Issues on the Implementation of General Imaging on Top of Probabilistic Datalog

Fabio Crestani
University of Padova
Italy

Thomas Rölleke
University of Dortmund
Germany

Abstract

In 1986 Van Rijsbergen proposed in [11] to interpret Information Retrieval as the process of selecting, for a query q, those documents d_i for which the logical formula $d_i \rightarrow q$ (d_i implies q) holds. The desired ranking of the documents according to their relevance to the query is obtained by computing the probabilities $P(d_i \rightarrow q)$.

Logical Imaging provides a way of evaluating the probability of a logical implication in the context of the Possible World semantics. In 1993 a model of IR based on Logical Imaging was proposed by Crestani and Van Rijsbergen [1]. Later they proposed an extension of this model using a generalised form of Logical Imaging [2].

Probabilistic Datalog, a probabilistic extension of the Datalog logical model of databases proposed by Fuhr in 1994 [3], enables the modelling of Information Retrieval as uncertain inference. The expressiveness of Probabilistic Datalog is such that it enables modelling both new models of hypermedia retrieval and classical probabilistic models of Information Retrieval.

In this paper we report on some results [9] and some open issues regarding the implementation of the General Imaging model on top of Probabilistic Datalog.

1 Information Retrieval by General Logical Imaging

Logical Imaging is a process developed in the framework of Modal Logic that enables the evaluation of a conditional sentence without explicitly defining the operator "$\to$" [10]. Imaging has been extended to the case where there is a probability distribution on the worlds by Lewis [7]. In this case the evaluation of $P(y \to x)$ causes a shift of the original probability P from a world w to the closest world w_y where y is true. Probability is neither created nor destroyed, it is simply moved from a "not-y-world" to a "y-world" to derive a new probability distribution P_y. This process is called "deriving P_y from P by imaging on y".

General Logical Imaging originated from an attempt to overcome one of the restrictive assumptions Lewis made for Stalnaker's semantics of conditionals [7]. The assumption is related to the "uniqueness" of the world w_y, that is the uniqueness of the world most similar to w where y is true. In [5] p. 110, Gärdenfors propose a generalisation of the Imaging process that does not rely on this assumption. The starting point of the generalisation is the use of a probability function to represent the fact that in any possible world w a proposition y is either true or false. In the case $P^w(y) = 1$ if y is true in w, and $P^w(y) = 0$ if y is false in w we go back to the classical definition of Imaging, in any other case, with $0 < P^w(y) < 1$, we state that y is only partially true in w. Lewis called such probability function "opinionated" because "it would represent the beliefs of someone who was absolutely certain that the world w was actual and who therefore held a firm opinion about every question" (see [7], p. 145).

Retrieval by General Logical Imaging (GLI) can be regarded a the process of applying General Imaging on d in order to evaluate the probability that a document d implies the query q. The focus point of the GLI model, proposed by Crestani and Van Rijbergen in [2], is the consideration that "an index term is a world". The following is a simplified formulation of the model:

$$P(d \to q) \quad = \quad \sum_t \sum_{t'} P_d^{t'}(t) * \mu(t') * \tau(t, q).$$

where: $P_d^{t'}(t)$ is the opinionated probability of term t' in t in relation to document d; $\tau(t, q)$ is a function with the values 1 if the term t is present in the query q, and 0 if the term t is not present; and $\mu(t')$ is the probability assigned to each term in the term space T[1].

The application of the above technique to IR requires an appropriate measure of similarity over the term space T to enable the identification of the set of closest terms to t where d is true, and an opinionated probability function determining the portion of the probability associated to a term t that need to be transfer to a term t'. An example of a very simple opinionated probability function is reported in Table 1.

[1] It should be noticed that the nature of the probability function $P(x)$ of $P(d \to q)$ is different from that of the probability function $\mu(x)$ used to assign the probability to index terms.

t_i	$P_d^{t_1}(t_i)$	$P_d^{t_2}(t_i)$	$P_d^{t_3}(t_i)$	$P_d^{t_4}(t_i)$	$P_d^{t_5}(t_i)$	$P_d^{t_6}(t_i)$
t_1	1.0	0.67	0	0.33	0	0
t_2	0	0	0	0	0	0
t_3	0	0	0	0	0	0
t_4	0	0	0	0	0	0
t_5	0	0	0.67	0.67	1.0	0
t_6	0	0.33	0.33	0	0	1.0

Table 1: An example of an opinionated probability function

t_i	$\mu(t_i)$	$\tau(t_i, d_1)$	t_d	$\mu_{d_1}(t_i)$	$\tau(t_i, q)$	$\mu_{d_1}(t_i) * \tau(t_i, q)$
t_1	0.2	1	t_1	0.333	1	0.333
t_2	0.1	0	t_1, t_6	0	0	0
t_3	0.05	0	t_5, t_6	0	0	0
t_4	0.2	0	t_5, t_1	0	1	0
t_5	0.3	1	t_5	0.467	0	0
t_6	0.15	1	t_6	0.2	1	0.2
$\sum_{t_i}$	1.0			1.0		0.533

Table 2: An example of the evaluation of $P(d_1 \rightarrow q)$ by general imaging on d

Table 2 reports an example of the evaluation of $P(d_1 \rightarrow q)$ by general imaging on d. A graphical interpretation of this process is depicted in Figure 1.

2 Probabilistic Datalog

Probabilistic Datalog (see [3]) is an extension of stratified Datalog (see [6]). The basic ideas of probabilistic Datalog are the assignment of probabilistic weights to facts and the computation of the weights of derived facts by means of intensional semantics.

Consider the following two uncertain facts:

```
0.7  docTerm(1,ir).
0.8  docTerm(1,db).
```

The first uncertain fact assigns the probability 0.7 to the proposition "document 1 provides information about the term ir". It is straight forward to regard `docTerm` as a probabilistic relation { (0.7 (1,ir)), (0.8 (1,db)) }. To refer to the probability of a fact we write $\omega($`docTerm(1,ir)`$) = 0.7$.

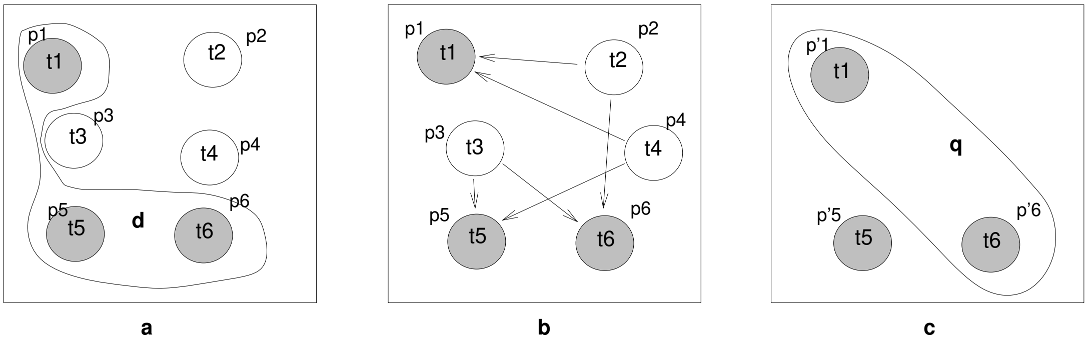

Figure 1: Graphical interpretation of the evaluation of $P(d_1 \rightarrow q)$ by general imaging on d.

Given this little program, we may formulate a rule for deriving documents about both ir and db:

```
q1(D) :- docTerm(D,ir) & docTerm(D,db).
```

Assuming the tuples of `docTerm` to be stochastically independent, we get the relation q1 = { (0.56 (1)) } where $\omega(\texttt{q1(1)}) = 0.56$. Using rules enables the formulation of any conjunction and disjunction of predicates.

Now we extend our program with further facts and rules expressing the link structure of a hypertext document.

```
0.5  link(2,1).
     about(D,T)  :-  docTerm(D,T).
     about(D,T)  :-  link(D,D1) & about(D1,T).
```

The predicate `about` indicates the supposition that a document D is about term T, if it refers to a document that is about T.

We rewrite `q1` and get

```
q2(D) :- about(D,ir) & about(D,db).
```

An evaluation based on extensional semantics would yield $\omega(\mathrm{q2}(2)) = 0.5 * 0.7 * 0.5 * 0.8 = 0.14$.

From a probabilistic point of view this weight is not correct, since the fact `link(2,1)` is considered twice.

Instead of just computing the weight of a derived fact as a function of the rule weight and the weight produced by evaluating the rule body, probabilistic Datalog uses so-called event expressions for computing the resulting weight of a fact. Thus we achieve an intensional semantics of the fact weights. The central point is the extension of the relations with a special attribute for event expressions (see [4]). The event expression of a fact p is denoted by $\eta(p)$. The weight of a fact is computed via the formula $\omega(p) = P(\eta(p))$ where P is a probability distribution on events. Thus we get the probabilistically correct weight of `q2(2)`:

$$\begin{aligned}
\omega(\texttt{q2(2)}) &= P(\eta(\texttt{q2(2)})) \\
&= P(\eta(\texttt{about(2,ir) \& about(2,db)})) \\
&= P(\eta(\texttt{about(2,ir)}) \wedge \eta(\texttt{about(2,db)})) \\
&= P(\eta(\texttt{link(2,1) \& docTerm(1,ir)}) \wedge \\
&\qquad \eta(\texttt{link(2,1) \& docTerm(2,db)})) \\
&= P(\texttt{link(2,1)} \wedge \texttt{docTerm(1,ir)} \wedge \\
&\qquad \texttt{link(2,1)} \wedge \texttt{docTerm(2,db)}) \\
&= 0.5 * 0.7 * 0.8 = 0.28.
\end{aligned}$$

The example illustrates the computation of the event expression. Since P is a function on event expressions, it detects the double occurrence of the event `link(2,1)` and computes the correct probability.

To explain what we mean by the probability of a fact, we use possible worlds semantics (see [8]).

The probability of a fact $\varphi(x)$ is computed by summing the probabilities of those possible worlds, where the fact is true.

$$\omega(\varphi(x)) = \sum_{w_i \in \mathcal{W}} \mu(w_i) * \begin{cases} 1, & \text{if } \varphi(x) \text{ is true at world } w_i \\ 0, & \text{if } \varphi(x) \text{ is not true at world } w_i \end{cases}$$

Probabilistic Datalog can also cope with a special case of dependent events, namely disjoint events. For example, the syntactical element `!pred(dk,_,_)` is used for declaring the disjointness of all facts `pred(X,Y,Z)` which share the same value of the attribute X. The attribute set defined by the disjointness clause is called disjointness key.

For an elaborated description of the complete syntax and semantics of probabilistic Datalog and the evaluation process refer to [3] and [4].

3 Modelling General Imaging using Probabilistic Datalog

Table 2 shows an example of computing the probability $P(d_1 \rightarrow q) = 0.533$. Assuming a document collection $D = d_1, ..., d_n$ we can derive a term vector for each document which represents the occurrence of a term within a document ($\tau(t_i, d_i)$). Given the probability distribution $\mu(t_i)$, a term t_i is regarded as a possible world. A document is regarded as a proposition which is true at a possible world, if the term occurs within the document. Column t_d assigns to each world where d_1 is not true the nearest worlds where d_1 is true. The probabilities of the non-d_1-worlds are transferred to the nearest d_1-worlds in column $\mu_{d_1}(t_i)$. For example, the probability of world t_2 is transferred to the worlds t_1 and t_6. The opinionated probability function $P_{d_1}^{t'}(t_i)$ transfers a portion of $\mu(t')$ to the world t_i. Considering the example given in table 1 the nearest world gets 2/3 and the second nearest gets 1/3 of the probability. The resulting probability $P(d_1 \rightarrow q) = 0.533$ is computed as the sum of the (new) probabilities of the d_1-worlds where q is true.

As an example, consider the computation of $\mu_{d_1}(t_1)$:

$$\begin{aligned}
\mu_{d_1}(t_1) &= \mu(t_1) * P_{d_1}^{t_1}(t_1) + \mu(t_2) * P_{d_1}^{t_2}(t_1) + \mu(t_4) * P_{d_1}^{t_4}(t_1) \\
&= 0.2 * 1.0 + 0.1 * 0.67 + 0.2 * 0.33 = 0.333
\end{aligned}$$

Now we are going to implement the above information on term probability, term occurrence and opinionated probability function in probabilistic Datalog.

```
0.2   term(t1).
0.1   term(t2).
0.05  term(t3).
0.2   term(t4).
0.3   term(t5).
0.15  term(t6).
      !term(_).
      docTerm(d1,t1).
      docTerm(d1,t5).
      docTerm(d1,t6).

0.67  transfer(d1,t2,t1).
0.33  transfer(d1,t2,t6).
0.67  transfer(d1,t3,t5).
0.33  transfer(d1,t3,t6).
0.67  transfer(d1,t4,t5).
0.33  transfer(d1,t4,t1).
      !transfer(dk,dk,_).

about(D,T) :- docTerm(D,T) & term(T).
about(D,T) :- transfer(D,T',T) & term(T').
```

Figure 2: A probabilistic Datalog program

This program defines the probabilities of the terms t_i (relation `term`) and the occurrence of the terms in a document (relation `docTerm`). The facts of `term` are declared to be disjoint (`!term(_)`). This corresponds to the disjointness of possible worlds concerning the Imaging model. For the semantics of probabilistic Datalog, disjointness means that there exists no world where more than one fact of the relation `term` is true. The relation `transfer` is used to determine the portion of the probability of a term t_i which does not occur in a document to be transferred to a term which does occur. Thus this relation is used to express the shifting of the probabilities of non-y-worlds to the y-worlds modelling the opinionated probability function. The facts of `transfer` are disjoint with respect to the same disjointness key. The clause (`transfer(dk,dk,_)`) indicates that the first and second attribute form the disjointness key.

The query

$$\texttt{?- about(}d_1\texttt{,}t_1\texttt{).}$$

yields (0.333 ()) as answer, since

$$\begin{aligned}
\omega(\texttt{about(}d_1\texttt{,}t_1\texttt{)}) &= P(\eta(\texttt{about(}d_1\texttt{,}t_1\texttt{)})) \\
&= P(\texttt{docTerm(}d_1\texttt{,}t_1\texttt{)} \wedge \texttt{term(}t_1\texttt{)} \vee \\
&\quad \texttt{transfer(}d_1\texttt{,}t_2\texttt{,}t_1\texttt{)} \wedge \texttt{term(}t_2\texttt{)} \vee \\
&\quad \texttt{transfer(}d_1\texttt{,}t_4\texttt{,}t_1\texttt{)} \wedge \texttt{term(}t_4\texttt{)}) \\
&= 1.0 * 0.2 + 0.67 * 0.1 + 0.33 * 0.2 = 0.333
\end{aligned}$$

We formulate the whole query for all documents about t_1, t_4, and t_6 as

$$\begin{array}{lll}
\texttt{q(D)} & \texttt{:-} & \texttt{about(D,}t_1\texttt{).} \\
\texttt{q(D)} & \texttt{:-} & \texttt{about(D,}t_4\texttt{).} \\
\texttt{q(D)} & \texttt{:-} & \texttt{about(D,}t_6\texttt{).} \\
 & \texttt{?-} & \texttt{q(D).}
\end{array}$$

The result is (0.533 (d_1)).

4 Implementing General Imaging on top of Probabilistic Datalog

In order to implement General Imaging on top of Probabilistic Datalog we need to produce the necessary Probabilistic Datalog facts. To do so we need:

1. a “prior” probability distribution over the term space;

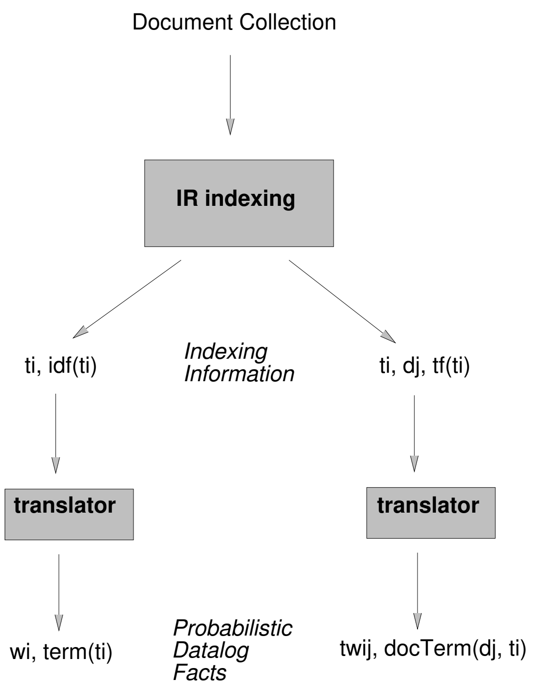

Figure 3: First phase of the construction of Probabilistic Datalog facts from IR indexing.

2. a measure of similarity between terms;
3. an opinionated probability function to direct and weight the transfer of probability between terms.

Once we have decided on these requirements, see for example in [2] for an experimental setting, we need to set up one or more processes that contruct the Probabilistic Datalog facts.

Figure 3 shows that in the first phase we may use information on inverse document frequency and term frequence to build the probabilistic relations `term` and `docTerm` as described in figure 2.

How can we now determine the weights of the transfer facts which model the opinionated function? The portion is computed by the formula

$$\omega(\texttt{transfer}(d, t_i, t_j)) = \frac{sim(t_i, t_j)}{\sum_{t|\tau(t,d)=1} sim(t_i, t)} \tag{1}$$

The portion depends both on the similarity of the terms and on the set of terms occuring in the document. Figure 4 depicts the building of the transfer facts with the corresponding weights.

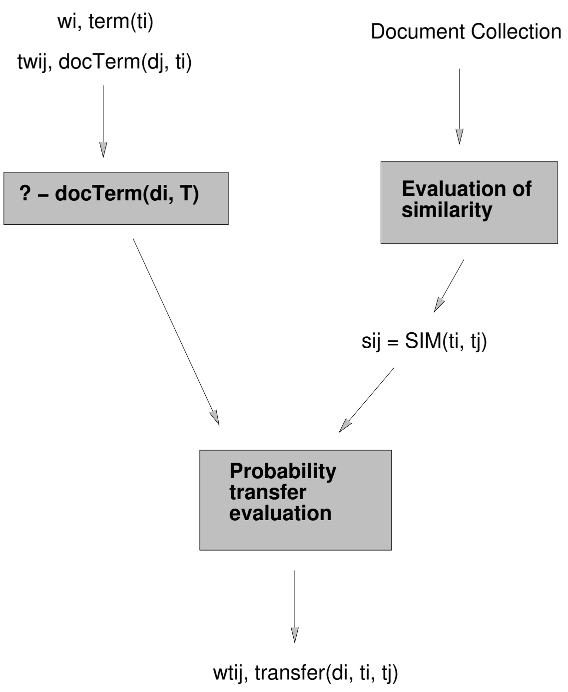

Figure 4: Second phase of the construction of Probabilistic Datalog facts from a index term similarity matrix.

We use the result of phase 1 to determine the set of terms that occur in a document. Given a similarity measure we can then compute the weight of the transfer facts as defined by equation 1.

5 Conclusions and Future Work

From the use of Probabilistic Datalog as an implementation platform for General Imaging we gain the possibility to combine the probability kinematics defined by General Imaging on terms with other probabilistic knowledge. The expressiveness of the knowledge representation and query language is increased, because typical IR knowledge and queries may be combined with typical DB knowledge and queries.

The 2-phases approach presented above allows for implementing any kind of probability function, since the computation of the weight is done externally. A subset of the possible transfer functions could be computed using Probabilistic Datalog itself. But for the chosen function given in equation 1 we have not found a solution to compute it internally.

From an implementation point of view we are going to investigate if the proposed implementation is feasible with respect to the amount of data generated by indexing a large data collection.

References

[1] F. Crestani and C.J. van Rijsbergen. Information Retrieval by Logical Imaging. *Journal of Documentation*, 51(1):1–15, 1995.

[2] F. Crestani and C.J. van Rijsbergen. Probability kinematics in information retrieval. In *Proceedings of ACM SIGIR*, pages 291–299, Seattle, WA, USA, July 1995.

[3] N. Fuhr. Probabilistic datalog - a logic for powerful retrieval methods. In *Proceedings of ACM SIGIR*, pages 282–290, Seattle, WA, USA, July 1995.

[4] N. Fuhr and T. Rölleke. A probabilistic relational algebra for the integration of information retrieval and database systems. Submitted for publication, 1995.

[5] P. Gärdenfors. *Knowledge in flux: modelling the dynamics of epistemic states*. The MIT Press, Cambridge, Massachusetts, USA, 1988.

[6] J. Hullman. *Principles of Database and Knowledge-Base Systems*, volume I. Computer Science Press, Rockville, MD, USA, 1988.

[7] D. Lewis. Probability of conditionals and conditionals probabilities. In W.L. Harper, R. Stalnaker, and G. Pearce, editors, *Ifs*, The University of Western Ontario Series in Philosophy of Science, pages 129–147. D.Reidel Publishing Company, Dordrecht, Holland, 1981.

[8] N.J. Nilsson. Probabilistic logic. *Artificial Intelligence*, 28:71–87, 1986.

[9] T. Rölleke. Imaging on top of probabilistic datalog. In *Proceedings of ACM SIGIR*, Seattle, WA, USA, July 1995. Poster session.

[10] R. Stalnaker. Probability and conditionals. In W.L. Harper, R. Stalnaker, and G. Pearce, editors, *Ifs*, The University of Western Ontario Series in Philosophy of Science, pages 107–128. D.Riedel Publishing Company, Dordrecht, Holland, 1981.

[11] C.J. van Rijsbergen. A non-classical logic for Information Retrieval. *The Computer Journal*, 29(6):481–485, 1986.