

Reasoning in artificial intelligence (ai): a review



**ASSIGN
BUSTER**

1: Introduction

Artificial Intelligence (AI) is one of the developing areas in computer science that aims to design and develop intelligent machines that can demonstrate higher level of resilience to complex decision-making environments (López, 2005 ^[1]). The computations that at any time make it possible to assist users to perceive, reason, and act forms the basis for effective Artificial Intelligence (National Research Council Staff, 1997 ^[2]) in any given computational device (e. g. computers, robotics etc.,). This makes it clear that the AI in a given environment can be accomplished only through the simulation of the real-world scenarios into logical cases with associated reasoning in order to enable the computational device to deliver the appropriate decision for the given state of the environment (López, 2005). This makes it clear that reasoning is one of the key elements that contribute to the collection of computations for AI. It is also interesting to note that the effectiveness of the reasoning in the world of AI has a significant level of bearing on the ability of the machine to interpret and react to the environmental status or the problem it is facing (Ruiz et al, 2005 ^[3]). In this report a critical review on the application of reasoning as a component for effective AI is presented to the reader. The report first presents a critical overview on the concept of reasoning and its application in the Artificial Intelligence programming for the design and development of intelligent computational devices. This is followed by critical review of selected research material on the chosen topic before presenting an overview on the topic including progress made to date, key problems faced and future direction.

2: Reasoning in Artificial Intelligence

2. 1: About Reasoning

Reasoning is deemed as the key logical element that provides the ability for human interaction in a given social environment as argued by Sincák et al (2004) ^[4]. The key aspect associated with reasoning is the fact that the perception of a given individual is based on the reasons derived from the facts that relative to the environment as interpreted by the individual involved. This makes it clear that in a computational environment involving electronic devices or machines, the ability of the machine to deliver a given reason depends on the extent to which the social environment is quantified as logical conclusions with the help of a reason or combination of reasons as argued by Sincák et al (2004).

The major aspect associated with reasoning is that in case of human reasoning the reasoning is accompanied with introspection which allows the individual to interpret the reason through self-observation and reporting of consciousness. This naturally provides the ability to develop the resilience to exceptional situations in the social environment thus providing a non-feeble minded human to react in one way or other to a given situation that is unique in its nature in the given environment. It is also critical to appreciate the fact that the reasoning in the mathematical perspective mainly corresponds to the extent to which a given environmental status can be interpreted using probability in order to help predict the reaction or consequence in any given situation through a sequence of actions as argued by Sincák et al (2004).

The aforementioned corresponds with the case of uncertainty in the environment that challenges the normal reasoning approach to derive a specific conclusion or decision by the individual involved. The introspective nature developed in humans and some animals provides the ability to cope with the uncertainty in the environment. This adaptive nature of the non-feeble minded human is the key ingredient that provides the ability to interpret the reasons to a given situation as opposed to merely following the logical path that results through the reasoning process. The reasoning in case of AI which aims to develop the aforementioned in the electronic devices to perform complex tasks with minimal human intervention is presented in the next section.

2. 2: Reasoning in Artificial Intelligence

Reasoning is deemed to be one of the key components to enable effective artificial programs in order to tackle complex decision-making problems using machines as argued by Sincák et al (2004). This is naturally because of the fact that the logical path followed by a program to derive a specific decision is mainly dependant on the ability of the program to handle exceptions in the process of delivering the decision. This naturally makes it clear that the effective use of the logical reasoning to define the past, present and future states of the given problem alongside the plausible exception handlers is the basis for successfully delivering the decision for a given problem in chosen environment. The key areas of challenge in the case of reasoning are discussed below (National Research Council Staff, 1997).

Adaptive Software - This is the area of computer programming under Artificial Intelligence that faces the major challenge of enabling the effective decision-making by machines. The key aspect associated with the adaptive software development is the need for effective identification of the various exceptions and the ability to enable dynamic exception handling based on a set of generic rules as argued by Yuen et al (2002) ^[5]. The concept of fuzzy matching and de-duplication that are popular in case of software tools used for cleansing data cleansing in the business environment follow the above-mentioned concept of adaptive software. This is the case there the ability of the software to decide the best possible outcome for a given situation is programmed using a basic set of directory rules that are further enhanced using references to a variety of combinations that comprise the database of logical combinations for reasons that can be applied to a given situation (Yuen et al, 2002). The concept of fuzzy matching is also deemed to be a major breakthrough in the implementation of adaptive programming of machines and computing devices in Artificial Intelligence. This is naturally because of the fact that the ability of the program to not only refer to a set of rules and associated reference but also to interpret the combination of reasons derived relative to the given situation prior to arriving on a specific decision. From the aforementioned it is evident that the effective development of adaptive software for an AI device in order to perform effective decision-making in the given environment mainly depends on the extent to which the software is able to interpret the reasons prior to deriving the decision (Yuen et al, 2002). This makes it clear that the adaptive software programming in artificial intelligence is not only deemed as an area of challenge but also the one with extensive scope for development to <https://assignbuster.com/reasoning-in-artificial-intelligence-ai-a-review/>

enable the simulation of complex real-world problems using Artificial Intelligence.

It is also critical to appreciate the fact that the adaptive software programming in the case of Artificial Intelligence is mainly focused on the ability to not only identify and interpret the reasons using a set of rules and combination of outcomes but also to demonstrate a degree of introspection. In other words the adaptive software in case of Artificial Intelligence is expected to enable the device to become a learning machine as opposed to an efficient exception handler as argued by Yuen et al (2002). This further opens room for exploring into knowledge management as part of the AI device to accomplish a certain degree of introspection similar to that of a non-feeble minded human.

Speech Synthesis/Recognition - This area of Artificial Intelligence can be deemed to be a derivative of the adaptive software whereby the speech/audio stream captured by the device deciphers the message for performs the appropriate task (Yuen et al, 2002). The speech recognition in the AI field of science poses key issues of matching, reasoning to enable access control/ decision-making and exception handling on top of the traditional issues of noise filtering and isolation of the speaker's voice for interpretation. The case of speech recognition is where the aforementioned issues are faced whilst in case of speech synthesis using computers, the major issue is the decision-making as the decision through the logical reasoning alone can help produce the appropriate response to be synthesised into speech by the machine.

The speech synthesis as opposed to speech recognition depends only on the adaptive nature of the software involved as argued by Yuen et al (2002). This is due to the fact that the reasons derived from the interpretation of the input captured using the decision-making rules and combinations for fuzzy matching form the basis for the actual synthesis of the sentences that comprises the speech. The grammar associated with the sentences so framed and its reproduction depends heavily on the initial decision of the adaptive software using the logical reasons identified for the given environmental situation. Hence the complexity of speech synthesis and recognition poses a great challenge for effective reasoning in Artificial Intelligence.

Neural Networks – This is deemed to be yet another key challenge faced by Artificial Intelligence programming using reasoning. This is because of the fact that neural networks aim to implement the local behaviour observed by the human brain as argued by Jones (2008) ^[6]. The layers of perception and the level of complexity associated through the interaction between different layers of perception alongside decision-making through logical reasoning (Jones, 2008). This makes it clear that the computation of the decision using the neural networks strategy is aimed to solving highly complex problems with a greater level of external influence due to uncertainties that interact with each other or demonstrate a significant level of dependency to one another. This makes it clear that the adaptive software approach to the development of the reasoned decision-making in machines forms the basis for neural networks with a significant level complexity and dependencies involved as argued by referrece8.

The Single Layer Perceptions (SLP) discussed by Jones (2008) and the representation of Boolean expressions using SLPs further makes it clear that the effective deployment of the neural networks can help simulate complex problems and also provide the ability to develop resilience within the machine. The learning capability and the extent to which the knowledge management can be incorporated as a component in the AI machine can be defined successfully through identification and simulation of the SLPs and their interaction with each other in a given problem environment (Jones, 2008).

The case of neural networks also opens the possibility of handling multi-layer perceptions as part of adaptive software programming through independently programming each layer before enabling interaction between the layers as part of the reasoning for the decision-making (Jones, 2008). The key influential element for the aforementioned is the ability of the programmer(s) to identify the key input and output components for generating the reasons to facilitate the decision-making.

The backpropagation or backward error propagation algorithm deployed in the neural networks is a salient feature that helps achieve the major aspect of learning from mistakes and errors in a given computer program as argued by Jones (2008). The backpropagation algorithm in the multi-layer networks is one of the major areas where the adaptive capabilities of the AI application program can be strengthened to reflect the real-world problem solving skills of the non-feeble minded human as argued by Jones (2008).

From the aforementioned it is clear that the neural networks implementation of AI applications can be achieved to a sustainable level using the backpropagation error correction technique. This self-correcting and learning system using the neural networks approach is one of the major elements that can help implement complex problems' simulation using AI applications. The case of reasoning discussed earlier in the light of the neural networks proves that the effective use of the layer-based approach to simulate the problems in order to allow for the interaction will help achieve reliable AI application development methodologies.

The discussion presented also reveals that reasoning is one of the major elements that can help simulate real-world problems using computers or robotics regardless of the complexity of the problems.

2. 3: Issues in the philosophy of Artificial Intelligence

The first and foremost issue faces in the case AI implementation of simulating complex problems of the real-world is the need for replication of the real-world environment in the computer/artificial world for the device to compute the reasons and derive upon a decision. This is naturally due to the fact that the simulation process involved in the replication of the environment for the real-world problem cannot always account for exceptions that arise due to unique human behaviour in the interaction process (Jones, 2008). The lack of this facility and the fact that the environment so created cannot alter itself fundamentally apart from being altered due to the change in the state of the entities interacting within the simulated environment makes it a major hurdle for effective AI application development.

Apart from the real-world environment replication, the issue faced by the AI programmers is the fact that the reasoning processes and the exhaustiveness of the reasoning is limited to the knowledge/skills of the analysts involved. This makes it clear that the process of reasoning depending upon non-feeble minded human's response to a given problem in the real-world varies from one individual to another. Hence the reasons that can be simulated into the AI application can only be the fundamental logical reasons and the complex derivation of the reasons' combination which is dependant on the individual cannot be replicated effectively in a computer as argued by López (2005).

Finally, the case of reasoning in the world of Artificial Intelligence is expected to provide a mathematical combination to the delivery of the desired results which cannot be accomplished in many cases due to the uniqueness of the decision made by the non-feeble minded individual involved. This poses a great challenge to the successful implementation of AI in computers and robotics especially for complex problems that has various possibilities to choose from as result.

3: Critical Summary of Research

3. 1: Paper 1 – Programs with Common Sense by Dr McCarthy

The rather ambitious paper presented by Dr McCarthy aims to provide an AI application that can help overcome the issues in speech recognition and logical reasoning that pose significant hurdles to the logical reasoning in AI application development. However, the approach to the delivery of the aforementioned in the form of an *advice taker* is a rather feeble approach to the AI representation of the solution to a problem of greater magnitude.

<https://assignbuster.com/reasoning-in-artificial-intelligence-ai-a-review/>

Even though the paper aims to provide an Artificial Intelligence application for verbal reasoning processes that are simple in nature, the fact that the interpretation of the verbal reasoning in the light of the given problem relative to an environment is not a simple component to be simulated with ease prior to achieving the desired outcome as discussed in section 2.

“ One will be able to assume that the advice taker will have available to it a fairly wide class of immediate logical consequences of anything it is told and its previous knowledge”. (Dr McCarthy, Pg 2). This statement by the author in the research paper provides room for the discussion that the *advice taker* program proposed by Dr McCarthy is aimed to deliver an AI application using knowledge management as a core component for logical reasoning. This is so because of the nature of the statement which implies that the *advice taker* program will be able to deliver its decision through access to a wide range of *immediate logical consequences of anything it is told and its previous knowledge*. This makes it clear that the *advice taker* software program is not a non-viable approach as the knowledge management strategy for logical reasoning is a component under debate as well as development over a wide range of scientific applications related problems simulation using AI. The Two Stage Fuzzy Clustering based on knowledge discovery presented by Qain in Da (2006) ^[7] is a classical example for the aforementioned. It is also interesting to note that the knowledge management aspect of artificial intelligence programming is mainly dependant on the speed related to the access and processing of the information in order to deliver the appropriate decision relative to the given problem (Yuen et al, 2002). A classical example for the aforementioned

would be the use of fuzzy matching for validation or suggestion list generation on Online Transaction Processing Application (OLTP) on a real-time basis. This is the scenario where a portion of the data provided by the user is interpreted using fuzzy matching to arrive upon a set of concrete choices for the user to choose from (Jones, 2008). The process of choosing the appropriate option from the given suggestion list by the individual user is the component that is being replaced using Artificial Intelligence in machines to choose the best fit for the given problem. The aforementioned is evident in case of the *advice taker* software program that aims to provide a solution for responding to verbal reasoning processes of the day-to-day life of a non-feeble minded individual.

The author's objective 'to make programs that learn from their experience as effectively as humans do', makes it clear that the knowledge management approach with the ability of the program to utilise a database type storage option to store/access its knowledge and previous experiences as part of the process. This makes it clear that the *advice taker* software maybe a viable option if the processing speed related to the retrieval and storage of information from a database of such magnitude which will grow in size at an exponential rate is made available for the AI application. The aforementioned approach can be achieved by the use grid computing technology as well as other processing capabilities with the availability of electronic components at affordable prices on the market. The major issue however is the design for such an application and the logical reasoning processes of retrieving such information to arrive at a decision for a given problem. Form the discussion presented in section 2 it is evident that the

complexity in the level of logical reasoning results in higher level of computation to account for external variants thus providing the decision appropriate to the given problem. This cannot be accomplished without the ability to deliver process through the existing logical reasons from the application's knowledgebase. Hence the processing speed and efficiency of computation in terms of both the architecture and software capability is a question that must be addressed to implement such a system.

Although the *advice taker* software is viable in a hardware architecture perspective, the hurdle is the software component that must be capable of delivering the abstraction level discussed by the author. This is because, the ability to change the behaviour of the system by merely providing verbal commands from the user which is the main challenge faced by the AI application developers. This is so because of the fact that the effective implementation of the aforementioned can be achieved only with the effective usage of the speech recognition and logical reasoning that is already available to the software for incorporating the new logical reason as an improvement or correction to the existing set-up of the application. This approach is the major hurdle which also poses the challenge of identifying the key speech patterns that are deemed to be such corrective commands over the statements' classification provided by the user author for providing information to the application. From the above arguments it can be concluded that the author's statement - " If one wants a machine to be able to discover an abstraction, it seems most likely that the machine must be able to represent this abstraction in some relative simple way" - is not a task that is easily realisable. It is also necessary to address the issue that the

abstractions that can be realised by the user can be realised by an AI application only if the application being used already has a set of reasons or room for learning the reasons from existing reasons prior to decision-making. This process can be accomplished only through complex algorithms as well as error propagation algorithms discussed in section 2. 3. This makes it clear that the realization of the *advice taker* software's capability to deliver to represent any abstraction in a relative simpler way is far fetched without the appropriate implementation of self-corrective and learning algorithms. The fact that learning is not only through capturing the previous actions of the application in similar scenarios but also to generate logical reasons based on the new information provided to the application by the users is an aspect of AI application which is still under development but the necessary ingredient for the *advice taker* software. However, considering the timeline associated with the research presented by Dr McCarthy and the developments till date, one can say that the AI application development has seen higher level of developments to interpret information from the user to provide an appropriate decision using the logical reasoning approach. The author's argument that for a machine to learn arbitrary behaviour simulating the possible arbitrary behaviours and trying them out is a method that is extensively used in the twenty-first century implementation of the artificial intelligence for computers and robotics. The knowledge developed in the machines programmed using AI is mainly through the use of the arbitrary behaviours simulated and their results loaded into the machine as logical reasons for the AI application to refer when faced with a given problem.

Form the arguments of the author on the five features necessary for an AI application hold viable in the current AI application development environment although the ability of the system to create subroutines which can be included into procedures as units is still a complex task. The magnitude of the processor speed and related requirements on the hardware architecture is the problem faced by the developers as opposed to the actual development of such a system. The author's statement that '*In order for a program to be capable of learning something it must first be capable of being told it*' is one of the many components of the AI application development that has seen tremendous development since the dawn of the twenty-first century (Jones, 2008). The multiple layer processing strategy to address complex problems in the real world that have influential variants both within the input provided as well as the output in the current state of AI application development is synonymous to the above statement by Dr McCarthy.

The neural networks for adaptive behaviour presented in great detail by Pfeifer and Scheier (2001) ^[8] further justifies the aforementioned. This also opens room for discussion on the extent to which the *advice taker* application can learn from experience through the use of neural networks as an adaptive behaviour component for programming robots and other devices facing complex real-world problems. This is the kind of adaptive behaviour that is represented by the *advice taker* application by Dr McCarthy who described it nearly half a century ago. The viability of using neural networks to take comments in the form of sentences (imperative or declarative) is

plausible with the use of the adaptive behaviour strategy described above using neural networks.

Finally, the construction of the *advice taker* described by the author can be met with in the current AI application development environment although the viability of the same would have been an enormous challenge at the time when the paper was published. The *advice taker* construction in the twenty-first century AI environment can be accomplished using either a combination of computers and robotics or one of the two as a sole operating environment. So development of the AI application either using computers or robotics for the delivery of the *advice taker* is plausible depending upon the delivery scope for the application and its operational environment. Some of the hurdles faced however would be with the speech recognition and the ability to distinguish imperative sentences to declarative sentences. The second issue faced in the case of the *advice taker* will be the scope of application as the simulation of various instances for generating the knowledge database is plausible only within the defined scope of the application's target environment as opposed to the non-feeble human mind that can interact with multiple environments at ease. The multiple layer neural networks approach may help tackle the problem only to a certain level as the ability to distinguish between different environments when formed as layers is not easily plausible without the knowledge on its interpretation stored within the system. Finally, a self-corrective system for AI application is plausible in the twenty-first century but the self learning system using the logical reasons provided is still scarce and requires a greater level of design resilience to account for input and output variants of

the system. The stimulus-response forms described by the author in the paper is realisable using the multiple layer neural networks implementation with the limitation on the scope of the *advice taker* restricted to a specific problem or set of problems. The adaptive behaviour simulated using the neural networks mentioned earlier justifies the ability to achieve the aforementioned.

3. 2: Paper 2 – A Logic for Default Reasoning

Default reasoning in the twenty-first century AI applications is one of the major elements that attribute to the effective functioning of the systems without terminating unexpectedly unable to handle the exception raised due to the combination of the logic as argued by Pfeifer and Scheier (2001). This is naturally because of the fact that the effective use of the default reasoning process in the current AI application development environment aims to provide default reasoning when an exhaustive list of the reasons that are simulated and rules combinations are effectively managed. However, the definition of exhaustive or the perception of an exhaustive list for the development in a given environment is limited to the number of simulations that the users can develop at the time of AI application design and the adaptive capabilities of the AI system post implementation (refernece8). This makes it clear that the effective use of the default reasoning in the AI application development can be achieved only through handling a wide variety of exceptional conditions that arise in the normal operating environment for the problem being simulated (Pfeifer and Scheier, 2001). In the light of the above arguments the assertion by the author on the default

reasoning as beliefs which may well be modified or rejected by subsequent observations holds true in the current AI development environment.

The default reasoning strategy described by the author is deemed to be a critical component in the AI application development mainly because of the fact that the defaulting reasons are not only aimed to prevent unhandled exceptions leading to abnormal termination of the program but also the effective learning from experience strategy implemented within the application. The learn from experience described in the section 2 as well as the discussion presented in section 3. 1 reveal that the assignment of a default reason for an adaptive AI application will provide room for identifying the exceptions that occur in the course of solving problems thus capturing new exceptions that can replace the existing default value. Furthermore, the fact that the effective use of the default reasoning strategy in AI applications also limits the learning capabilities of the application in cases where the adaptive behaviour of the system is not effective although preventing abnormal termination of the system using the default reason.

The logical representation of the exceptions and defaults and the interpretation used by the author to interpret the phrase 'in the absence of any information to the contrary' as 'consistent to assume' justifies the aforementioned. It is further evident from the arguments of the author that the default reason creation and its implementation into the neural network as a set of logical reasons are complex than the typical case wise conditional analysis on establishing a given condition holds true to the situation on hand. Another interesting factor to the aforementioned is the fact that the definition of the conditions must incorporate room for partial success owing

<https://assignbuster.com/reasoning-in-artificial-intelligence-ai-a-review/>

to the fact that the typical logical approach of success or failure do not always apply to the AI application problems. Hence it is necessary to ensure that the application is capable of accommodating partial success as well as accounting for a concrete number to the given problem in order to generate an appropriate decision. The discussion on the non-monotonic character of the application defines the ability to effectively formulate the condition for default reasoning rather than merely defaulting due to the failure of the system to accommodate for the changes in the environment as argued by Pfeifer and Scheier (2001). Carbonell (1980) ^[9] further argues that the type hierarchies and their influence on the AI system have a significant bearing on the default reasoning strategies defined for a given AI application. This is naturally because of the fact that the introduction of the type hierarchies in the AI application will provide the application to not only interpret the problem against the set of rules and reference data stored as reasons but also assign it within the hierarchy in order to identify the viability of applying a default reason to the given problem. The arguments of Carbonell (1980) on Single-Type and Multi-Type inclusion with either strict or non-strict partitioning justify the above-mentioned argument. It is further critical to appreciate the fact that the effective implementation of the type hierarchy in a logical reasoning environment will provide the AI application with greater level of granularity to the definition and interpretation of the reasons pertaining to a given problem (Pfeiffer and Scheier, 2001). It is this state of the AI application that can help achieve a significant level of independence and ability to interact effectively in the environment with minimal human intervention. The discussion on the inheritance mechanisms presented by Carbonell (1980) alongside the implementation of the inheritance properties

<https://assignbuster.com/reasoning-in-artificial-intelligence-ai-a-review/>

as the basis for the implementation of AI systems in the twenty-first century (Pfeifer and Scheier, 2001) further justify the need for default reasoning as an interactive component as opposed to a problem solving constant to prevent abnorm