Synthesis Lectures on
Artificial Intelligence and Machine Learning

SYNTHESIS
COLLECTION OF TECHNOLOGY

Vaishak Belle

Toward Robots That Reason: Logic, Probability & Causal Laws

Springer

Synthesis Lectures on Artificial Intelligence and Machine Learning

This series publishes short books on research and development in artificial intelligence and machine learning for an audience of researchers, developers, and advanced students.

Vaishak Belle

Toward Robots That Reason: Logic, Probability & Causal Laws

Springer

Vaishak Belle
School of Informatics
University of Edinburgh
Edinburgh, UK

ISSN 1939-4608 ISSN 1939-4616 (electronic)
Synthesis Lectures on Artificial Intelligence and Machine Learning
ISBN 978-3-031-21002-0 ISBN 978-3-031-21003-7 (eBook)
https://doi.org/10.1007/978-3-031-21003-7

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Artificial Intelligence (AI) is widely acknowledged as a new kind of science that will bring about the next technological revolution. The vast majority of exciting reports that come our way about the use of AI in applications, however, are concerned with a very narrow technological capability: predicting future instances based on previously observed data. But AI, as understood by both scientists and science fiction writers, is clearly much broader.

This book is on the science of general-purpose, open-ended computational entities that deliberate and learn. Although such an agenda raises numerous philosophical and technical concerns that have no easy answers, some ideas have emerged that are attempting to tackle fundamental representational and reasoning problems for rational agents operating in complex, uncertain environments. The book builds on two such major developments in AI:

(a) the longstanding goal of integrating logic and probability for commonsense reasoning over noisy data; and
(b) theories of actions, dynamic laws and planning to achieve objectives in a changing world.

To that end, it presents the mathematical machinery for a logical language that integrates quantifiers, probabilities, actions, plans and programs.

Indeed, the unification of logic and probability has enjoyed a lot of attention in mathematics, logic, computer science and game theory. In AI, for example, areas such as statistical relational learning, neuro-symbolic systems, probabilistic databases, among others, are motivated by the need to incorporate noise and probabilistic uncertainty with logical knowledge and deductive machinery. Much of this work is limited to a *static* state of affairs, and so reasoning about a changing world via actions and plans is the next frontier. Many formalisms, moreover, limit the expressiveness for computational reasons, but this leaves open what a general account of first-order logic, probability and actions looks like. That is what this book seeks to address.

Who is this book for? Graduate students and researchers in computer science, artificial intelligence, philosophy, logic, robotics and statistics in the least would find the material useful. For instance:

- In computer science, epistemic logic has provided a formal model to capture distributed systems, multi-agent systems, privacy, cryptography and security. The integration of probability demonstrates how uncertainty could be further captured.
- In philosophy, the unification of logic and probability has been of interest from the point of epistemology and language. It will be useful to see a working proposal of such a unification for problems in AI.
- In statistics, a great amount of effort goes into capturing appropriate assumptions as well as exploiting tractable properties when modelling time and dynamics. The development of a language where no effort is needed in understanding how the dependencies between variables might change over actions is a fresh perspective on representational matters. Tractable properties are also shown to emerge as a special case when the appropriate limitations hold.
- In robotics, there is an increasing interest in autonomy and enabling commonsense reasoning along with real-time behavior. The book's focus on arbitrary actions and programmatic abstractions for partially specified behavior against a first-order knowledge base might be indicative of what is possible with very expressive languages. Clearly, that level of expressiveness does not entail real-time behavior, so a middle ground needs to be sought. But a roboticist can now revisit the expressiveness-tractability tradeoff by better articulating what is possible with representational richness.

A background in first-order logic is all that is needed to go through the contents of this book. It might even be possible to grasp the thrust of chapters without any knowledge of formal logic, but the reading of equations will require some understanding of logic. (We provide a brief introduction to first-order logic in Chap. 3).

For general audiences, Chaps. 1 and 2 motivate the scientific program. For researchers in statistical relational learning, these chapters also position the technical work in the book against popular relational languages in the machine learning community.

For readers better acquainted with modal logic, Chap. 10 might be a more comfortable read. For researchers in automated planning, Chap. 8 on belief-level regression might be especially interesting. For researchers in agent programming, Chap. 9 on programmatic abstractions for agent design under uncertainty might be worthwhile.

I hope this book motivates you to think about the many beautiful ways in which logic and probability interact.

Edinburgh, UK
September 2022

Vaishak Belle

Acknowledgments

Almost the entirety of the technical material in this book is a result of collaborations that began during my time at the University of Toronto. I owe immense intellectual debt to my co-authors and mentors: Hector Levesque, Sheila McIlraith and Gerhard Lakemeyer.

My thoughts on statistical relational learning and how it connects to the scientific program advocated in this book was significantly shaped during my time at KU Leuven. I am grateful to the energetic group of co-authors and mentors I had there. I am especially grateful to Luc De Raedt.

Many students and colleagues at the University of Edinburgh helped me sharpen my thoughts on logic and learning. I am grateful for their insights and feedback.

Over the years, many of my peers from the knowledge representation and artificial intelligence community have become close friends. I want to thank them for their encouragement and camaraderie.

The writing of this book, as well as the research carried out within would not be possible without generous funding from several institutions: DAAD (Germany), NSERC (Canada), FWO (Belgium), UKRI (UK) and the Royal Society (UK).

This book is dedicated to my wife, Sukanya, and my daughter, Bodhi, for their unending love.

Edinburgh, UK
September 2022

Vaishak Belle

Contents

About the Author

Vaishak Belle Ph.D., is a Chancellor's Fellow and Reader at the School of Informatics, University of Edinburgh (UK). He is an Alan Turing Institute Faculty Fellow, a Royal Society University Research Fellow, and a member of the Royal Society of Edinburgh's Young Academy of Scotland.

He has co-authored over 50 scientific articles on AI, and along with his co-authors, he has won several best paper awards. At the University of Edinburgh, he directs a research lab on artificial intelligence, specialising in the unification of logic and machine learning. He joined the University of Edinburgh in 2016, after postdoctoral fellowships at the University of Toronto (Canada) and KU Leuven (Belgium).

1 Introduction

> *They don't have intelligence. They have what I call "thintelligence". They see the immediate situation. They think narrowly and they call it being focused. They don't see the surround. They don't see the consequences.*
>
> —***Michael Crichton**, Jurassic Park*

Artificial Intelligence (AI) is widely acknowledged as a new kind of science that will bring about (and is already enabling) the next technological revolution. Virtually every week, exciting reports come our way about the use of AI for drug discovery, game playing, stock trading and law enforcement. And virtually all of these are mostly concerned with a very narrow technological capability, that of predicting future instances based on past instances. Identifying statistical patterns, correlations, and associations are, without doubt, extremely useful. In the first instance, they are needed in applications to inspect features and properties of interest in observed data. But AI, as understood by both scientists and science fiction writers, is clearly much broader. In fact, pattern recognition, machine learning and finding associations by mining data are closely related subfields of AI. Put differently, from first principles, what distinguishes big-data analysis from AI is that the set of capabilities we wish to enable with the latter. We are not interested in a "thintelligence", but rather a general-purpose, autonomous computational entity that, in the very least, has agency.

1.1 A Science of Agency, Deliberation and Learning

To develop a science of agency with deliberation and learning, we need to address several critical philosophical concerns, including:

- Which capabilities are of interest?
- What sort of framework allows us to capture those capabilities?

V. Belle, *Toward Robots That Reason: Logic, Probability & Causal Laws*,
Synthesis Lectures on Artificial Intelligence and Machine Learning,
https://doi.org/10.1007/978-3-031-21003-7_1

- How are we to reason about the system's uncertainty about the world, and the laws that govern it?
- Which of those capabilities and laws can be codified, using mathematical language, and how is that language defined?

It should not come as a surprise that there is no consensus yet on how such questions should be answered. To wit, consider the simple capability of reasoning about and manipulating ordinary things, as might be expected of a robotic caretaker servicing an office, for example. It might be tasked with cleaning up rooms, delivering coffee to individuals issuing such requests, and so on. For one thing, uncertainty could range from disjunctive (e.g., either-or) to existential (e.g., there is someone with a certain property) to probabilistic (e.g., one event is more likely than the other), in addition to other notions. For another, if the functionality is to be addressed in a general way, a wide range of technical concerns arise. In the very least, consider:

- Should the system's behavior be learnt entirely from data, or only partially?
- If the latter, what knowledge does a system need to have in advance (e.g., provided by a modeler) versus what can be acquired by observations?
- What kind of semantics governs the updating of a priori knowledge given new and possibly conflicting observations?
- How does the system generalize from low-level observations to high-level structured knowledge?

Technical solutions to such concerns need to be further embedded in a society, where compliance with cultural and social norms is surely demanded. To reiterate our point above more bluntly, we are yet to identify any single framework or language that is shown to be appropriate for AI systems, understood so broadly.

Be that as it may, some ideas have emerged that are attempting to tackle fundamental representational and reasoning problems for rational agents operating in complex, uncertain environments. This book builds on two such major developments in AI:

- The longstanding goal of integrating logic and probability for commonsense reasoning over noisy data.
- And, models of actions and planning to achieve objectives in a changing world.

We believe that what is needed as a key driver towards general-purpose AI, such as autonomous open-ended robots, is a framework that unifies:

$$\textit{(logic + probability) + actions}.$$

However, as is clear from our many open-ended questions above, a precise understanding of what level of expressiveness is needed for commonsensical rational agents is lacking.

This means we should strive for an application-independent, general-purpose language. A language that combines abstract, logical reasoning with probabilistic data. A language for reasoning about objects and their properties. A language that allows for reasoning about past events and hypothetical futures. A language that can express recursive plans of action. A language for refining knowledge with new but imprecise and noisy observations.

We motivate a proposal for such a unification in this book. We will begin by positioning historical developments, and then turn to the issues of big data and knowledge acquisition, which are important but orthogonal concerns for this book.

1.2 Logic Meets Probability

In the early days of AI, John McCarthy put forward a profound idea to realize artificial intelligence (AI) systems: he posited that what the system needs to know could be represented in a *formal language*, and a general-purpose algorithm would then conclude the necessary actions needed to solve the problem at hand. The main advantage is that the representation can be scrutinized and *understood by external observers*, and the system's behavior could be *improved* by making statements to it. Numerous such languages emerged in the years to follow, but first-order logic remained at the forefront as a general and powerful option. Propositional and first-order logic continue to serve as the underlying language for several areas in AI, including constraint satisfaction, automated planning, database theory, ontology specification, verification, and knowledge representation.

One of the main arguments against a logical approach is that in practice, there is pervasive uncertainty in almost every domain of interest: these can be in the form of measurement errors (e.g. readings from a thermometer), the absence of categorical assertions (e.g. smoking may be a factor for cancer, but cancer is not an absolute consequence for smokers), and the presence of numerous "latent" factors, including causes that the modeler may simply not have taken into account, all of which question the legitimacy of the model. The upshot is that on the one hand, logic was seen as an inappropriate tool, as it is "rigid" (sentences always evaluate to true or false), "brittle" (sentences in the knowledge base must be true in all possible worlds) and discrete (as opposed to the continuous error profiles for thermometers). On the other, the knowledge of the system, as posited in the declarative approach, may not only be incomplete but may be impossible to specify a priori (e.g., consider the many dimensions to telling a system on what constitutes as a face in high-resolution photographs).

Modeling uncertain worlds needed a rigorous formulation, and this came in the form of probabilistic models, including ones admitting a graphical representation, such as Bayesian networks. Such models allow one to effectively factorize the joint distribution over random variables. What makes such models particularly attractive is that both the probabilities of the variables in a given model, as well as the dependencies themselves can be learnt from data, thereby circumventing the requirement that the model needs to be provided by some omniscient modeler. Probabilistic models, obtained either by explicit specification or implicitly induced by means of modern machine learning methods such as deep learning, have supercharged the application of statistical methods in language understanding, vision and data analysis more generally.

Despite the success of probabilistic models, we observe that they are essentially propositional, but are nonetheless deployed in an inherently relational world. That is, they easily make sense of "flat" data, where atomic events are treated as independent random variables. But the environment that a robot operates has things in it: some objects may be inside another, others on top, some fragile and some heavy. We would need to reason about the properties of these things to manipulate and transport them successfully. Likewise, in medical records, it makes little sense to treat individual entries on patient symptoms as independent, since it ignores relationships between co-occurring symptoms, and family history. This encouraged the design of probabilistic concept languages, culminating in the area of statistical relational learning, neuro-symbolic AI and many other hybrid formalisms integrating probabilistic observations and high-level reasoning and/or planning. These formalisms borrow syntactic devices from finite-domain first-order logic to define complex interactions between random variables in large-scale models over classes and hierarchies.

With so many formalisms to choose from, which language shall we work with? The main thing to note is that, in often distinct ways, these languages are carefully designed to balance expressiveness versus computational efficiency for the application context at hand. However, as a result, we are left with limited languages that offer some benefits over propositional approaches but are overly restrictive for other concerns. A central problem with such probabilistic concept languages is their very controlled engagement with first-order logic: by almost exclusively considering finite-domain relational logic, succinct modelling may be admitted, but it is ultimately no more powerful than propositional logic from an expressiveness viewpoint. In some programmatic approaches, moreover, logical connectives such as disjunctions are also disallowed.

Interestingly, such probabilistic concept languages are drawn from earlier, more general, studies on unifying first-order logic and probability, such as the works of Nilsson, Bacchus and Halpern. McCarthy and Hayes, in fact, were the first to suggest the following:

(i) *It is not clear how to attach probabilities to statements containing quantifiers in a way that corresponds to the amount of conviction people have.*
(ii) *The information necessary to assign numerical probabilities is not ordinarily available. Therefore, a formalism that required numerical probabilities would be epistemologically inadequate.*

Translating such sentiments to a desiderata of sorts, one might say a general-purpose language should support full first-order logic, but also allow probabilistic assertions. In other words, it should allow a purely probabilistic specification, if the application demands it and the information available allows it. Analogously, such a language should allow a purely logical specification, if no probabilistic information is available. And, of course, everything in between: for example, it should be possible to have an initial database consisting of only first-order formulas, then gradually add purely probabilistic formulas, and obtain appropriate conclusions from that resulting database. Moreover, from such a general language, one may then determine which fragment is sufficient for the application at hand, and constrain the language accordingly. This is an important advantage with rich languages.

Naturally, the downside of working with such a powerful language is that we will not be able to say very much about efficient computation is every instance. With a specific fragment in mind, that is possible. But not in general. Since the game here is to really understand the principles and theory behind integrating logic and probability, we will accept the matter, and consider concrete computational strategies at a later stage.

Not surprisingly, we will be in a very similar position with regards to reasoning about world dynamics. We will aim for a general language, on the one hand, and focus on computation only with appropriate fragments.

1.3 Actions

Reasoning about events, actions, plans and programs has a long history in computer science and AI. Similar to the many proposals in the literature for commonsense reasoning, we have plenty of formalisms to choose from for capturing actions. Formalisms such as temporal logics allow us to reason about time, including the positioning of properties in the current and future states (e.g., the variable will never go above the value of 100). Markov process allows the stochastic modelling of sequential events. When coupled with a reward function, they can be used to compute the sequence of actions to be taken by an agent to maximize the overall reward. Planning languages such as STRIPS describe the current state of the system as a database, and by means of a synthesis algorithm, a sequence of actions can be produced that changes the state to a desired database.

Similar to the observation we just made about probabilistic logical representation languages, there are very many models of actions, with some limitation on what kinds of things can be expressed. As scientists consider more challenging applications, a new feature would be considered desirable to add, and inevitably a new modelling language would be introduced, with a corresponding semantics. Of course, there is no way we can completely future-proof a language against all possible desirable features. But at least we can consider a language that is powerful enough to reason about features such as:

- Causal laws relating actions and effects.
- Internal actions that can be performed by the robot to the change the world state, sensing actions that do not change the world state but only what the robot knows, and exogenous actions that affect the world but are performed without the robot's control (and possibly without its knowledge).
- Reasoning about the past, hypothetical and counterfactual events.

- Reasoning about the beliefs, desires and intentions of all the agents in the environment.
- Reasoning about discrete and continuous, noise-free and noisy actions and sensors.
- Expressing atomic actions, sequential plans, recursive plans, program-like plans, and partial instructions for the robot to execute.

What are after, then, is a unifying "theory of dynamics", and again, we turn to first-order logic but now extended for actions. In other words, it would be ideal if first-order logic provided the substrate that allows us to reason about both probabilities and actions, which would then count as a general proposal in line with our aims. As we shall see, such a possibility does exist, and is a fairly simple extension to the one of most popular knowledge representation languages: the situation calculus. Originally postulated by McCarthy, and later revised by Reiter, it has enjoyed considerable attention as an important knowledge representation language with extensions for time, plans, programs, inductive definitions, abstraction, rewards and high-level control. Basically, initial knowledge is a standard (unrestricted) first-order theory, over which we define actions and effects. Actions result in some formulas in the theory changing values, depending on which predicates are affected by an action. The key feature, like temporal and dynamic logics, however, and unlike dynamic Bayesian networks and planning formalisms, is that the underlying language allows us to reason about arbitrary trajectories of actions. So, one can reason about the past and the future.

So, the situation calculus has all the expressiveness of standard first-order logic together with a theory of actions. All of that is studied comprehensively in the introductory book by Raymond Reiter. What this book is about is further extending that framework for reasoning about probabilities in a general way.

1.4 Some Related Areas

Before going further, it might be useful to position some existing, well-established areas in the context of our discussion on first-order probabilistic languages. They help motivate the kind of generality we are aiming for.

- **Classical databases.** Databases are defined over a relational logical language and a finite set of constants, the latter denoting the individuals and values in the database. A database is equivalent to a finite and consistent set of atoms, with the understanding that all atoms not mentioned in that set are false.
  Transactional databases allow for the execution of *commands,* which amount of adding new atoms to the set, and deleting others.
  For example, a university database might consist of all of the enrolled students in the current year, along with their phone numbers and details of the courses they are undertaking. Matriculating a new student would mean the addition of this student to the set of

enrolled students, and adding the relevant personal and course details. When a student graduates, they would be removed from the set of enrolled students.

- **Incomplete databases.** Uncertainty about the truth of atoms might mean we entertain disjunctive knowledge. For example, while scanning a handwritten text, we might be unsure about the first name of a student with ID #243: it could either be *Mary* or *May*. Each alternative together with the remaining facts correspond to a finite and consistent set of atoms, so the representation is allowing for multiple possible worlds.
- **Probabilistic databases.** Perhaps the student with ID #243 had to submit multiple forms, and although the handwritten text is problematic to scan in all of them, it might seem more probable that it is *Mary* rather than *May*. Probabilistic databases allow probabilities on atoms, and more generally on possible worlds.
- **Probabilistic relational languages.** To accord probabilities to both atoms and formulas, formalisms such as Markov logic networks and relational Bayesian networks have emerged. These can be seen to extend standard probabilistic formalisms such as Markov and Bayesian networks with a relational syntax allowing for an easier way to define intricate probabilistic models involving entities and their relationships. Alternatively, probabilistic logic programming allow modelers to decorate Horn rules with probabilities, to similar effect.

  For example, when examining electronic health records, it is useful to learn predictive models that can natively understand the relationships between patients, diagnoses, prescribed medications, family history and the progression of symptoms over months and years. Here, a probabilistic relational model can be built to leverage such relationships.

  Actions can be further defined for such models: for example, given probabilistic variables for the positions of objects, noisy actions that move these objects to other locations would affect the distributions of those variables. The distribution would need to account for the error profile of the actions to capture the probabilistic nature of the current position.
- **Automated planning.** Perhaps the simplest model for automated planning can be defined using classical databases. Such a database could denote the initial state, and actions such as moving from one location to another, picking up and dropping objects, and so on, can be seen as transactions over such databases amounting to the addition and deletion of facts.

  A more involved planning language might allow for uncertainty over the initial state (using multiple possible worlds), as well as involved actions that are *context-dependent*. For example, if the floor is slippery, a move action may cause the robot to fall rather than just move ahead. Probabilities can be further accorded to the possible worlds as well as the unintended effects of actions, thereby necessitating the need for dynamic probabilistic relational languages.

  Usually a solution to an automated planning problem is simply a sequence of prescribed actions that transform the initial state to a desired goal state. However, when there is uncertainty about the initial state, or about the type of observations that the agent might encounter, solutions can be iterative or even recursive, resembling a program with if-then-else and while loops.