Taming Data Explosion in Probabilistic Information Integration

Ander de Keijzer, Maurice van Keulen, and Yiping Li

Faculty of EEMCS, University of Twente
POBox 217, 7500AE Enschede, The Netherlands
{a.dekeijzer,m.vankeulen,liy}@ewi.utwente.nl

Abstract. Data integration has been a challenging problem for decades. In autonomous data integration, i.e., without a user to solve semantic uncertainty and conflicts between data sources, it even becomes a serious bottleneck. A probabilistic approach seems promising as it does not require extensive semantic annotations nor user interaction at integration time. It simply teaches the application how to generically cope with uncertainty. Unfortunately, without any world knowledge, uncertainty abounds as almost everything becomes (theoretically) possible and maintaining all possibilities produces huge volumes of data. In this paper, we claim that simple and generic knowledge rules are sufficient to drastically reduce uncertainty, hence tame data explosion to a manageable size.

1 Introduction

Information integration largely remains a labor-intensive manual task. At best, tools assist users in the integration with suggestions of matching data items and attributes, or with performing schema and data conversions based on given rules. The need for human interaction is illustrated by the data integration challenges given by [Lev99]: (1) overlapping and contradictory data, (2) semantic mismatches among sources, and (3) different naming conventions for data values These challenges require a human's world knowledge to make concrete decisions. Only *exact* decisions can unambiguously determine the resulting data items. Even AI techniques cannot make such decisions with certainty.

Our work focuses on *autonomous* information integration. In applications like ambient intelligence, where devices have their own databases and network connectivity is ad hoc, devices need to exchange and integrate information whenever the opportunity arises and without human interaction. Hence, we approach information integration differently: any decision that needs world knowledge, is not resolved, but all possible outcomes are stored with an associated probability.

Unfortunately, without any kind of world knowledge, huge information sources would be produced in this way. This is due to the fact that many things, however remotely possible, are indeed *in principle* possible. In [KKA05], we calculated that for two information sources with each five data items, there are in theory 1546 possibilities how these may combine. In this paper, we show that this data explosion can be greatly reduced by using simple and generic knowledge rules.

The paper is organized as follows. First, we position our work among related research and summarize our probabilistic XML integration approach. Section 4 subsequently examines a movie information integration scenario. Section 5 introduces simple knowledge rules and attempts to quantify their effect.

2 Related Work

For a survey on information integration, we refer to [DH05]. We distinguish between schema and data integration and focus on the latter. We deal with the aforementioned data integration challenges by explicitly handling the inherent uncertainties using a probabilistic database approach. Suciu's SIGMOD'05 tutorial comes with an extensive bibliography on the topic of probabilistic data management [SD05]. Originally, work concentrated on relational databases, but in [KKA05] we argue that XML expresses uncertainty in a more natural way. Other probabilistic XML databases are, for example, PXML [HGS03] and ProTDB [NJ02]. Many results from the logic programming and artificial intelligence communities carry over to our probabilistic XML approach.

Schema matching techniques [RB01] can often be adapted and applied to probabilistic databases. For example, duplicate detection, matching and classification techniques can be used to find and assign probabilities to different representations of the same real-world object (rwo).

Finally, an important source of schema and data integration techniques can be drawn from the Semantic Web community. Approaches mostly attempt to sufficiently annotate data with meaning and world knowledge. We approach data integration from the other end: our approach is independent of any world knowledge, but adding some can be used to restrict uncertainty. We believe this to be a more practical approach than to always require enough annotation to take away all uncertainty. Moreover, in this paper we claim that only simple and generic world knowledge statements suffice.

3 Information Integration using Probabilistic XML

In an ordinary XML document, all information is certain. When XML information sources contain data on the same rwos, conflicts may occur. Consider, e.g., two address books: one claims that a person's name is 'John' while the other claims it is 'Jon'. Therefore, after data integration, there may exist more than one possibility for a certain text node, or in general, for entire subtrees. We model this uncertainty in a probabilistic XML tree with three kinds of nodes: (1) probability nodes ($\bigtriangledown$), (2) possibility nodes ($\circ$), which have an associated probability, and (3) ordinary XML nodes ($\bullet$). The children of a probability node enumerate all possibilities. Figure 1 shows a probabilistic XML tree illustrating uncertainty about the name of a person.

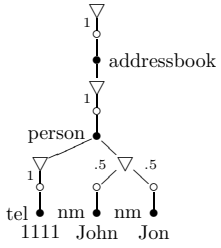

Fig. 1. Example probabilistic XML tree.

A probabilistic XML tree can be seen as a device's knowledge about the 'real world'. The probabilistic XML tree of Figure 1 says that in the real world, there exists with certainty one person with telephone number "1111" and named either "John" or "Jon". A possible database instantiation is called *possible world*. The answer to a query on a probabilistic XML tree can be determined by executing the query on each possible world separately.

As an example of data integration, consider two address books containing the addresses of a "John_1" and "Rita_1", and a "Jon_2" and "Rita_2", respectively. Without world knowledge, even "John_1" may refer to the same rwo as "Rita_2" resulting in an integration result with 7 possible worlds (see Table 1).

Person 1	Person 2	Person 3	Person 4
John_1	Rita_1	Jon_2	Rita_2
John_1=Jon_2	Rita_1	Rita_2	
John_1=Rita_2	Rita_1	Jon_2	
Rita_1=Jon_2	John_1	Rita_2	
Rita_1=Rita_2	John_1	Jon_2	
John_1=Rita_2	Jon_2=Rita_1		
John_1=Jon_2	Rita_1=Rita_2		

Table 1. Possible worlds

In [KKA05], we give a formalization of notions like probabilistic XML tree, probabilistic information integration, and related properties.

4 Movie database scenario

We investigate data explosion in a scenario in which we integrate four data sources on the web containing movie information (see Table 2). We show that simple and generic world knowledge statements can greatly reduce the number of possibilities without negative effects on querying. More details can be found in [KKL06].

Source	#movies
Internet Movie Database (`http://www.imdb.com`)	470,000
All Movie Guide (`http://www.allmovie.com`)	290,000
Yahoo! movies (`http://movies.yahoo.com`)	*unknown*
Simply Scripts (`http://www.simplyscripts.com`)	1,500

Table 2. Some movie sources

Attribute	Comparison
Title/Year	Exactly equal in all three sources.
Genre	IMDb: Action, Adventure, Drama, Fantasy, Sci-Fi, Thriller. AMG: Adventure, Monster Film, Period Film. Yahoo: Action/Adventure, Romance, Thriller, Remake.
Cast	IMDb 15 people, AMG 11 (all also in IMDb and Yahoo), and Yahoo 13 (extra 2 are different from the 4 extra of IMDb). 3 differences in spelling.
Location	IMDb: "New Zealand / USA". AMG: "New Zealand". Yahoo: "Wellington, New Zealand (Campertown Studios - Stone Street Studios)".
Plot summary	All three sources have a different description or plot summary.

Table 3. Comparison of information on the 2005 movie "King Kong".

The main cause for data explosion is the *semantic equality problem*: How to decide whether or not two data items refer to the same rwo? Any movie element, however remotely possible, may *in theory* be semantically equal to any other movie element from another source. The existence of keys or key-like attributes can almost completely avoid this problem. For movies we found two candidates: Semantic equality of nearly all movies can be established with the IMDb-number or the combination of title and year. Note that probabilistic integration only calls for safely ruling out possibilities, hence does not require a perfect key. For example, the movie title alone would reduce the number of possible matches for IMDb's "King Kong/2005" with AMG from 290,000 to 3.

Other causes for data explosion are differences in attribute existence and values. However, these uncertainties are local for an attribute and storage overhead is expected to be small using the compact representation of [KKA05]. Querying the resulting integrated source is not expected to suffer significantly from the incurred uncertainty. Items can still be found, some items may only have a reduced probability. Table 3 illustrates these conclusions for one movie. For genre, cast and transcript, we use a generic rule for lists of text nodes: no semantic equality if two strings sufficiently mismatch. For example, integrating IMDb's and Yahoo's genre attributes results in only three uncertainties: 'Action/Adventure' is the same as 'Action' or 'Adventure', or is an entirely different genre. For location and plot summary, we use a generic cardinality rule: the schema requires only one value, hence a different value between sources means another possibility. Observe that a query asking for movies filmed in New Zealand containing a predicate like `location='New Zealand'`, *will find the movie* "King Kong" in the integrated source.

5 Simple Knowledge Rules

The framework of [KKA05] is independent of any world knowledge for integration of information sources. In theory, any element from one information source may refer to the same rwo as any element of another. Hence theoretically, the number of possibilities in the resulting information source is huge. To more concretely quantify the effects of simple knowledge rules on the number of possibilities, we conducted experiments on the two example data sources of [KKA05] which contained four and two addresses, respectively (see Figure 2 for the DTD of both sources).

```
<! DOCTYPE persons [
  <! ELEMENT persons (person*) >
  <! ELEMENT person (firstname, lastname,
                     phone, room)>
  <! ELEMENT firstname (#PCDATA) >
  <! ELEMENT lastname (#PCDATA) >
  <! ELEMENT phone (#PCDATA) >
  <! ELEMENT room (#PCDATA) >
]>
```

Fig. 2. DTD of example sources

We defined several simple knowledge rules that are based on *numbers* of attributes being equal between elements and on *key-like* attributes. Some of the knowledges rules and resulting number of possible worlds can be found in Table 4. The simplest of the knowledge rules, the single element rule, reduced the number of possible worlds from 1546 to 39 (±97.5%). The actual knowledge introduced is very safe and minimal: if two data items do not agree on any attribute, we decide that they do not refer to the same real-world object. Further reductions to 15 or even 3 possible worlds can be obtained.

Name	**Rule**	**#pw**
Ignorance	No world knowledge, i.e., any two elements may refer to the same rwo	1546
Single element	Elements do not refer to the same rwo, if *none of the children* have the same value	39
50%	Elements do not refer to the same rwo, if *less than 50% of the children* have the same value	15
Firstname	The firstname attribute is considered a key, i.e., elements do not refer to the same rwo, if the firstnames disagree	15
Lastname	Analogously for lastname	3
Combination 1	50% and firstname rule	15
Combination 2	50% and lastname rule	3
Combination 3	firstname and lastname rule	3

Table 4. Knowledge rules and resulting number of possible worlds (#pw)

We should, however, avoid adding world knowledge that does not hold in general. For example, if document 1 would have had the data item 'John Kingship / phone=4030 / room=3035', it is actually very likely that this data item does *not* refer to the same rwo as 'Allen Kingship / phone=2020 / room=3035'. The 50% rule is in this case not a good knowledge rule, because it rules out possibilities that are likely to be true. Good knowledge rules for probabilistic integration are safe rules that have little or no false positives.

6 Conclusion and future work

In this paper we have shown that data explosion in probabilistic information integration can be reduced drastically by introducing safe, simple and generic knowledge rules. In the movie database scenario, we looked at some real-life data to be able to investigate the uncertainty occurring in practical information integration. We showed that although much conflicting information can be found, there is enough solid ground. It is expected that the remaining uncertainty need not be resolved to be able to effectively answer the usual queries.

Although probabilistic information integration can function without user interaction at integration time, user interaction may still be beneficial. A user could indicate that certain possibilities are nonsense. In such a case, those possibilities can be eliminated from the source. As future research, we will investigate if user statements about a *query result* can be used to reduce uncertainty.

References

[DH05] A. Doan and A. Halevy. Semantic integration research in the database community: Brief survey. *AI Magazine, Sp.Issue on Semantic Integration*, 2005.

[HGS03] E. Hung, L. Getoor, and V.S. Subrahmanian. PXML: A probabilistic semistructured data model and algebra. In *Proc. ICDE Conf., Bangalore, India*, pages 467–, Mar. 2003.

[KKA05] M. van Keulen, A. de Keijzer, and W. Alink. A probabilistic xml approach to data integration. In *Proc. ICDE Conf., Tokyo, Japan*, pages 459–470, 2005.

[KKL06] A. de Keijzer, M. van Keulen, and Y. Li. Taming data explosion in probabilistic information integration. Technical Report ????, Centre for Telematics and Information Technology, Enschede, The Netherlands, 2006.

[Lev99] A.Y. Levy. Combining artificial intelligence and databases for data integration. In *Artificial Intelligence Today*, LNCS 1600, pages 249–268. 1999.

[NJ02] A. Nierman and H.V. Jagadish. ProTDB: Probabilistic data in XML. In *Proc. VLDB Conf., Hong Kong, China*, 2002.

[RB01] E. Rahm and P.A. Bernstein. A survey of approaches to automatic schema matching. *VLDB Journal*, 10(4):334–350, Dec. 2001.

[SD05] D. Suciu and N.N. Dalvi. Foundations of probabilistic answers to queries. In *Proc. SIGMOD Conf.*, page 963, 2005. Bibliographic notes to this tutorial at `http://www.cs.washington.edu/homes/suciu/tutorial-sigmod2005-bib.pdf`.