Recent Advances in Querying Probabilistic Knowledge Bases

Stefan Borgwardt[1], **İsmail İlkan Ceylan**[2], **Thomas Lukasiewicz**[2]
[1] Faculty of Computer Science, Technische Universität Dresden, Germany
[2] Department of Computer Science, University of Oxford, UK
stefan.borgwardt@tu-dresden.de, ismail.ceylan@cs.ox.ac.uk, thomas.lukasiewicz@cs.ox.ac.uk

Abstract

We give a survey on recent advances at the forefront of research on probabilistic knowledge bases for representing and querying large-scale automatically extracted data. We concentrate especially on increasing the semantic expressivity of formalisms for *representing* and *querying* probabilistic knowledge (i) by giving up the closed-world assumption, (ii) by allowing for commonsense knowledge (and in parallel giving up the tuple-independence assumption), and (iii) by giving up the closed-domain assumption, while preserving some computational properties of query answering in such formalisms.

1 Introduction

In the recent years, there has been a strong interest in building *large-scale probabilistic knowledge bases (PKBs)* from data in an automated way, which has resulted in several systems, such as DeepDive [De Sa *et al.*, 2016; 2017], NELL [Mitchell *et al.*, 2015], Microsoft's Probase [Wu *et al.*, 2012], and Google's Knowledge Vault [Dong *et al.*, 2014]. These systems continuously crawl the Web and extract *structured information*, and thus populate their databases with millions of entities and billions of tuples. A recent survey on such systems has been given in [Hossain and Schwitter, 2018].

To what extent can these search and extraction systems help with real-world use cases? This turns out to be an open-ended question. For example, DeepDive is used to build knowledge bases for domains such as paleontology, geology, medical genetics, and human movement [Ku *et al.*, 2015; Peters *et al.*, 2014]. Google's Knowledge Vault has compiled more than a billion facts from the Web and is primarily used to improve the quality of search results on the Web. Currently, it can even estimate the trustworthiness of more than 119M sources [Dong *et al.*, 2015].

From a broader perspective, the quest for building large knowledge bases serves as a new dawn for artificial intelligence (AI) research. Fields such as information extraction, natural language processing (e.g., question answering), relational and deep learning, knowledge representation and reasoning, and databases are taking initiative towards a common goal. Querying large-scale probabilistic knowledge bases is commonly regarded to be at the heart of these efforts.

Beyond all these success stories, however, probabilistic knowledge bases still lack the fundamental machinery to convey some of the valuable knowledge hidden in them to the end user [Weikum *et al.*, 2016], which seriously limits their potential applications in practice. These problems are rooted in the semantic expressivity of *probabilistic databases* (PDBs) [Imieliński and Lipski, 1984; Fuhr and Rölleke, 1997; Suciu *et al.*, 2011], which are used for encoding most probabilistic knowledge bases. Most relational PDB management systems are restricted in their expressivity in three dimensions: the *relational dimension*, the *knowledge dimension*, and the *probabilistic dimension*.

More specifically, along the relational dimension, PDB systems are restricted by two shortcomings, all of which are inherited from classical databases:

- The *closed-world assumption* states that any fact that is not entailed by the knowledge base is false, i.e., has probability 0.
- The *closed-domain assumption* fixes the domain of discourse to a *finite* set of known constants, i.e., those that appear in the database.

In the knowledge dimension,

- the *lack of commonsense knowledge* means that PDB systems cannot infer new facts from stated ones in the way a human does intuitively.

All three limitations have been crucial for the efficiency, and hence the success, of classical database systems. They are also well-justified in an environment where the data are well-curated and of high quality, e.g., for small-to-mid-scale companies' internal knowledge management systems. However, in recent years, it has been increasingly recognized that when dealing with automatically constructed large-scale knowledge bases, these restrictions need to be relaxed to allow for more flexibility and expressivity [Weikum *et al.*, 2016].

There is an additional tradeoff between expressivity and efficiency along the probabilistic dimension of PDBs:

- The most popular PDB model employs the *tuple-independence assumption*, i.e., it views every extracted fact as an *independent* Bernoulli variable, which means that all facts in the database are assumed to be probabilistically independent.

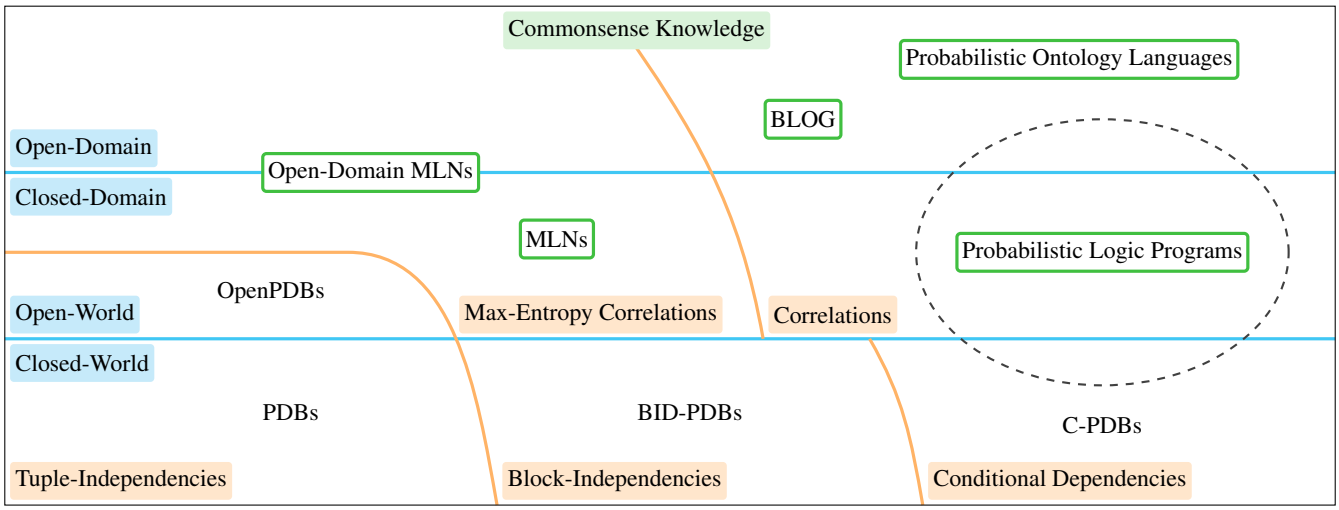

Figure 1: Landscape of formalisms for probabilistic knowledge bases. The horizontal lines (in blue) represent the relational dimension, the vertical lines (in orange) the probabilistic dimension, and the models in green boxes include some form of commonsense knowledge (encoded by logical formulas). We classify some closed-domain models as open-world in the sense that these representations allow to assign positive probabilities to facts that are not explicitly inferred (unlike closed-world, closed-domain models). More conventional open-world models are those that also operate over open domains.

Again, this has been crucial for the success of PDB systems; nonetheless, this assumption is unnatural in many real-world applications, which need to express at least some correlations or conditional dependencies between pieces of information.

Following the recent paradigms in querying probabilistic data and knowledge bases [Ceylan, 2017], we survey the landscape of current research on relaxing these four restrictions towards more expressive probabilistic knowledge base formalisms. These restrictions are tightly connected, as we illustrate in Figure 1. Along the relational dimension (blue), we observe an increase in expressivity from closed-world, closed-domain PDBs (bottom) to open-world, open-domain approaches (top). In the knowledge dimension (green), we distinguish models that can encode commonsense knowledge. Along the probabilistic dimension (orange), we compare tuple-independent models (left) to ones that allow correlations between facts and explicit conditional dependencies (right). This classification clearly abstracts away from other subtle differences among these formalisms; more details are given in the relevant sections.

In the rest of this paper, we first introduce the basic PDB models, highlighting the development of successful systems, and then address relaxing the closed-world assumption, the lack of commonsense knowledge, and finally the closed-domain assumption. The discussion of probabilistic dependencies is internal to each section.

2 Probabilistic Databases

Tuple-independent probabilistic databases (PDBs) [Dalvi and Suciu, 2007] are a simple and intuitive model for adding probabilities to classical databases, interpreted using the standard possible-worlds semantics. A possible world is a combination of true and false facts, and is assigned a probability by multiplying the probabilities of the true facts with the complementary probabilities of those facts that are believed to be false. A query (i.e., formula) is evaluated by summing up the probabilities of all those worlds where it is satisfied.

Example 1. Consider the following tuple-independent PDB $\mathcal{P}$, which assigns the probability 0.5 to several self-explaining facts

$$\mathsf{composer}(\mathsf{haydn}){:}\,0.5,\ \mathsf{teacherOf}(\mathsf{haydn},\mathsf{beethoven}){:}\,0.5,$$
$$\mathsf{knows}(\mathsf{haydn},\mathsf{beethoven}){:}\,0.5,\ \mathsf{friendOf}(\mathsf{haydn},\mathsf{mozart}){:}\,0.5,$$

e.g., Joseph Haydn was recognised to be a composer with a certainty of 0.5. The conjunctive query

$$Q_1 = \exists x\,(\mathsf{knows}(x,\mathsf{beethoven}) \wedge \mathsf{composer}(x)),$$

is evaluated to a probability of 0.25, since it is satisfied in all worlds in which the first and the third facts are true. ∎

Dalvi and Suciu [2012] have shown that evaluating unions of conjunctive queries is either possible in polynomial time (*safe*), or it is #P-hard (*unsafe*), and gave an effective algorithm to recognise and evaluate safe queries. However, the class of safe queries is small, and it is crucial to achieve scalability for unsafe queries, using the insights provided by the structure of safe queries. Beyond exact query evaluation, it is well known that unions of conjunctive queries can be approximately evaluated in polynomial time, since such queries are positive (existential) DNF formulas, for which polynomial time approximation schemes exist [Karp *et al.*, 1989].

The tuple-independent PDB model has been extended towards expressing probabilistic dependencies. In *block-independent-disjoint (BID) PDBs* [Dalvi *et al.*, 2009], each relation is grouped into blocks, and the facts of each block are assumed to be disjoint (i.e., mutually exclusive), while the facts of different blocks are assumed to be independent. *Conditional PDBs (C-PDBs)* and *U-databases* allow to encode further dependencies between facts by annotating them with propositional formulas [Green and Tannen, 2006;

Antova *et al.*, 2008]; this allows them to represent any probability distribution over the set of possible worlds.

Query evaluation over PDBs is closely related to the task of *weighted (first-order) model counting* [Gribkoff *et al.*, 2014b; Beame *et al.*, 2017]. For solving either task, the method of *knowledge compilation* has proven fruitful, where the PDB is translated into a different representation over which queries can be evaluated more efficiently. While the translation may be time-consuming, this effort is paid off when a large number of queries needs to be answered over the same PDB. In this way, knowledge compilation solvers can be applied to tasks that are PP-complete [Park and Darwiche, 2004], such as threshold query evaluation over PDBs [Jha and Suciu, 2013; Olteanu and Schleich, 2016]. Tensor factorization has recently also been applied to PDBs [Krompaß *et al.*, 2014].

There exists a multitude of PDB systems, such as MystiQ [Boulos *et al.*, 2005], SPROUT [Fink *et al.*, 2011], SlimShot [Gribkoff and Suciu, 2016], MayBMS [Antova *et al.*, 2008], and Tuffy [Niu *et al.*, 2011], which are based on a variety of exact and approximate query evaluation techniques. A recent survey on query answering over probabilistic data has been given in [Van den Broeck and Suciu, 2017].

3 Open-World Assumption

Most real-world probabilistic knowledge bases encode only a portion of the real world, and this description is, in most cases, *incomplete*. However, for computational efficiency reasons, PDBs typically lack a suitable handling of incompleteness. In the query semantics, most of these systems employ the *closed-world assumption*, i.e., any fact that is not entailed by the knowledge base is assigned the probability 0, and thus assumed to be impossible, although it actually has some unknown probability in $[0, 1]$. Hence, many queries evaluate to the probability 0, which makes it impossible to distinguish queries that should *intuitively* differ.

Example 2. We illustrate the effects of the closed-world assumption on the PDB $\mathcal{P}$ from above, where all missing facts have the probability 0, i.e., they are false. The following two queries both evaluate to the probability 0 over the PDB:

$$Q_2 = \exists x\,(\mathsf{teacherOf}(x, \mathsf{beethoven}) \wedge \mathsf{bornIn}(x, \mathsf{austria})),$$
$$Q_3 = \exists x\,(\mathsf{person}(x) \wedge \neg\mathsf{person}(x)).$$

In particular, the fact bornIn(haydn, austria) is assumed to have the probability 0 (i.e., to be false); however, this assumption is likely incorrect. Indeed, bornIn(haydn, austria) may even have the probability 1 (i.e., may be true), which would result in Q_2 having the probability 0.5. In contrast, Q_3 is unsatisfiable and should always have the probability 0, no matter how incomplete the PDB is. That is, the closed-world assumption forces a very flat representation, which makes it impossible to even distinguish a satisfiable query from an unsatisfiable one. ∎

Since [Reiter, 1980], the closed-world assumption has been widely criticised in the context of classical data and knowledge bases, and alternatives have been proposed. Similar arguments apply to PDBs and probabilistic knowledge bases. In fact, for real-world applications of PDBs, it is common to use the so-called *local closed-world assumption* [Dong *et al.*, 2014], a relaxed version that is, however, still distant from the open-world assumption, which allows all facts to have a non-zero probability, even if they do not follow from the database.

An alternative approach is to set the probabilities of facts that are not in the database to a default probability interval. This approach, i.e., allowing unknown facts to take on probabilities from an interval, belongs to the area of imprecise probabilities [Levi, 1980]. The resulting inference over sets of probability distributions is harder than inference over a single distribution; for efficiency reasons, one is thus interested in closed and convex sets of distributions in the spirit of credal networks [Cozman, 2000]. In *open-world PDBs (OpenPDBs)* [Ceylan *et al.*, 2016a], the dichotomy result and algorithm for PDBs [Dalvi and Suciu, 2012] were lifted to the case where unknown facts take on a value from a default probability interval, without decreasing the class of safe queries.

4 Commonsense Knowledge

Another limitation of PDBs is the *lack of commonsense knowledge*. This shows up in real-world applications of PDBs: nowadays, results of Web search usually come with structured information boxes whenever possible, e.g., the search for "Mozart" or "Beethoven" results in a box containing basic information about them, such as their date of birth and their compositions. This information is linked to the underlying knowledge base [Dong *et al.*, 2014], but when it comes to query answering, these systems lack a means of intuitive reasoning. This problem is also closely linked to the tuple-independence assumption in PDBs.

Example 3. The query Q_4 asking whether there is a composer who knows both Mozart and Beethoven,

$$\exists x(\mathsf{composer}(x) \wedge \mathsf{knows}(x, \mathsf{beethoven}) \wedge \mathsf{knows}(x, \mathsf{mozart}))$$

is evaluated to the probability 0, since the fact knows(haydn, mozart) is missing from the knowledge base. Even using open-world PDBs, this fact would only be assigned a default probability interval, i.e., would be equally likely as knows(haydn, elvis). However, the PDB $\mathcal{P}$ actually contains concrete information about this fact, namely that Haydn was a friend of Mozart. The only missing link is the commonsense knowledge that friends of course know each other. Furthermore, under tuple-independence, the query

$$Q_5 = \exists x\,(\mathsf{teacherOf}(x, \mathsf{beethoven}) \wedge \mathsf{knows}(x, \mathsf{beethoven}))$$

evaluates to the probability $0.5 \cdot 0.5 = 0.25$. But, if Haydn is a teacher of Beethoven, then he also knows Beethoven, i.e., these tuples are not independent. So, the probability of Q_5 should actually be 0.5. ∎

As we can see, adding commonsense knowledge to a probabilistic knowledge base can help improving query answers. Reasoning exploits such basic knowledge to deduce implicit consequences from data, and this kind of knowledge is essential for querying large-scale PDBs in an uncontrolled environment, i.e., the internet. Realistic data models should thus incorporate commonsense knowledge, which is also inherently connected to giving up the tuple-independence assumption of standard PDBs. Indeed, this knowledge automatically induces implicit dependencies between facts, which means

that the tuple-independence assumption does not hold anymore (cf. Figure 1).

The need to relax the tuple-independence assumption and to allow for representing commonsense knowledge has already been recognized in several recent approaches to PKBs, e.g., based on *Markov logic networks (MLNs)* [Richardson and Domingos, 2006; Jha and Suciu, 2013; Gribkoff and Suciu, 2016]. There, the PDB is viewed as a set of weighted facts in an MLN; additional soft/hard constraints are imposed through a set of weighted/unweighted rules. This is closely related to *maximum-entropy (ME)* models, where the probability distribution is obtained as the one of maximum entropy satisfying some constraints; intuitively, such a model makes the least assumptions on dependencies between tuples, and leaves all unaffected tuples independent [Jaynes, 1957].

In particular, the recent system SlimShot [Gribkoff and Suciu, 2016] reduces such PKBs to tuple-independent PDBs with additional logical constraints; it also provides certain accuracy guarantees; but the numbers and sizes of the resulting independent probabilistic facts grow very quickly with the number of logically related facts. Other related approaches based on Markov logic networks, like Tuffy [Niu *et al.*, 2011] and DeepDive [De Sa *et al.*, 2016; 2017], use Markov Chain Monte Carlo (MCMC) for probabilistic inference, or a variant called MC-SAT [Poon and Domingos, 2006]. In general, however, such approximating algorithms do not provide any accuracy guarantees.

Other probabilistic formalisms that allow to encode commonsense knowledge, but still make the closed-domain assumption, are based on (function-free) *probabilistic logic programming*, covering positive as well as more general logic programs, such as normal and disjunctive normal logic programs. In particular, in probabilistic logic programming under the *distribution semantics* (e.g., *probabilistic Datalog* [Fuhr, 1995], *Bayesian logic programs* [Kersting and De Raedt, 2001], *ProbLog* [De Raedt *et al.*, 2007], and *probabilistic description logic programs* [Lukasiewicz *et al.*, 2011])—which is perhaps closest in spirit to PDBs—different joint instantiations of random variables specify subsets of a logic program and thus their least Herbrand models. Along with probability distributions over the values of each random variable and an independence assumption on the random variables, this then generates a probability for each joint instantiation, and thus overall a probability distribution (or a set of probability distributions) over all Herbrand models.

Observe that this is fundamentally different from the probabilistic semantics above that are based on Markov logic networks or maximum-entropy models. Furthermore, this distribution semantics for probabilistic logic programs is an example where the independence of a collection of random variables is not in conflict with a large amount of commonsense knowledge, as the latter is defined on top of instantiations of the random variables. Note also that probabilistic logic programs under the distribution semantics allow to encode explicit conditional dependencies between the probabilistic facts, similar to Bayesian networks; for a recent survey on probabilistic logic programming under the distribution semantics and their ability to encode Bayesian networks, see [De Raedt and Kimmig, 2015]. Importantly, many approaches to probabilistic logic programming also make a closed-world assumption (cf. Figure 1), e.g., the distribution semantics assigns the probability 0 to facts that are not entailed in any possible world, and approaches to probabilistic logic programming with nonmonotonic negation in rule bodies employ negation as failure as part of their semantics.

There are also many alternative approaches to probabilistic logic programming that are not based on the distribution semantics, such as the ones in [Lukasiewicz, 2001; Kern-Isberner and Lukasiewicz, 2004; Lukasiewicz, 2008b]; they are based on a language of conditional constraints, which specifies a convex set of probability distributions over a finite set of Herbrand models, the maximum-entropy model in that convex set of probability distributions, and a probabilistic default semantics, respectively. The approach in [Kern-Isberner and Lukasiewicz, 2004] is similar to the probabilistic semantics above that are based on MLNs or maximum-entropy models; however, a crucial difference is that conditional probabilities are a modeling primitive, while the above MLNs and maximum-entropy models only model unconditional probabilities as primitives.

5 Open-Domain Assumption

All the approaches in the previous section have in common that they also allow for modelling logic-based commonsense knowledge. However, since they are based on *grounding* universally quantified variables in first-order formulas with known constants of a finite domain, they essentially only allow for encoding *propositional* logical knowledge, and not fully fledged first-order knowledge, as it already occurs in (rather restricted) ontology languages that are used to formulate commonsense knowledge. In particular, e.g., standard MLNs cannot express full existential quantification, which may introduce unknown individuals, as illustrated by the following example.

Example 4. Consider the constraint "every employee has a private address". Unless the private address of employee A is explicitly mentioned in the PDB, in an MLN, this constraint means that A must be assigned a private address that is either (a) a private address of another employee that is mentioned in the data, or (b) a completely different object, e.g., an employee or a department. Obviously, neither (a) nor (b) reflects the intended meaning. ∎

Interpreting databases under fully fledged first-order commonsense knowledge in the form of ontologies is closely related to ontology-based data access (OBDA) [Poggi *et al.*, 2008], which has been very widely studied in the context of classical databases, and also addresses the need for open-world, open-domain querying. Ontology-based access to probabilistic data has been studied for the lightweight description logics $\mathcal{EL}$ and *DL-Lite*, and the data complexity dichotomy in PDBs has been lifted [Jung and Lutz, 2012]; they also describe the case of a more expressive ontology language that causes all CQs of a certain form to become #P-hard. The paper [Borgwardt *et al.*, 2017] considers the more expressive languages of the Datalog$^{\pm}$ family and provides results both relative to PDBs and OpenPDBs. It shows that the semantic

differences between these formalisms lead to different results, and also identifies tractable cases.

Most of the recent work on probabilistic query answering using ontologies is based on lightweight ontology languages, such as the approaches to *Bayesian description logics* in [d'Amato *et al.*, 2008; Ceylan and Peñaloza, 2015; 2017], which combine the description logics of the *DL-Lite* family and the description logic $\mathcal{EL}$, respectively, with Bayesian networks [Pearl, 1988]. The underlying probabilistic semantics can be generalized to other ontology languages and graphical probabilistic models as well. For example, a closely related approach is the one to probabilistic Datalog$^{\pm}$ in [Gottlob *et al.*, 2013], which combines Datalog$^{\pm}$ with Markov logic networks [Richardson and Domingos, 2006]. In [Ceylan *et al.*, 2016b], the computational complexity of query answering in probabilistic Datalog$^{\pm}$ under the possible worlds semantics is investigated. In closely related work [Lukasiewicz *et al.*, 2016; Lukasiewicz and Predoiu, 2016], the computational complexity of query answering over annotation-based PDBs under Datalog$^{\pm}$ is explored.

There are many alternative approaches to probabilistic ontology languages that are based on a different probabilistic semantics (for a survey, see [Lukasiewicz and Straccia, 2008]), such as the approach in [Lukasiewicz, 2008a], which uses a probabilistic default reasoning semantics, and which allows to represent and reason about terminological probabilistic knowledge about classes of individuals, as well as assertional probabilistic knowledge about single individuals.

The need for the open-domain assumption has also been recognized in statistical relational formalisms. The probabilistic programming language BLOG [Milch *et al.*, 2005] allows to encode probabilistic models with unknown objects (i.e., objects that may not be known a priori and may not be directly and uniquely identified), and thus to perform inference over open domains. In [Singla and Domingos, 2007], also MLNs were extended to an open domain by allowing universal quantifiers to range over an infinite domain (but not existential quantifiers as used in Example 4). Rather recently, a probabilistic extension of Datalog is proposed in [Bárány *et al.*, 2017] to also cope with infinite domains.

There are also many approaches to open-domain probabilistic logic programming (with function symbols) under the distribution semantics, such as *probabilistic Horn abduction* [Poole, 1993], *PRISM* [Sato and Kameya, 1997], the *independent choice logic* [Poole, 1997], and *P-log* [Baral *et al.*, 2009]. The main idea behind the underlying probabilistic semantics is that a well-definedness condition ensures finite probability distributions even in the presence of function symbols, namely, by guaranteeing that there are only finitely many different possible worlds (often along with assuming that their canonical Herbrand models are also finite).

6 Alternative Inference Tasks

In addition to query evaluation and computing conditional probabilities, a lot of research effort has also focused on *explaining* probabilistic knowledge bases. The explanation of the behaviour of AI systems is widely regarded as an important item on the agenda of AI research in general. In probabilistic models, this encompasses research on established inference tasks such as *maximum a posteriori* (MAP) computations and *most probable explanations* (MPE), e.g., in probabilistic graphical models [Koller and Friedman, 2009], as well as recent trends towards notions of *causality*, e.g., in database research [Meliou *et al.*, 2010; Kanagal *et al.*, 2011].

In MPE and MAP, one is interested in finding the *most plausible* instantiation of a set of probabilistic variables that explains a given observation. In our setting, this corresponds to identifying tuples that contribute the most to the satisfaction of an observed query.

Example 5. One may ask, for example, what are the tuples from $\mathcal{P}$ that are responsible for the query Q_1—the answer being, of course, composer(haydn) and knows(haydn, beethoven). If the PDB contains more composers that knew Beethoven, the task is to find the combination of tuples with the highest probability. This task is more important in the presence of commonsense knowledge, where it is not always so clear which tuples contribute to satisfying the query.

For MLNs and probabilistic logic programming approaches, this is one of the central inference tasks investigated [Richardson and Domingos, 2006; De Raedt and Kimmig, 2015]. For tuple-independent PDBs, first investigations into this problem for a variety of database and ontology-based query languages have been made in [Gribkoff *et al.*, 2014a; Ceylan *et al.*, 2017].

More generally, research on causality in (probabilistic) databases also tries to explain negative query answers, as well as rank the causes according to their *responsibility*, a measure of the influence they have on the query result [Meliou *et al.*, 2010; Kanagal *et al.*, 2011]

7 Summary and Outlook

We have surveyed recent advances in querying probabilistic knowledge bases. We focused on research efforts to increase the semantic expressivity of formalisms for representing and querying probabilistic knowledge. Our survey does not provide a deep discussion in terms of the scalability issues in each of the above formalisms, which is still a very active research area, and there are big challenges remaining in this regard. On the other hand, expressive semantic representations do not necessarily lead to a computational overhead; for example, it is well-known that some ontology languages admit rewritings into unions of conjunctive queries; meaning that they can essentially be evaluated on PDB systems using existing approximation schemes.

All the above topics are also subject to heavy present research activities on representing and querying probabilistic knowledge. Despite all the described advances, further research efforts will be necessary to unlock the full potential of probabilistic knowledge bases in applications in practice.

Acknowledgments

This work was supported by the German Research Foundation (DFG) within the project BA 1122/19-1 (GOASQ), by The Alan Turing Institute under the UK EPSRC grant EP/N510129/1, and by the EPSRC grants EP/R013667/1, EP/L012138/1, and EP/M025268/1.

References

[Antova *et al.*, 2008] Lyublena Antova, Thomas Jansen, Christoph Koch, and Dan Olteanu. Fast and simple relational processing of uncertain data. In *Proc. of ICDE*, pages 983–992, 2008.

[Baral *et al.*, 2009] Chitta Baral, Michael Gelfond, and J. Nelson Rushton. Probabilistic reasoning with answer sets. *Theory and Practice of Logic Programming*, 9(1):57–144, 2009.

[Bárány *et al.*, 2017] Vince Bárány, Balder ten Cate, Benny Kimelfeld, Dan Olteanu, and Zografoula Vagena. Declarative probabilistic programming with Datalog. *ACM TODS*, 42(4):22:1–22:35, 2017.

[Beame *et al.*, 2017] Paul Beame, Jerry Li, Sudeepa Roy, and Dan Suciu. Exact model counting of query expressions: Limitations of propositional methods. *ACM TODS*, 42(1):1:1–1:46, 2017.

[Borgwardt *et al.*, 2017] Stefan Borgwardt, İsmail İlkan Ceylan, and Thomas Lukasiewicz. Ontology-mediated queries for probabilistic databases. In *Proc. of AAAI*, pages 1063–1069, 2017.

[Boulos *et al.*, 2005] Jihad Boulos, Nilesh Dalvi, Bhushan Mandhani, Shobhit Mathur, Chris Ré, and Dan Suciu. MYSTIQ: A system for finding more answers by using probabilities. In *Proc. of ACM SIGMOD*, pages 891–893, 2005.

[Ceylan and Peñaloza, 2015] İsmail İlkan Ceylan and Rafael Peñaloza. Probabilistic query answering in the Bayesian description logic $\mathcal{BEL}$. In *Proc. of SUM*, pages 21–35. Springer, 2015.

[Ceylan and Peñaloza, 2017] İsmail İlkan Ceylan and Rafael Peñaloza. The Bayesian ontology language $\mathcal{BEL}$. *J. Autom. Reasoning*, 58(1):67–95, 2017.

[Ceylan *et al.*, 2016a] İsmail İlkan Ceylan, Adnan Darwiche, and Guy Van den Broeck. Open-world probabilistic databases. In *Proc. of KR*, pages 339–348. AAAI Press, 2016.

[Ceylan *et al.*, 2016b] İsmail İlkan Ceylan, Thomas Lukasiewicz, and Rafael Peñaloza. Complexity results for probabilistic Datalog$^{\pm}$. In *Proc. of ECAI*, pages 1414–1422, 2016.

[Ceylan *et al.*, 2017] İsmail İlkan Ceylan, Stefan Borgwardt, and Thomas Lukasiewicz. Most probable explanations for probabilistic database queries. In *Proc. of IJCAI*, pages 950–956, 2017.

[Ceylan, 2017] İsmail İlkan Ceylan. *Query Answering in Probabilistic Data and Knowledge Bases*. PhD thesis, Technische Universität Dresden, 2017.

[Cozman, 2000] Fabio Gagliardi Cozman. Credal networks. *AIJ*, 120(2):199–233, 2000.

[Dalvi and Suciu, 2007] Nilesh Dalvi and Dan Suciu. Efficient query evaluation on probabilistic databases. *The VLDB J.*, 16(4):523–544, 2007.

[Dalvi and Suciu, 2012] Nilesh Dalvi and Dan Suciu. The dichotomy of probabilistic inference for unions of conjunctive queries. *J. ACM*, 59(6):1–87, 2012.

[Dalvi *et al.*, 2009] Nilesh Dalvi, Christopher Ré, and Dan Suciu. Probabilistic databases: Diamonds in the dirt. *Comm. ACM*, 52(7):86–94, 2009.

[d'Amato *et al.*, 2008] Claudia d'Amato, Nicola Fanizzi, and Thomas Lukasiewicz. Tractable reasoning with Bayesian description logics. In *Proc. of SUM*, pages 146–159, 2008.

[De Raedt and Kimmig, 2015] Luc De Raedt and Angelika Kimmig. Probabilistic (logic) programming concepts. *Mach. Learn.*, 100(1), 2015.

[De Raedt *et al.*, 2007] Luc De Raedt, Angelika Kimmig, and Hannu Toivonen. ProbLog: A probabilistic Prolog and its application in link discovery. In *Proc. of IJCAI*, pages 2462–2467, 2007.

[De Sa *et al.*, 2016] Christopher De Sa, Alex Ratner, Christopher Ré, Jaeho Shin, Feiran Wang, Sen Wu, and Ce Zhang. DeepDive: Declarative knowledge base construction. *SIGMOD Rec.*, 45(1):60–67, 2016.

[De Sa *et al.*, 2017] Christopher De Sa, Alex Ratner, Christopher Ré, Jaeho Shin, Feiran Wang, Sen Wu, and Ce Zhang. Incremental knowledge base construction using DeepDive. *The VLDB J.*, 26(1):81–105, 2017.

[Dong *et al.*, 2014] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge Vault: A web-scale approach to probabilistic knowledge fusion. In *Proc. of ACM SIGKDD*, pages 601–610, 2014.

[Dong *et al.*, 2015] Xin Dong, Evgeniy Gabrilovich, Kevin Murphy, Van Dang, Wilko Horn, Camillo Lugaresi, Shaohua Sun, and Wei Zhang. Knowledge-based trust: Estimating the trustworthiness of web sources. *Proc. of VLDB*, 8(9):938–949, 2015.

[Fink *et al.*, 2011] Robert Fink, Andrew Hogue, Dan Olteanu, and Swaroop Rath. Sprout2: A squared query engine for uncertain web data. In *Proc. of ACM SIGMOD*, pages 1299–1302, 2011.

[Fuhr and Rölleke, 1997] Norbert Fuhr and Thomas Rölleke. A probabilistic relational algebra for the integration of information retrieval and database systems. *ACM TOIS*, 15(1):32–66, 1997.

[Fuhr, 1995] Norbert Fuhr. Probabilistic Datalog—A logic for powerful retrieval methods. In *Proc. of SIGIR*, pages 282–290, 1995.

[Gottlob *et al.*, 2013] Georg Gottlob, Thomas Lukasiewicz, Maria Vanina Martinez, and Gerardo I. Simari. Query answering under probabilistic uncertainty in Datalog$^{\pm}$ ontologies. *Ann. Math. Artif. Intell.*, 69(1):37–72, 2013.

[Green and Tannen, 2006] Todd Green and Val Tannen. Models for incomplete and probabilistic information. In *Proc. of EDBT Workshops*, pages 278–296. Springer, 2006.

[Gribkoff and Suciu, 2016] Eric Gribkoff and Dan Suciu. SlimShot: In-database probabilistic inference for knowledge bases. *Proc. of VLDB*, 9(7), 2016.

[Gribkoff *et al.*, 2014a] Eric Gribkoff, Guy Van den Broeck, and Dan Suciu. The most probable database problem. In *Proc. of BUDA*, 2014.

[Gribkoff *et al.*, 2014b] Eric Gribkoff, Guy Van den Broeck, and Dan Suciu. Understanding the complexity of lifted inference and asymmetric weighted model counting. In *Proc. of UAI*, pages 280–289, 2014.

[Hossain and Schwitter, 2018] Bayzid Ashik Hossain and Rolf Schwitter. A survey on automatically constructed universal knowledge bases. *Sem. Web*, 2018. Submitted, available online.

[Imieliński and Lipski, 1984] Tomasz Imieliński and Witold Lipski. Incomplete information in relational databases. *J. ACM*, 31(4):761–791, 1984.

[Jaynes, 1957] E. T. Jaynes. Information theory and statistical mechanics. *Physics Review. Series II*, 106(4):620–630, 1957.

[Jha and Suciu, 2013] Abhay Jha and Dan Suciu. Knowledge compilation meets database theory: Compiling queries to decision diagrams. *Theor. Comput. Syst.*, 52(3), 2013.

[Jung and Lutz, 2012] Jean Christoph Jung and Carsten Lutz. Ontology-based access to probabilistic data with OWL QL. In *Proc. of ISWC, Part I*, pages 182–197, 2012.

[Kanagal *et al.*, 2011] Bhargav Kanagal, Jian Li, and Amol Deshpande. Sensitivity analysis and explanations for robust query evaluation in probabilistic databases. In *Proc. of SIGMOD*, pages 841–852, 2011.

[Karp *et al.*, 1989] Richard M. Karp, Michael Luby, and Neal Madras. Monte-Carlo approximation algorithms for enumeration problems. *J. Alg.*, 10(3):429 – 448, 1989.

[Kern-Isberner and Lukasiewicz, 2004] Gabriele Kern-Isberner and Thomas Lukasiewicz. Combining probabilistic logic programming with the power of maximum entropy. *AIJ*, 157(1/2):139–202, 2004.

[Kersting and De Raedt, 2001] Kristian Kersting and Luc De Raedt. Towards combining inductive logic programming with Bayesian networks. In *Proc. of ILP*, pages 118–131, 2001.

[Koller and Friedman, 2009] Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[Krompaß *et al.*, 2014] Denis Krompaß, Maximilian Nickel, and Volker Tresp. Querying factorized probabilistic triple databases. In *Proc. of ISWC*, pages 114–129, 2014.

[Ku *et al.*, 2015] Joy P. Ku, Jennifer L. Hicks, Trevor Hastie, Jure Leskovec, Christopher Ré, and Scott L. Delp. The mobilize center: An NIH big data to knowledge center to advance human movement research and improve mobility. *J. Am. Med. Inform. Assoc.*, 22(6):1120–1125, 2015.

[Levi, 1980] Isaac Levi. *The Enterprise of Knowledge*. MIT Press, 1980.

[Lukasiewicz and Predoiu, 2016] Thomas Lukasiewicz and Livia Predoiu. Complexity of threshold query answering in probabilistic ontological data exchange. In *Proc. of ECAI*, pages 1008–1016, 2016.

[Lukasiewicz and Straccia, 2008] Thomas Lukasiewicz and Umberto Straccia. Managing uncertainty and vagueness in description logics for the semantic web. *J. Web Sem.*, 6(4):291–308, 2008.

[Lukasiewicz *et al.*, 2011] Thomas Lukasiewicz, Livia Predoiu, and Heiner Stuckenschmidt. Tightly integrated probabilistic description logic programs for representing ontology mappings. *Ann. Math. Artif. Intell.*, 63(3/4):385–425, 2011.

[Lukasiewicz *et al.*, 2016] Thomas Lukasiewicz, Maria Vanina Martinez, Livia Predoiu, and Gerardo I. Simari. Basic probabilistic ontological data exchange with existential rules. In *Proc. of AAAI*, pages 1023–1029. AAAI Press, 2016.

[Lukasiewicz, 2001] Thomas Lukasiewicz. Probabilistic logic programming with conditional constraints. *ACM TOCL*, 2(3):289–339, 2001.

[Lukasiewicz, 2008a] Thomas Lukasiewicz. Expressive probabilistic description logics. *AIJ*, 172(6/7):852–883, 2008.

[Lukasiewicz, 2008b] Thomas Lukasiewicz. Probabilistic description logic programs under inheritance with overriding for the Semantic Web. *Int. J. Approx. Reasoning*, 49(1):18–34, 2008.

[Meliou *et al.*, 2010] Alexandra Meliou, Wolfgang Gatterbauer, Katherine F. Moore, and Dan Suciu. The complexity of causality and responsibility for query answers and non-answers. *Proc. of VLDB*, 4(1), 2010.

[Milch *et al.*, 2005] Brian Milch, Bhaskara Marthi, Stuart J. Russell, David Sontag, Daniel L. Ong, and Andrey Kolobov. BLOG: Probabilistic models with unknown objects. In *Proc. of IJCAI*, pages 1352–1359. Professional Book Center, 2005.

[Mitchell *et al.*, 2015] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling. Never-Ending Learning. In *Proc. of AAAI*, pages 2302–2310. AAAI Press, 2015.

[Niu *et al.*, 2011] Feng Niu, Christopher Ré, AnHai Doan, and Jude W. Shavlik. Tuffy: Scaling up statistical inference in Markov logic networks using an RDBMS. *Proc. of VLDB*, 4(6):373–384, 2011.

[Olteanu and Schleich, 2016] Dan Olteanu and Maximilian Schleich. Factorized databases. *SIGMOD Rec.*, 45(2):5–16, 2016.

[Park and Darwiche, 2004] James D. Park and Adnan Darwiche. A differential semantics for jointree algorithms. *AIJ*, 156(2):197–216, 2004.

[Pearl, 1988] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems*. 1988.

[Peters *et al.*, 2014] Shanan E. Peters, Ce Zhang, Miron Livny, and Christopher Ré. A machine reading system for assembling synthetic paleontological databases. *PLoS ONE*, 9(12), 2014.

[Poggi *et al.*, 2008] Antonella Poggi, Domenico Lembo, Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Linking data to ontologies. *J. Data Sem.*, 10:133–173, 2008.

[Poole, 1993] David Poole. Probabilistic Horn abduction and Bayesian networks. *AIJ*, 64(1):81–129, 1993.

[Poole, 1997] David Poole. The independent choice logic for modelling multiple agents under uncertainty. *AIJ*, 94(1/2):7–56, 1997.

[Poon and Domingos, 2006] H Poon and Pedro Domingos. Sound and efficient inference with probabilistic and deterministic dependencies. In *Proc. of AAAI*, pages 458–463, 2006.

[Reiter, 1980] Raymond Reiter. A logic for default reasoning. *AIJ*, 13(1):81–132, 1980.

[Richardson and Domingos, 2006] Matthew Richardson and Pedro Domingos. Markov logic networks. *ML*, 62(1):107–136, 2006.

[Sato and Kameya, 1997] Taisuke Sato and Yoshitaka Kameya. PRISM: A language for symbolic-statistical modeling. In *Proc. of IJCAI*, pages 1330–1335, 1997.

[Singla and Domingos, 2007] Parag Singla and Pedro Domingos. Markov logic in infinite domains. In *Proc. of UAI*, pages 368–375, 2007.

[Suciu *et al.*, 2011] Dan Suciu, Dan Olteanu, Christopher Ré, and Christoph Koch. *Probabilistic Databases*, volume 3 of *Synthesis Lectures on Data Management*. Morgan & Claypool, 2011.

[Van den Broeck and Suciu, 2017] Guy Van den Broeck and Dan Suciu. Query processing on probabilistic data: A survey. *Foundations and Trends® in Databases*, 7(3/4):197–341, 2017.

[Weikum *et al.*, 2016] Gerhard Weikum, Johannes Hoffart, and Fabian Suchanek. Ten years of knowledge harvesting: Lessons and challenges. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 39(3):41–50, 2016.

[Wu *et al.*, 2012] Wentao Wu, Hongsong Li, Haixun Wang, and Kenny Q. Zhu. Probase: A probabilistic taxonomy for text understanding. In *Proc. of ACM SIGMOD*, pages 481–492, 2012.