Automated Reasoning About an Uncertain Domain

Flávio S. Corrêa da Silva † Dave Robertson † Paul Chung †
† Dept. of AI, Univ. of Edinburgh † AIAl
80 South Bridge, Edinburgh, Scotland EH1 1HN
email: fcs@alpna.ed.ac.uk dr@alpna.ed.ac.uk chung@aiai.ed.ac.uk

Abstract

In this paper we introduce a resolution-based logic programming language that handles probabilities and fuzzy events. The language can be viewed as a simple knowledge representation formalism, with the features of being operational and presenting a complete declarative semantics. An extended version of this paper can be found in [3].

1 Introduction

Designing expert systems requires tools for representing and manipulating statements of uncertainty. Representation tools must be expressive, with at least the expressive power of first-order logic, and precise, that is, they must present a well-specified declarative semantics. Uncertainty, on the other hand, has an ambivalent character: it is at the same time part of the represented objects and a property of these objects. As a result, a precise representation of uncertainty must include a description of how it is measured and of how it attaches to these objects. The requirement for manipulation tools stresses the need for computational tractability: if automated reasoning for expert systems is a goal, then tools which cannot be implemented are unsatisfactory.

This paper introduces a language presenting conjointly the following features:

• it is an extension of - and therefore at least as expressive as - first-order logic (or at least the part of it that admits being automated);

• it contains a precise specification for the uncertainty measures under consideration and how they attach to other elements in the domain;

• it is computable, and an implementation of it is available.

Uncertainty is a multi-faceted concept. This paper focuses on uncertainty measures attached directly to objects in the domain. Even so, at least two different types of measures can be identified: i) probability measures and its extensions; ii) measures of proximity and similarity. There is no reason to believe that they are mutually exclusive, so the way they may interact is also identified.

The concern with combined representations of multiple measures of uncertainty stems from the works by Bacchus [1] and Halpern [7]. In contrast with our work, which deals with two different measures on the domain, their works emphasise probability measures on the domain and on possible worlds. A similar emphasis for measures of similarity is found in [5].
The concern for automated formal reasoning with a well-defined declarative semantics is also found in [4] and [8]. In [4] the concern is with *possibilistic logic programming*, thus placing the uncertainty measure on possible worlds, whereas the approach presented in [8] is based on the existence of a complete lattice of truth-values, an assumption which may not be necessarily fulfilled per se. The approach in this paper contrasts with the latter by avoiding this assumption as a definite one, and by construing uncertainty measures as inherent domain properties, rather than syntactical constructs which find their counterparts in a model.

Section 2 introduces a specification of the uncertainty measures and how they interact with each other. Section 3 presents the representation tool, constructed as a logic programming language. The presentation is incremental. First an "uncertainty-free" language is selected. Then it is extended to encompass measures of similarity. Finally, it is extended to admit probabilities. Some conclusions are drawn in section 4.

2 The Uncertainty Model

Probability measures are assumed to be finite and discrete. A *finite discrete probability space* is a pair \((D, \mathcal{P})\) where \(D\) is a finite sample space and \(\mathcal{P}\) is a discrete probability measure (that is, \(X \subseteq D, \mathcal{P}(X) = \sum_{d \in X} \mathcal{P}(d)\), where \(\mathcal{P}(d)\) is known for all \(d \in D\). \(X\) is called an *event* of \(D\).

A conditional probability measure of an event \(X\) given an event \(Y\) is defined as:

\[
\mathcal{P}(X|Y) = \frac{\mathcal{P}(X \cap Y)}{\mathcal{P}(Y)}, \mathcal{P}(Y) > 0,
\]

\[
\mathcal{P}(X|Y) = 0, \mathcal{P}(Y) = 0.
\]

The restriction on \(D\) being finite ensures the computability of probability measures. The restriction on discrete probability measures will be justified after fuzzy events are introduced (see below).

Recalling that an event specifies a class of objects in \(D\), *fuzzy set theory* was developed to treat ill-defined classes by allowing fractional memberships: a fuzzy membership function measures the degree to which an element belongs to a class or, alternatively, the degree of similarity between the class to which the element belongs and a reference class. Formally, a fuzzy subset \(F\) of a referential set \(D\) is defined by an arbitrary mapping \(\mu_F : D \rightarrow [0, 1]\). \(\mu_F(d) = 1\) corresponds to the intuitive notion that \(d \in F\) and \(\mu_F(d) = 0\) to the notion that \(d \not\in F\).

Set-theoretic operations can be extended to fuzzy sets by means of *triangular norms and conorms* [9, 13]. A norm of two membership functions corresponds to the generalised operation of intersection for fuzzy sets, and the conorm to the generalised operation of union. If these set operations are required to be *distributive* and *idempotent*, in order to be kept as close as possible to standard set operations, then the only possible triangular norm and conorm are, respectively, \(T = \min\) and \(T = \max\) (see [9] for a proof of this result). On the other hand, the simplest existing complementation function is the function \(C(x) = 1 - x\). Henceforth, these functions are adopted as extended set operations.

In [9] the concept of algebra (an algebra \(\chi\) is a set of subsets of \(D\) such that \(D \in \chi\) and \(\chi\) is closed under complementation and union [6, 12]) is extended for fuzzy sets, and in [11, 13, 15] the definition of the probability of a fuzzy event is presented, reputed as originally by Zadeh. Given a probability space \((D, \chi, \mathcal{P})\) and assuming that the fuzzy set \(F\) belongs to the algebra \(\chi\), the probability of the event \(F\) is given by the Lebesgue-Stieltjes integral \(\mathcal{P}(F) = \int_D \mu_F(d) d\mathcal{P}\). For a finite discrete probability space \((D, \mathcal{P})\), this integral turns to \(\mathcal{P}(F) = \sum_{d \in F} \mu_F(d) \times \mathcal{P}(d)\).