

Statistical Relational Artificial Intelligence

Logic, Probability, and Computation

Synthesis Lectures on Artificial Intelligence and Machine Learning

Editors

Ronald J. Brachman, *Yahoo! Labs*

William W. Cohen, *Carnegie Mellon University*

Peter Stone, *University of Texas at Austin*

Statistical Relational Artificial Intelligence: Logic, Probability, and Computation

Luc De Raedt, Kristian Kersting, Sriraam Natarajan, and David Poole

2016

Representing and Reasoning with Qualitative Preferences: Tools and Applications

Ganesh Ram Santhanam, Samik Basu, and Vasant Honavar

2016

Metric Learning

Aurélien Bellet, Amaury Habrard, and Marc Sebban

2015

Graph-Based Semi-Supervised Learning

Amarnag Subramanya and Partha Pratim Talukdar

2014

Robot Learning from Human Teachers

Sonia Chernova and Andrea L. Thomaz

2014

General Game Playing

Michael Genesereth and Michael Thielscher

2014

Judgment Aggregation: A Primer

Davide Grossi and Gabriella Pigozzi

2014

[An Introduction to Constraint-Based Temporal Reasoning](#)

Roman Barták, Robert A. Morris, and K. Brent Venable

2014

[Reasoning with Probabilistic and Deterministic Graphical Models: Exact Algorithms](#)

Rina Dechter

2013

[Introduction to Intelligent Systems in Traffic and Transportation](#)

Ana L.C. Bazzan and Franziska Klügl

2013

[A Concise Introduction to Models and Methods for Automated Planning](#)

Hector Geffner and Blai Bonet

2013

[Essential Principles for Autonomous Robotics](#)

Henry Hexmoor

2013

[Case-Based Reasoning: A Concise Introduction](#)

Beatriz López

2013

[Answer Set Solving in Practice](#)

Martin Gebser, Roland Kaminski, Benjamin Kaufmann, and Torsten Schaub

2012

[Planning with Markov Decision Processes: An AI Perspective](#)

Mausam and Andrey Kolobov

2012

[Active Learning](#)

Burr Settles

2012

[Computational Aspects of Cooperative Game Theory](#)

Georgios Chalkiadakis, Edith Elkind, and Michael Wooldridge

2011

[Representations and Techniques for 3D Object Recognition and Scene Interpretation](#)

Derek Hoiem and Silvio Savarese

2011

[A Short Introduction to Preferences: Between Artificial Intelligence and Social Choice](#)

Francesca Rossi, Kristen Brent Venable, and Toby Walsh

2011

Human Computation

Edith Law and Luis von Ahn
2011

Trading Agents

Michael P. Wellman
2011

Visual Object Recognition

Kristen Grauman and Bastian Leibe
2011

Learning with Support Vector Machines

Colin Campbell and Yiming Ying
2011

Algorithms for Reinforcement Learning

Csaba Szepesvári
2010

Data Integration: The Relational Logic Approach

Michael Genesereth
2010

Markov Logic: An Interface Layer for Artificial Intelligence

Pedro Domingos and Daniel Lowd
2009

Introduction to Semi-Supervised Learning

Xiaojin Zhu and Andrew B. Goldberg
2009

Action Programming Languages

Michael Thielscher
2008

Representation Discovery using Harmonic Analysis

Sridhar Mahadevan
2008

Essentials of Game Theory: A Concise Multidisciplinary Introduction

Kevin Leyton-Brown and Yoav Shoham
2008

A Concise Introduction to Multiagent Systems and Distributed Artificial Intelligence

Nikos Vlassis
2007

Intelligent Autonomous Robotics: A Robot Soccer Case Study
Peter Stone
2007

Copyright © 2016 by Morgan & Claypool

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopy, recording, or any other except for brief quotations in printed reviews, without the prior permission of the publisher.

Statistical Relational Artificial Intelligence: Logic, Probability, and Computation

Luc De Raedt, Kristian Kersting, Sriraam Natarajan, and David Poole

www.morganclaypool.com

ISBN: 9781627058414 paperback

ISBN: 9781627058421 ebook

DOI 10.2200/S00692ED1V01Y201601AIM032

A Publication in the Morgan & Claypool Publishers series

SYNTHESIS LECTURES ON ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Lecture #32

Series Editors: Ronald J. Brachman, *Yahoo! Labs*

William W. Cohen, *Carnegie Mellon University*

Peter Stone, *University of Texas at Austin*

Series ISSN

Print 1939-4608 Electronic 1939-4616

Statistical Relational Artificial Intelligence

Logic, Probability, and Computation

Luc De Raedt
KU Leuven, Belgium

Kristian Kersting
Technical University of Dortmund, Germany

Sriraam Natarajan
Indiana University

David Poole
University of British Columbia

*SYNTHESIS LECTURES ON ARTIFICIAL INTELLIGENCE AND MACHINE
LEARNING #32*



MORGAN & CLAYPOOL PUBLISHERS

ABSTRACT

An intelligent agent interacting with the real world will encounter individual people, courses, test results, drugs prescriptions, chairs, boxes, etc., and needs to reason about properties of these individuals and relations among them as well as cope with uncertainty.

Uncertainty has been studied in probability theory and graphical models, and relations have been studied in logic, in particular in the predicate calculus and its extensions. This book examines the foundations of combining logic and probability into what are called relational probabilistic models. It introduces representations, inference, and learning techniques for probability, logic, and their combinations.

The book focuses on two representations in detail: Markov logic networks, a relational extension of undirected graphical models and weighted first-order predicate calculus formula, and Problog, a probabilistic extension of logic programs that can also be viewed as a Turing-complete relational extension of Bayesian networks.

KEYWORDS

probabilistic logic models, relational probabilistic models, lifted inference, statistical relational learning, probabilistic programming, inductive logic programming, logic programming, machine learning, Prolog, Problog, Markov logic networks

Contents

	Preface	xiii
1	Motivation	1
	1.1 Uncertainty in Complex Worlds	1
	1.2 Challenges of Understanding StarAI	3
	1.3 The Benefits of Mastering StarAI	5
	1.4 Applications of StarAI	5
	1.5 Brief Historical Overview	12
	 PART I Representations	 15
2	Statistical and Relational AI Representations	17
	2.1 Probabilistic Graphical Models	17
	2.1.1 Bayesian Networks	18
	2.1.2 Markov Networks and Factor Graphs	20
	2.2 First-Order Logic and Logic Programming	22
3	Relational Probabilistic Representations	27
	3.1 A General View: Parameterized Probabilistic Models	28
	3.2 Two Example Representations: Markov Logic And ProbLog	34
	3.2.1 Undirected Relational Model: Markov Logic	35
	3.2.2 Directed Relational Models: ProbLog	37
4	Representational Issues	45
	4.1 Knowledge Representation Formalisms	45
	4.2 Objectives for Representation Language	46
	4.3 Directed vs. Undirected models	48
	4.4 First-Order Logic vs. Logic Programs	50
	4.5 Factors and Formulae	51
	4.6 Parameterizing Atoms	52

4.7	Aggregators and Combining Rules	54
4.8	Open Universe Models	58
4.8.1	Identity Uncertainty	58
4.8.2	Existence Uncertainty	60
4.8.3	Ontologies	62

PART II Inference 63

5	Inference in Propositional Models	65
5.1	Probabilistic Inference	65
5.1.1	Variable Elimination	66
5.1.2	Recursive Conditioning	66
5.1.3	Belief Propagation	68
5.2	Logical Inference	69
5.2.1	Propositional Logic, Satisfiability, and Weighted Model Counting	69
5.2.2	Semiring Inference	70
5.2.3	The Least Herbrand Model	72
5.2.4	Grounding	74
5.2.5	Proving	75
6	Inference in Relational Probabilistic Models	77
6.1	Grounded Inference for Relational Probabilistic Models	77
6.1.1	Weighted Model Counting	77
6.1.2	WMC for Markov Logic	77
6.1.3	WMC for ProbLog	78
6.1.4	Knowledge Compilation	79
6.2	Lifted Inference: Exploiting Symmetries	80
6.2.1	Exact Lifted Inference	83
6.3	(Lifted) Approximate Inference	87

	PART III Learning	91
7	Learning Probabilistic and Logical Models	93
7.1	Learning Probabilistic Models	93
7.1.1	Fully Observed Data and Known Structure	94
7.1.2	Partially Observed Data with Known Structure	95
7.1.3	Unknown Structure and Parameters	95
7.2	Logical and Relational Learning	96
7.2.1	Two Learning Settings	97
7.2.2	The Search Space	98
7.2.3	Two Algorithms: Clausal Discovery and FOIL	99
7.2.4	From Propositional to First-Order Logic	101
7.2.5	An ILP Example	102
8	Learning Probabilistic Relational Models	105
8.1	Learning as Inference	105
8.2	The Learning Problem	106
8.2.1	The Data Used	106
8.3	Parameter Learning of Relational Models	108
8.3.1	Fully Observable Data	108
8.3.2	Partially Observed Data	109
8.3.3	Learning with Latent Variables	111
8.4	Structure Learning of Probabilistic Relational Models	112
8.4.1	A Vanilla Structure Learning Approach	113
8.4.2	Probabilistic Relational Models	114
8.4.3	Boosting	117
8.5	Bayesian Learning	119
	PART IV Beyond Probabilities	121
9	Beyond Basic Probabilistic Inference and Learning	123
9.1	Lifted Satisfiability	123
9.2	Acting in Noisy Relational Worlds	124
9.3	Relational Optimization	129
10	Conclusions	135

Bibliography	139
Authors' Biographies	169
Index	173

Preface

This book aims to provide an introduction that can help newcomers to the field to get started, to understand the state-of-the-art and the current challenges and be ready for future advances. It reviews the foundations of StarAI, motivates the issues, justifies some choices that have been made, and provides some open problems. Laying bare the foundations will hopefully inspire others to join us in exploring the frontiers and the yet unexplored areas.

The target audience for this book consists of advanced undergraduate and graduate students and also researchers and practitioners who want to get an overview of the basics and the state-of-the-art in StarAI. To this aim, Part I starts with providing the necessary background in probability and logic. We then discuss the representations of relational probability models and the underlying issues. Afterward, we focus first on inference, in Part II, and then on learning, in Part III. Finally, we touch upon relational tasks that go beyond the basic probabilistic inference and learning tasks as well as some open issues.

Researchers who are already working on StarAI—we apologize to anyone whose work we are accidentally not citing—may enjoy reading about parts of StarAI they are less familiar with.

We are grateful to all the people who contributed to the development of statistical relational learning and statistical relational AI. This book is made possible by you.

We also thank the reviewers for their valuable feedback and our co-authors, who accompanied us on our StarAI adventures, such as Laura Antanas, Udi Apsel, Babak Ahmadi, Hendrik Blockeel, Wolfram Burgard, Maurice Bruynooghe, David Buchman, Hung H. Bui, Peter Carbonetto, Alexandru Cocora, Fabrizio Costa, Michael Chiang, Walter Daelemans, Jesse Davis, Nando de Freitas, Kurt De Grave, Tinne De Laet, Bart Demoen, Kurt Driessens, Saso Dzeroski, Thomas G. Dietterich, Adam Edwards, Alan Fern, Daan Fierens, Paolo Frasconi, Roman Garnett, Amir Globerson, Bernd Gutmann, Martin Grohe, Fabian Hadiji, McElory Hoffmann, Manfred Jaeger, Gerda Janssens, Thorsten Joachims, Saket Joshi, Leslie Kaelbling, Andreas Karwath, Arzoo Katiyar, Seyed M. Kazemi, Angelika Kimmig, Jacek Kisynski, Tushar Khot, Stefan Kramer, Gautam Kunapuli, Chia-Li Kuo, Tobias Lang, Niels Landwehr, Daniel Lowd, Catherine A. McCarty, Theofrastos Mantadelis, Wannes Meert, Brian Milch, Martin Mladenov, Bogdan Moldovan, Roser Morante, Plinio Moreno, Marion Neumann, Davide Nitti, Phillip Odom, Jose Oramas, David Page, Andrea Passerini, Rui Pimentel de Figueiredo, Christian Plagemann, Tapani Raiko, Christopher Re, Kate Revoredo, Achim Rettinger, Ricardo Rocha, Scott Sanner, Vitor Santos Costa, Jose Santos-Victor, Erkal Selman, Rita Sharma, Jude W. Shavlik, Prasad Tadepalli, Nima Taghipour, Ingo Thon, Hannu Toivonen, Pavel Tokmakov, Sunna Torge, Marc Toussaint, Volker Tresp, Tinne Tuytelaars, Vincent Van Asch, Guy Van den Broeck, Martijn van Otterlo, Joost Vennekens, Jonas Vlasselaer, Zhao Xu, Shuo Yang, and Luke Zettlemoyer.

Thanks for all the encouragement and fun! Thanks to the StaRAI lab at Indiana for proofreading the book.

Last but not least, we also thank our families and friends for their patience and support. Thanks!

LDR and KK thank the European Commission for support of the project FP7-248258-First-MM. KK further thanks Fraunhofer Society, ATTRACT Fellowship "STREAM", the German Science Foundation, DFG KE 1686/2-1, as part of the DFG Priority Programme 1527, and the German-Israeli Foundation for Scientific Research and Development, GIF 1180/2011. SN thanks Army Research Office (ARO) grant number W911NF-13-1-0432 under the Young Investigator Program and the National Science Foundation grant no. IIS-1343940. LDR thanks the Research Foundation Flanders, and the KULeuven BOF fund for their support. DP thanks the Natural Sciences and Engineering Research Council of Canada (NSERC) for ongoing support.

Luc De Raedt, Leuven, Belgium
Kristian Kersting, Dortmund, Germany
Sriram Natarajan, Bloomington, USA
David Poole, Vancouver, Canada
February 2016

CHAPTER 1

Motivation

There are good arguments that an intelligent agent that makes decisions about how to act in a complex world needs to model its uncertainty; it cannot just act pretending that it knows what is true. An agent also needs to reason about individuals (objects, entities, things) and about relations among the individuals.

These aspects have often been studied separately, with models for uncertainty often defined in terms of features and random variables, ignoring relational structure, and with rich (logical) languages for reasoning about relations that ignore uncertainty. This book studies the integration of the approaches to reasoning about uncertainty and reasoning about individuals and relations.

1.1 UNCERTAINTY IN COMPLEX WORLDS

Over the last 30 years, **Artificial Intelligence** (AI) has evolved from being skeptical, even hostile, to the use of probability to embracing probability. Initially, many researchers were skeptical about **statistical AI** because probability seemed to rely on too many numbers and did not deal with the complexities of a world of individuals and things. But the use of probabilistic graphical models, exploiting probabilistic independencies, has revolutionized AI. The independencies specified in such models are natural, provide structure that enables efficient reasoning and learning, and allow one to model complex domains. Many AI problems arising in a wide variety of fields such as machine learning, diagnosis, network communication, computer vision, and robotics have been elegantly encoded and solved using probabilistic graphical models.

Meanwhile, there have also been considerable advances in **logical AI**, where agents reason about the structure of complex worlds. One aspect of this is in the semantic web and the use of ontologies to represent meaning in diverse fields from medicine to geology to the products in a catalogue. Generally, there is an explosive growth in the amount of heterogeneous data that is being collected in the business and scientific world. Example domains include biology and chemistry, transportation systems, communication networks, social networks, and robotics. Like people, intelligent agents should be able to deal with many different types of knowledge, requiring structured representations that give a more informative view of the AI task at hand.

Moreover, reasoning about individuals and relations is all about reasoning with regularities and symmetries. We lump individuals into categories or classes (such as “person” or “course”) because the individuals in a category share common properties—e.g., there are statements that are true about all living people such as they breathe, they have skin and two biological parents. Similarly for relations, there is something in common between Sam being advised by Professor

2 1. MOTIVATION

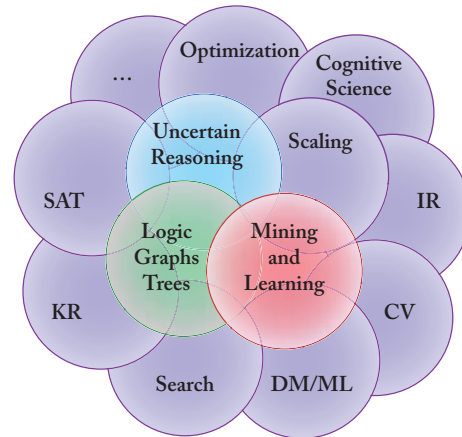


Figure 1.1: Statistical Relational Artificial Intelligence (StarAI) combines probability, logic, and learning and covers major parts of the AI spectrum.

Smith and Chris being advised by Professor Johnson; there are statements about publishing papers, working on a thesis and projects that are common among the “advised by” relationships. We would like to make predictions about two people about whom all we know may be only their advisory relationships. It is these commonalities and regularities that enable language to describe the world. Reasoning about regularities and symmetries is the foundation of logics built on the predicate calculus, which allows statements about all individuals.

Thus, to deal with the real world we actually need to exploit uncertainty, independencies, and symmetries and tackle a long standing goal of AI, namely unifying first-order logic—capturing regularities and symmetries—and probability—capturing uncertainty and independence. Predicate logic and probability theory are not in conflict with each other, they are synergistic. Both extend propositional logic, one by adding relations, individuals, and quantified variables, the other by allowing for measures over possible worlds and conditional queries. This may explain why there has been a considerable body of research in combining both of them over the last 25 years, evolving into what has come to be called **Statistical Relational Artificial Intelligence (StarAI)**; see also Fig. 1.1:

the study and design of intelligent agents that act in worlds composed of individuals (objects, things), where there can be complex relations among the individuals, where the agents can be uncertain about what properties individuals have, what relations are true, what individuals exist, whether different terms denote the same individual, and the dynamics of the world.

The basic building block of StarAI are relational probabilistic models—we use this term in the broad sense, meaning any models that combine relations and probabilities. They can be seen

as combinations of probability and predicate calculus that allow for individuals and relations as well as probabilities. In building on top of relational models, StarAI goes far beyond reasoning, optimization, learning, and acting optimally in terms of a fixed number of features or variables, as it is typically studied in machine learning, constraint satisfaction, probabilistic reasoning, and other areas of AI. In doing so, StarAI has the potential to make AI more robust and efficient.

This book aims to provide an introduction that can help newcomers to the field get started, understand the state-of-the-art and the current challenges, and be ready for future advances. It reviews the foundations of StarAI, motivates the issues, justifies some choices that have been made, and provides some open problems. Laying bare the foundations will hopefully inspire others to join us in exploring the frontiers and the yet unexplored areas.

The target audience for this book consists of advanced undergraduate and graduate student as well as researchers and practitioners who want to get an overview of the basics and the state-of-the-art in StarAI. To this aim, Part I starts with providing the necessary background in probability and logic. We then discuss the representations of relational probability models and the underlying issues. Afterward, we focus first on inference, in Part II, and then on learning, in Part III. Finally, we touch upon relational tasks that go beyond the basic probabilistic inference and learning tasks as well as some open issues.

Researchers who are already working on StarAI—we apologize to anyone whose work we are accidentally not citing—may enjoy reading about parts of StarAI with which they are less familiar.

1.2 CHALLENGES OF UNDERSTANDING STARAI

Since StarAI draws upon ideas developed within many different fields, it can be quite challenging for newcomers to get started.

One of the challenges of building on top of multiple traditions is that they often use the same vocabulary to mean different things. Common terms such as “variable,” “domain,” “object,” “relation,” and “parameter” have come to have accepted meanings in mathematics, computing, statistics, logic, and probability, but their meanings in each of these areas is different enough to cause confusion. We will be clear about the meaning of these when using them. For instance, we follow the logic tradition and use the term “individuals” for things. They are also called “objects,” but that terminology is often confusing to people who have been brought up with object-oriented programming, where objects are data structures and associated methods. For example, a person individual is a real person and not a data structure that encapsulates information about a person. A computer is not uncertain about its own data structures, but can be uncertain about what exists and what is true in the world.

Another confusion stems from the term “relational.” Most existing datasets are, or can be, stored in relational databases. Existing machine learning techniques typically learn from datasets stored in relational datasets where the values are Boolean, discrete, ordinal, or continuous. However, in many datasets the values are the names of individuals, as in the following example.

4 1. MOTIVATION

<i>student</i>	<i>course</i>	<i>grade</i>
s_1	c_1	a
s_2	c_1	c
s_1	c_2	b
s_2	c_3	b
s_3	c_2	b
s_4	c_3	b
s_3	c_4	?
s_4	c_4	?

Figure 1.2: An example dataset that is not amenable to traditional classification.

Example 1.1 Consider learning from the dataset in Fig. 1.2. The values of the Student and the Course attributes are the names of the students (s_1, s_2, s_3 and s_4) and the courses (c_1, c_2, c_3 and c_4). The value of the grade here is an ordinal (a is better than b which is better than c). Assume that the task is to predict the grade of students on courses, for example predicting the grade of students s_3 and s_4 on course c_4 . There is no information about course c_4 , and students s_3 and s_4 have the same average (they both have one “ b ”); however, it is still possible to predict that one will do better than the other in course c_4 . This can be done by learning how difficult each course is, and how smart each student is, given the data about students, the courses they take, and the grades they obtain. For example, we may learn that s_1 is intelligent, s_2 is not as intelligent, course c_2 is difficult and course c_3 is not difficult, etc. This model then allows for the prediction that s_3 will do better than s_4 in course c_4 .

Standard textbook supervised learning algorithms that learn, e.g., a decision tree, a neural network, or a support vector machine (SVM) to predict *grade* are not appropriate; they can handle ordinals, but cannot handle the names of students and courses. It is the relationship among the individuals that provides the generalizations from which to learn. Traditional classifiers are unable to take into account such relations. This also holds for learning standard graphical models, such as Bayesian networks. These approaches make what can be seen as a single-table single-row assumption, which requires that each instance is described in a single row by a fixed set of features and all instances are independent of one another (given the model). This clearly does not hold in this dataset as the information about student s_1 is spread over multiple rows, and that about course c_1 as well. Furthermore, tests on $student = s_1$ or $course = c_3$ would be meaningless if we want to learn a model that generalizes to new students.

StarAI approaches take into account the relationships among the individuals as well as deal with uncertainty.

1.3 THE BENEFITS OF MASTERING STARAI

The benefits of combining logical abstraction and relations with probability and statistics are manifold.

- When learning a model from data, relational and logical abstraction allows one to reuse experience in that *learning about one entity improves the prediction for other entities*. This can generalize to objects that have never been observed before.
- Logical variables, which are placeholders for individuals, allow one to make abstractions that apply to all individuals that have some common properties.
- By using logical variables and unification, one can specify and reason about regularities across different situations using rules and templates rather than having to specify them for each single entity separately.
- The employed and/or learned knowledge is often declarative and compact, which potentially makes it easier for people to understand and validate.
- In many applications, background knowledge about the domain can be represented in terms of probability and/or logic. Background knowledge may improve the quality of learning: the logical aspects may focus the search on the relevant patterns, thus restricting the search space, while the probabilistic components may provide prior knowledge that can help avoid overfitting.

Relational and logical abstraction have the potential to make statistical AI more robust and efficient. Incorporating uncertainty makes relational models more suitable for reasoning about the complexity of the real world. This has been witnessed by a number of real-world applications.

1.4 APPLICATIONS OF STARAI

StarAI has been successfully applied to problems in citation analysis, web mining, natural language processing, robotics, medicine bio- and chemo-informatics, electronic games, and activity recognition, among others. Let us illustrate using a few examples.

Example 1.2 Mining Electronic Health Records (EHRs) As of today, EHRs hold over 50 years of recorded patient information and, with increased adoption and high levels of population coverage, are becoming the focus of public health analyses. Mining EHR data can lead to improved predictions and better disease characterization. For instance, Coronary Heart Disease (CHD) is a major cause of death worldwide. In the U.S., CHD is responsible for approximated 1 in every 6 deaths with a coronary event occurring every 25 s and about 1 death every minute based on data current to 2007. Although a multitude of cardiovascular risks factors have been identified, CHD actually reflects complex interactions of these factors over time. Thus, early detection

6 1. MOTIVATION

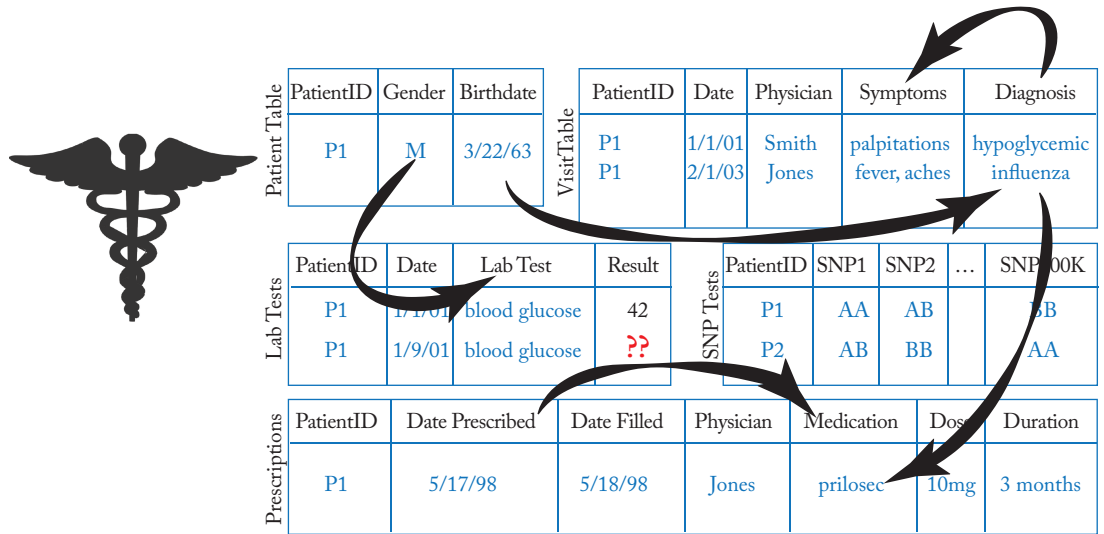


Figure 1.3: Electronic Health Records (EHRs) are relational databases capturing noisy and missing information with probabilistic dependencies (the black arrows) within and across tables.

of risks will help in designing effective treatments targeted at youth in order to prevent cardiovascular events in adulthood and to dramatically reduce the costs associated with cardiovascular diseases.

Doing so, however, calls for StarAI. As illustrated in Fig. 1.3, EHR data consists of several diverse features (e.g., demographics, psychosocial, family history, dietary habits) that interact with each other in many complex ways making it relational. Moreover, like most data sets from biomedical applications, EHR data contains missing values, i.e., all data are not collected for all individuals. And, EHR data is often collected as part of a longitudinal study, i.e., over many different time periods such as 0, 5, 10 years, etc., making it temporal. Natarajan et al. [2013] demonstrated that StarAI can uncover complex interactions of risk factors from EHRs. The learned probabilistic relational model performed significantly better than traditional non-relational approaches and conforms to some known or hypothesized medical facts. For instance, it is believed that females are less prone to cardiovascular issues than males. The relational model identifies sex as the most important feature. Similarly, in men, it is believed that the low- and high-density lipoprotein cholesterol levels are very predictive, and the relational model confirms this. For instance, the risk interacts with a (low-density lipoprotein) cholesterol level in year 0 (of



Figure 1.4: Populating a knowledge base with probabilistic facts (or assertions) extracted from dark data (e.g., text, audio, video, tables, diagrams, etc.) and background knowledge.

the study) for a middle-aged male in year 7, which can result in a relational conditional probability

$$0.79 = P(\text{risk}(\text{Person}) \mid \text{sex}(\text{Person}, \text{male}) \wedge (35 \leq \text{age}(\text{Person}, \text{year } 7) < 45) \wedge \neg(0 \leq \text{ldl}(\text{Person}, \text{year } 0) < 100)) .$$

The model also identifies complex interaction between risk factors at different years of the longitudinal study. For instance, smoking in year 5 interacts with cholesterol level in later years in the case of females, and the triglyceride level in year 5 interacts with the cholesterol level in year 7 for males. Finally, using data such as the age of the children, whether the patients owns or rents a home, their employment status, salary range, their food habits, their smoking and alcohol history, etc., revealed striking socio-economic impacts on the health state of the population.

Example 1.3 Extracting value from dark data Many companies' databases include natural language comments buried in tables and spreadsheets. Similarly, there are often tables and figures embedded in documents and web pages. Like dark matter, dark data is this great mass of data buried in text, tables, figures, and images, which lacks structure and so is essentially unprocessable by traditional methods. StarAI helps bring dark data to light (see, e.g., [Niu et al., 2012, Venugopal et al., 2014]), making the knowledge base construction, as illustrated in Fig. 1.4 feasible. The resulting relational probabilistic models are richly structured with many different entity types with complex interactions.

To carry out such a task, one starts with transforming the dark data such as text documents into relational data. For example, one may employ standard NLP tools such as logistic regression, conditional random fields and dependency parsers to decode the structure (e.g., part-of-speech tags and parse trees) of the text or run some pattern matching techniques to identify candidate

8 1. MOTIVATION

entity mentions and then store them in a database. Then, every tuple in the database or result of an database query is a random variable in a relational probabilistic model. The next step is to determine which of the individuals (entities) are the same as each other and same as the entities that are already known about. For instance, a mention of an entity may correspond to multiple candidate entities known from some other database such as Freebase. To determine which entity is correct, one may use the heuristic that “if the string of a mention is identical to the canonical name of an entity, then this mention is likely to refer to this entity.” In the relational probabilistic model, this may read as the quantified conditional probability statement:

$$\forall Entity, \forall Mention P(entityMentioned(Entity, Mention) \mid \exists String mentionText(Mention, String) \wedge entityName(Entity, String)) = 0.8 .$$

Given such conditional probabilities, the marginal probability of every tuple in the database as well as all derived tuples such as *entityMentioned* can be inferred.

We can also model the relationship between learned features using rich linguistic features for trigger and argument detection and type labeling using weighted logical formulae (defined in Chapter 3) such as:

$$0.73 : word(Sentence, PosInSentence, Word) \wedge \\ triggerType(Sentence, PosInSentence, Type) \wedge \\ DependencyType(Sentence, PosInDependencyTree, PosInSentence, DepType) \wedge \\ ArgumentRole(Sentence, PosInDependencyTree, PosInSentence, ArgRole) .$$

For each sentence, this assigns probability to the join of a word at some position in a sentence and the dependency type label that connects the word token to the argument token in the dependency parse tree with the trigger types and argument types of the two tokens. Here, *triggerType* denotes the prediction from a pre-trained support vector machine for triggering some information extracted from the text. This way, high-dimensional, sophisticated features become available to the relational probabilistic model. Moreover, as with Google’s Knowledge Vault [Dong et al., 2014], the parameters of the relational probabilistic model or even (parts) of its structure can be learned from data.

Example 1.4 Language modeling The aim of language modeling is to estimate a distribution over words that best represents the text of a corpus. This is central to speech recognition, machine translation, and text generation, among others, and the parameters of language models are commonly used as features or as initialization for other natural language processing approaches. Examples include the word distributions learned by probabilistic topic models, or the word embeddings learned through neural language models. In practice, however, the size of the vocabulary traditionally limited the distributions applicable for this task: specifically, one has to either resort to local optimization methods, such as those used in neural language models, or work with heavily constrained distributions. Jernite et al. [2015] overcame these limitations. They model the entire

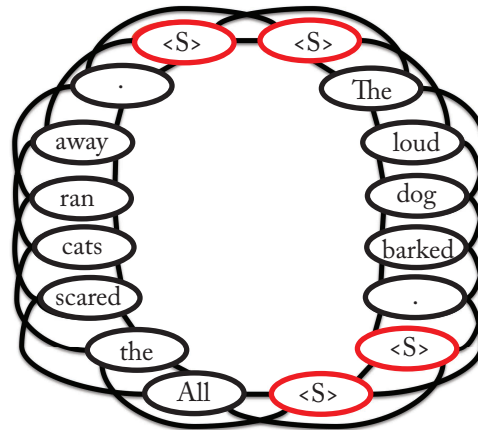


Figure 1.5: Instance of the cyclic language model [Jernite et al., 2015] with added start and end $\langle S \rangle$ tokens based on two sentences. Each word of a sentence is a random variable and probabilistically depends on the words being at most two steps away. The model exhibits symmetries that are exploitable during parameter estimation using lifted probabilistic inference.

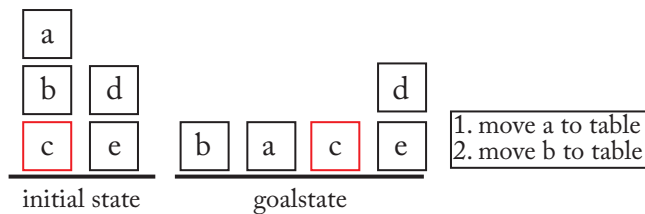


Figure 1.6: A Blocks World planning problem and a plan to solve the goal to have c clear.

corpus as an undirected graphical model, whose structure is illustrated in Fig. 1.5. Because of parameter sharing in the model, each of the random variables is indistinguishable and by exploiting this symmetry, Jernite et al. derived an efficient approximation of the partition function using lifted variational inference with complexity independent of the length of the corpus.

Example 1.5 Planning under uncertainty The Blocks World, see Fig. 1.6, consists of a number of blocks stacked into towers on a table large enough to hold them all. The positioning of the towers on the table is irrelevant. The planning problem is to turn an initial state of the blocks into a goal state, by moving one block at a time from the top of a tower onto another tower or to the table. However, the effects of moving a block may not be perfectly predictable. For example,

10 1. MOTIVATION

the gripper of a robot may be slippery so moving a block may not succeed. This uncertainty compounds the already complex problem of planning a course of action to achieve the goal.

The classical representation and algorithms for planing under uncertainty require the enumeration of the state space. Thus, although arguably a simple planning problem, we get already a rather large representation. For instance, if there are just 10 blocks, there are more than 50 million states. More importantly, we lose the structure implicit in the problem description. Decision-theoretic planning [Boutilier et al., 1999] generalized state-based planning under uncertainty to include features. But features are not enough to exploit all of the structure of the world. Each Blocks World planning instance is composed of these same basic individuals relations, but the instances differ in the arrangements of the blocks. We would like the agent to make use of this structure in terms of computing plans that apply across multiple scenarios. This can be achieved using relational models. For instance, the following parameterized plan, which says that we do nothing if block c is clear. If c is not clear than we move a clear block B that is above c to the floor, achieves the goal to have the block c clear

$$\begin{aligned} &doNothing \text{ if } clear(c) \\ &move(B, floor) \text{ if } \exists B \text{ above}(B, c) \wedge clear(B) \wedge \neg clear(c) , \end{aligned}$$

no matter how many blocks there are.

Example 1.6 Robotics As robots start to perform everyday manipulation tasks, such as cleaning up, setting a table or preparing simple meals, they must become much more knowledgeable than they are today. Typically, everyday tasks are specified vaguely and the robot must therefore infer what are the appropriate actions to take and which are the appropriate individuals involved in order to accomplish these tasks. These inferences can only be done if the robot has access to a compact and general knowledge base [Beetz et al., 2010, Tenorth et al., 2011]. StarAI provides the means for action-centered representation, for the automated acquisition of concepts through observation and experience, for reasoning about and managing uncertainty, and for fast inference.

These are only few examples for the many StarAI applications. It is clear that they all fit the definition of StarAI given above, that is, there are individuals that are connected to one another through relations and there is uncertainty that needs to be taken into account.

There are many more applications of StartAI. For instance, Getoor et al. [2001c] used statistical relational models to estimate the result size of complex database queries. Segal et al. [2001] employed probabilistic relational models for clustering gene expression data and to discover cellular processes from gene expression data [Segal et al., 2003]. Getoor et al. [2004] used probabilistic relational models to understand tuberculosis epidemiology. McGovern et al. [2003] estimated probabilistic relational trees to discover publication patterns in high-energy physics. Neville et al. [2005] used probabilistic relational trees to learn to rank brokers with respect to the probability that they would commit a serious violation of securities regulations in the near future.

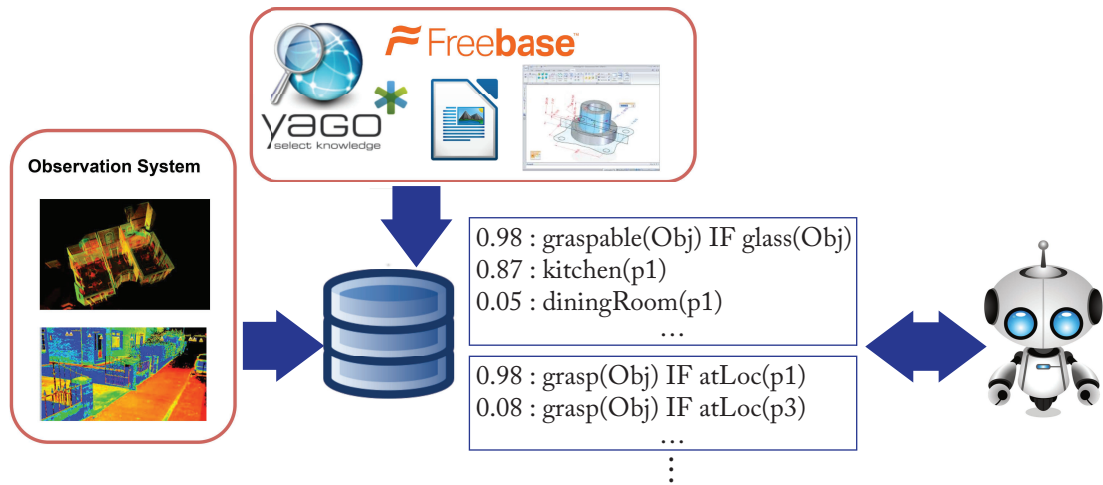


Figure 1.7: The robot's relational probabilistic model is populated with the data produced by its perception and experience (robot log data, human motion tracking, environment information, etc.) as well as with facts (or assertions) extracted from other dark data.

Anguelov et al. [2005] used relational Markov networks for segmentation of 3D scan data. Relational Markov networks have also been used to compactly represent object maps and to estimate trajectories of people by Limketkai et al. [2005]. Kersting et al. [2006] employed relational hidden Markov models for protein fold recognition. Singla and Domingos [2006] proposed a Markov logic model for entity resolution (ER), the problem of finding records corresponding to the same real-world entity. Markov logic has also been used for joint inference for event extraction [Poon and Domingos, 2007, Riedel and McCallum, 2011]. Poon and Domingos [2008] showed how to use Markov logic to perform joint unsupervised coreference resolution. Nonparametric relational models have been used for analyzing social networks [Xu et al., 2009a]. Kersting and Xu [2009] and Xu et al. [2010] used relational Gaussian processes for learning to rank search results. Poon and Domingos [2009] showed how to perform unsupervised semantic parsing using Markov logic networks, and Davis and Domingos [2009] used MLNs to successfully transfer learned knowledge among molecular biology, social network and Web domains. Yoshikawa et al. [2009] used Markov logic for identifying temporal relations in text, Meza-Ruiz and Riedel [2009] for semantic role labeling, and Kiddon and Domingos [2011] for biomolecular event prediction tasks. Niepert et al. [2010] used Markov logic for ontology matching. Tenenbaum et al. [2011] demonstrated that relational models can address some of the deepest questions about the nature and origins of human thought. Verbeke et al. [2012] used statistical relational learning for identifying evidence based medicine categories. Schiegg et al. [2012] segmented motion capture data using Markov logic mixture of Gaussian processes making use of both continuous and declarative

12 1. MOTIVATION

observations. [Song et al. \[2013\]](#) presented a Markov logic framework for recognizing complex events from multimodal data. Statistical relational learning has also been employed for learning to predict heart attacks [[Weiss et al., 2012](#)], onset of Alzheimer’s [[Natarajan et al., 2012a](#)], for analyzing pathways and omics experiments [[De Maeyer et al., 2013](#)], and extracting adverse drug events from text [[Odom et al., 2015](#)]. [Lang et al. \[2012\]](#) applied statistical relational learning to robot manipulations tasks, [Nitti et al. \[2014\]](#) to robot tracking tasks, and [Moldovan and De Raedt \[2014\]](#) to affordance learning. Generally, statistical relational methods are essential for enabling autonomous robots to do the right thing to the right object in the right way [[Tenorth and Beetz, 2013](#)]. [Hadji et al. \[2015\]](#) used relational label propagation to track migration among computer scientists. [Kersting et al. \[2015\]](#) used relational linear programs for topic classification in bibliographic networks, and [Toussaint \[2015\]](#) used relational mathematical programs for combined robot task and motion planning. [Foulds et al. \[2015\]](#) showcased the benefit of relational probabilistic languages for general-purpose topic modeling. [Khot et al. \[2015b\]](#) used Markov logic to answer elementary-level science questions using knowledge extracted automatically from textbooks, and [Mallory et al. \[2016\]](#) used DeepDive to extract both protein–protein and transcription factor interactions from over 100,000 full-text PLOS articles.

It is our belief that there will be many more exciting techniques and applications of StarAI in the future.

1.5 BRIEF HISTORICAL OVERVIEW

Before starting the technical introduction to StarAI, we briefly sketch some key streams of research that have led to the field of StarAI. Note that a full historical overview of this rich research field is out of the scope of this book.

StarAI and SRL (Statistical Relational Learning) basically integrate three key technical ingredients of AI: logic, probability, and learning. While each of these constituents has a rich tradition in AI, there has been a strong interest in pairwise combinations of these techniques since the early 1990s.

- Indeed, various techniques for *probabilistic learning* such as gradient-based methods, the family of EM algorithms or Markov Chain Monte Carlo methods have been developed and exhaustively investigated in different communities, such as in the Uncertainty in AI community for Bayesian networks and in the Computational Linguistics community for Hidden Markov Models. These techniques are not only theoretically sound, they have also resulted in entirely new technologies for, and revolutionary novel products in computer vision, speech recognition, medical diagnostics, troubleshooting systems, etc. Overviews of probabilistic learning can be found in [Koller and Friedman \[2009\]](#), [Bishop \[2006\]](#), [Murphy \[2012\]](#).
- Inductive logic programming and relational learning techniques studied *logic learning*, i.e., learning using first order logical or relational representations. Inductive Logic Programming

has significantly broadened the application domain of data mining especially in bio- and chemo-informatics and now represent some of the best-known examples of scientific discovery by AI systems in the literature. Overviews of inductive logic learning and relational learning can be found in this volume and in [Muggleton and De Raedt \[1994\]](#), [Lavraç and Dzeroski \[1994\]](#), [De Raedt \[2008\]](#).

- *Probabilistic logics* have been also studied from a knowledge representational perspective [[Nilsson, 1986](#), [Halpern, 1990](#), [Poole, 1993b](#)]. The aim of this research was initially more a probabilistic characterization of logic than suitable representations for learning.

In the late 1990s and early 2000s, researchers working on these pairwise combinations started to realize that they also need the third component. Influential approaches of this period include the Probabilistic Relational Models [[Friedman et al., 1999](#)], which extended the probabilistic graphical model approach toward relations, the probabilistic logic programming approach of [Sato \[1995\]](#), which extended the knowledge representation approach of [Poole \[1993b\]](#) with a learning mechanism, and approaches such as Bayesian and stochastic logic programs [[Kersting and De Raedt, 2001](#), [Cussens, 2001](#)] working within an inductive logic programming tradition. Around 2000, Lise Getoor and David Jensen organized the first workshop on “Statistical Relational Learning” at AAAI which gathered researchers active in the area for this time. This was the start of a successful series of events that continues till today (since 2010 under the header “StarAI”).

In the early days, many additional formalisms were contributed such as RBNs [[Jäger, 1997](#)], MMMNs [[Taskar et al., 2004](#)], SRMs [[Getoor et al., 2001c](#)], MLNs [[Richardson and Domingos, 2006](#)], BLOG [[Milch et al., 2005](#)], RDNs [[Neville and Jensen, 2004](#)], and IBAL [[Pfeffer, 2001](#)], many of which are described in the book edited by [Getoor and Taskar \[2007\]](#). Especially influential was the Markov Logic approach of [Richardson and Domingos \[2006\]](#), as it was an elegant combination of undirected graphical models with relational logic. In this early period, research focused a lot on representational and expressivity issues, often referring to the “alphabet-soup” of systems, reflecting the fact that many systems have names that use three or four letters from the alphabet. Around 2005, it was realized that the commonalities between the different systems are more important than their (often syntactic) differences, and research started to focus on key open issues surrounding learning and inference. One central question is that of lifted inference [[Poole, 2003](#)], that is, whether one can perform probabilistic inference without grounding out the probabilistic relational model first. Other questions concern the relationship to existing probabilistic and logical solvers and the continuing quest for efficient inference techniques. Finally, research on SRL and StarAI has also inspired the addition of probabilistic primitives to programming languages, leading to what is now called probabilistic programming. Although some of the early formalisms [[Poole, 1993b](#), [Sato, 1995](#), [Pfeffer, 2001](#)] already extend an existing programming language with probabilities, and hence, possess the power of a universal Turing machine, this stream of research became popular since BLOG [[Milch et al., 2005](#)] and Church [[Goodman et al., 2008](#)].