Knowledge representation and reasoning (KRR) is a subfield of artificial intelligence (AI) that is concerned with understanding, designing, and implementing ways of representing information in computers so that agents can use that information to reason and solve problems [3]. Representation of knowledge and reasoning is central to the entire field of artificial intelligence [3]. The agents build primarily around this paradigm are often referred to as knowledge-based agents. The central component of a knowledge-based agent is its knowledge base, or KB [1]. A knowledge base is a set of sentences. Each sentence is expressed in a language called a knowledge representation language and represents some assertion about the world [1]. Inferencing and decision procedures constitute the reasoning component of a knowledge-based system. Deriving information that is implied by the information that is already present is a form of reasoning [3]. Inference must obey the primary requirement that the new sentences should follow logically from the previous ones [3]. Knowledge representation is hard because a "minimal" common-sense system must "know" something about cause-and-effect, time, purpose, locality, process, and types of knowledge. It also needs ways to acquire, represent, and use such knowledge [7].

There are several types of knowledge, such as declarative/descriptive and procedural [3]. There are also several formats for knowledge representation such as frame networks, declarative languages (logics), imperative languages (Java, Python, etc.), semantic networks, product rules, neural networks, genetic algorithms etc. [3]. Each of knowledge representation formats has its own advantages and limitations. The limitations are caused by two of the most fundamental problems in the field of expert systems, namely knowledge acquisition and representation and search[2]. In this paper we seek to answer five questions: What are some logics? How are they used for knowledge representation? What is a frame system? How is a frame system used for knowledge representation? What are the advantages and disadvantages of using each type of system?

There are several different types of logics, including: propositional, first order, second order, modal, fuzzy etc. [5]. We shall only concern ourselves with propositional and first order logics. Logic was the dominant paradigm in AI before the 1990s, but it had some drawbacks due to it being deterministic and rule based [4]. Despite these drawbacks it is very expressive and compact [4]. The goals of logical languages are to represent and reason about knowledge in the real world [4], there is a direct relation between the goals of KRR and logical languages. In propositional language there are propositional symbols (A, B, etc.) and logical connectives (not, and or, implication, bidirectional implication) [4]. A logical formula compactly represents a set of models [4]; for example, if we have propositional symbols P and Q, then P V Q represents all the worlds where P is true, or Q is true. In a trivial manner true represents all models, and false represents no models. In the logical paradigm, each sentence in a KB can be thought of as a logical formula that describes a set of models. As you add new information to your knowledge base it shrinks to represent the models where both your current knowledge and the new information are both true, in other words M(KB) intersect M(f) (M being a function that returns the set of models where the logical formula passed in is true). There are a couple of ways in which this can change the KB. One means is entailment, which is when the KB does not change at all because M(KB) is a subset of M(f) [4]. Another case is contradiction, which is when M(KB) intersect M(f) is empty [4]. The final case is contingency, which is when the empty set is a strict subset of M(KB) intersect M(f) and that is a strict subset of M(KB) [4]. There is also a relationship between contradiction and entailment KB contradicts f if KB entails not(f) [4]. There are two primary operations that can be performed on a KB, ASK and TELL. The TELL operation adds information to the KB by means of and(KB, f) resulting one of the previously mentioned cases. One can also ask the KB a question f, ask(KB, f), if f is entailed by the KB the answer is yes (or true), if f is contradicted by the KB, then the answer is no, but if f is contingent on the KB the answer is ‘I don’t know’ [4]. We can then reduce contingency, entailment, and contradiction to one primitive: satisfiability. A knowledge base is satisfiable if M(KB) is not empty, it entails f if KB union not(f) is not satisfiable, it contradicts f if KB union not f is satisfiable and KB union f is not satisfiable, and f is contingent if KB union not f is satisfiable and KB union f is satisfiable [4]. For logical inferencing using propositional logic our agent can use either modus ponens or resolution [6]. Modus Ponens is sound and complete for propositional logic with horn clause, and resolution is complete for propositional logic in general. Unfortunately, resolution has exponential time complexity while Modus Ponens is linear [6]. We will focus our attention on resolution, which is both sound and complete for all propositional logic. Resolution works by operating on a disjunction of atomic symbols formulas p1,…, pk and q1,…,qk where px and qx represent the same logic symbol r and p = r and q = not r, the those two symbols cancel one another. A resolution-based inference algorithm starts by adding not f to the knowledge base converts all formulas into conjunctive normal form (CNF), repeatedly applies the resolution rule, and returns entailment if it derives false. If it doesn’t derive false, it can rerun the algorithm with f and return contradiction if it derives false and contingency otherwise. CNF is a conjunction of clause, where each each clause is a disjunction of atomic symbols e.g., P or not P [6]. Every formula in propositional logic can be converted into conjunctive normal form [6]. So, any KB where the information is encoded in propositional logic can be evaluated using resolution. But unfortunately, propositional logic is limited in its expressiveness. Thus first-order logic adds variables, functions, and quantification. First-order logic has two types of quantifiers, universal and existential. The universal quantifier argue that every member of a group meets a condition, and the existential quantifier argues at least one member of a group meets a condition. (Add goldbach’s conjecture to the video, every even integer greater than 2 is the sum of two primes: for all x there exist y, z where Even(x) and Greater(x, 2) then Equals(x, Sum(y, z)) and Prime(y) and Prime(z) [6]) We also impose the restriction that there is a one-to-one mapping from object to constant symbol in first-order logic, which brings about the idea of propositionalization where first-order logic is just syntactic sugar for propositional logic and as a result we can use any inference algorithm for propositional logic on first-order logic [6].

Logic as a form of knowledge representation may seem very attractive. But, in relation to these systems, some people believe in simple cases one can get such systems to "perform," but as we approach reality the obstacles become overwhelming. The problem of finding suitable axioms–the problem of "stating the facts" in terms of always-correct, logical, assumptions is very much harder than is generally believed [7]. Another formalism for representing knowledge is referred to as a frame. A frame is a data-structure for representing a stereotyped situation, like being in a certain kind of living room, or going to a child’s birthday party [7]. We can think of a frame as a network of nodes and relations [2]. Collections of related frames are linked together into frame-systems . The effects of important actions are mirrored by transformations between the frames of a system [7]. A frame's terminals are normally already filled with "default" assignments [7]. The "top levels" of a frame are fixed and represent things that are always true about the supposed situation. The lower levels have many terminals–"slots" that must be filled by specific instances or data. Each terminal can specify conditions its assignments must meet. (The assignments themselves are usually smaller "sub-frames.") Simple conditions are specified by markers that might require a terminal assignment to be a person, an object of sufficient value, or a pointer to a sub-frame of a certain type. More complex conditions can specify relations among the things assigned to several terminals [2, 7]. The frame system supports the so-called closed world inferring paradigm (Detwiler et al. 2016), where all facts that are presented in the system are true. If some fact is not presented, that means that it is untrue. It allows avoiding errors in inferring mechanism related to the knowledge representation format [2]. Other formats as, for example, ontology, may support the open-world paradigm, where all facts that are not presented may also be true [2]. A frame-based knowledge base is one of the typical models or a part of such models for knowledge representation in expert and decision- making systems [2]. For example, in [8] the authors developed a question answering (QA) system using a textual KB constructed from a biology textbook, and in [9] the authors developed an emergency management system using a KB constructed from domain expertise. In order to support the decision-makers in the evaluation of this raw information, knowledge-based systems (KBS) are good candidates, as they are able to integrate both theoretical and common-sense knowledge directly taken from the expert decision-makers. Furthermore, KBS can provide explanations of their recommendations; this is of fundamental importance in any emergency domain, as the responsible personnel cannot adopt a decision without fully understand it [9]. In general, the design of a knowledge model is based on a sequence of refinement steps, starting from a general valid reasoning method capable of meeting the goals of the target application [9]. An example is a KB as a cache of answers from a QA system [8].

"The frame begins the verification process by checking any sample features that it already has on hand - features that arrived in the first wave or were obtained while testing previous hypotheses. Then, if the hypothesis has not already been accepted or rejected, the frame begins asking questions to get more information about features of the sample. The nature of these questions will vary according to the problem domain: A doctor program might order some lab tests, a vision program might direct its low-level components to look at some area more closely. Sometimes a question will recursively start another recognition process: 'This might be a cow–see if that part is an udder.' "The order in which the questions are asked is determined by auxiliary information in the frame. This information indicates which features are the most critical in the verification at hand, how these priorities might be affected by information already present, and how much each question will cost to answer. As each new feature of the sample is established, its description is added to a special packet of information about the sample, along with some indication of where the information came from and how reliable it is. This packet can be taken along if the system moves to another hypothesis. Sometimes unsolicited information will be noticed along the way; it, too, is tested and thrown into the pot [7].

Sometimes the sample will appear to fit quite well into some category, but

there will be one or two serious violations. In such a case the system will

consider possible excuses for the discrepancies: Perhaps the cow is purple

because someone has painted it. Perhaps the patient doesn't have the expected

high blood pressure because he is taking some drug to suppress it. If a

discrepancy can be satisfactorily explained away, the system can accept the

hypothesis after all. Of course, if the discrepancies suggest some other

hypothesis, the system will try that first and resort to excuses only if the new

hypothesis is no better. Sometimes two categories will be so close together that

they can only be told apart by some special test or by paying particular

attention to some otherwise insignificant detail. It is a simple enough matter

for both of the frames to include a warning of the similarity and a set of

instructions for making the discrimination. In medicine, such testing is called

differential diagnosis. [7]

Bibliography

[1] Russell, S. J., & Norvig, P. (2010). *Artificial Intelligence: A modern approach*. Prentice-Hall.

[2] Nazaruks, V., & Osis, J. (2017). A survey on domain knowledge representation with frames. *Proceedings of the 12th International Conference on Evaluation of Novel Approaches to Software Engineering*. https://doi.org/10.5220/0006388303460354

[3] Chaturvedi, V. (n.d.). *Understanding Artificial Intelligence and machine learning*. Udemy. Retrieved September 26, 2022, from https://www.udemy.com/course/understanding-artificial-intelligence-and-machine-learning/

[4] stanfordonline. (2020, December 17). *Logic 1 - propositional logic | Stanford CS221: Ai (Autumn 2019)*. YouTube. Retrieved September 26, 2022, from https://www.youtube.com/watch?v=xL0kNw5TudI

[5] YouTube. (2016, January 11). *An introduction to formal logics*. YouTube. Retrieved September 26, 2022, from https://www.youtube.com/watch?v=\_MhgsoPHvYo&list=PLJ5C\_6qdAvBG8HP77xIOVxH4Jzq3RQ9dI

[6] stanfordonline. (2020, December 17). *Logic 2 - first-order logic | Stanford CS221: Ai (Autumn 2019)*. YouTube. Retrieved September 26, 2022, from https://www.youtube.com/watch?v=\_Iz83hfkFds

[7] *A framework for representing knowledge Marvin Minsky mit-AI laboratory ...* (n.d.). Retrieved September 28, 2022, from https://courses.media.mit.edu/2004spring/mas966/Minsky%201974%20Framework%20for%20knowledge.pdf

[8] Inc., P. C. V., Clark, P., Inc., V., Inc., P. H. V., Harrison, P., Washington, N. B. U. of, Balasubramanian, N., Washington, U. of, Washington, O. E. U. of, Etzioni, O., & Metrics, O. M. V. A. (2012, June 1). *Constructing a textual KB from a biology textbook: Proceedings of the joint workshop on automatic knowledge base construction and web-scale knowledge extraction*. DL Hosted proceedings. Retrieved September 28, 2022, from https://dl.acm.org/doi/10.5555/2391200.2391214

[9] Hernández, Josefa & Serrano, Juan. (2001). Knowledge-based models for emergency management systems. Expert Systems with Applications. 20.