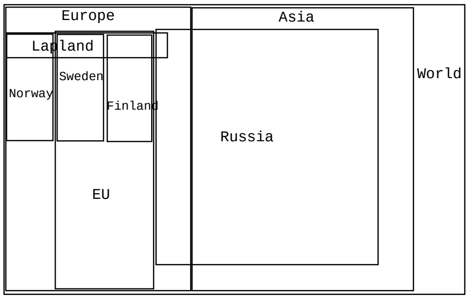
A Method for Modeling Uncertainty in Semantic Web Taxonomies

Markus Holi and **Eero Hyvönen**[1]

Abstract. Semantic web ontologies are based on crisp logic, and do not provide well-defined means for expressing uncertainty needed when modeling the real world. To address this problem, this paper presents a new probabilistic method to model degrees of subsumption, i.e., overlap between concepts. We propose a notation, based on set theory, by which concepts can be quantified, and partial subsumption represented in a taxonomy. Based on this notation, a probabilistic method for computing degrees of overlap between the concepts is presented. Overlap is quantified by transforming the taxonomy into a Bayesian network. The degree of overlap is a simple, well-defined measure of conceptual similarity. It can be applied, for example, in information retrieval to computing the relevance relation of the result set.

1 UNCERTAINTY IN ONTOLOGIES



```
World 37*23 = 851
Europe 15*23 = 345
Asia 18*23 = 414
EU  8*21 = 168
Sweden 4*9 = 36
Finland 4*9 = 36
Norway 4*9 = 36
Lapland 13*2 = 26  Lapland&(Finland | Sweden | Norway) = 8
Lapland&EU = 16 Lapland&Russia = 2
Russia 18*19 = 342  Russia&Europe = 57  Russia&Asia = 285
```

Figure 1. A Venn diagram illustrating countries, areas, their overlap, and size in the world.

Taxonomic concept hierarchies constitute an important part of the RDF(S) [2] and OWL [1] ontologies used on the semantic web. For example, subsumption hierarchies based on the *subClassOf* or *partOf* properties are widely used.

Relations among real life entities are always a matter of degree. Subsumption hierarchies, on the other hand, are crisp in principle. Thus, they fail to describe important aspects of real life concepts, and relations between them. This is an important drawback, that hinders the usability of ontology based information retrieval systems [12].

For example, the Venn diagram of figure 1 illustrates some countries and areas in the world. A crisp *partOf* meronymy cannot express the simple fact that Lapland partially overlaps Finland, Sweden, Norway, and Russia, or that Russia is to some degree part of both Europe and Asia. Furthermore, it is not possible to quantify the coverage and the overlap of the areas involved.

Selected	**Referred**	**Overlap**
Lapland	World	26/851 = 0.0306
	Europe	26/345 = 0.0754
	Asia	0/414 = 0.0
	EU	16/168 = 0.0953
	Norway	8/36 = 0.2222
	Sweden	8/36 = 0.2222
	Finland	8/36 = 0.2222
	Russia	2/342 = 0.0059

Table 1. The *overlap table* of Lapland according to figure 1.

To address these foundational problems, this paper presents a new probabilistic method to represent overlap in taxonomies, and to compute the overlap between a *selected* concept and every other - *referred* - concept in the taxonomy. In effect, an *overlap table* is created for the selected concept. The overlap table can be created for every concept of a taxonomy. For example, table 1 present the overlap table of Lapland based on the the Venn diagram of figure 1. The Overlap column lists values expressing the mutual overlap of the selected concept and the other - referred - concepts, i.e., $Overlap = \frac{|Selected \cap Referred|}{|Referred|}$. These values can be used as natural measure of mutual overlap.

Intuitively, the overlap value has the following meaning: High values imply, that the meaning of the selected concept approaches the meaning of the referred one. The value is 0 for disjoint concepts (e.g., Lapland and Asia) and 1, if the referred concept is subsumed by the selected one.

The overlap value is useful, for example, in information retrieval. Assume that an ontology contains individual products manufactured in the different countries and areas of figure 1. The user is interested in finding objects manufactured in Lapland. The overlap values of the table 1 then tell how well the annotations “Finland”, “EU”, “Asia”, etc., match with the query concept “Lapland” in a well-defined probabilistic sense, and the hit list can be sorted into an order of relevance

[1] University of Helsinki, Helsinki Institute for Information Technology (HIIT), P.O. Box 26, 00014 UNIVERSITY OF HELSINKI, FINLAND, http://www.cs.helsinki.fi/group/seco/ email: firstname.lastname@cs.helsinki.fi

accordingly.

It is mathematically easy to compute the overlap tables, if a Venn diagram (the sets) is known. In practice, the Venn diagram may be difficult to create from the modeling view point, and computing with explicit sets is computationally complicated and inefficient. For these reasons our method calculates the overlap values from a taxonomic representation of the Venn diagram.

Our method consists of two parts:

1. A graphical notation by which partial subsumption and concepts can be represented in a quantified form. The notation can be represented easily in RDF(S).
2. A method for comoputing degrees of overlap between the concepts of a taxonomy. Overlap is quantified by transforming the taxonomy first into a Bayesian network [6].

In the following, the graphical notation is presented first, and then the method for computing overlaps is described. After this an implementation of the method is described. Finally, related work is discussed, lessons learned are summarized, and directionss of further research are outlined.

2 REPRESENTING OVERLAP

In RDFS and OWL a concept class refers to a set of individuals. Subsumption reduces essentially into the subset relationship between the sets corresponding to classes [1]. A taxonomy is therefore a set of sets and can be represented, e.g., by a Venn diagram.

If A and B are sets, then A must be in one of the following relationships with B.

1. A is a subset of B, i.e. $A \subseteq B$.
2. A partially overlaps B, i.e. $\exists x, y : (x \in A \wedge x \in B) \wedge (y \in A \wedge y \notin B)$.
3. A is disjoint from B, i.e. $A \cap B = \emptyset$.

Based on these relations, we have developed a simple graph notation for representing uncertainty and overlap in a taxonomy as an acyclic *overlap graph*. Here concepts are nodes, and a number called *mass* is attached to each node. The mass is a measure of the size of the set corresponding to the node concept. A solid directed arc from concept A to B denotes crisp subsumption $A \subseteq B$, a dashed arrow denotes disjointness $A \cap B = \emptyset$, and a dotted arrow represents quantified partial subsumption between concepts, which means that the concepts partially overlap in the Venn diagram. The amount of overlap is represented by the *partial overlap value* $p = \frac{n(A\cap B)}{n(A)}$.

In addition to the quantities attached to the dotted arrows, also the other arrow types have implicit overlap values. The overlap value of a solid arc is 1 (crisp subsumption) and the value of a dashed arc is 0 (disjointness). The quantities of the arcs emerging from a concept must sum up to 1. This means that either only one solid arc can emerge from a node or several dotted arcs (partial overlap). In both cases, additional dashed arcs can be used (disjointness). Intuitively, the outgoing arcs constitute a quantified partition of the concept. Thus, the dotted arrows emerging from a concept must always point to concepts that are mutally disjoint with each other.

Notice that if two concepts overlap, there must be a directed (solid or dotted) path between them. If the path includes dotted arrows, then (possible) disjointness between the concepts must be expressed explicitly using the disjointness relation. If the directed path is solid, then the concepts necessarily overlap.

For example, figure 2 depicts the meronymy of figure 1 as an overlap graph. The geographic sizes of the areas are used as masses and the partial overlap values are determined based on the Venn diagram. This graph notation is complete in the sense that any Venn diagram can be represented by it. However, sometimes the accurate representation of a Venn diagram requires the use of auxiliary concepts, which represent results of set operations over named sets, for example $A \setminus B$.

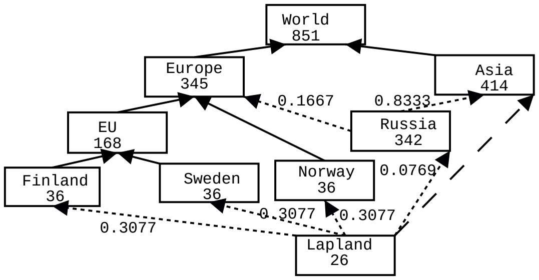


Figure 2. The taxonomy corresponding to the Venn diagram of figure 1.

3 SOLID PATH STRUCTURE

Our method creates an overlap table (cf. figure 1) for each concept in the taxonomy. Computing the overlaps is easiest when there are only solid arcs, i.e., complete subsumption relation, between concepts. If there is a directed solid path from A (selected) to B (referred), then overlap $o = \frac{m(A\cap B)}{m(B)} = \frac{m(A)}{m(B)}$. If the solid path is directed from B to A, then $o = \frac{m(A\cap B)}{m(B)} = \frac{m(B)}{m(B)} = 1$. If there is not a directed path between A and B, then $o = \frac{m(A\cap B)}{m(B)} = \frac{m(\emptyset)}{m(B)} = 0$.

If there is a mixed path of solid and dotted arcs between A and B, then the calculation is not as simple. Consider, for example, the relation between $Lapland$ and EU in figure 2. To compute the overlap, we have to follow all the paths emerging from $Lapland$, take into account the disjoint relation between $Lapland$ and $Asia$, and sum up the partial subsumption values somehow.

To exploit the simple solid arc case, a taxonomy with partial overlaps is first transformed into a *solid path structure*, in which crisp subsumption is the only relation between the concepts. The transformation is done by using to the following principle:

Transformation Principle 1 *Let A be the direct partial subclass of B with the partial overlap value o. In the solid path structure the partial subsumption is replaced by an additional* middle concept, *that represents $A \cap B$. It is marked to be the subclass of both A and B.*

For example, the taxonomy of figure 2 is transformed into the solid path structure of figure 3. The original partial overlaps of Lapland and Russia are transformed into crisp subsumption by using middle concepts.

The transformation is specified in algorithm 1. The algorithm processes the overlap graph T in a breadth-first manner starting from the root concept. A concept c is processed only after all of it super concepts (partial or complete) are processed. Because the graph is acyclic, all the concept will eventually be processed.

Each processed concept c is written to the solid path structure SPS. Then each arrow emerging from c is processed in the following way. If the arrow is solid, indicating subsumption, then it is written

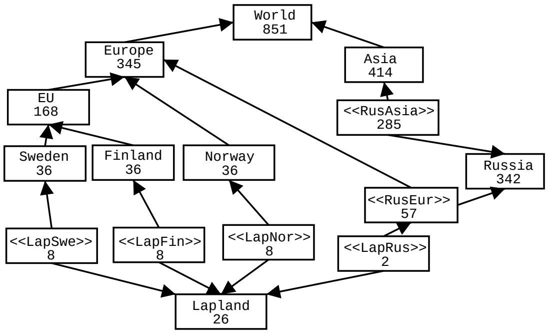


Figure 3. The taxonomy of figure 2 as a solid path structure.

into the solid path structure as such. If the arrow is dotted, indicating partial subsumption, then a middle concept $newMc$ is added into the solid path structure. It is marked to be the complete subconcept of both c and the concept p to which the dotted arrow points in T. The mass of $newMc$ is $m(newMc) = |c \cap p|$.

```
Data: OverlapGraph T, RootConcept r
Result: SolidPathStructure SPS
SPS := empty;
foreach concept c in T do
    foreach complete or partial superconcept p of c in T do
        if p connected to its superconcepts through middle concepts in SPS then
            mc := the middle concept that c overlaps;
            if c complete subconcept of p then
                mark c to be complete subconcept of mc in SPS;
            else
                newMc := middle concept representing $c \cap p$;
                mark newMC to be complete subconcept of c and mc in SPS;
            end
        else
            if c complete subconcept of p then
                mark c as complete subconcept of p in SPS;
            else
                newMc .= middle concept representing $c \cap p$;
                mark newMc to be complete subconcept of c and p in SPS;
            end
        end
    end
end
```

Algorithm 1: Creating the solid path structure

However, if p is connected to its superconcepts (partial or complete) with a middle concept structure, then the processing is not as simple. In that case c has to be connected to one of those middle concepts. The right middle concept is found by using the dashed arrows emerging from c. The right middle concept mc is the one that is not subsumed by a concept that is marked to be disjoint from c in the overlap graph. This is the middle concept that c overlaps. Notice, that if the overlap graph is an accurate representation of the underlying Venn diagram, then mc is the only middle concept that fulfils the condition.

If c is a complete subconcept of p in the overlap graph T, then c is marked to be the complete subconcept of mc SPS. If c is a partial subconcept of p in T, then it connected to mc with a middle concept structure.

If c was connected directly to p, instead of mc, then the information conveyed in the dashed arrows, indicating disjointness between concepts would have been lost. For example, in figure 3 $Lapland$ was connected directly to $Russia$, then the information about the disjointness of $Lapland$ and $Asia$ would have been lost.

4 COMPUTING THE OVERLAPS

Based on the solid path structure, the overlap table values $o = P(A|B)$ for a selected concept A and a referred concept B could be calculated by the algorithm 2, where notation X_s denotes the set of (sub)concepts subsumed by the concept X.

```
if $B \subset A$ then
    $o := 1$
else
    $C = A_s \cap B_s$
    if $C = \emptyset$ then
        $o := 0$
    else
        $o := \frac{n(\bigcup C)}{n(B)}$
    end
end
```

Algorithm 2: Computing the overlap

The overlap table for A could be implemented by going through all the concepts of the graph and calculating the overlap value according to the above algorithm. However, because the overlap values between concepts can be interpreted as a conditional probabilities, we chose to use the solid path structure as a Bayesian network topology and let the efficient evidence propagation algorithms developed for Bayesian networks [6] to take care of the overlap computations. Furthermore, we saw a Bayesian representation of the taxonomy valuable as such. The Bayesian network could be used for example in user modelling [10].

Probabilistically, concepts can be interpreted as boolean binary variables. If A is the selected concept and B is the referred one, then the overlap value o can be interpreted as the conditional probability $P(B = true|A = true) = \frac{|s(A) \cap s(B)|}{|s(B)|} = o$, where $s(A)$ and $s(B)$ are the sets depicted by the random variables A and B. In the context of information retrieval the meaning of the conditional probability could be described as the probability that a person interested data belonging to category A will also be interested in data belongin to category B.

The joint probability distribution of the Bayesian network is defined by conditional probability tables (CPT) $P(A|B_1, B_2, \ldots B_n)$ for nodes with parents $B_i, i = 1 \ldots n$, and by prior marginal probabilities set for nodes without parents. The CPT $P(A|B_1, B_2, \ldots B_n)$ for a node A can be constructed by enumerating the value combinations (true/false) of the parents $B_i, i = 1 \ldots n$, and by assigning:

$$P(A=true|B_1=b_1,\dots B_n=b_n)=\frac{n(\bigcup\{B_i:b_i=true\})}{n(A)} \qquad (1)$$

The value for the complementary case $P(A = false|B_1 = b_1,\dots B_n = b_n)$ is obtained simply by subtracting from 1.

If A has no parents, then $P(A = true) = \lambda$, where λ is a very small non-zero probability, because we want the posterior probabilities to result from conditional probabilities only, i.e., from the overlap information.

The whole overlap table of a concept can now be determined efficiently by using the solid path structure as a Bayesian network with its conditional and prior probabilities. By instantiating the selected concept node and all the concepts subsumed by as evidence (their values are set "true"), the propagation the algorithm returns the overlap values as posterior probabilities of concept nodes.

Notice, that a Bayesian network created in the above method does not calculate the posterior exactly by the definition of the conditional probability above. A small inaccuracy, is attached to each value, as the result of the λ prior probablity that was given to the parentless variables. The λ is result of the fact that a prior probability of zero can not be given. Despite this small inaccuracy we decided to define the Bayesian network in the above manner for the following reasons.

The solid path structure, interpreted as a Bayesian network, has the following usefule characteristics. First, disjoint concepts are d-separated, overlapping concepts are not. Thus, d-separation can be taken as an indication of disjointness. Second, the conditional probability tables can be created easily based on the masses of the concepts.

First, to be able to easily use the the solid path structure as the topology of the Bayesian network. The CPTs can be calculated directly based on the masses of the concepts. Second, with this definition the Bayesian evidence propagation algorithm returns the overlap values readily as posterior propabilities. We experimented with various ways to construct a Bayesian network according to probabilistic interpretations of the Venn diagram. However, no one of these constructions did not answer to our needs as well as the construction defined above.

Third, in the solid path structure d-separation indicates disjointness between concepts. We see this as a useful characteristic, because it makes the simultaneous selection of two or more disjointed concepts possible. If the Venn diagram was taken as the set theoretic descriptioni of the joint probability distribution, then the disjointness should be stated explicitly in the Bayesian network.

5 IMPLEMENTATION

The presented method has been implemented as a proof-of-concept.

5.1 Overlap Graph

Overlap graphs are represented as RDF(S) ontologies in the following way. Concepts are represented as RDF(S) classes[2] The concept masses are represented using a special *Mass* class. Its two properties, subject and mass, that tell the concept resource in question and mass as a numeric value, respectively. The subsumption relation can be implemented with a property of the users choice. Partial subsumption is implemented by a special *PartialSubsumption* class with three properties: subject, object and overlap. The subject property points to the direct partial subclass, the object to the direct partial superclass, and overlap is the partial overlap value. The disjointness arc is implemented by the disjointFrom property used in OWL.

5.2 Overlap Computations

The architecture of the implementation can be seen in figure 4. The input of the implementation is an RDF(S) ontology, the URI of the root node of the overlap graph, and the URI of the subsumption property used in the ontology. Additionally, also an RDF data file that contains data records annotated according to the ontology may be given. The output is the overlap tables for every concept in the taxonomy extracted from the input RDF(S) ontology. Next, each submodule in the system is discussed briefly.

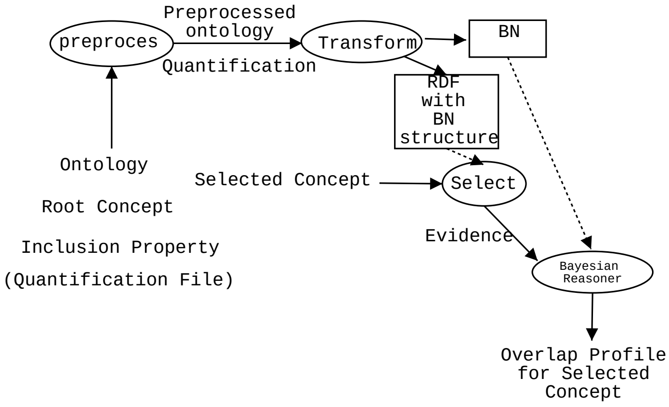


Figure 4. The architecture of the implementation.

The *preprocessing* module transforms the taxonomy into a predefined standard form. If an RDF data file that contains data records annotated according to the ontology is given as optional input, then the preprocessing module determines the mass of each concepts of the taxonomy based on these annotations. The value is the number of direct and indirect instances of the concept. The quantification principle is illustrated in figure 5.

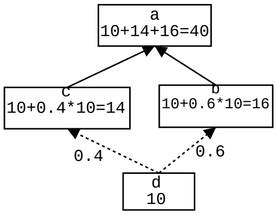


Figure 5. Quantification of concepts. The number of direct instances of each concept is 10.

The *transformation* module implements the transformation algorithm, and defines the CPTs of the resulting Bayesian network. In addition to the Bayesian network, it creates an RDF graph with an identical topology, where nodes are classes and the arcs are represented by the *rdf:subClassOf* property. This graph will be used by the

[2] Actually, any resources including instances could be used to represent concepts.

selection module that expands the selection to include the concepts subsumed by the selected one, when using the Bayesian network.

The *Bayesian reasoner* does the evidence propagation based on the selection and the Bayesian network. The selection and Bayesian reasoner modules are operated in a loop, where each concept in the taxonomy is selected one after the other, and the overlap table is created.

The *preprocessing*, *transformation*, and selection modules are implemented with SWI-Prolog[3]. The Semantic Web package is used. The *Bayesian reaoner* module is implemented in Java, and it uses the Hugin Lite 6.3[4] through its Java API.

6 DISCUSSION

6.1 Related Work

The problem of representing uncertain or vague inclusion in ontologies and taxonomies has been tackled also by using methods of fuzzy logic [3, 4, 13] and rough sets [11, 8]. We chose to use crisp set theory and Bayesian networks, because of the sound mathematical foundations they offer. The calculations are simple, but still enable the representation of overlap and vague subsumption between concepts. The Bayesian network representation of a taxonomy is useful not only for the matching problem we discussed, but can also be used for other reasoning tasks [10].

The work that is closest to ours is that of Ding et al. [5]. They present principles and methods to convert an OWL ontology into a Bayesian network. Their methods are based on probabilistic extensions to description logics [9, 7]. The approach is quite different from ours, in a number of ways. First, their aim is to create a method to transform any OWL ontology into a Bayesian network. Our goal is not to transform existing ontologies into Bayesian networks, but to create a method by which overlap between concepts could be represented and computed from a taxonomical structure. However, we designed the overlap graph and its RDF(S) implementation so, that it is possible, quite easily, to convert an existing crisp taxonomy to our extended notation.

Second, in Ding et al.'s approach, probabilistic information must be added to the ontology by the human modeler that needs to know probability theory. In our approach, the taxonomies can be constructed without virtually any knowledge of probability theory or Bayesian networks. Third, the created Bayesian network in their approach is the goal of the work. In our method, the Bayesian network is merely a background tool to help in uncertainty modeling. Fourth, the actual transformation of subsumption relations (subclass) is done quite differently in Ding's work.

6.2 Lessons Learned

Overlap graphs are simple and can be represented in RDF(S) easily. Using the notation does not require knowledge of probability or set theory. The notation enables the representation of any Venn diagram, but there are set structures, which lead to complicated representations.

Such a situation arises, for example, when three or more concepts mutually partially overlap each other. In these situations some auxiliary concepts have to be used. We are considering to extend the notation so that this kind of situations could be represented better. On the other hand, we do not think such situations are frequent in real-world taxonomies.

The Bayesian network structure that is created with the presented method is only one of the many possibilities. This one was chosen, because it can be used for computing the overlap tables in a most direct manner. However, it is possible that in some situations different Bayesian reprepresentation of the would be better.

We see the principle of modeling uncertainty on the basis of the set theoretic structure of the concepts as the most valuable contribution of this work. The actual notations and transformations can be seen as first guesses on deploying the principle.

6.3 Future Work

We intend to apply the overlap calculation in various realistic application situatioins. Also the refinement of the taxonomy language is considered to enhance its usability. The transformation of the taxonomy to alternative Bayesian network structures is an issue of future work, as well as trying the Bayesian network as a basis for personalization.

ACKNOWLEDGEMENTS

Our research was funded mainly by the National Technology Agency Tekes.

REFERENCES

[1] *OWL Web Ontology Language Guide*. http://www.w3.org/TR/2003/CR-owl-guide-20030818/.

[2] *RDF Vocabulary Description Language 1.0: RDF Schema*. http://www.w3.org/TR/rdf-schema/.

[3] G. Akrivas, M. Wallace, G. Andreou, G. Stamou, and S. Kollias. Context - sensitive semantic query expansion. In *Proceedings of the IEEE International Conferrence on Artificial Intelligence Systems (ICAIS)*, 2002.

[4] R.A. Angryk and F.E. Petry. Consistent fuzzy concept hierarchies for attribute generalization. In *Proceeding of the IASTED International Conference on Information and Knowledge Sharing (IKS' 03)*, 2003.

[5] Zhongli Ding and Yun Peng. A probabilistic extension to ontology language owl. In *Proceedings of the Hawai'i Internationa Conference on System Sciences*, 2004.

[6] F. V. Finin and F. B. Finin. *Bayesian Networks and Decision Graphs*. Springer-Verlag, 2001.

[7] R. Giugno and T. Lukasiewicz. P-shoq(d): A probabilistic extension of shoq(d) for probabilistic ontologies in the semantic web. INFSYS Research Report 1843-02-06, Technische Universität Wien, 2002.

[8] J.Pawlak. Rough sets. *Internation Journal of Information and Computers*, 1982.

[9] D. Koller, A. Levy, and A. Pfeffer. P-classic: A tractable probabilistic description logic. In *Proceedings of AAAI-97*, 1997.

[10] A. Kuenzer, C. Schlick, F. Ohmann, L. Schmidt, and H. Luczak. An empirical study of dynamic bayesian networks for user modeling. In R. Schafer, M.E. Muller, and S.A. Macskassy, editors, *Proc. of the UM'2001 Workshop on Machine Learning for User Modeling*, 2001.

[11] H. Stuckenschmidt and U. Visser. Semantic translation based on approximate re-classification. In *Proceedings of the 'Semantic Approximation, Granularity and Vagueness' Workshop*, 2000.

[12] M. Wallace, G. Akrivas, and G. Stamou. Automatic thematic categorization of documents using a fuzzy taxonomy and fuzzy hierarchical clustering. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2003)*, 2003.

[13] L. Zadeh. Fuzzy sets. *Information and Control*, 1965.

[3] http://www.swi-prolog.org/

[4] http://www.hugin.com/