Can (Should) We Incorporate Common Sense Into AI/ML Solutions?

John A. Stankovic, University of Virginia

Artificial intelligence and machine learning have been addressing how to incorporate common sense into solutions for many decades. While some progress has been made, due to the incredibly complex issues involved, we are far from a general solution.

Artificial intelligence (AI) and machine learning (ML) have made tremendous progress since their inception. However, solutions using AI/ML are subject to hallucinations, errors, and nonsensical outputs, some of which can be life threatening, especially in application domains, such as smart health and smart cities. Many efforts are being made to address these problems. One direction is incorporating common sense into the solutions.

Digital Object Identifier 10.1109/MC.2026.3677545
Date of current version: 28 July 2026

Since common sense has been researched for decades, there are many approaches and results. Research directions include training large language models, employing knowledge graphs, using formal logics, multimodalities, and neurosymbolic AI, and employing crowd sourcing and expert knowledge. Recent works in the literature on common sense for AI/ML discuss general issues[1],[2] describe curated common sense data sets[1] and apply neurosymbolic techniques[3] and logic, such as answer set programming.[4] However, despite these advances the problem remains open. This leads one to ask, is it even possible to achieve general common sense in AI/ML? If it is possible, or possible to some degree, should we even try to incorporate common sense, or is there a better approach?

The main goals of this article are 1) to investigate what is the state of common sense research in AI/ML via the novel perspective of can we or should we incorporate any such solutions into systems, 2) propose key research directions with examples, and 3) highlight various open questions.

DEFINITIONS

Psychologists and AI researchers have had difficulty in defining common sense. It is one of those terms that is difficult to define, but you know it when you see it. However, with vague definitions can we determine if an AI/ML solution achieves common sense? Nevertheless, here are two typical definitions.

Definition 1: Common sense is truths that are self-evident.[5] Note that this definition focuses on common sense knowledge. But, as we will see, a major complexity is that common sense knowledge is highly personal.

Definition 2: Common sense is the ability to reason intuitively about everyday situations and events.[1] Note that this definition is about common sense reasoning. Complexity here arises from many factors including that intuition is involved and that context plays a significant role.

In summary, what makes defining common sense so difficult is that it is a function of the domain, context, personal beliefs, culture, intuition and more, each of which can be incredibly complex. What is considered common sense also changes over time. More detailed discussions of these issues are in the next two sections.

CAN WE INCORPORATE COMMON SENSE INTO AI/ML SOLUTIONS

To answer this question in a general sense let us consider a complexity theory argument. We know from complexity theory[6] that some problems are intractable, meaning that the solution times grow exponentially with the problem size. I postulate that the same is true for common sense solutions.

Common sense is a function of domain, culture, personal beliefs, age, gender, time, location, history, current events, social status, emotions, intuition, etc. This is an open-ended list and provides important context for common sense. If AI/ML solutions don't cover the context correctly we can have nonsensical results. Further, correlations exist among these factors giving rise to exponential growth in the number of things to consider. Consequently, we can consider handling all the potential cases as intractable.

As a concrete example, consider an AI/ML solution that knows about gravity and a person is said to drop an item. The system may conclude that it drops quickly to the ground. But what if the person is lying down, in space, is scuba diving, jumping on a trampoline, the item was a feather, and it was windy, etc. If any of this context is missing for developing the solution, then a nonsensical answer may appear.

For intractable problems, they are addressed in many ways, for example, with heuristics, with approximations, or by constraining the general problem. This needs to be better recognized and applied to common sense.

Also, in complexity theory, we find that many problems are tractable, for example, they can be solved in linear, log time, etc. Is it possible to create complexity classes for common sense where perhaps by constraining the problem enough it becomes tractable with guarantees on solutions being sensical and thereby avoiding hallucinations and errors? Is there even one such class for the simplest problems? While it is possible that negative results will appear for this approach, the process may provide insights on other ways to categorize common sense tractability. This is an open problem.

SHOULD WE INCORPORATE COMMON SENSE INTO AI/ML SOLUTIONS

At first, it may seem like a poor question to ask if we should incorporate common sense into AI/ML solutions, especially since attempts have been ongoing since the inception of AI. However, the question has key nuances that bring to light interesting questions.

For example, a recent study[5] has shown that in general there is often little consensus as to whether an output from an AI/ML solution is failing at common sense. This study uses curated data sets that are specifically built (some over many years) to codify common sense knowledge. Yet, the study showed that, in general, people do not agree on whether a particular output makes common sense. This is troubling because why should we work to ensure common sense when, in the end, there is no clear agreement that it is achieved? The study also showed that for the simpler type of common sense problem, such as for technology facts, the agreement was in the 80% range. When human behavior and social norms were key aspects of the problems, agreement dropped to near 50%. Again, what good would it be to incorporate common sense at a high cost (execution time/memory/power) only to have little agreement on whether the output made common sense?

This referenced study[5] also showed that common sense was highly personalized. This implies that solutions must learn a user's preferences so that outputs will make common sense to them. Note that common sense to someone may still be a wrong output in terms of accuracy. How and when can we override a supposedly common sense output?

Referring to the notion of common sense complexity classes mentioned in the previous section, if we can find problem classes which are tractable, then maybe there will be high agreement that a solution adheres to (personalized) common sense.

It is important to stress that the real problem to solve is to avoid hallucinations, errors, and nonsensical outputs. If common sense can contribute to that, then indeed it should be incorporated.

SOLUTION DIRECTIONS

To avoid hallucinations and wrong answers, it is necessary (but not sufficient) to follow a robustness methodology. Robustness includes many steps, such as data cleaning, data augmentation, adversarial training, early stopping to avoid overfitting, using robust loss functions, hyper-parameter optimization, uncertainty quantification, and use of formal methods to specify properties that the resultant models must have. Robustness also refers to how sensitive are the trained models to input data and new situations not explicitly trained for. While there is some overlap between common sense and robustness, common sense is not the same as robustness. Common sense relies on general knowledge and intuition, so it is likely that common sense solutions are still required to supplement robustness.

Formal methods play a role in both robustness and common sense. Applying formal methods can minimize hallucinations, errors, and nonsensical outputs, but it is necessary to know the properties you want to guarantee ahead of time, and it is difficult to encode all the underlying assumptions and contexts that go along with a particular property. The use of formal methods can be implemented by a student-teacher approach.[7]

To illustrate how we might develop common sense solutions, consider three examples of smart agents. The examples are chosen to highlight different aspects of common sense. The first example concerns smart agents to aid a first responder at an emergency response incident. This type of application requires immediate answers as the first responder is administering aid. The second example considers smart health where safety critical issues are foremost. The third example is a smart agent for a smart city. Here, it is necessary to forecast how the state of the city will evolve over some future time period, such as 6 or 24 hours, given significant uncertainties of what might occur in the city. Suggestions and solutions from all three types of smart agents must be robust and convey common sense answers. However, 100% guarantees are not likely.

To better follow the examples, each of them follows a proposed common sense solution architecture shown in Figure 1.

First responders example

A smart agent can aid an emergency responder in assessing the condition of the patient and choosing the correct treatment protocols to administer. To create such an agent, start with a large language model (LLM) to have a basis of language understanding. Tune such a model to support the style

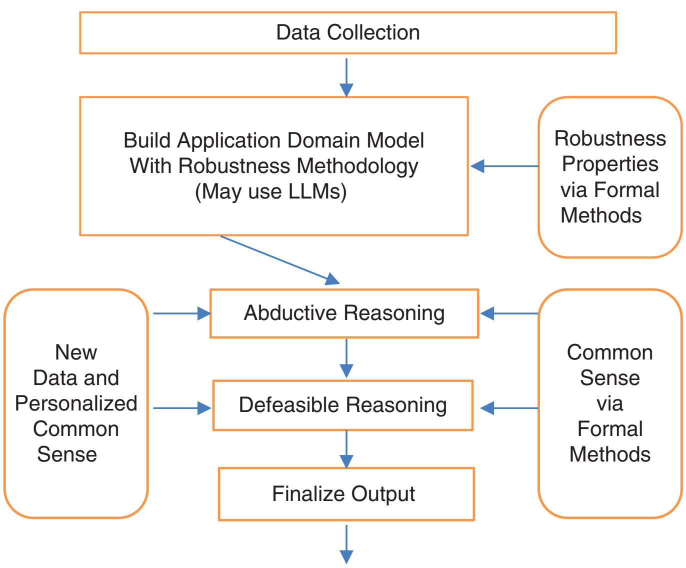

FIGURE 1. Proposed common sense architecture.

of question-and-answer interactions. Further, tune the model to medical terminology and information. However, general medical information is not adequate for first response applications, which have very specialized terms and semantics. Hence, tune again using language from the first response domain.

At this point, solutions should then be enhanced to support robustness using the methodology steps as described previously. It is highly encouraged to apply formal logic to both describe robustness requirements and embed that knowledge into the models. Constraints might include difficult to remember information, such as "never administer a certain drug to a diabetic," or obvious information, such as "always check the air passageway for an unconscious patient." While this latter information is obvious, it can be forgotten under very stressful incidents. One approach for implementing this utilizes a student teacher network.

At this point, the solution should reduce hallucinations, errors, nonsensical results, but one can still expect such outputs at times. Also, since this is a quite complicated application, there is not likely always a single viable answer. To strengthen the solution with some degree of common sense I believe it is necessary to apply abductive[8] and defeasible[9] reasoning.

Abductive reasoning will produce multiple plausible suggestions for the first responder. Presenting (some of) them to the first responders enables a runtime human check on nonsensical outputs. The smart agent may suggest giving the patient oxygen, starting an IV, and having the patient stand up. Having the patient stand may be nonsense. Importantly, implementations of abductive reasoning in this application has to be very efficient to react in real time.

As the patient is undergoing a medical emergency and as treatment is applied, there may be significant new information (for example, maybe cardiac arrest happens while being treated for dizziness) that requires stepping back and supplying very new suggestions. This capability can be supported by defeasible logic programming and defeasible reasoning solvers. This requires explicitly representing defeasible rules and exceptions and deciding how to integrate processing those rules in an efficient manner and together with the other architectural components of the solution. This would allow, for example, a reset on what treatments to propose due to a new event, such as cardiac arrest.

What is important about defeasible reasoning solvers is that they 1) handle incomplete or conflicting information, 2) allow conclusions to be retracted when new evidence occurs, and 3) use tentative reasoning via specified rules and preferences.

Smart health example

Consider a smart health application for an elderly person living alone in a private home. The smart home may have in-situ sensors measuring environmental factors and activities of daily living, and on-body sensors measuring physiological states of the elderly person. There may also be a smart agent that verbally interacts with the person. While many of the interactions may mainly be helpful with respect to daily activities, there is a safety critical aspect if a serious medical issue arises. It is important that the solution recognizes such conditions and acts appropriately in real time.

Since the agent and person will interact via voice, LLMs should be used to understand the semantics of language. Additional model training needs to include information about medical emergencies, such as cardiac arrest, falling down and being hurt, blood pressure very high, and depression. Robustness of the models can be improved by adding constraints via formal models. Such constraints should be as comprehensive as possible. The list of constraints include what drugs must be taken, drug dosage must not exceed a certain amount, talk of suicide, and many others. While there is no guarantee that every safety critical situation is covered, the designers of this solution, by using formal methods, can at least explicitly understand what emergencies are indeed covered. It is likely that the most common medical emergencies can be covered and, although not ideal, for more rare emergencies, their coverage can improve over time.

Advice, whether it is for emergencies or daily living, will often have multiple plausible suggestions. Using abductive reasoning, for daily activities, the agent can present various options to the patient. This avoids monotony for the patient on interactions with the agent. For safety critical events, the solutions should (automatically) include actions for the emergency, such as notifying a doctor, calling for an ambulance, or turning off the stove.

Defeasible reasoning is also necessary. There are many activities of daily living that have similar profiles and only after enough data are collected can one distinguish between them. For example, cooking and making coffee have actions that overlap. Washing and showering have some similar movements. Various exercises may have substeps that are the same. Keeping track of eating, hygiene, and exercising are important aspects of an in-home smart health app that provides suggestions based on these assessments. As new conditions are recognized, suggestions made by the agent may have to be completely revised.

As is true for all the examples, since the systems are dynamic, new data can include new core knowledge, personalized common sense and context. This requires dynamic updates to knowledge needed by, at least, abductive and defeasible rules which guide their reasoning.

Smart cities example

An AI/ML agent helping decision makers of smart cities tries to predict the future city states so that actions can be taken now to mitigate future problems. For example, based on current traffic, air quality index, time of day, weather predictions, special events like a baseball game, etc., the smart agent predicts if there will be traffic congestion and air quality violations in the next 6 hours. If so, actions can be suggested to reduce the projected congestion and improve air quality. Suggestions should be robust in the sense if changes occur (say an accident), then the solution should not have to change too dramatically, such as all the traffic light timings and all the police locations have to change. Also, nonsensical solutions must be avoided, for example, directing traffic down a one-way street the wrong way.

For smart cities, it is necessary to begin with a large data set of the past history of the operation of the city. Since there is great uncertainty in such a complex and dynamic environment, solutions addressing uncertainty, such as using a Bayesian neural net are suggested.[10] Sensor data collected to represent the current states of the city may have errors or be out-of-date. Using standard techniques, such as a long short-term memory (LSTM) predictions into the future are possible, but an LSTM only predicts a single possible future trajectory for the smart city. This does not adequately address the uncertainty of what might happen in the future in such a dynamic environment. Instead, using a Bayesian recurrent neural network can capture the inherent future uncertainty by generating what is called a flowpipe. A flowpipe contains many possible future trajectories so that one has a form of confidence level of future trajectories. A flowpipe can be integrated with using formal specification of the requirements and thereby detect requirement violations probabilistically. It is possible that none, some, or all flowpipe trajectories violate the requirements. This approach incorporates a significant degree of robustness. The solution then should be integrated with abductive and defeasible reasoning to support adding common sense.

Abductive reasoning is necessary because smart cities are so complex with many alternative solutions for a given situation. For example, if a major incident occurs there can be multiple hypotheses of how to act; the number of ambulances and police cars and from which hospitals and police stations should be deployed, what streets to block, how to inform the community, etc., each with different costs and predictions of successful outcomes. Decision makers can be presented with plausible action plans and choose one of them. One key complexity is that it is possible that multiple decision makers are involved each with their own personalized common sense related to their own objectives. For example, a traffic engineer may want to choose a solution that reduces driving delays while an environmental engineer may try to reduce pollution. Conflicts must be detected and resolved, possibly by the smart agent or by the decision makers or both.

Defeasible reasoning is necessary to enable reacting to major changes in the state of the city. For example, a major incident report may turn out to be a false alarm, or a subsequent city-wide or localized area blackout may occur.

OPEN QUESTIONS

While significant progress has been made on understanding and developing AI/ML solutions for common sense, many open problems remain. Several important ones are described next.

It seems that one of the most important issues is *how to represent and use context in all its complexity*. Such context will need to be dynamic, personalized, nuanced, and applied at differing levels of detail. For example, if common sense is represented in a knowledge graph, that graph may need to contain context about a person's health, such as age, gender, comorbidities, drug prescriptions, allergies, where they live, local hospital, psychological state, etc. This information may have to often be updated and new context added as a function of a person's changing health. How new context affects common sense seems open ended. Nuances of how the various context interacts with other context has to be inferred as it is not likely that all possible context interactions will be part of the knowledge graph. Research in this area is found under the term context engineering.[11]

In many applications, solutions will require iterative sequences of predictions into the future. The core of this problem is *how to address uncertainty in its many forms*. This ranges from environmental uncertainty to human behavior and emotional dynamics uncertainty. Some current solutions simply output the highest probability result. This is not adequate. Other solutions present multiple feasible hypotheses. However, new research should better integrate the hypothesis generation and scoring with robustness and personalized common sense. Addressing uncertainty is a very active research area.

As abductive reasoning will likely be at the core of common sense solutions, as it was true for partially dealing with uncertainty, it is necessary to better solve the problems of *how to score and choose the plausible hypotheses derived by AI/ML solutions*. Some solutions might involve interactions with users to help choose among the alternatives. However, since a user's common sense can be inaccurate, how can the agent decide between its own preferences for a solution versus being preempted by the user's common sense? As in the aforementioned smart city example, conflicts among the hypotheses must be resolved when multiple decision makers are involved. Solutions must also allow for defeasible reasoning as time elapses.

Defeasible reasoning is essential. Compared to other AI/ML topics, it has received too little attention. Open questions include *how to represent defeasible rules and exceptions, how to efficiently handle these rules and conflicts among them at runtime, and how to integrate defeasible reasoning into a comprehensive robust and common sense architecture*. Learning from large data alone may find some common sense facts and support some common sense reasoning, but additional facts, rules, exceptions, missing context, support for inference, etc. will likely require a multicomponent architecture that integrates, at least, large language models, robustness solutions, and abductive and defeasible components.

Can we develop complexity classes for common sense solutions that can identify when and where common sense will likely be correct and others where they are intractable? *Can we define the boundaries of what a particular solution can and cannot do*? Perhaps various logics will be able to do this at least for some classes of problems. For example, logic can define various constraints, such as a "solution should not send a police car on a call down a one-way street the wrong way," and such constraints can be guaranteed to be adhered to in a final output. But, logic alone may not be capable of specifying certain types of constraints and adding every possible constraint you can think of to frame a solution does not scale.

How can neuroscience, brain-memory concepts, and results from psychology contribute to the fundamental understanding of common sense? As these results appear, it is likely to contribute to the way in which the components of the proposed architecture are realized.

How will reinforcement learning play a role in supporting common sense solutions?[12] This may be especially important in personalizing common sense. However, reinforcement learning solutions that learn everything from scratch will not likely be able to or take too long to learn enough common sense to be effective. Reinforcement learning would also have to be integrated with the other architectural components described in this article. Some application types, for example, safety critical ones, may not be suitable for reinforcement learning.

FINAL REMARKS

The proposed architecture in this article is a more comprehensive collection of modules (see Figure 1) than in current solutions. Four main common sense architectures are often used today. One, neurosymbolic approaches combine deep neural networks with rule-based AI. Two, world models contain an internal representation of the environment, such as intuitive physics. Three, LLMs try to learn common sense from large data sets. Four, multimodal solutions incorporate visual, audio and text information as inputs to the models. The proposed architecture could use any of these four approaches as the main imodule labeled "Build Application Domain Model With Robustness Methodology."

Since common sense solutions are highly personalized, other issues must be simultaneously handled, such as privacy, security, data collection, user consent, and how and when to override user's common sense.

There are research works that argue that formal logics are too brittle and suffer from scaling issues to be used for common sense. However, formal logics can support various aspects of robustness and abductive and defeasible reasoning and, therefore, can and should complement common sense solutions.

Common sense requires intuitive reasoning and is generative and instantaneous. Solutions cannot weigh all (or even many) choices like an ML model might do with probabilistic scores on every choice. Efficient real-time components and architectures must be developed.

ABOUT THE AUTHOR

JOHN A. STANKOVIC is the BP America Professor Emeritus in the Computer Science Department at the University of Virginia, Charlottesville, VA 22903 USA. His research interests include developing intelligent-based cyberphysical systems. Stankovic received his Ph.D. in computer science from Brown University. He is a Fellow of both the IEEE and the ACM. Contact him at stankovic@cs.virginia.edu.

This article focused on major aspects of the approach to common sense, including context, personal views, robustness, abductive and defeasible reasoning, and the need for comprehensive architectures. There are many other important issues that are outside the scope of this article, including context engineering, abstract thinking, non-monotonic logic, negative case understanding, real-time processing, counterfactual knowledge, naïve knowledge of physics, knowledge of social norms, reasoning in unseen situations, spatial and temporal reasoning, and for systems to know what they don't know.

Achieving common sense in AI/ML is incredibly complex. If successful, a large reduction of hallucinations, errors, and nonsensical outputs would occur. This article argues that, in a general sense, it is not likely possible to develop a general solution for common sense due to considerable nuances and context required, and because in many domains common sense is highly personal. The article also suggests that development of improved robustness techniques can mitigate some or even many of the nonsensical results today's models sometimes exhibit. Nevertheless, since common sense is not the same thing as robustness, adding solutions for common sense, especially targeted for the domain of the application, will be helpful and feasible. It is also suggested that comprehensive architecture is required with at least the following components: robustness, abductive and defeasible reasoning, and formal methods. Since this is such an active field, many topics could not be covered and are mentioned in open questions and final remarks.

ACKNOWLEDGMENT

Thanks to C. Lu, T. Iqbal, Y. Ji, and M. Ma for their comments on drafts of this article. Thanks also to the insightful comments from the referees.

REFERENCES

1. Y. Choi, "The curious case of commonsense intelligence," *Daedalus*, vol. 151, no. 2, pp. 139–155, 2022.
2. E. Davis and G. Marcus, "Commonsense reasoning and commonsense knowledge in artificial intelligence," *Commun. ACM*, vol. 58, no. 9, pp. 92–103, 2015.
3. X.-W. Yang et al., "Neuro-symbolic artificial intelligence: Towards improving the reasoning abilities of large language models," in *Proc. 34th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2025, pp. 10,770–10,778.
4. Y. Zeng, A. Rajashekharan, K. Basu, H. Wang, J. Arias, and G. Gupta, "A reliable common-sense reasoning socialbot built using LLMs and goal-directed ASP," 2024, *arXiv:2407.18498*.
5. E. Whiting and D. J. Watts, "A framework for quantifying individual and collective common sense," *Proc. Natl. Acad. Sci.*, vol. 121, no. 4, 2024, Art. no. e2309535121.
6. R. Downey, *Computability and Complexity: Foundations and Tools for Pursuing Scientific Applications*. Cham, Switzerland: Springer-Verlag, 2024.
7. M. Ma, J. Gao, L. Feng, and J. Stankovic, "STLnet: Signal temporal logic enforced multivariate recurrent neural networks," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2021, pp. 14,604–14,614.
8. Z. Dai, Q. Xu, Y. Yu, and Z.-H. Zhou, "Bridging machine learning and logical reasoning by abductive learning," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, 2019, pp. 2815–2826.
9. O. Arieli, K. van Berkel, and C. Straßer, C, "Defeasible normative reasoning: A proof-theoretic integration of logical argumentation," in *Proc. 38th AAAI Conf. Artif. Intell.*, 2024, pp. 10,450–10,458, doi: 10.1609/aaai.v38i9.28913F.
10. M. Ma, J. Stankovic, E. Bartocci, and L. Feng, "Predictive monitoring with logic-calibrated uncertainty for cyber-physical systems," "*ACM Trans. Embedded Comput. Syst.*, vol. 20, no. 5s, pp. 1–25, 2021, doi: 10.1145/3477032.
11. L. Mei et al., "A survey of context engineering for large language models," 2025, *arXiv:2507.13334*.
12. A. Garcez, A. R. R. Dutra, and E. Alonzo, "Towards symbolic reinforcement learning with common sense," 2018, *arXiv:1804.08597*.