ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 1, January 2018

Artificial Intelligence Using Forward Chaining and Backward Chaining

Kajal Jaisinghani[1]

Assistant Professor, Department of Computer Science, Birla College of Arts Science and Commerce, Maharashtra, India.[1]

ABSTRACT: Artificial Intelligence(AI) research is advancing the frontier of computing by endowing machines with the abilities to solve problems that require high-level sometimes human expertise, perform complex tasks autonomously, learn from experience, interact and collaborate seamlessly with people, and cope effectively with uncertainty and missing information. The field was founded on the claim that human intelligence "can be so precisely described that a machine can be made to simulate it". This raises philosophical arguments about the nature of the mind and the ethics of creating artificial beings endowed with human-like intelligence, issues which have been explored by myth, fiction and philosophy since antiquity.

KEYWORDS: Artificial intelligence, Expert system, inference rule, forward and backward chaining, ontology, semantic field, prolog.

I. INTRODUCTION

Knowledge representation and knowledge engineering are central to AI research. Many of the problems machines are expected to solve will require extensive knowledge about the world. Among the things that AI needs to represent are: objects, properties, categories and relations between objects situations, events, states and time causes and effects knowledge about knowledge (what we know about what other people know)and many other, less well researched domains. A representation of "what exists" is ontology: the set of objects, relations, concepts, and properties formally described so that software agents can interpret them. The semantics of these are captured as description logic concepts, roles, and individuals, and typically implemented as classes, properties, and individuals in the Web Ontology Language. The most general ontologies are called upper ontologies, which attempt to provide a foundation for all other knowledge by acting as mediators between domain ontologies that cover specific knowledge about a particular knowledge domain (field of interest or area of concern).
The fields of artificial intelligence, the Semantic Web, systems engineering, software engineering, biomedical informatics, library science, enterprise bookmarking, and information architecture all create ontologies to limit complexity and organize information. The ontology can then be applied to problem solving.

II. DOMAIN ONTOLOGY

A domain ontology (or domain-specific ontology) represents concepts which belong to part of the world. Particular meanings of terms applied to that domain are provided by domain ontology. For example, the word card has many different meanings. Ontology about the domain of poker would model the "playing card" meaning of the word, while ontology about the domain of computer hardware would model the "punched card" and "video card" meanings.
Since domain ontologies represent concepts in very specific and often electric ways, they are often incompatible. As systems that rely on domain ontologies expand, they often need to merge domain ontologies into a more general representation. This presents a challenge to the ontology designer. Different ontologies in the same domain arise due to different languages, different intended usage of the ontologies, and different perceptions of the domain (based on cultural background, education, ideology, etc.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 1, January 2018

III. KNOWLEDGE INFERENCE - FORWARD AND BACKWARD CHAINING

An Inference Engine is a tool from artificial intelligence. The first inference engines were components of expert systems. The typical expert system consisted of a knowledge base and an inference engine. The knowledge base stored facts about the world. The inference engine applied logical rules to the knowledge base and deduced new knowledge. This process would iterate as each new fact in the knowledge base could trigger additional rules in the inference engine. Inference engines work primarily in one of two modes either special rule or facts: forward chaining and backward chaining. Forward chaining starts with the known facts and asserts new facts. Backward chaining starts with goals, and works backward to determine what facts must be asserted so that the goals can be achieved.

IV. FORWARD CHAINING

Forward Chaining is one of the two main methods of reasoning when using an inference engine and can be described logically as repeated application of modus ponens. Forward chaining is a popular implementation strategy for expert systems, business and production rule systems. Forward chaining starts with the available data and uses inference rules to extract more data (from an end user, for example) until a goal is reached. An inference engine using forward chaining searches the inference rules until it finds one where the antecedent (If clause) is known to be true. When such a rule is found, the engine can conclude, or infer, the consequent (Then clause), resulting in the addition of new information to its data

A Horn clause C is called definite it contains exactly one positive literal, i.e., implications of type are not possible.

If the knowledge base consists of Horn clauses only, then generalized modus ponens can be used just like modus ponens to infer statements iteratively by forward chaining.

Inference engines will iterate through this process until a goal is reached.

For example, suppose that the goal is to conclude the color of a pet named Fritz, given that he croaks and eats flies, and that the rule base contains the following four rules:

- If X croaks and X eats flies - Then X is a frog
- If X chirps and X sings - Then X is a canary
- If X is a frog - Then X is green
- If X is a canary - Then X is yellow

Let us illustrate forward chaining by following the pattern of a computer as it evaluates the rules.

Assume the following facts:

Fritz croaks

Fritz eats flies

With forward reasoning, the inference engine can derive that Fritz is green in a series of steps:

1. Since the base facts indicate that "Fritz croaks" and "Fritz eats flies", the antecedent of rule #1 is satisfied by substituting Fritz for X, and the inference engine concludes:

Fritz is a frog

2. The antecedent of rule #3 is then satisfied by substituting Fritz for X, and the inference engine concludes Fritz is green

The name "forward chaining" comes from the fact that the inference engine starts with the data and reasons its way to the answer, as opposed to backward chaining, which works the other way around. In the derivation, the rules are used in the opposite order as compared to backward chaining. In this example, rules #2 and #4 were not used in determining that Fritz is green.

Because the data determines which rules are selected and used, this method is called data-driven, in contrast to goal-driven backward chaining inference. The forward chaining approach is often employed by expert systems, such as CLIPS.

One of the advantages of forward-chaining over backward-chaining is that the reception of new data can trigger new inferences, which makes the engine better suited to dynamic situations in which conditions are likely to change.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 1, January 2018

V. BACKWARD CHAINING

It is also called as backward reasoning .It is an inference method that can be described colloquially as working backward from the goal(s). It is used in automated theorem proves, inference engines, proof assistants and other artificial intelligence applications.
In game theory, its application to (simpler) sub games in order to find a solution to the game is called backward induction. In chess, it is called retrograde analysis, and it is used to generate table bases for chess endgames for computer chess.
Backward chaining is implemented in logic programming by SLD resolution. Both rules are based on the modus ponens inference rule. It is one of the two most commonly used methods of reasoning with inference rules and logical implications – the other is forward chaining. Backward chaining systems usually employ a depth-first search strategy, e.g. Prolog.

VI. CONCLUSION

The field of artificial intelligence gives the ability to the machines to think analytically, using concepts. Tremendous contribution to the various areas has been made by the Artificial Intelligence techniques from the last 2 decades. Artificial Intelligence will continue to play an increasingly important role in the various fields. This paper is based on the concept of Artificial intelligence, areas of artificial intelligence and the artificial intelligence techniques.

REFERENCES

[1] N Ramesh, C Kambhampati, JRT Monson, PJ Drew, "Artificial intelligence in medicine", 2004.
[2] Charles Weddle, Graduate Student, Florida State University "Artificial Intelligence and Computer Games", unpublished.
[3] C. Sampada,, et al, "Adaptive Neuro-Fuzzy Intrusion Detection Systems", Proceedings: International Conference on Information Technology: Coding and Computing.
[4] M. Dyer, A. Frieze and R. Kannan, A random polynomial time algorithm for approximating the volume of convex bodies, in: Proceedings Annual ACM Symposium on the Theory of Computing (1989) 375- 381.
[5] Y. Freund, Boosting a weak learning algorithm by majority, in: Proceedings 3rd Annunl Workshop on Compututional Learning Theory, San Francisco, CA (Morgan Kaufmann, San Mateo, CA, 1990) 202-216.
[6] H. Drucker, C. Cortes, L.D. Jackel, Y. LcCun and V. Vapnik, Boosting and other machine learning algorithms, in: Proceedings 11th International Conference on Machine Learning, New Brunswick, NJ (Morgan Kaufmann, San Mateo, CA,1994) 53-61.
[7] P. Clark and T. Niblett, The CN2 induction algorithm, Machine Learning 3 (1989) 261-284. 10. D.A. Cohn, Z. Ghahramani and Jordan, Active learning with statistical models, J. Artif Intell. Research 4 (1996) 129-145. 12.
[8] R. E. Bellman and L. A. Zadeh, 'Local and fuzzy logics', in Modern Uses of Multiple-Valued Logic, eds., J. M. Dunn and Epstein, pp. 103–165. D. Reidel, Dordrecht, (1977).
[9] N. D. Belnap, 'A useful four-valued logic', in Modern Uses of MultipleValued Logic, eds., J. M. Dunn and G. Epstein 7–37. D. Reidel, Dordrecht, (1977).
[10] J. Ben-Naim and H. Prade, 'Evaluating trustworthiness from past performances; interval-based approaches' , Ann. Math. Artif. Intell., 64(2- 3), 247–268, (2012) .
[11] J. Y. Halpern, Reasoning About Uncertainty, MIT Press, Cambridge, MA, 2003.
[12] M. E. Bratman, Faces of Intention, Cambridge University Press, 1999.
[13] C. Domshlak, E. Hullermeier, S. Kaci, and H. Prade, 'Preferences in ¨ AI: An overview', Artifificial Intelligence, 175, 1037–1052, [14]
[14]P. Dellunde and L. Godo, 'Introducing grades in deontic logics', in Proc. of 9th Intl. Conf. on Deontic Logic in Computer Science, , DEON 2008 July 15-18, 2008., eds., R. van der Meyden and L. van der Torre, volume 5076 of Lecture Notes in Computer Science, pp. 248–262, (2008).
[15] A. Casali, L. Godo, and C. Sierra, 'A graded BDI agent model to represent and reason about preferences', Artificial Intelligence, 175, 1468–1478, (2011)