Valid or Complete Information in Databases —A Possibility Theory-Based Analysis—

Didier DUBOIS Henri PRADE

Institut de Recherche en Informatique de Toulouse (IRIT) - CNRS
Université Paul Sabatier, 118 route de Narbonne
31062 Toulouse Cedex 4 – France
Email: Henri.Prade@irit.fr

Abstract

The validity of the information stored in a database may be guaranteed or not, according to the family of items which is considered. The information available in the database may be complete or not, as well, for a given type of items. The paper discusses how these forms of uncertainty can be represented in the framework of possibility theory, and how queries to a database where information is neither necessarily complete and nor valid, can be handled using possibilistic logic. One benefit of the possibilistic modelling is to allow for the use of graded levels of validity, and of graded levels of certainty that the information is complete.

Key-Words: Valid information; (in)complete information; uncertainty; possibility theory; possibilistic logic.

1 - Introduction

The validity of the information stored in a database can be often asserted, but not always, depending on the reliability of the sources which are feeding the database. Dually, the information available on a given topic may be known as complete in the sense that all the true information pertaining to this topic is stored in the base, i.e., no missing items can be true; but often the completeness of the information cannot be asserted.

Demolombe (1996a) gives the following illustrative example:

> "Let us take an example where a database contains information about flights which is represented by the relation schema: F(#Flight, Departure-city, Arrival-city, Company, Day). Assume that the information about validity is: *all the tuples in the relation F corresponding to flights whose departure city or arrival city is Paris, represent true facts of the world*, and the information about completeness is *all the true facts of the world corresponding to flights whose company is Air France are represented by a tuple in the relation F*."

As pointed out, already a long time ago, in the database literature (e.g., Motro, 1986, 1989), it might be desirable to inform the user about the validity and completeness of the retrieved information, when answering a given query. Thus, considering the previous example, Demolombe (1996a) writes

> "Now, if one asks the standard query: *what are the flights from Paris to London?*, the answer to the corresponding validity query is: *all the tuples in the answer are valid, and the answer to the completeness query is the answer is complete for all the tuples where the company is Air France.*"

Several modelling frameworks have been proposed for handling various types of uncertainty in databases; see (Motro and Smets, 1997) for introductions to these different approaches. Possibility theory (Dubois and Prade, 1988) has been shown of particular interest for modelling partially-known attribute values (whose precise values are pervaded with imprecision and uncertainty), as well as standard null values (e.g., Bosc and Prade, 1997). Moreover, the modelling of uncertainty provided by possibility theory can remain purely qualitative and is suitable for the representation of partial ignorance. It is thus tempting to investigate how validity and completeness issues could be captured in the framework of possibility theory.

The paper is organized in the following way. Section 2 discusses the representation of validity and completeness information using possibility theory, where validity and certainty of completeness can be easily graded according to the sources providing the pieces of data stored in the base. Section 3 suggests how possibilistic logic can handle queries w.r.t. to validity and completeness information. Section 4 briefly outlines directions for further research.

2 - Representing Validity and Completeness Information

2.1 - Possibility Theory

In possibility theory, the assessment of uncertainty is based on a [0,1]-valued measure of possibility Π and an associated measure of necessity N, such that the following duality relation holds,

$$N(\varphi) = 1 - \Pi(\neg\varphi). \qquad (1)$$

This relation expresses that the more impossible '$\neg\varphi$', the more certain 'φ' is. Besides, N satisfies the min-decomposability property

$$N(\varphi \wedge \psi) = \min(N(\varphi), N(\psi)) \qquad (2)$$

and $N(\bot) = 0$, where $\bot$ denotes the contradiction. Moreover it enables the user to introduce intermediary states between the three basic epistemic states: φ is true $N(\varphi) = \Pi(\varphi) = 1$; φ is false $N(\varphi) = \Pi(\varphi) = 0$ and φ is unknown, i.e., $N(\varphi) = 0$ and $\Pi(\varphi) = 1$. These intermediary states are

- φ is believed, or accepted: $N(\varphi) > 0$ and $\Pi(\varphi) = 1$
  (since min $(N(\varphi), N(\neg\varphi)) = 0$, due to (2)-(3) and (1))
- φ is disbelieved: $\Pi(\varphi) < 1$ and $N(\varphi) = 0$ (since $N(\neg\varphi) > 0 \Rightarrow N(\varphi) = 0$).

A possibility measure Π can be defined from a possibility distribution π on the set of interpretations of the language under consideration. Namely, in a finite setting

$$\Pi(\varphi) = \max_{\omega\models\varphi} \pi(\omega) \text{ and } N(\varphi) = \min_{\omega \not\models \varphi} (1 - \pi(\omega)).$$

The possibility distribution π rank-orders the interpretations according to their plausibility. The use of the real interval [0,1] is not compulsory; any discrete, totally ordered, scale of the form $1 = \alpha_1 > \ldots > \alpha_n = 0$ can be used as well (then the complementation to 1 in (1) is replaced by the order-reversing operation *n* of the scale defined by $n(\alpha_i) = \alpha_{n-i+1}$, $\forall$ i = 1,n).

2.2 - Graded Validity and Completeness

Let s(x) be a statement corresponding to a potential or existing tuple x of the relational database R under consideration (e.g., in the example of the introduction, s(x) pertains to the description of a flight x in terms of the attributes "Flight_number", "Departure_city", "Arrival_city", "Company" and "Day"). Let R(x) denotes the fact the corresponding tuple is stored in the database, i.e., belongs to the relation R.

Then, it is reasonable to admit that what is stored in the database corresponds to accepted beliefs, while what is not stored in the database is somewhat disbelieved. Formally speaking, we have

$$R(x) \Rightarrow N(s(x)) > 0 \quad (3)$$

and

$$\neg R(x) \Rightarrow \Pi(s(x)) < 1 \quad (4)$$

where $N(s(x)) > 0$ (resp. $\Pi(s(x)) < 1$) expresses that the statement s(x) corresponding to x is believed to be true (resp. is somewhat disbelieved, i.e., is believed to be false due to (1)).

Some tuples in the database are asserted as being valid. Let Val be the property describing these tuples, then we have for those tuples x

$$R(x) \wedge Val(x) \Rightarrow N(s(x)) = 1 \quad (5)$$

which expresses that if the tuple appears in the database and belongs to a family of tuples asserted as valid, then it is completely certain that it represents true information.

For other tuples, described as satisfying a property Comp, the information, is guaranteed to be complete; it means that

$$\neg R(x) \wedge Comp(x) \Rightarrow \Pi(s(x)) = 0 \quad (6)$$

i.e., if x is a tuple which satisfies Comp and which does not appear in the database, then it is false since the database is supposed to contain all true tuples which satisfies Comp (e.g., in the example of the introduction, the predicate 'Comp' corresponds to the property of being an Air France flight).

Thus we can distinguish between i) statements which are surely true (since the corresponding information is in the database and is validated), ii) statements which are believed although they might be false (they correspond to pieces of information in the database which are not validated), iii) statements which are disbelieved although they might be true (since the corresponding information is not in the database), and iv) statements which are surely false (since they do not correspond to a piece of information stored in the base and the base is known to have a complete information on the topic under consideration).

In the possibility theory framework, it is easy to grade the levels of certainty, or the levels of possibility. Indeed, (3) and (4) can be modified into

$$R(x) \Rightarrow N(s(x)) > \alpha > 0 \quad (7)$$

and $$\neg R(x) \Rightarrow \Pi(s(x)) < \beta < 1 \qquad (8)$$

for expressing that our level of certainty that a tuple in database corresponds to a true statement, is at least α, and the level of possibility that a tuple x not in the database corresponds to a true statement, is upper bound by β (i.e., the certainty that the statement corresponding to x is false is at least equal to $1 - \beta$).

Besides, the reliability of the stored information depends on the sources which provides it. Thus, (5) can be generalized into

$$R(x) \wedge \mathrm{Val}_i(x) \Rightarrow N(s(x)) \geq \alpha_i > 0, i = 1,k \qquad (9)$$

where the tuples which satisfy the property Val_i are validated by source i whose level of reliability is α_i. It is assumed that the sources can be rank-ordered according to their reliability and that there exists a more reliable source which provides pieces of information which are completely certain, i.e., $\alpha_1 = 1 > \alpha_2 >\ldots > \alpha_k > 0$. For the sake of coherence, it is assumed that $\min_{i=1,k} \alpha_i = \alpha_k = \alpha$ where α appears in (7). Indeed, any information in a database is at least as certain as the certainty level attached to the information provided by the least reliable source which feeds the database.

Similarly for completeness information, (6) is generalized into

$$\neg R(x) \wedge \mathrm{Comp}_j(x) \Rightarrow \Pi(s(x)) \leq \beta_j < 1, j = 1,\ell \qquad (10)$$

where the tuples which satisfy the property Comp_j are assumed to be complete in the database with a certainty level at least equal to $1 - \beta_j$. The β_j's are supposed to be rank-ordered such that $\beta_1 = 0 \leq \beta_2 \leq\ldots \leq \beta_\ell$, with $\max_j \beta_j = \beta_\ell = \beta$ for the sake of coherence with (8).

This enables us to distinguish between sets of pieces of information which are more or less certainly valid, or more or less certainly complete.

3 - Processing Queries in Possibilistic Logic

3.1. - Possibilistic Logic

The reader is referred to Dubois, Lang and Prade (1994a and b) for introductory and detailed presentations respectively. In the following, we only recall some basic points.

A possibilistic logic formula is made of a pair constituted by of a classical logic formula φ and a weight α belonging to a totally ordered scale, e.g., [0,1], or a finite scale. (φ,α) is semantically interpreted as a constraint of the form $N(\varphi) \geq \alpha$ where N is a necessity measure. Then the following resolution rule is in agreement with this semantics:

$$(\varphi,\alpha), (\psi,\beta) \vdash (\text{Resolvent}(\varphi,\psi), \min(\alpha,\beta)).$$

In particular we have the cut rule

$$\begin{array}{c} (\neg\varphi \vee \psi, \alpha) \\ (\varphi \vee \theta, \beta) \\ \hline (\psi \vee \theta, \min(\alpha,\beta)) \end{array}.$$

Refutation can be extended to this framework. Let K be a possibilistic knowledge base made of a set of possibilistic logic formulas put under clausal form (this can be always done). Then proving (φ,α) from K, which can be written symbolically $K \vdash (\varphi,\alpha)$, amounts to prove $(\bot,\alpha)$, where $\bot$ denotes the empty clause, by applying the resolution rule to $K \cup \{(\neg\varphi,1)\}$ repeatedly. Moreover we have, $(\varphi,\alpha') \vdash (\varphi,\alpha)$ iff $\alpha \leq \alpha'$; besides, if $K \vdash (\varphi,\alpha)$ and $K \vdash (\varphi,\alpha')$, then $K \vdash (\varphi, \max(\alpha,\alpha'))$. So we are looking for the refutation which provides the greatest lower bound. This syntactic machinery is sound and complete with respect to a semantics in terms of a possibility distribution encoding an ordering on the interpretations, in agreement with the interpretation of (φ,α) as $N(\varphi) \geq \alpha$.

Let us assume that a possibilistic knowledge base contains the two formulas $(\neg\varphi \vee \psi, \alpha)$, $(\neg\varphi \vee \neg\varphi' \vee \psi, \alpha')$ with $\alpha' > \alpha$. Then if we add the information $(\varphi,1)$, we can infer that (φ,α), while if we have both $(\varphi,1)$ and $(\varphi',1)$, we can infer a more certain conclusion, namely (ψ,α') using the more "specific" clause $(\neg\varphi \vee \neg\varphi' \vee \psi, \alpha')$, since $\alpha' > \alpha$.

Besides, in possibilistic logic it is always possible to move literals from the formula slot to the weight slot. Indeed, it can be shown that there is a semantic equivalence between $(\neg\varphi \vee \psi, \alpha)$ and $(\psi, \min(\alpha, \varphi[\omega]))$ for instance, where $\varphi[\omega]$ denotes the truth-value (1 for 'true', 0 for 'false') of φ for an interpretation ω. It means that saying that $\neg\varphi \vee \psi$ is at least α-certain is semantically equivalent to say that ψ is a-certain provided that φ is true. This remark can be exploited when some literal cannot be eliminated in the resolution process, and more generally in hypothetical reasoning.

3.2 - Application to query evaluation

The machinery briefly recalled in 3.1 can be applied to querying a database w.r.t. to validity and completeness issues. Let W(x) be a predicate expressing that the information reported in tuple x is indeed true in the real world. Then the validity and completeness information can be encoded by the following possibilistic knowledge base.

$$\begin{array}{l} \{(\neg R(x) \vee W(x), \alpha), \\ (\neg R(x) \vee \neg Val_i(x) \vee W(x), \alpha_i) \text{ for } i = 1,k, \\ (R(x) \vee \neg W(x), 1 - \beta) \\ (R(x) \vee \neg Comp_j(x) \vee \neg W(x), 1 - \beta_j) \text{ for } j = 1,\ell\}. \end{array}$$

It expresses that

- if a tuple x appears in the (relational) database (i.e., R(x) holds), then this information is true in the world with certainty α;
- if moreover x satisfies the validity condition Val_i it is certain at degree $\alpha_i \geq \alpha$ that it represents a true information, for i = 1,k; in particular for i = 1, the validity is completely guaranteed ($\alpha_1 = 1$);
- if a tuple x does not appear in the base, it is not true in the world with certainty $1 - \beta$;
- if moreover x satisfies the completeness condition Comp_j, it is certain that it is not a true information with certainty degree $1 - \beta_j \geq 1 - \beta$; for j = 1, it is totally certain that the information x in the base, such that $\mathrm{Comp}_1(x)$ is true, is complete ($\beta_1 = 0$).

Let us go back to the example of Section 1. It writes,

$$\begin{aligned}K = \{ \ & (\neg R(x) \vee W(x), \alpha),\\ & (\neg R(x) \vee \neg \mathrm{Depart}(x, \mathrm{Paris}) \vee W(x), 1),\\ & (\neg R(x) \vee \neg \mathrm{Arriv}(x, \mathrm{Paris}) \vee W(x), 1),\\ & (R(x) \vee \neg W(x), 1 - \beta),\\ & (R(x) \vee \neg \mathrm{Comp}(x, \mathrm{Air\ France}) \vee \neg W(x), 1)\}.\end{aligned}$$

Then let us consider a flight which flies from Paris, i.e., it corresponds to tuples which are described by the possibilistic formula

$$f = \{(\mathrm{Depart}(x, \mathrm{Paris}), 1)\}.$$

If we are asking what is the valid information, i.e., the x's such that W(x) is true, we proceed by refutation, adding the formula

$$(\neg W(x),\ 1)$$

to $K \cup f$. Then by applying the resolution principle repeatedly and moving literals in the weight slot when necessary, we get the empty clause

$$(\bot, R(x)[\omega])$$

which means that we are certain that the information about flight x is valid provided that it is in the base. Note that there is another way to get the empty clause but with a smaller weight $(\bot, \min(\alpha, R(x)[\omega]))$ by using the general rule $(\neg R(x) \vee W(x), \alpha)$ only.

If we now ask what are the tuples in the base for which the information is complete. These tuple x are such as $\neg W(x) \vee R(x)$, i.e., if x is true in the world, it is in the basis. Thus, proceeding by refutation from K, we add

$$(\neg R(x),\ 1)$$

and
$$(W(x),\ 1)$$

to K and we get by resolution

$$(\bot,\ \max(\beta,\ \mathrm{Comp}(x,\ \text{Air France})[x]),$$

i.e., we obtain the Air France flights (and by default, if Comp(x, Air France)[x]) = 0 any information is complete with degree $1 - \beta$).

4 - Concluding Remarks

We have suggested a simple way for modeling validity and completeness information in the possibility theory framework, in this preliminary study. A careful comparison with the modal logic-based approach proposed by Demolombe (1996a, b) is still to be done. However possibility theory-based logic have connections with conditional modal logic (e.g., Fariñas del Cerro and Herzig, 1991), and we should not be too much surprized by the existence of a possibilistic logic approach to the handling of valid and/or complete information in databases.

One benefit of the possibilistic approach is to allowed for graded level of certainty. Due to the inference mechanism of possibilistic logic, the most certain conclusions with respect to the available information, can be obtained. It would be also possible to keep explicitly track of the sources providing the information. This can be done by dealing with generalized possibilistic formulas of the form $(\varphi, (\alpha^1 / s_1, \alpha^2 / s_2, \ldots, \alpha^m / s_m))$ with the following intended meaning: φ is true with certainty α^1 according to source $s_1, \ldots,$ with certainty α_m according to source s_m. Thus, all the information provided by a source has not necessarily the same level of reliability (a source may be more reliable on some topics than others), and the certainty labels associated with formulas are now only partially ordered. However the basic possibilistic machinery can be extended to this framework; see (Dubois, Lang and Prade, 1992) for details. See also Demolombe (1997) for another approach.

The validity information can be also easily incorporated into the framework proposed by (Prade and Testemale, 1984) for handling ill-known attribute values. Indeed in this approach the available information about attribute values in a tuple is represented by means of possibility distributions on the attribute domains. Asserting the validity of a tuple, as being certain at least at level α, then amounts to modifying each possibility distribution π into $\pi' = \max(\pi, 1 - \alpha)$ (since there is a possibility $1 - \alpha$ that the information is not valid and thus that the value of the attribute is outside the (fuzzy) set of values restricted by π). Then the evaluation of queries is made in terms of possibility and necessity measures, Π and N, that each tuple satisfies the requirement. It can be easily checked that if $\Pi(Q)$ and N(Q) are the evaluations of a query Q based on π, the evaluation incorporating the validity assessment (based on π') are given by $\max(\Pi(Q), 1 - \alpha)$ and $\min(N(Q), \alpha)$ respectively. The latter expresses that, even if N(Q) is high (which means that according to the information represented by π we are certain that Q is satisfied), we cannot be more certain of the relevance of the tuple w.r.t. the query than its validity degree α.

References

Bosc P., Prade H. (1997) An introduction to the fuzzy set and possibility theory-based treatment of flexible queries and uncertain or imprecise databases. In: Uncertainty Management in Information Systems — From Needs to Solutions (A. Motro, P. Smets, eds.), Kluwer Academic Publ., Boston, 283-324.

Demolombe R. (1996a) Answering queries about validity and completeness of data: From modal logic to relational algebra. In: Flexible Query-Answering Systems (Proc. of the 1996 Workshop FQAS'96, Roskilde, Denmark, May 22-24, 1996) (H. Christiansen, H.L. Larsen, T. Andreasen, eds.), Roskilde University, Denmark, 265-276.

Demolombe R. (1996b) Validity queries and completeness queries. In: Foundations of Intelligent Systems (Proc. of the 9th Inter. Symp. ISMIS'96, Zakopane, Poland, June 1996) (Z.W. Ras, M. Michalewicz, eds.), Lecture Notes in Artificial Intelligence, Vol. 1079, Springer Verlag, Berlin, 253-263.

Demolombe R. (1997) Formalizing the reliability of agent's information. 4th ModelAge Workshop on Formal Models of Agents, Sienna, Italy, Jan. 15-17.

Dubois D., Lang J., Prade H. (1992) Dealing with multi-source information in possibilistic logic. Proc. of the 10th Europ. Conf. on Artificial Intelligence (ECAI'92) (B. Neumann, ed.), Vienna, Austria, Aug. 3-7, 38-42.

Dubois D., Prade H. (with the collaboration of Farreny H., Martin-Clouaire R., Testemale C.) (1988) Possibility Theory — An Approach to Computerized Processing of Uncertainty. Plenum Press, New York.

Fariñas del Cerro L., Herzig A. (1991) A modal analysis of possibility theory. Proc. of the Inter. Workshop on Fundamentals of Artificial Intelligence Reserch (FAIR'91) (P. Jorrand, J. Kelemen, eds.), Smolenice Castle, Czechoslovakia, Sept. 8-12, 1991, Lecture Notes in Computer Sciences, Vol. 535, Springer Verlag, Berlin, 11-18.

Motro A. (1986) Completeness information and its application to query processing. Proc. of the 12th Inter. Conf. on Very Large Data Bases, 170-178.

Motro A. (1989) Integrity = validity + completeness. ACM Trans. on Database Systems, 14(4), 480-502.

Motro A., Smets P. (1997) Uncertainty Management in Information Systems — From Needs to Solutions. Kluwer Academic Publ., Boston.

Prade H., Testemale C. (1984) Generalizing database relational algebra for the treatment of incomplete/uncertain information and vague queries. Information Sciences, 34, 115-143.