Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 11 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

An Extension of Description Logic Al

By F. Mamache

Université des Sciences et de la Technologie Houari Boumediene Alger Algérie

Abstract - The research in the domain of knowledge representation and reasoning has always concentrated on the methods that give a good description in the domain where they are able to be used to construct intelligent applications. Description Logics are a family of languages of knowledge representation which can be used to represent knowledge of a field of applications by clear, formal and structured means. In this paper, we give an overview of what are Description Logics and their actual applications in different fields and a brief idea of extensions of Description Logic AL, as we also introduce two operators, the operator less and operator more, which allow us to obtain a new extension of the Description Logic AL.

Keywords : Artificial Intelligence, Description Logics, Knowledge Representation, Semantic, Subsumption, Classification.

GJCST-C Classification : D.2.2

AN EXTENSION OF DESCRIPTION LOGIC AL

An Extension of Description Logic AL

F. Mamache

Abstract - The research in the domain of knowledge representation and reasoning has always concentrated on the methods that give a good description in the domain where they are able to be used to construct intelligent applications. Description Logics are a family of languages of knowledge representation which can be used to represent knowledge of a field of applications by clear, formal and structured means. In this paper, we give an overview of what are Description Logics and their actual applications in different fields and a brief idea of extensions of Description Logic AL, as we also introduce two operators, the operator less and operator more, which allow us to obtain a new extension of the Description Logic AL.

Keywords : Artificial Intelligence, Description Logics, Knowledge Representation, Semantic, Subsumption, Classification.

I. Introduction

Research in the domain of knowledge representation and reasoning always concentrates on the methods that give a good description in the domain where they are able to be used to construct intelligent applications. By intelligent applications, we refer to systems able to find implicit consequences to represent knowledge explicitly.

Description Logic systems produce to their users' possibilities of varied inferences that deduct the implicit knowledge of the knowledge represented explicitly. Description Logics are a family of languages of knowledge representation which can be used to represent the knowledge of a field of applications by clear, formal and structured means.

These are logical formalisms of representation which distinguish themselves from Networks and Frames by their formal semantic that is based on logic.

In this paper, we give an overview of what are Description Logic and their applications in different fields. We notice several domains of applications, some include Software Engineering, Configuration, Medicine, Numeric libraries and Information Systems based on Web. there exists other domains of applications where the Description Logics have an significant role, as the field which include the Treatment of Natural Language and Management of Database. We give in this paper a brief idea of extensions of Description Logic AL, as we also introduce two operators, operator less and operator more, which allow us to obtain a new extension of the Description Logic AL.

II. Origin of Description Logics

Description Logics Dls or terminology logics are a family of languages of knowledge representation which can be used to represent the knowledge of a field of applications by clear, formal and structured means. Description Logics difier of their predecessors, such as Networks and Frames, given that they are equipped of formal logic based on semantic. We find three generations of systems. In the following, we will see their historic evolution.

a) Pre-description logic systems

Description Logics are formalisms of knowledge representation based on KL-One language. KL-One language is considered as root of the family of all languages. The Networks that are at the origin of the language KL-One, were introduced in 1966 as a representation of the basic concepts of the English words, and become a popular type of structures to represent a wide variety of concepts of the applications in Arti_cial Intelligence.

KL-One language introduced most key notions of Dls:

- Notion of concepts and roles
- Notions of restrictionvalue and the restrictionnumber that has an important role in the usage of the roles in the de_nition of the concepts and,
- Inference of subsumption and classi_cation. Kl−One is based on the subsumption : it's a system of structured inheritance and it is at the origin of a family of languages such as : KL-Two, Krypton, Loom, Kandor, Back, Nikl, Classic and Kriss.

b) Description logics Systems

The last pre-Description logics originate directly from KL-One that itself is a direct result from formal analysis. Description Logics systems that will follow as future generation will result from more theoretical research on terminology logics than of examination consequences of KLOne and of other latest systems. We can notice three approaches for the implementation of the reasoning services :

- The first one can be considered as limited and complete or as systems that are studied by restriction of the set of the concepts so that the subsumption can be calculated eficiently, possible in polynomial time. The system Classic is an example of this approach.
- The second approach designated as expressive and incomplete, since the idea is to furnish an expressive language and an effective reasoning. The inconvenience is, nevertheless, that the

LRIA Laboratoire de Recherche en Intelligence Arti_cielle. Faculté d'Electronique et d'Informatique. Département d'Informatique, USTHB, BP 32 El Alia, Bab Ezzouar, Alger (Email : amamache @usthb.dz).

algorithm of reasoning proves to be incomplete in these systems. An example of this system is the system Loom

- In the third approach, we have the characterized systems as being expressive and complete. They are not effective like those of the preceding approaches.

c) Current Description Logics systems

In the current generation of Knowledge Representation Systems based on the DLs (DLKRS), the need of complete algorithms of the expressive languages became focal points. The expressivity of the language of Description Logics is necessary to reason on the data models. The semi-structured data contributed to the identification of the most of the important extensions for practical applications.

III. Introduction to Description Logics

a) Introduction

A knowledge system is a program able to reason on an application domain to solve a particular problem, using knowledge related to the studied field. The knowledge of the domain is represented by entities which have syntactic descriptions which are associated to semantics. It does not exist any universal method to conceive such systems, but there is a stream of current and active research developed that were nourished by the studies carried out on the logic of the predicates, the networks semantic and the languages of Frames. This research gave rise to a family of languages of representation called Description Logics. In the formalism of Description Logics, a concept allows to represent a set of individuals, while a role represents a binary relation between individuals. A concept corresponds to a generic entity of an application domain and an individual to a particular entity, i.e, instance of a concept. Concepts, roles and individuals obey to the following principles:

- Concept and a role possess a structural description, elaborated from some constructors. A semantic is associated to each description of concept and role by an interpretation. The manipulations operated on the concepts and roles, are realized in agreement with this semantic.
- The knowledge are taken into account according to several levels : The representation and the manipulation of concepts and roles result from terminological level, the description and the manipulation of individuals result from factual level or assertions level. The terminological level is qualified by T-Box and the factual level by A-Box.
- Subsumption allows organizing concepts and roles by generality level: intuitively, a concept C subsumes a concept D if C is more general than D in the view where the set of the individuals represented by C contains the set of the individuals represented by D. A knowledge basis is composed of a hierarchy of concepts and of a hierarchy of roles.
- The operations which are at the basis of the terminological reasoning are the classification and instantiation. Classification applies to the concepts, if necessary to the roles and allows determining the position of a concept and of a role in their respective hierarchies. Instantiation allows finding the concepts of which an individual is susceptible to be an instance.

b) Basis of Description Logics

The basic sets that are defined and used in Description Logic are concepts and roles. Concept denotes a set of individuals and a role denotes a binary relation between individuals. Concept possesses a structured description which is constructed using a set of constructors introducing the roles associated to the concept and the restrictions attached to these roles. The restrictions carry generally on the co-domains of the role, which is the concept which the role establishes a relation, and the cardinality of the role, which fixes the minimal and maximal number of elementary values that, can take the role. The elementary values are instances of concepts or many values that result from basic types as integer, real, and chains of characters.

The concepts can be primitive or defined. The primitive concepts are comparable to atoms and are used as a basis for construction of the definite concepts. A role can be primitive or defined and can have a structural description, where appear the properties associated to the role.

The constructor and indicates that a concept is constructed from a conjunction of concepts that are the ascendants of the new concept- and the constructor all specifies the co-domain of a relation. The constructor not express the negation and does apply only to primitive constructors. The constructors at−last and at−most specify the cardinality of the role which they are associated and respectively indicate the minimum number and the maximum number of elementary values of the role.

The associated characteristics to a primitive concept are necessary: an individual x that is an instance of a primitive concept P possesses the characteristics of P. The associated characteristics to a defined concept D are necessary and sufficient: an individual x that is an instance of a defined concept D possesses the characteristics of D, and inversely, the fact that an individualy possesses the set of the associated characteristics to D suffices to infer that y is an instance of D. This distinction is at the basis of the classification process. Concepts are defined in a declarative manner (in a declaratory way) and the (installation) set up of the defined concepts in the hierarchy of the concepts is carried out under the check (control) of the classification process.

c) Description of concepts and roles : syntax

There is several description languages of concepts and roles. In follows, we introduce a minimal language called AL, which is enriched progressively by new constructors. The language AL is based on the languages FL and FL^- presented below, which are the languages for which were established the first theoretical results on the Dls.

C,D→	A|
Top|	⊤
Botto|⊥	
(and C D)|	C⊓D
(not A)|	¬A
(all r C)|	∀r.C
(some r)	∃r
Lispian*syntax*	Germany*syntax*

The grammar of the description language of AL, with Lispian and Germany syntaxes, C and D are concepts names, A a primitive concept name and r a primitive role name.

- The constructor Top (>) denotes the most general concept.
- The BOTTOM concept (?) denotes the least specific concept. Intuitively, the Top extension includes all possible individuals while that of BOTTOM is empty.
- The operator of conjunction: The operator and (u) allows us to build a new concept corresponding to conjunction of definite concepts. Example: The concept Person and Mother gives a new concept Female.
- The constructor not (¬) corresponds to the negation and relates only to the primitive concepts. Example: The concept Person and not Female can be expressed by: Person u ¬ Female.
- The operator of disjunction: The operator or (t) allows us to build a new concept corresponding to disjunction of definite concepts. Example: The concept person that are Male or Female can be represented by: Male t Female.
- Restrictions of roles: The connectors at−last, at−most and all are called restrictions of roles.

Restrictions of cardinality at−last (≥) and at−most (≤) specify the cardinality of role with which they are associated and indicate the minimal and maximum number of elementary values of the role. They limit the sets of values max and min of a role on a concept or an individual. Example: The concept: (≥3 has Child) ⊓ (≤2 has Female Relative) represent the concept: an individual having at − least 3 children and more 2 daughters.

To represent concepts like "In the system, there is less equations than unknowns ", and "an individual having more girls than boys" where the minimal number and the maximum number are not known, we thought to introduce others restrictions operators.

The constructors less and more indicate the cardinality of the role to which they are associated without specifying the minimal number or the maximum number of elementary values of the role. Example: The concept: (system (has (equations) < (unknowns)) (system (less (equations, unknowns))) represent the concept:" the system has less equations than unknowns ". Example: The concept: (has Child (daughters) > (sons)) (has Child (more (daughters, sounds))) represent the concept: "an individual having more daughters than sons ".

- The universal quantification all (∀r.c) specifies the co-field of role r. Example: The concept: (All children are female) is expressed by: ∀ has Child. Female.
- The existential quantification some (∃r) introduced the role r and affirms the existence of (less) one couple of individuals in relation via r. The operator of restriction of existential values: Allows to write the concept (an individual having a girl) like ' 9 has Child. Female'. Language AL = {> ?, u B, ¬ A, 8 r :C, 9 r} can be enriched by the following constructors:
- The negation of primitive or defined concepts, which is noted (not C) or ¬ C. The corresponding extension of AL is ALL = AL [{¬ C}.
- The disjunction of concepts, which is noted (or C D) or C t D. The corresponding extension of AL is ALU = AL [{C t D}.
- The typed existential quantification, noted (c − some r C) or 9 r: C. The corresponding extension of AL is ALE = AL [{9 r: C}.
- The typed existential quantification 9 r: C introduces a role r of co-field C and imposes the existence of less one couple of individuals (x, y) in relation by the role r, where C is the type of y.
- The cardinality on the roles is noted (at − leastnr) or ≤ nr, and (at − mostnr) or ≥ nr.The corresponding extension of AL is ALN = AL [{≤ nr, ≥ nr}.
- The constructors ≤ nr and ≥ nr fix the cardinality minimum and maximum elementary values numbers of the role which they are associate. In particular, construction (∃r) is equivalent to construction (≥1 r).
- The comparison of the cardinality on the roles is noted r_1 less r_2 or $r_1 < r_2$, and (r_1 more r_2) or $r_1 > r_2$. The corresponding extension of AL is ALC = AL [{$r_1 < r_2$, $r_1 > r_2$}.
- The conjunction of roles is noted (and r_1 r_2) or $r_1 \setminus r_2$, the roles r_1 and r_2 being primitive. The corresponding extension of AL is ALR = AL [{$r_1 \setminus r_2$}.

Global Journal of Computer Science and Technology (C) Volume XII Issue XI Version I Year 2012

Global Journal of Computer Science and Technology (C) Volume XII Issue XI Version I Year 2012

d) Concepts and roles description : Semantic

i. *Interpretation in ALLNRC*

A semantic is associated to descriptions of concepts and roles: Concepts are interpreted like subsets of a field of interpretation _ and roles like subsets of product $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

The concepts are interpreted like subsets of interpretation field $\Delta^{\mathcal{I}}$ and roles like subsets of product $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

For a concept C, CI corresponds to the subset of the elements of field $\Delta^{\mathcal{I}}$, and for a role r, rI corresponds to the subset of the couples of elements of product $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

The following definition is given within the framework of language ALCNRI Definition 1 (Interpretation) An interpretation I = (I,.I) is the data of a set called interpretation field and a interpretation function .I which fact of corresponding to a concept a subset of $\Delta^{\mathcal{I}}$ and to a role a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, so that following equations are satisfied:

$$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$$
$$\bot^{\mathcal{I}} = \emptyset$$
$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$
$$(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$$
$$(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} - C^{\mathcal{I}}$$
$$(\forall\, r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \;/\; \forall\, y : (x,y) \in r^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}}\}$$
$$(\exists\, r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \;/\; \exists\, y : (x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$$
$$(\geq nr)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \;/ |\{y \in \Delta^{\mathcal{I}} \;/\; (x,y) \in r^{I}\} \mid \geq n\}$$
$$(\leq nr)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \;/\; |\{y \in \Delta^{\mathcal{I}} \;/\; (x,y) \in r^{I}\} \mid \leq n\}$$
$$(r_1 > r_2)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \;/ |\{y \in \Delta^{\mathcal{I}} : (x,y) \in r_1^{\mathcal{I}}\} \mid > |\{z \in \Delta^{\mathcal{I}} : (x,z) \in r_2^{\mathcal{I}}\}|\}$$
$$(r_1 < r_2)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \;/ |\{y \in \Delta^{\mathcal{I}} : (x,y) \in r_1^{\mathcal{I}}\} \mid < |\{z \in \Delta^{\mathcal{I}} : (x,z) \in r_2^{\mathcal{I}}\}|\}$$
$$(r_1 \sqcap \cdots \sqcap r_n)^{\mathcal{I}} = r_1^{\mathcal{I}} \cap \cdots \cap r_n^{\mathcal{I}}$$

IV. Modelling in Description Logics

At the beginning, DLs were regarded particularly as effective for fields where knowledge could be organized in a hierarchical structure, based on the relation ' is-a '.

The ability to represent and reason on taxonomies in DLs, justified their use as language of modelling in the study and maintenance of organisms of structured knowledge in a hierarchical way as well as their adoption like language of representation for formal ontology.

So that the designers are able to use DLs to model applications, it is significant that the concepts of Description logic are easily understandable; this will facilitate the use of the effective tools.

There are two principal alternatives to grow the use of DLs like language of modelling:

i. To provide a syntax which be like the natural language,
ii. To implement interfaces where the user can specify the structures of representation through graphic operations.

To model in DLs requires of the designer to specify the concepts of the field of discussion, to characterize their relationships to the other concepts and to specify also individuals.

V. Applications Developed With Description Logic Systems

We notice several applicability, some including Software, Engineering, Configuration, Medicine, Numerical Libraries and Information systems based on Web. There is several other applicability where DLs play a significant role, as the fields which include Treatment of Natural Language and Management of the Data bases. Some applications, whose creation lasted several years, arrived only at the level of prototype, but several among have the totality of the industrial systems several projects on the treatment natural language based on DLs were undertaken; some reached the level of industrial applications. We will see now, briefly, some fields of research which have relation with DLs.

a) The natural language

The use of DLs in the treatment of the natural language for knowledge representation can be used to communicate the meaning of the sentences. This knowledge is typically concerned by the meaning of the words (dictionary), and by the context i.e. a representation of the situation and the field of dialogue. The expressivity of the natural language also carries out to investigations concerning the extensions of DLs, such as for example it reason by defect. Work on the natural language required construction ontology.

b) Management of Data Bases

Knowledge and reasoning systems based on Dls, DL – KRS, management of data bases systems DBMS are present and very useful. A DBMS takes care of the persistence of the data and the management of a broad quantity of these data, whereas a DL – KRS manages intentional knowledge by keeping the base of knowledge in memory DLs are equipped with tools of reasoning which can revive the phase of conceptual modelling of some advantages, compared with traditional languages whose role is limited, concerning modelling. The second aspect of the improvement of the DBMS with DLs requires the query language.

c) Software Engineering

The Software Engineering is one of the first applicability of DLs. The principal idea was to implement an information system Software or a system which could help the developer of the software to find information in a wide Software system. One of the most original applications of DLs is Lassiesystem. Lassiesystem had a considerable success but ended up falling because of difficulty of the maintenance of its knowledge base. The idea of an information Software system and use of DLs survived like particular application and was used later by others Systems.

d) Configuration

The task of the configuration is to find a set of components which can be suitably connected in order to carry out a system which satisfies a given specification. The task of the configuration appears in many industrial fields like telecommunication, car industry and constructions of buildings. By using DLs, we can exploit the capacity to classify the components and to organize in a taxonomy.

e) Medicine

Medicine is also a field where the expert systems were developed since 1980, however, the complexity of the medical field requires a variety in the use of the DL – KRS. The need to deal with large range for knowledge bases (100000 concepts) leads to development of specialized systems such as Galen.

VI. Conclusion

Description Logics are responsible for several basic concepts in Knowledge Representation and Reasoning. The most significant aspect of work on DLs was certainly the union between the theory and practice. Descriptions Logics are not only theoretical formalism reserved to the theorists of Knowledge Representation, research around Description Logics is very active and has practical and theoretical aiming. Thus, the construction of systems dealing with the real problems is in the center of the concerns of many research tasks. Description Logics are not fixed formalisms; they are sufficiently flexible to accept the introduction of new constructors, able to meet particular needs. In this paper, we introduced two new operators, the operator less and the operator more, who allowed us to obtain a new extension of the logic of description AL. These operators will find certainly an applicability in one of the fields quoted previously.

References Références Referencias

1. R.J. Brachman, J.G.Schmolze : An overview of the KL-ONE knowledge representation system Cognitive Science ,9, 171-216(1985).
2. R.J. Brachman, al : Living with Classic : When and how to use a KL-ONE like language PIn J. F. SOWA, Ed., Principles of semantic networks, chapter 14, 401-456 (1991).
3. Donini, al : Reasoning in description logics. In G. Brewka, editor, Principles of Knowledge Representation, CSLI Publications, Stanford (CA), USA, 191-236 (1996).
4. E. Franconi: Description Logics (http ://www.cs.man.ac.uk/òfranconi).
5. A. Napoli: Une brève introduction aux logiques de descriptions Rapport de Recherche RR-3314, INRIA, (1997).
6. D. Nardi, R.J.Brachman: An Introduction to Description Logics In the Description Logic Handbook, edited by F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider, Cambridge University Press, 5-44 (2002).
7. W.A Woods, J.G. Schmolze: The KL-ONE family Computers, Mathematics with Applications, 23(25) :133-177, (1992).

This page is intentionally left blank