e. there is no fixed answer for both the above statements.

The standard and traditional knowledge representation and reasoning fail in the cases troubled with uncertainty.

Basic Concepts Related to Knowledge Based Agents

Knowledge base

The knowledge base or a KB sits at the heart of a knowledge based agent. The KB can be defined as a set of sentences. These sentences though aren't written in natural language such as English, but in special kind of a language, known as knowledge representation language. Propositional logic and predicate(first order)logic are examples of knowledge representation languages. The statements in KB are assertions written in crisp fashion about the world in which the agent is supposed to work. Typically, a KB contains domain specific knowledge.

Inference

In simple words, the inference can be defined as the process of deriving new statements from the old ones.

The basic statements that aren't derived from any statement in knowledgebase are termed as axioms, while the ones that are derived from the old ones, are called inferred statements. The inferences are drawn based on strong inference rules of logical reasoning.

e.g.

i. Marcus is a man. All men are mortal. – Both are axioms.

ii. Marcus is mortal. – inferred statement from above two old statements in KB.

Inference procedures are domain-independent phenomenon.

Inference Algorithm

An inference algorithm is a procedure following which new sentences are derived from the KB. Mathematically, its written as,

$$KB \vdash_i S$$

It is read as S is derived/inferred from KB using inference procedure i.

Entailment

Entailment is a logical concept. A sentence is said to be entailed when it logically follows another sentence. Mathematically it is represented as,

$$\alpha \models \beta$$

The above statement is read as- α entails β. i.e.in every world where α is true, β is also true.

If a knowledgebase entails a statement, it might have been inferred by some inference procedure. Basically the entailment is a semantic relation.

e.g. i. Sham baked the pizza. $\models$ The pizza is baked

ii. Sharvari and Shamal ate ice cream. $\models$ Shamal ate ice cream.

The conceptual difference between entailment and inference is that inference procedures cannot essentially find everything that can be entailed and not all inference procedures are particularly correct or sound.

Implementation of Knowledge based agent

The knowledge based systems are designed and developed in either of the two ways-

i. Knowledge level approach – The questions discussed here are what the agent knows, what are its goals and it fixes agent behavior regardless of how it is implemented.
e.g. The knowledge level approach specifies sub-goals and can have mechanisms to tell if achieving a particular sub-goal is necessary in getting to final outcome.

ii. Implementation level – This approach tries to answer how the agents are implemented in detail, like the data structures and algorithms used during implementation.
E.g. The Travelling Salesperson Problem solution uses pixels or graph to represent cities and if the solution is computed using neural networks, machine learning or genetic algorithms.

Knowledge Representation Languages –

Typically, the artificial intelligence knowledge is represented using propositional and predicate logic methods.
Propositional logic is simple, though very restrictive in representing the knowledge. It typically uses a word or a symbol in combination with logical operators to represent the knowledge. It doesn't take into consideration parts of the statement as subject, verb, object etc. The truth value of the complex statements depends on the truth values of atomic sentences those make up that complex sentence.
Every sentence in propositional logic has truth value either true or false.

e.g.

Sentence	Propositional Logic Representation1	Propositional Logic Representation2
Rimi is hungry	P	RimiHungry
Rimi gets angry	Q	RimiAngry
Raja barks	R	RajaBark
if Rimi is hungry then she gets angry	$P \rightarrow Q$	RimiHungry → RimiAngry
if Rimi gets angry then raja barks	$Q \rightarrow R$	RimiAngry → RajaBark
Tom barks	S	TomBark

The limitation of propositional logic is, when symbols are used to represent the sentence, one cannot understand much of context from the mere symbol. The another and major disadvantage of using propositional logic is- one cannot generalise and relate statements with each other. In the above set, Raja and Tom both are barking, but the symbols R and S do not give any clue to equate them.

Predicate logic, also known as first order logic, is quite expressive compared to propositional logic. It breaks every natural language statement into subject, object and adjective to make predicates out of them. It represents relationship between entities mentioned in the statement and helps the logical reasoning process. Thus the created predicate represents properties of the subject with respect to rest of the elements in the statement.

e.g.

Sentence	Predicate Logic Representation
Rimi is hungry	Hungry(Rimi)
Rimi gets angry	Angry(Rimi)
Raja barks	Bark(Raja)
if Rimi is hungry then she gets angry	Hungry(Rimi) → Angry(Rimi)
if Rimi gets angry then raja barks	Angry(Rimi) → Bark(Raja)
Tom barks	Bark(Tom)

The predicate logic statements in the above example represent the relationship of being hungry, angry or barking action related to their subjects. The statement has good readability in the sense that one can create a good natural language sentence from the predicate logic representation. Also, predicate logic incredibly helps in generalization. In the given example, the relations/predicates will remain the same even if there are thousands of hungry or angry subjects, only the predicate object will change and predicate- number of elements in predicate will remain the same. Here, the predicate Bark() represents both, Tom and Raja.

Sentences represented in propositional logic or predicate logic both, are called well formed formula.

A Generic Knowledge-Based Agent.

The knowledge is vast and keeps on changing. This makes it quite natural for a designer to miss certain facts and rules during the knowledge based agent's implementation process and discover the errors later. Also in exploration environment, the new facts are realized during the execution and those must be added to the KB before the results are computed which must consider the newly added facts. The knowledge based agents achieve query results and updation of KB by using two actions used to communicate with the KB, namely- Ask and Tell.

Ask- this instruction mines answer to the query from KB.

KB is asked a question or a query in knowledge representation language, in a syntax that's understood by the KB. KB responds to the query with axiom or with a inferred sentence. The inference rules make sure that the response and the knowledge stored in KB do not contradict each other.

Tell- the agent Tells KB about a new piece of information, result of its action on environment or some change in task environment using the Tell action.

The KB is *Tell*ed(told) in knowledge representation language, in a syntax that's understood by the KB. The KB adds this new information to its existing knowledge and uses it next time for drawing inferences.

Algorithm Knowledge-based-Agent(percept) returns an action

static: *KB* - a knowledge base
t - a counter to represent time
t ← 0
Tell(KB, represent-percept-sentence(percept, t))
Action ← Ask(KB, represent-Action-Query())
Tell(KB, represent-Action-Sentence(action, t)
t ← t +1
Return action

The knowledge based agent perceives environment and TELLs about it to KB. The KB is then ASKed which action to take and finally TELLs KB that it is taking the given action. Actually the ASK operation may involve lot of computations by referring all the facts in knowledgebase to come to a suggestive action to perform. The functions represent-percept-sentence() and represent-Action-Sentence() take the burden of representing percept and action into appropriate knowledge representation language, following syntax and semantics of the same. Similarly, the represent-Action-Query() writes a query in knowledge representation language to ask the KB which action it should take.

A knowledgebase agent must be able to:

- Represent states, KB and actions,
- Incorporate new percepts
- Update internal representations of the world
- Deduce hidden properties of the world
- Deduce appropriate actions

inference rules

The process of deriving new statements follows certain rules. Though the concept of these inference rules remains the same, the representation of the definitions vary a bit from language to language. This section introduces the basic concepts of inference rules and they'll be elaborated more in the following chapter. Typically these inference rules are used to derive new statements or simplify the old ones so as to help in inferring new statements.

Modus Ponens

This is one of the most simple and commonly used inference rule. The rule states that, 'if a statement implies another one, and the statement is given, then the implied statement is taken true'.
i.e. if A → B, and A is given, then B is true.

e.g. Assume the given facts are as- If Rimi is angry then Raja barks. Rimi is angry.

Here, if 'Rimi is angry' part implies Raja barks. Then according to the modus ponens rule, as Rimi is angry is given, the fact of Raja barks is taken true.

Modus Tollens

This rule is also known as law of contrapositive. It states that if a statement implies another one, then negation of implied statement implies negation of the given statement.
i.e. if A → B, then ~B → ~A.

e.g. Assume the given facts are as- If Rimi is angry then Raja barks. Rimi is angry.
Here, if Rimi is angry implies Raja barks.
Then according to modus tollens,
Raja isn't barking ➔ Rimi is not angry.

AND-elimination

According to this rule, if two atomic sentences are connected together by AND connective, they can be separated by eliminating AND, and can be used further individually to infer new statements.

e.g. 'Today is Tuesday and it's raining today' represents two statements connected by AND. The result of AND -elimination on this example would be
Statement 1- Today is Tuesday
Statement 2- It's raining today

Bidirectional-elimination

Some compound statements are so complicated that they imply each other in recursive manner. Such statements cannot be used by inference methods unless they are simplified. The Bidirectional-elimination rule gives us two separate sentences connected by AND, and each part has some sort of one directional implication. One could further perform AND-elimination rule to separate those two implications.

e.g. Dead is not alive and not alive is dead. Logically, this sentence could also be represented as, Dead ←→ ~Alive.

The effect of Bidirectional-elimination on this example would be,
(Dead → ~Alive) AND (~Alive → Dead)

Applying AND-elimination rule to above sentences yields two separate sentences,

i. Dead → ~Alive
ii. ~Alive → Dead

Forward Chaining

Forward chaining, backward chaining and resolution are popular inference methods. Forward chaining starts with the given facts and answers the query by repeatedly applying Modus-Ponens. To be able to apply modus-ponens, sometimes one may have to simplify the statements using bidirectional-elimination, AND-elimination and modus tollens beforehand on the sentences in the KB. One can use all of these one after the other, or the only appropriate rules can be used. Forward chaining is data oriented approach and takes more time and processing to reach to the conclusion.

e.g. Consider a small set of statements to answer if Raja is barking.

i. When Rimi is hungry, she gets angry.
ii. When Rimi is angry, Raja barks.
iii. Rimi is hungry.

One can use forward chaining on the given statements to infer if Raja is barking as,

→ From statements i and iii,
iv. Rimi is angry

→ from statements ii and iv,
Raja barks is true.

In forward chaining, KB $\cup$ new inferred statement is not a contradiction.

Backward Chaining

Backward chaining is called goal oriented inference method as it works backward from the goal to KB. As it focuses on exact goals, it doesn't wander in irrelevant facts with futile results, but works only on precise statements to give solution in minimum time and processing. Human mind works in a backward chaining fashion.

In backward chaining If the goal query - say Q - is known to be true, then no work is needed.

Otherwise, the algorithm finds those implications in the knowledge base that conclude Q. If all the premises of one of those implications can be proved true (by backward chaining), then Q is true.

This inference method is typically used for theorem proving. It begins work with a list of goals and works backwards repeatedly to see if the newly inferred statements take it to the facts that are in unison with the given KB.

e.g. Consider a small set of statements to answer if Raja is barking.

i. When Rimi is hungry, she gets angry.
ii. When Rimi is angry, Raja barks.
iii. Rimi is hungry.

One can use backwards chaining on the given statements to infer if Raja is barking as,

→ Here for Raja is barking to be true, then as per (ii), Rimi is angry has to be true.

→ Applying backward reasoning again, for Rimi is Angry to be true, then as per (i), she is hungry has to be true.

→Fortunately, it is true as per the (iii) given statement.

i.e. starting from the goal, by repeatedly following backward direction, if the inference method proves that all the implications concluding goal are true, the goal is proved to be true using backward chaining. If the backward chaining process infers a fact which is contradicts with any of the statements in KB, the goal is assumed to be failed.

Resolution

This method of inference is also called as proof by negation or proof by refutation. Resolution process negates the goal and adds it to KB. Then all the statements including negated goal are converted into Conjunctive Normal Form (CNF). Further, until all the resolvable pairs of clauses are exhausted, find resolvable clauses and resolve them by repeatedly adding resolution results to the knowledge base. If the process ends with a NIL i.e. empty clause/ contradiction, then negation of goal is false. i.e. Goal is true.

The example on resolution will be discussed in next chapter as it needs the introduction to knowledge representation and CNF.

Properties Of Inference Algorithms

Inference Algorithms hold two important properties- Soundness and Completeness.

The soundness is a truth-preserving characteristic. An inference algorithm is said to be sound iff it derives only the entailed statements.

An inference algorithm is said to be complete iff it derives all the possible entailed statements.

e.g. Consider a statement – Mary went to city mall yesterday evening and got a very beautiful red purse on discount that went perfect with her new dress.

This statement can derive various new statements from the given statement. Moreover, if it has more information on purchases, accessories, styles, human behaviour in the knowledgebase, the number of new statements it can derive increases more and more. Some of the new derivable statements are-

- Mary purchased a purse.
- Last evening Mary wasn't home.
- There's a mall named city mall.
- City mall was open last evening.
- City mall has at least one shop that sells purses.
- There is discount sale on purses.
- Mary has at least one new dress.
- Mary didn't have a red purse.
- There was at least one woman at mall last evening.
- There was at least one woman named Mary at the mall last evening.
- Mary doesn't mind carrying a purse.

There is scope to derive some more statements too. If an inference procedure derives only correct new statements, it will hold valid soundness property and if an inference procedure derives all the possible new statements, it will hold valid completeness property. It's possible for an inference procedure to be only complete or only sound or have both the properties.

Forward chaining and backward chaining are sound and complete with horn clauses. Resolution is sound and complete for propositional logic and is sound for predicate logic.

Summary

- The problem-solving agents are not enough to solve all sorts of problems in AI, especially the problems wherein role of knowledge is significant.
- Such problems need to be handled in a special way by using knowledge-based agents.
- Problem-solving agents use lmited and fixed knowledge, they cannot extract and use knowledge about hidden aspects of the environment, they aren't very strong with partially observable and exploration environment s, also they face various limitations in processing natural language.
- The knowledge based agents use strong mathematical background of logic.
- The knowledge based agents have limitations in representing informal knowledge and they cannot express uncertainty in the statements.
- Knowledge base, inference algorithms, entailment, knowledge representation languages, inference rules, forward chaining, backward chaining and resolution are very important aspects of knowledge based agents.
- The inference algorithms have two properties- completeness and soundness
- The knowledge-based agents work on knowledge base (KB)
- The KB is nothing but a set of sentences represented in special language, called knowledge representation language.
- KB= domain specific knowledge + problem specific information + observations during the course of solution.
- The knowledge-based agents perform two operations on the KB - **Tell** and **Ask.**
- The **Tell** operation inserts a new sentence into the KB.
- The **Ask** operation answers the query.
- The **ask** operation involves a lot of computation to deduce the appropriate response.
- An agent's designer builds initial knowledge into the KB.
- The knowledge-based agents can learn new knowledge using Tell function.
- The generic architecture of knowledge-based agents works with two basic functions- ASK and tell.
- Knowledge base agents do represent states, knowledge base, actions, they incorporate new updates, internal representations of the world, deduce hidden properties and also deduce appropriate actions to be taken in the task environment.
- Modus ponens, modus tollens, AND-elimination, bidirectional elimination are some of the important inference rules used by knowledge based agents.
- The knowledge based agents are implemented at two different levels- knowledge level and implementation level. Typically hybrid approach works the best in design and implementation of knowledge based agents.

Test Your Skills

Multiple Choice Questions

1. Which of the following is not a Goal-based agent?
   (a) Inference
   (b) Searching
   (c) Planning
   (d) Conclusion
   e) All of the above

2. Knowledge and reasoning are more important while working with ___________________ environment.
   a) Fully Observable
   b) Partially Observable
   c) Neither Fully nor Partially Observable
   d) Fully and Partially Observable, both

3. Verdict announced by a judge on a particular case is based on -

   a) Only current case, witness statements and evidences
   b) Current case details, witness statements , evidences plus some knowledge from the law books
   c) Current case details, witness statements , evidences, some knowledge from the law books and knowledge earned through experience
   d) All of the mentioned

4. State whether True or False. A knowledge-based agent combines domain specific knowledge with current precepts to infer hidden aspects of the current state prior to selecting actions.
   a) True
   b) False

5. Choose the correct option.

   A) Knowledge base (KB) contains of set of statements but not expressed in natural language.

   B) Inference is defined as deriving a new sentences from the old ones contained in KB.
      a) A is true, B is true
      b) A is false, B is false
      c) A is true, B is false
      d) A is false, B is true

6. if $\alpha \vDash \beta$, then in every world wherein α is _____ , β is _____.
   a) True, true
   b) True, false
   c) False, true
   d) False, false

7. Knowledge based agents can express uncertainty in knowledge

   a) True
   b) False

8. Resolution is based on the concept of

a. Proof by negation
b. Proof by refutation
c. Proof by KB ∪ ~Goal is contradiction
d. All of the above
e. None of the above

9. Assume three inference algorithms Algo1, Algo2 and Algo3. Suppose the maximum possible correct inferences from a given statement are 10.

Algo1 gives 15 inferences out of which 10 are correct and 5 are incorrect.
Algo2 gives exactly 10 and all of them are correct.
Algo3 gives 17 out of which 7 are correct and rest are incorrect.
Which one of the following statements about Algo1, Algo2 and Algo3 are correct?

a. Algo1 - not complete, sound; Algo2 is not complete and is sound, Algo3 is complete and sound.
b. Algo1 - complete, not sound; Algo2 is complete and sound, Algo3 is neither complete nor sound.
c. Algo1 - not complete and not sound; Algo2 not complete and is sound, Algo3 is complete and sound.
d. All of the above
e. None of the above

10. state true or false. If a knowledgebase entails a statement, it might have been inferred by some inference procedure.

a) True
b) False

Theoretical questions

1. What is the significance of using knowledge-based agents over problem-solving agents?
2. Discuss the limitations of logic and knowledge based agents.
3. Differentiate between formal and informal logic with suitable examples.
4. What is uncertainty in the knowledge? Elaborate the same with a suitable example.
5. Explain the relationship between inference and entailment.
6. justify why the predicate logic is better than propositional logic for knowledge representation and reasoning process.
7. Describe the working of a generic knowledge-based agent.
8. State and explain inference rules modus ponens, modus tollens, AND-elimination and bidirectional elimination with suitable example.
9. Explain the concept of Forward chaining and backward chaining with suitable example.

10. " Third year B division of Computer Engineering branch has operating systems lab in room no. B-310 at 3:00 p.m.". Derive the maximum sentences that can be inferred from the given sentence.