



HAL
open science

Semantic and flexible query processing of medical images using ontologies

Yahia Chabane

► **To cite this version:**

Yahia Chabane. Semantic and flexible query processing of medical images using ontologies. Medical Imaging. Université Blaise Pascal - Clermont-Ferrand II, 2016. English. NNT : 2016CLF22784 . tel-01814913

HAL Id: tel-01814913

<https://theses.hal.science/tel-01814913>

Submitted on 13 Jun 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



N° d'ordre: D. U : 2784
ED SPIC: 788



UNIVERSITÉ Blaise Pascal Clermont-Ferrand II
ÉCOLE DOCTORALE
SCIENCES POUR L'INGENIEUR DE CLERMONT
FERRAND

T H È S E

Présentée par

M. Yahia CHABANE

soutenue le

19 Décembre 2016

en vue de l'obtention du grade de

DOCTEUR de l'UNIVERSITÉ BLAISE PASCAL

Spécialité : Informatique

Arrêté du 07 août 2006

Titre :

**Semantic and flexible query processing of
medical images using ontologies.**

MEMBRES du JURY :

Mme. Hélène JAUDOIN	Maître de conférence	Université de Rennes 1	(<i>Examinatrice</i>)
M. Mohand-Saïd HACID	Professeur	Université Claude Bernard	(<i>Rapporteur</i>)
M. Ladjel BELLATRECHE	Professeur	ISAE - ENSMA	(<i>Rapporteur</i>)
M. Farouk TOUMANI	Professeur	Université Blaise Pascal	(<i>Directeur de thèse</i>)
M. Christophe REY	Maître de conférence	Université Blaise Pascal	(<i>Encadrant</i>)

Acknowledgements

Je tiens à remercier mon directeur de thèse Mr. Farouk TOUMANI de m'avoir accepté et pour son suivi, ses orientations et ses précieux conseils, et qui a su me faire profiter de sa grande expérience.

Je remercie mon encadrant Mr. Christophe REY pour son aide et sa disponibilité.

Je remercie Mme. Hélène JAUDOIN, d'avoir acceptée de faire partie de mon jury de thèse et mes rapporteurs M. Mohand-Saïd HACID et M. Ladjel BELLATRECHE.

Je remercie M. Laurent D'ORAZIO, porteur du projet SYSEO qui a financé ma thèse, pour son aide et sa gentillesse.

Je remercie M. Alain QUILLOT, qui a accepté de m'accueillir au LIMOS, et à tous les membres et personnels de ce laboratoire qui ont toujours fait preuve de gentillesse.

Un grand merci à toutes ma familles pour leur présence, leur préoccupation, leurs encouragements et le souci qu'ils se sont fait pour moi.

Enfin, je remercie, de tout coeur, tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

Yahia CHABANE

Dedication

A ma mère qui s'est sacrifiée pour nous offrir une vie meilleure.

A mon père qui m'a toujours soutenu.

A mon épouse pour sa patience.

A mes deux frères, à toute ma famille.

A tous mes amis.

Je dédie ce travail.

Yahia CHABANE

Abstract

Efficiently querying images using an image retrieval system is a long standing and challenging research problem.

In the medical domain, images are increasingly produced in large quantities due their increasing interest for many medical practices such as diagnosis, report writing and teaching. This thesis proposes a semantic-based gastroenterological images annotation and retrieval system based on a new polyp ontology that can be used to support physicians to decide how to deal with a polyp. The proposed solution uses a polyp ontology and rests on an adaptation of standard reasonings in description logic to enable semi automatic construction of queries and image annotation.

A second contribution of this work lies in the proposition of a new approach for computing relaxed answers of ontological queries based on a notion of an edit distance of a given individual w.r.t. a given query. Such a distance is computed by counting the number of elementary operations needed to be applied to an ABox in order to make a given individual a correct answer to a given query. The considered elementary operations are adding to or removing from an ABox, assertions on atomic concept, a negation of an atomic concept or an atomic role. The thesis proposes several formal semantics for such query approximation and investigates the underlying decision and optimisation problems.

Keywords: ontological queries, relaxation reasoning, semantic medical image retrieval, polyp ontology.

Résumé

L'interrogation efficace d'images en utilisant un système de recherche d'image est un problème qui a attiré l'attention de la communauté de recherche depuis une longue période.

Dans le domaine médical, les images sont de plus en plus produites en grandes quantités en raison de leur intérêt croissant pour de nombreuses pratiques médicales comme le diagnostic, la rédaction de rapports et l'enseignement. Cette thèse propose un système d'annotation et recherche sémantique d'images gastroentérologiques basé sur une nouvelle ontologie des polypes qui peut être utilisée pour aider les médecins à décider comment traiter un polype. La solution proposée utilise une ontologie de polype et se base sur une adaptation des raisonnements standard des logiques de description pour permettre une construction semi-automatique de requêtes et d'annotation d'images.

Une deuxième contribution de ce travail consiste dans la proposition d'une nouvelle approche pour le calcul de réponses relaxées des requêtes ontologiques basée sur une notion de distance entre un individu donné et une requête donnée. Cette distance est calculée en comptant le nombre d'opérations élémentaires à appliquer à une ABox afin de rendre un individu donné x , une réponse correcte à une requête. Ces opérations élémentaires sont l'ajout à ou la suppression d'une ABox, d'assertions sur des concepts atomiques (ou leur négation) et/ou des rôles atomiques. La thèse propose plusieurs sémantiques formelles pour la relaxation de requêtes et étudie les problèmes de décision et d'optimisation sous-jacents.

Mots-clés : Requêtes ontologiques, raisonnement de relaxation, recherche sémantique d'images médicales, ontologie de polype.

Contents

1	Introduction	1
1.1	Context of the study and the statement of the problem	1
1.2	Main contributions	3
1.3	Organization of the thesis	5
2	Foundations of Description Logics	7
2.1	Introduction	8
2.2	Introductory example	8
2.3	Syntax	11
2.3.1	TBox	11
2.3.2	ABox	12
2.3.3	DL Languages	13
2.4	Semantics	14
2.4.1	Open World Assumption Semantics	16
2.4.2	Closed World Assumption Semantics	16
2.5	Reasoning Tasks	18
2.5.1	Reasoning Tasks for Concepts	18
2.5.2	Reasoning Tasks for ABoxes	20
2.5.2.1	Instance Retrieval	21
2.5.2.2	Variants of instance retrieval	21
2.6	The DL Family	21
2.6.1	Monotonic DL	22
2.6.2	Non-monotonic DL	22
2.7	Conclusion	22
3	A Description Logic Based Approach For Semantic Image Retrieval	25
3.1	Introduction	26
3.2	Image retrieval	26
3.2.1	Image retrieval techniques	26
3.2.1.1	Content-based image retrieval	27
3.2.1.2	Semantic-based image retrieval	28

3.2.2	Facets of image retrieval	28
3.2.2.1	Query modalities	29
3.2.2.2	Images annotation	30
3.2.2.3	Images similarity	31
3.2.3	Semantic image retrieval using DL	33
3.2.4	Medical image retrieval	35
3.3	Semantic gastroenterological image retrieval	36
3.3.1	Ontology	39
3.3.2	Annotation	42
3.3.3	Reasoning	45
3.3.4	Prototype and experiments	46
3.4	Conclusion	48
4	\mathcal{W}Relaxation : A Relaxed Answer For Ontological Queries	51
4.1	Introduction	52
4.2	Using \mathcal{W} Relaxation with a medical images ontology	53
4.3	Motivation and main ideas	54
4.4	A new relaxation based on the ABox	56
4.4.1	\mathcal{W} Relaxation by considering separately the individuals	57
4.4.1.1	\mathcal{W}^+ Relaxation: relaxation by adding assertions	57
4.4.1.2	\mathcal{W}^- Relaxation: relaxation by deleting assertions	59
4.4.1.3	\mathcal{W} Relaxation: relaxation by adding/deleting assertions	60
4.4.2	\mathcal{W} Relaxation by considering a set of individuals	62
4.4.2.1	\mathcal{W}^+ Relaxation: relaxation by adding assertions	62
4.4.2.2	\mathcal{W}^- Relaxation: relaxation by deleting assertions	63
4.4.2.3	\mathcal{W} Relaxation: relaxation by adding/deleting assertions	64
4.5	Comparison w.r.t. existing reasonings	64
4.5.1	ABox abduction	65
4.5.2	Other reasonings	67
4.5.2.1	Instance retrieval reasoning	67
4.5.2.2	Approximation	68
4.5.2.3	ABox revision/update/erasure	71
4.6	Conclusion	73

5	\mathcal{W}Relaxation: Algorithms And Results	75
5.1	Introduction	76
5.2	Decision Problems	76
5.3	Optimisation Problems	77
5.3.1	Computing minimal \mathcal{W} for \mathcal{W}^+ Relaxation	77
5.3.2	Computing minimal \mathcal{W} for \mathcal{W}^- Relaxation	82
5.3.3	Computing minimal \mathcal{W} for \mathcal{W} Relaxation	83
5.3.4	Complexity	87
5.4	Optimisation problems related to \mathcal{W} Relaxation of set of individuals . . .	88
5.5	Conclusion	89
6	Conclusions and Future Work	91
	List of figures	93
	List of tables	97
A	Appendix	109
A.1	Proofs	109
A.1.1	Proof of Theorem 5.2.2	109
A.1.2	Proof of Theorem 5.2.4	110
A.1.3	Proof of Lemma 5.3.2	110
A.2	Other problems definition	112
A.2.1	Optimisation problems	113

Introduction

1.1 Context of the study and the statement of the problem

Textual data are very limited for information representation. Therefore, multimedia technology, mainly images, are increasingly used. Works in this research area are multiple [28, 60] and the problems are various [68, 52, 62]. Image databases now represent very large volume of information but they are unfortunately poorly exploited. Querying and managing image databases is becoming a big challenge in computer science world. To effectively manage and use these image databases, an image retrieval system is required. This is why it has become a very important area for a long time. An image retrieval system manages access to several images, the images are represented and indexed according to the used approach and compared using various approaches to generated queries using specific techniques. In summary, an image retrieval system manages the image representation, the query construction and the selection of images.

In the medical domain, images are increasingly produced in large quantities. These images are stored for long durations, the hospitals must keep an images database measured in tera bytes. Images are becoming increasingly important and necessary for some medical practices such as diagnoses, report writing and teaching. Management and access to these images become capital but also complex. Important decisions are taken based on these images like for example, the gastroenterology [23, 24], where physicians, by exploiting past experiences try to retrieve images of previously treated cases similar to the present treated case to decide whether they remove a type of tumor called a polyp or not.

Medical images are commonly targeted by image retrieval systems and the medical domain is one of the main application cited by Content Based Image Retrieval (CBIR) systems [68, 81], but the most current access to medical images [68] is based on the patient's ID and other characteristics present in a DICOM (Digital Imaging and COmmunications in Medicine) file. DICOM [1] is a standard for the exchange of medical

images, information about patients and other information (concerning the examination for example) which are stored with the image in the same file. These files are stored and managed by systems called PACS (Picture Archiving and Communication System).

The SYSEO project is one of the projects supported by the ANR¹ in the framework of the technologies for the health and the autonomy (TecSan) program. It deals mainly with problems of data management domain, medical images domain and human computer interaction domain. This project aims at addressing several challenges underlying the production, the storage, the retrieval, the sharing and the manipulation of medical images. It introduces a software solution to improve the quality of gastroenterology examination by comparing the images taken during the examination with the old images of the patient and proposes a collaborative platform for mutual assistance between physicians. This software solution is also used for the images archiving, the teaching and the patient monitoring.

The medical practices and the particularity of the gastroenterology domain is one of the challenges of the SYSEO project. Images, for example, are acquired, interpreted and operated directly by the physician in real time. The chain of the process proposed by SYSEO, must respect the constraints of the gastroenterology domain and the examinations, like the real time constraint and the confidentiality of the patients data, to preserve the very limited duration of an examination and the number of treated patients per day and protect the patients data.

This work investigates one of the challenges of SYSEO project, which is semantic image retrieval. Indeed, we propose to physicians an effective and flexible semantic gastroenterological images annotation and retrieval system based on a new polyp ontology system. This system is mainly used for teaching, research and diagnoses. The problems of ontology construction, image annotation using an ontology, reasoning on an ontology are scrutinized and a description logics approach based is proposed. Description logics [15] are used as formalism to construct the polyp ontology, represent images and for reasoning about knowledge. A polyp ontology is constructed which will be used by an image retrieval system. This ontology is the knowledge base of the system, it is used to define the representation of an image and the important elements of gastroenterology as standard classifications of polyps and for reasoning about all these definitions.

Semantic medical image retrieval generally uses medical ontologies. The purpose of medical ontologies is to provide a common vocabulary with formal definitions. These

¹Agence Nationale de la Recherche. <http://www.agence-nationale-recherche.fr/>

definitions can be used to represent images and medical assessments. It helps to standardize the interpretation of medical images for better treatment. The challenges of these approaches are: the tradeoff between the expressivity of image representations and reasoning performance in terms of relevance and computation time, adapted interfaces to medical domain and conditions of the examinations and the construction of ontology (reuse ? adapt ? create ? ...) and its maintenance for approaches based on ontologies.

The context of medical domain and medical practices is very complicated and special, there is a gap between the theoretical approaches and the implemented systems. Several powerful techniques in other domains have not been tested. For example, the non utilisation of user feedback [68] justified by their absence due to the medical practices and the very limited time. Another example is the lack of investment in interfaces adapted to the working conditions [68]. The evaluation techniques and benchmarks are also lacking in the medical domain, although in 2004 a medical image retrieval task was added to ImageCLEF which is a part of the Cross Language Evaluation Forum (CLEF) [4].

1.2 Main contributions

In this thesis, two main problems are investigated: a semantic gastroenterological image retrieval and a relaxed answers for ontological queries. The main contributions of the work are structured as follows:

1. *WR*elaxation

A new approach is proposed for computing relaxed answers of ontological queries based on a notion of an edit distance of a given individual w.r.t. a given query. Such a distance is computed by counting the number of elementary operations needed to be applied to an ABox in order to make a given individual a correct answer to a given query. The considered elementary operations are added to or removed from an ABox, assertions of the form of an atomic concept (or its negation) and/or atomic roles. The main goal is to increase the number of concepts instances according to some defined criteria. In particular, we investigate:

- *WR*elaxation by considering separately the individuals: The general idea is to determine the minimum knowledge to add to or to remove from an ABox so that a given individual can be considered as an instance of a given concept. Such a quantity of knowledge is defined by counting the elementary operations

performed over the ABox.

The individuals are integrated into the instances of a concept one by one.

- *WRelaxation by considering a set of individuals*: After having presented the limits of the integration of individuals one by one, we redefine the WRelaxation by taking into account the interaction between the elementary operations which allow to integrate each individual. The objective is to form a single consistent ABox which allows for the integration of a set of individuals in the relaxation of a given concept.
- *Algorithms for decision problems*: we explore a decision problem related to deciding the existence of \mathcal{W} operations needed to make a given individual a instance of a given concept C , and we reduce these problems to known problems of checking the consistency of an *ABox* w.r.t. a *TBox*.
- *Algorithms for optimisation problems*: we study some optimisation problems, related to the computation of the minimal distance \mathcal{W} , needed to make a given individual a instance of a given concept C , and related to computation of relaxed answers by considering a set of individuals, in the case of the three WRelaxation. We propose three algorithms inspired from tableau algorithm to solve some optimisation problems, related to the computation of the minimal distance \mathcal{W} .
- *Application of WRelaxation*: we present the utilisation of WRelaxation with a medical images ontology to improve the semantic images retrieval and to enrich and/or to correct an existing images annotations database.

2. Semantic gastroenterological image retrieval

We propose and implement a semantic gastroenterological images annotation and retrieval system based on a new polyp ontology to help physicians, to decide whether they remove a polyp or not. The system includes:

- *Ontology*: It represents the knowledge base of the system, and it is used to define the representation of an image and the important elements of gastroenterology as standard classifications of polyps and for reasoning about all these definitions (WRelaxation, classification, instance retrieval, etc.).

We use description logics as formalism to define the polyp ontology. The knowledge collected about polyps is used for diagnosing cancer which is based

on the analysis of polyps present in an organ. The principle is to clearly identify the type and the state of the polyp by retrieving its standard classes used in gastroenterology.

- *Annotation*: An annotation (or query) of an image is manually generated using an interactive interface. We use semantic information to describe the content of the image, i.e., concepts, individuals or roles of the polyp ontology.
- *Reasoning*: we have implemented in our system three types of reasoning: a standard reasoning which is the instance retrieval and two other reasonings, a local subsumees retrieval and a local subsumers retrieval which are a slight modifications of the standard subsumption reasoning. These reasonings allow us to make a classification and an image retrieval.

1.3 Organization of the thesis

This thesis is organized as follows:

Chapter 2 This chapter presents the foundations of description logics, which include the syntax, two assumption semantics, the reasoning and the monotony property.

Chapter 3 This chapter introduces the two main techniques of image retrieval, the different facets of image retrieval, then the medical image retrieval and an overview of a semantic image retrieval using description logics, and finally, the proposed approach of semantic gastroenterological images retrieval based on a new polyp ontology.

Chapter 4 We highlight our main contribution in this chapter, which is a new reasoning for computing relaxed answers of ontological queries. We start by presenting the motivations and the general idea of our reasoning. Then, the definition of our reasoning and its implementation in the medical domain. Finally, we compare our reasoning with other approaches.

Chapter 5 In this chapter, a decision and optimization problems for \mathcal{WR} Relaxation are described, algorithms to solve these problems are proposed.

Chapter 6 This chapter is the conclusion of the thesis, we synthesizes and summarizes the major perspectives of this work.

Foundations of Description Logics

Contents

2.1	Introduction	8
2.2	Introductory example	8
2.3	Syntax	11
2.3.1	TBox	11
2.3.2	ABox	12
2.3.3	DL Languages	13
2.4	Semantics	14
2.4.1	Open World Assumption Semantics	16
2.4.2	Closed World Assumption Semantics	16
2.5	Reasoning Tasks	18
2.5.1	Reasoning Tasks for Concepts	18
2.5.2	Reasoning Tasks for ABoxes	20
2.6	The DL Family	21
2.6.1	Monotonic DL	22
2.6.2	Non-monotonic DL	22
2.7	Conclusion	22

2.1 Introduction

Description logics (DL) are a knowledge representation and reasoning formalism [15] heavily used in semantic web area. DL form the basic of Web Ontology language (OWL) [6], one of the main standards in semantic technologies. When modelling knowledge using DL, we define sets of elements and relations between these elements. The elements are called *individuals*, the sets of elements are *concepts*, and the sets of pairs of elements are *roles*.

For description logics, a domain of interest is defined by: an extentional part, which is a set of individuals organized in/by an intentional part, which is a set of concepts and roles. A concept is a category of individuals. The concepts can be defined using other concepts and roles and are structured as a hierarchy by axioms. Binary relations between individuals are defined by roles. An axiom represent a relation between two components of the knowledge base. The axioms are generally subset relations between concepts, subset relations between roles, the membership of individuals to concepts and the membership of pairs of individuals to roles.

A description logic based knowledge base \mathcal{KB} is a pair $(TBox \mathcal{T}, ABox \mathcal{A})$. \mathcal{T} is the set of concepts, roles, concept axioms and role axioms of the knowledge base, \mathcal{A} is the set of individuals and individual axioms of the knowledge base.

We present in this chapter the syntax and semantics of a very expressive DL, namely the $\mathcal{ALCHOQ}(\neg)$ language (see the section 2.6 for DL languages) [15]. A language \mathcal{L}_1 is more expressive than another language \mathcal{L}_2 if all knowledge bases of \mathcal{L}_2 can be defined in \mathcal{L}_1 and there exist a knowledge base of \mathcal{L}_1 that cannot be defined in \mathcal{L}_2 [15]. There is a trade-off between expressiveness of a language and the complexity of reasoning in this language. Reaching a good compromise between the expressiveness of the language and reasoning about language has always been a challenge for knowledge representation languages [15].

2.2 Introductory example

Let a DL based knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ represented in Table 2.1. It describes briefly football.

\mathcal{T} is the *TBox*. It contains definitions of concepts, of roles, of concept axioms and of role axioms. \mathcal{A} is the *ABox*. It contains assertions of concepts for the definition of

\mathcal{T}	
<i>Person</i>	$\equiv Man \sqcup Woman$
<i>Athlete</i>	$\equiv Person \sqcap \exists play.Sport$
<i>SoccerPlayer</i>	$\equiv Athlete \sqcap \exists play.\{football\}$
<i>TeamSport</i>	$\equiv Sport \sqcap \geq 4 participant.Athlete$
<i>FootballTeam</i>	$\equiv = 1 play.TeamSport \sqcap \exists use.Ball \sqcap \forall engage.SoccerPlayer$
<i>FootballChampionship</i>	$\equiv Competition \sqcap \geq 3 haveCandidat.FootballTeam$
<i>SerieA</i>	$\sqsubseteq FootballChampionship$
<i>UefaChLeagueCup</i>	$\sqsubseteq Cup$

\mathcal{A}
<i>Woman(van), Athlete(bolt), TeamSport(football), SoccerPlayer(lopez), SoccerPlayer(torres), play(federer, tennis), SoccerPlayer(elshaarawy), TeamSport(rugby), FootballTeam(acMilan), SerieA(serieA_2014), Cup(uefaChLeagueCup_1994), cupCount(acMilan, 29), winCup(acMilan, uefaChLeagueCup_1994), winUefaChLeagueCup(acMilan, 1994), winUefaChLeagueCup(acMilan, 2007), uefaChLeagueCupCount(acMilan, 7)</i>

Table 2.1: Example of DL based knowledge base.

concept instances and assertions of roles for the definition of role instances. What follows is a more detailed view of the syntax and meaning of some definitions:

“*Person* $\equiv Man \sqcup Woman$ ”: is a definition of the concept *Person*. “ \equiv ” is a concept axiom called *equivalence*. It is used to indicate an equivalence between a concept in its right and another concept in its left. Generally the concept in its right is a concept name (named concept) and the concept in its left is a concept description built from concepts (and / or roles), and concepts (and / or roles) constructors. “ \sqcup ” is a concept constructor called *disjunction*. It is used to build a new concept from two concepts. The new concept is the union of the two sets of individuals of the two concepts.

In the definition of the second concept *Athlete* $\equiv Person \sqcap \exists play.Sport$, the concept constructor “ \sqcap ”, called *conjunction*, is used. It allows the construction of a new concept with two concepts. The new concept is the intersection of the two sets of individuals of the two concepts. “ \exists ” is a concept constructor using a role *R* and a concept *C*. All individuals of this new concept are connected by the role *R* to at least one individual of the concept *C*. Therefore, the concept *Athlete* is a set of individuals of type *Person* which are connected by the role *play* to at least one individual of type *Sport*. In other words, athletes are people who practice at least one sport. Note that the concept constructor “ \exists ” may not use a concept. In this case, the individuals of the built concept need just to

have a link with any other individual. Note also that the concept C may not be a concept name. For the definition of the concept *SoccerPlayer* for example, the concept is built directly and represents a single individual. The concept constructor “ $\{\}$ ” uses a set of individuals to create the concept representing them. “ \forall ” is a concept constructor using also a role R and a concept C . All individuals of a concept built with the constructor “ \forall ” are connected by the role R only to individuals of the concept C . Individuals of the concept “ $\forall engage.SoccerPlayer$ ” are connected by the role *engage* only to individuals of the concept *SoccerPlayer*. In the example, the football teams are defined by the concept *FootballTeam*, it must use at least a ball and engage only football players. By this definition, a football team does not have a coach or a president.

Other concepts constructors, “ \leq ”, “ \geq ” and “ $=$ ” (with a role R and sometimes a concept C) use constraints on the role cardinality to build a concept. These constraints are on the number of relations that an individual must have with other individuals. For example, the individuals of the concept “ $\geq 4 participant.Athlete$ ” must have at least four relations using the role *participant* with individuals of the concept *Athlete*. For “ $= 1 play.TeamSport$ ”, the individuals of this concept must have exactly one relations using the role *play* with individuals of the concept *TeamSport*.

A concept axiom is used in the example. It is the subsumption axiom “ \sqsubseteq ” which is used to define a subsumption relation between two concepts C and D . “*SerieA* \sqsubseteq *FootballChampionship*” means that all individuals of *SerieA* are also individuals of *FootballChampionship*.

The individual axioms are defined in the *ABox*. The axiom “ $C(a)$ ” defines the membership of an individual a to a concept C (we say that a is an instance of C) and another axiom “ $R(a, b)$ ” defines the membership of two individuals a and b to a role R (we say that the pair (a, b) is an instance of R). In the example, the individual *bolt* is an instance of the concept *Athlete* and the individual *federer* is connected to the individual *tennis* by the role *play*. Note that other individual axioms can be deduced when combining the *ABox* and the *TBox*. For example, as the individual *serieA_2014* is an instance of the concept *SerieA* and the concept *SerieA* is subsumed by the concept *FootballChampionship* then the individual *serieA_2014* is also instance of the concept *FootballChampionship*. Note also that the two instances of a role are generally individuals but the second may not be an individual. We can use concrete domains, which include integers, real, etc. For the axiom “*winUefaChLeagueCup(acMilan, 1994)*”, 1994 is not an individual but an integer.

We have illustrated in this example that the knowledge of a domain can be defined and represented by a set of concepts, of roles and of individuals. Now, we present in detail the components of a description logic system. A description logic system is defined by the knowledge base together with reasoning algorithms (exploitation of the knowledge base). The knowledge base (TBox and ABox) is defined using a language which includes constructors, axioms and used semantics. Reasoning algorithms are techniques for knowledge exploitation of a knowledge base to deduce (infer) implicit knowledge.

2.3 Syntax

The syntax of a description logic language includes the definition of role constructors, of concept constructors and the definition of role axioms, of concept axioms and of individual axioms. The sets of concept names, of role names and of individual names are respectively denoted N_C , N_R and N_I . We now present the syntax of different parts of the $\mathcal{ALCHOQ}(\neg)$ language which is used in the next chapter.

2.3.1 TBox

The *TBox* of a knowledge base contains all concepts and roles, their definitions and their axioms. The concepts (or roles) are defined using other concepts (roles) and concepts (or roles) constructors. Some concepts (or roles) have not a formal definitions, they are called atomic or primitive concepts (or roles). An atomic concept (or role) is intuitive, it is defined by the set of individuals which it represents and is used to formally define other concepts.

The $\mathcal{ALCHOQ}(\neg)$ *TBox* defines one role constructor which is the role complement. Let R be a description role. $S = \neg R$ is a new role defined using the role complement.

Let A be an atomic concept, C and D two concept descriptions, R a role, i_1, \dots, i_n individuals and n an integer. Concept descriptions in $\mathcal{ALCHOQ}(\neg)$ are formed using the concept constructors of Table 2.2. The first two constructors are interesting for the reasonings, they respectively define the most general concept of the terminology and the empty concept. The third constructor defines a concept that cannot be defined using other concepts or roles. The other constructors use operators, concepts and roles to define concepts. The semantics of these constructors is given in the section 2.4.

The $\mathcal{ALCHOQ}(\neg)$ *TBox* defines two concept axioms and two role axioms:

Concept constructors	Name
\top	Universal concept
\perp	Bottom concept
A	Atomic concept
$\neg C$	Full negation
$C \sqcap D$	Intersection
$C \sqcup D$	Union
$\exists R.C$	Qualified existential quantification
$\forall R.C$	Value restriction
$\geq nR.C$	Qualified at-least restrictions
$\leq nR.C$	Qualified at-most restrictions
$one-of(i_1, \dots, i_n)$	Individuals set

Table 2.2: Concept constructors of $\mathcal{ALCHOQ}(\neg)$

- The concept equality axiom and the concept inclusion axiom. We can define by the concept inclusion axiom a concept hierarchy. Let C and D be two concept descriptions. $C \sqsubseteq D$ is a concept inclusion axiom. For this axiom, D is a more general concept than C . All instances of C are also instances of D . The concept equality axiom defines equivalence relations between concepts. $C \equiv D$ defines equivalence relation between C and D . All instances of C are also instances of D and all instances of D are also instances of C .
- The role equality axiom and the role inclusion axiom. We can define by the role inclusion axiom a role hierarchy. Let R and S be two role descriptions. $R \sqsubseteq S$ is a role inclusion axiom. For this axiom, S is a more general role than R . All R couples are also S couples. The role equality axiom defines equivalence relations between roles. $R \equiv S$ defines equivalence relation between R and S . All R couples are also S couples and all S couples are also R couples.

2.3.2 ABox

The *ABox* of a knowledge base contains all assertional axioms (individual axioms). Let C be a concept description, R be a role description, i and j be two individuals. The individual axioms of $\mathcal{ALCHOQ}(\neg)$ are listed in Table 2.3. The first and second axioms are used respectively to define the instances of a concept and the instances of a role. The two following axioms are used to define the individuals non-instance of a concept and the individuals non-instance of a role. The last two axioms define the equality and

the non-equality between individuals, which are very useful for reasoning on the concepts defined with operators like qualified restrictions, value restriction etc.

Individual axioms	Name
$C(i)$	Concept assertion
$R(i, j)$	Role assertion
$\neg C(i)$	Negated concept assertion
$\neg R(i, j)$	Negated role assertion
$i = j$	Individuals equality
$i \neq j$	Individuals difference

Table 2.3: Individual axioms of $\mathcal{ALCHOQ}(\neg)$

We define below some functions that are used throughout this thesis:

- *Conc*: returns all concepts of a given *TBox*.
- *Ind*: returns all individuals of a given *ABox*.
- *InsR*: returns all instances of a given concept, i.e., $InsR(C, \mathcal{KB}) = \{a \in Ind(\mathcal{A}) \mid \mathcal{KB} \models C(a)\}$

2.3.3 DL Languages

A description logic language is defined by the set of its role axioms, concept axioms, individual axioms and the set of its concept constructors and role constructors. The basic DL language is \mathcal{AL} [15]. A concept description of \mathcal{AL} can be an atomic concept, a top concept, a bottom concept, a negation of atomic concept, an intersection of two concept descriptions. A concept description [42] could also be formed using a value restriction or a limited existential quantification. All roles of \mathcal{AL} are atomic.

\mathcal{AL} can be extended by adding concept or role constructors. Generally, each extension is assigned a letter or a symbol. For concept constructors, the letters/symbols are written after the starting \mathcal{AL} , for role constructors, the letters/symbols are written between parentheses at the end of the language name. For example, $\mathcal{ALCHOQ}(\neg)$ is an extension of \mathcal{AL} by adding the concept complement (C), the role hierarchy (H), the nominals (O), the qualified number restrictions(Q) and the role complement.

A description logic complexity navigator is proposed in [94]. DL languages are formed by selecting a set of constructors and axioms. The name of the formed language is also given.

Among the most popular DL languages, we cite the *DL-Lite* family [21, 22]. The *DL-Lite* family is a hierarchy of DL languages. It captures the main notions of both ontologies and conceptual modeling formalisms used in databases and software engineering. The core language of this family is *DL-Lite_{core}*. Its syntax is the basic syntax of the whole family, all other languages are derived from this language (extension of the set of constructors or/and the set of axioms).

Let A be an atomic concept, P be an atomic role, P^- be the inverse of the atomic role P , C be a general concept, E be a general role and B be a basic concept, that is an atomic or a concept of the form $\exists R$, where R is a basic role, that is an atomic role or the inverse of an atomic role.

Concepts and roles of *DL-Lite_{core}* language are formed according the following syntax [21]:

$$\begin{array}{ll} B \longrightarrow A \mid \exists R & R \longrightarrow P \mid P^- \\ C \longrightarrow B \mid \neg B & E \longrightarrow R \mid \neg R \end{array}$$

The *DL-Lite_{core}* *TBox* defines a set of concept inclusion axioms of the form: $B \sqsubseteq C$ (B is a basic concept and C is a general concept) and the *DL-Lite_{core}* *ABox* is formed by a finite set of membership assertions on atomic concepts and on atomic roles, of the form: $A(a)$ or $P(a, b)$.

The *DL-Lite_R* extends *DL-Lite_{core}* with the role inclusion axiom of the form: $R \sqsubseteq E$ (R is a basic role and E is a general role). *DL-Lite_R* is a strict subset of OWL Lite [6].

The *DL-Lite_F* extends *DL-Lite_{core}* with the ability of specifying functionality on roles or on their inverses, of the form: (funct R) where R is a basic role. *DL-Lite_F* can be seen as an extension of the ontology language RDFS [9].

Let \mathcal{L} be a DL language, we denote by \mathcal{L} -concept a concept expressed using \mathcal{L} . Similarly, we denote by \mathcal{L} -TBox (\mathcal{L} -ABox) a TBox (an ABox) which all of these axioms are expressed using \mathcal{L} , and by \mathcal{L} -KB a knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ if \mathcal{T} is a \mathcal{L} -T and \mathcal{A} is a \mathcal{L} -A.

2.4 Semantics

The semantics of a DL is given in terms of interpretations, where an interpretation \mathcal{I} is defined by [15]:

- an *interpretation domain* $\Delta^{\mathcal{I}}$ which defines the elements of the domain which are represented by the knowledge base.

- an *interpretation function* \cdot^I which assigns to every element (concept, role or individual) of the knowledge base an element of the interpretation domain Δ^I :
 - assigning for every individual name $a \in N_I$ an element $a^I \in \Delta^I$ from the interpretation domain.
 - assigning for every concept name $A \in N_C$ a set of elements $A^I \subseteq \Delta^I$ from the interpretation domain.
 - assigning for every role name $r \in N_R$ a set of ordered pairs of elements $r^I \subseteq \Delta^I \times \Delta^I$ from the interpretation domain.

The interpretation function is extended to $\mathcal{ALCHOQ}(\neg)$ concept description by the inductive definitions of Table 2.4 [15].

Concept constructors	Semantics
\perp	$\perp^I = \emptyset$
\top	$\top^I = \Delta^I$
A	$A^I \subseteq \Delta^I$
$\neg C$	$(\neg C)^I = \Delta^I \setminus C^I$
$C \sqcap D$	$(C \sqcap D)^I = C^I \cap D^I$
$C \sqcup D$	$(C \sqcup D)^I = C^I \cup D^I$
$\exists R.C$	$(\exists R.C)^I = \{x \in \Delta^I \mid \exists y : (x, y) \in R^I \wedge y \in C^I\}$
$\forall R.C$	$(\forall R.C)^I = \{x \in \Delta^I \mid \forall y, (x, y) \in R^I \rightarrow y \in C^I\}$
$\geq nR.C$	$(\geq nR.C)^I = \{x \in \Delta^I \mid \text{Card}(\{y \in C^I \mid (x, y) \in R^I\}) \geq n\}$
$\leq nR.C$	$(\leq nR.C)^I = \{x \in \Delta^I \mid \text{Card}(\{y \in C^I \mid (x, y) \in R^I\}) \leq n\}$
$\text{one-of}(i_1, \dots, i_n)$	$\text{one-of}(i_1, \dots, i_n)^I = \{i_1^I, \dots, i_n^I\}$

Table 2.4: Semantics of $\mathcal{ALCHOQ}(\neg)$

The semantics of the other components is given in terms of satisfiability. An interpretation I satisfies a [15]:

- concept inclusion axiom $C \sqsubseteq D$ iff $C^I \subseteq D^I$
- concept equality axiom $C \equiv D$ iff $C^I = D^I$
- role inclusion axiom $R \sqsubseteq S$ iff $R^I \subseteq S^I$
- role equality axiom $R \equiv S$ iff $R^I = S^I$
- concept assertion axiom $C(a)$ iff $a^I \in C^I$

- role assertion axiom $R(a, b)$ iff $(a^I, b^I) \in R^I$
- $TBox \mathcal{T}$ iff it satisfies every axiom of \mathcal{T} .
- $ABox \mathcal{A}$ iff it satisfies every assertion of \mathcal{A} .
- $ABox \mathcal{A}$ with respect to a $TBox \mathcal{T}$ iff in addition to satisfying \mathcal{A} , it satisfies \mathcal{T} .
- knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ iff it satisfies both \mathcal{T} and \mathcal{A} .

If an interpretation I satisfies an axiom ax or a knowledge base \mathcal{KB} , then it is a model of ax noted $I \models ax$ or of \mathcal{KB} noted $I \models \mathcal{KB}$. A knowledge base \mathcal{KB} is satisfiable iff a model M such that $M \models \mathcal{KB}$ exists.

The entailment in DL aims at making explicit the implicit knowledge. Two semantics are defined for the entailment in the context of description logics, the open world assumption (OWA) [15] and the closed world assumption (CWA) [61, 13, 37]. OWA and CWA are two standard assumptions upon which entailments in a DL system may be based. The difference is that for the CWA, any knowledge which cannot be entailed is false, but for OWA, any knowledge which cannot be entailed may be true. For the same knowledge base, we can entail different knowledge depending on which assumption is made.

2.4.1 Open World Assumption Semantics

Open world assumption is a widely used semantic in the context of description logics. An $ABox$ represents many different interpretations. The absence of information in an $ABox$ only indicates lack of knowledge. A knowledge not belonging to the knowledge base can be true or false. Only, a part of the knowledge of the world is represented.

The entailment in the context of OWA is defined by [15]:

Definition 2.4.1 *A knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ entails an axiom α (written $\mathcal{KB} \models \alpha$) iff for every model M of \mathcal{KB} , $M \models \alpha$ (i.e., $M \models \mathcal{KB}$ implies $M \models \alpha$).*

2.4.2 Closed World Assumption Semantics

Closed world assumption semantics is illustrated by the databases theory [63, 11]. Knowledge must be explicitly represented or inferred. Anything that is not inferred is considered false.

Classical entailment model [37] of a closed world is based on a closure operation on the knowledge base. The closed world assumption of a knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ is a set of assertions defined by $CWA(\mathcal{KB}) = \{\neg p \mid p \in AS \text{ and } \mathcal{KB} \not\models p\}$ where AS is the set of all assertions $A(a)$ and $R(a, b)$ constructable from atomic concepts A and atomic roles R . The entailment in the context of CWA is defined by [37] as follows:

Definition 2.4.2 *A knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ entails an assertion α considering closed world assumption (written $\mathcal{KB} \models_{cwa} \alpha$) iff $\mathcal{KB}^+ \models \alpha$, with $\mathcal{KB}^+ = \mathcal{KB} \cup CWA(\mathcal{KB})$.*

Let us consider a knowledge base \mathcal{KB} , with a classical entailment model of CWA, for every atomic concepts A and for every individuals i , we have $\mathcal{KB} \models A(i)$ or $\mathcal{KB} \models \neg A(i)$. The problem with this model is that we can have both $\mathcal{KB} \models C(i)$ and $\mathcal{KB} \models \neg C(i)$ for a concept C . A simple example of this case is illustrated in Example 1.

Example 1 *Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be the following knowledge base.*

$\mathcal{T} = \{C \equiv A \sqcup B\}$ and $\mathcal{A} = \{C(a)\}$.

$CWA(\mathcal{KB}) = \{\neg A(a), \neg B(a)\}$.

\mathcal{KB} is inconsistent. We have $\mathcal{KB} \models_{cwa} C(a)$, $\mathcal{KB} \models_{cwa} \neg A(a)$ and $\mathcal{KB} \models_{cwa} \neg B(a)$ therefore $\mathcal{KB} \models_{cwa} \neg C(a)$.

\mathcal{KB} entails $C(a)$ and $C \equiv A \sqcup B$ then \mathcal{KB} entails $A(a)$ or \mathcal{KB} entails $B(a)$ certainly, but as the case is non-deterministic, we can have three possible cases of entailment:

- $\mathcal{KB} \models_{cwa} A(a)$ and $\mathcal{KB} \models_{cwa} B(a)$
- $\mathcal{KB} \models_{cwa} \neg A(a)$ and $\mathcal{KB} \models_{cwa} B(a)$
- $\mathcal{KB} \models_{cwa} A(a)$ and $\mathcal{KB} \models_{cwa} \neg B(a)$

Therefore, at least one of the two assertions $A(a)$ and $B(a)$ must be entailed, but none of these two assertions cannot be entailed from the \mathcal{KB} and $\neg A(a)$ and $\neg B(a)$ are added to \mathcal{KB} what makes \mathcal{KB}^+ inconsistent.

To tackle this problem, the generalized closed world assumption (GCWA) is proposed in [37]. GCWA is a generalization of CWA which ensures the consistency of a knowledge base.

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a consistent DL based knowledge base and let AS be the set of all assertions of the form $A(a)$ or $R(a, b)$ with A an atomic concept and R an atomic

role and a, b two individuals. Let PK be the set of all positive clauses constructed from AS . A positive clause in this case is a disjunction of assertions $A(a)$ and $R(a, b)$. The generalized closed world assumption set of \mathcal{KB} , annotated $GCWA(\mathcal{KB})$, is defined as follows: $GCWA(\mathcal{KB}) = \{\neg p \mid p \in AS, \forall K \in PK, \mathcal{KB} \not\models p, \text{ and if } \mathcal{KB} \models (p \vee K) \text{ then } \mathcal{KB} \models K\}$. The entailment in the context of GCWA is defined by [37]:

Definition 2.4.3 *A knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ entails an assertion α considering generalized closed world assumption (written $\mathcal{KB} \models_{gcwa} \alpha$) iff $\mathcal{KB}^* \models \alpha$, with $\mathcal{KB}^* = \mathcal{KB} \cup GCWA(\mathcal{KB})$.*

Back to the introductory example (section 2.2): $\{\neg Man(acMilan), \neg Woman(acMilan), \neg winCup(acMilan, van), \neg winCup(acMilan, bolt), \neg winCup(acMilan, football), \neg winCup(acMilan, lopez), \neg winCup(acMilan, torres), \neg winCup(acMilan, federer), \neg winCup(acMilan, tennis), \neg winCup(acMilan, elshaarawy), \neg winCup(acMilan, rugby), \neg winCup(acMilan, acMilan), \neg winCup(acMilan, serieA_2014)\} \subset GCWA(\mathcal{KB})$

Hence, $\mathcal{KB} \models_{gcwa} (\neg SoccerPlayer \sqcap \forall winCup.UefaChLeagueCup)(acMilan)$

In this thesis, we use the context of the Open World Assumption (OWA) and the Generalized Closed World Assumption (GCWA) approaches.

2.5 Reasoning Tasks

Reasoning means making explicit the implicit knowledge that is logically inferred from the knowledge contained in a $TBox$ and an $ABox$.

2.5.1 Reasoning Tasks for Concepts

Concepts are the intentional part of the knowledge base. They allow to regroup individuals, to structure and organize a domain. The properties of concepts and the relations between concepts are source of logical inferences.

For example, testing the satisfiability of a concept; a concept C is *satisfiable* w.r.t. a knowledge base \mathcal{KB} if there exist an interpretation that is a model of \mathcal{KB} such that C denotes a nonempty set in this interpretation. When a concept C is unsatisfiable, the knowledge: *for all individuals i of the $ABox$, i is not a instance of C* becomes explicit (logically inferred) by this property.

Another example is the *subsumption problem*. The problem is to try to retrieve a subsumption property between two concepts. A concept C is *subsumed* by a concept D if for every model of \mathcal{KB} the instance of C is a subset of the instance of D . If a concept C is subsumed by a concept D , the knowledge ‘for all instances i of C , i is an instance of D ’ is logically inferred.

Let a knowledge base \mathcal{KB} . Some properties of concepts are [15]:

- **Satisfiability:** A concept C is *satisfiable* w.r.t. a knowledge base \mathcal{KB} if there exist a model I of \mathcal{KB} such that the interpretation of C in \mathcal{I} is nonempty i.e. $C^{\mathcal{I}} \neq \emptyset$.
- **Subsumption:** A concept C is *subsumed* by a concept D (denoted $C \sqsubseteq D$) w.r.t. a knowledge base \mathcal{KB} (denoted $C \sqsubseteq_{\mathcal{KB}} D$ or $\mathcal{KB} \models C \sqsubseteq D$) if for every model I of \mathcal{KB} the interpretation of C in \mathcal{I} is a subset of the instances of D i.e. $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$.
- **Equivalence:** Two concepts C and D are *equivalent* w.r.t. a knowledge base \mathcal{KB} if for every model I of \mathcal{KB} the interpretation of C in \mathcal{I} is equal to the interpretation of D in \mathcal{I} i.e. $C^{\mathcal{I}} = D^{\mathcal{I}}$.
- **Disjointness:** Two concepts C and D are *disjoint* w.r.t. a knowledge base \mathcal{KB} if for every model I of \mathcal{KB} the intersection of the interpretation of C in \mathcal{I} and the interpretation of D in \mathcal{I} is empty i.e. $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$.

The four properties are related to one another. If the language allows, a property can be reduced to another. Generally, the reasoning algorithms are implemented for one property and derived for the other properties.

For two concepts C and D , if the language provides the intersection operator “ \sqcap ” and contains the unsatisfiable concept “ \perp ”, we have the following reduction to subsumption [15]:

- C is unsatisfiable $\Leftrightarrow C \sqsubseteq \perp$.
- C and D are equivalent $\Leftrightarrow C \sqsubseteq D$ and $D \sqsubseteq C$.
- C and D are disjoint $\Leftrightarrow C \sqcap D \sqsubseteq \perp$.

The reduction to unsatisfiability is possible if the language provides the intersection operator and the negation operators; for two concepts C and D , we have [15]:

- C is subsumed by $D \Leftrightarrow C \sqcap \neg D$ is unsatisfiable.

- C and D are equivalent \Leftrightarrow both $(C \sqcap \neg D)$ and $(\neg C \sqcap D)$ are unsatisfiable.
- C and D are disjoint $\Leftrightarrow C \sqcap D$ is unsatisfiable.

2.5.2 Reasoning Tasks for ABoxes

Reasoning tasks for *ABoxes* are about individuals which are the extensional part of the knowledge base. The main *ABox* properties which are sources of inferences are [15]:

- **Consistency:** An *ABox* \mathcal{A} is consistent w.r.t. a *TBox* \mathcal{T} , if there exists an interpretation \mathcal{I} that is a model of both \mathcal{A} and \mathcal{T} i.e. $\mathcal{I} \models (\mathcal{T}, \mathcal{A})$ (or $\mathcal{T} \models_{\mathcal{I}} \mathcal{A}$).
- **Concept instantiation:** An individual a is an instance of a concept C if for every model \mathcal{I} of \mathcal{KB} , \mathcal{I} is a model of $C(a)$ i.e. $\mathcal{KB} \models C(a)$.
- **Role instantiation:** Two individuals a and b are an instance of a role R if for every model \mathcal{I} of \mathcal{KB} , \mathcal{I} is a model of $R(a, b)$ i.e. $\mathcal{KB} \models R(a, b)$.

The modelization of a domain is performed manually by humans, it is important to check the consistency of this modelization. Indeed, the description of a domain should not be contradictory. This verification is carried out by testing the consistency of the *ABox*. An inconsistent *ABox* entails a false knowledge, which will make the system useless.

A reasoning problem for concepts can be reduced to a reasoning problem for ABoxes. For example, the concept satisfiability problem can be reduced to an *ABox* consistency problem, we have [15]:

- C is satisfiable iff $\mathcal{A} \cup \{C(a)\}$ is consistent.

The concept instantiation and the role instantiation are an *instance checking problem* which is based on an *ABox* inference. It is the check whether an assertion α is entailed by an *ABox* \mathcal{A} w.r.t. a *TBox* \mathcal{T} i.e. $\mathcal{A} \models_{\mathcal{T}} \alpha$.

We can reduce this problem if the language allows, to the consistency problem for *ABox*. We have [15]:

- $\mathcal{A} \models C(a)$ iff $\mathcal{A} \cup \{\neg C(a)\}$ is inconsistent.

We now present two reasoning approaches for *ABoxes*. The first reasoning is the instance retrieval which is based on instance checking. The second reasoning is a relaxation reasoning which aims at relaxing a logical inference.

2.5.2.1 Instance Retrieval

Let $\mathcal{KB} = (TBox, ABox)$ be a description logic based knowledge base. The instance retrieval problem [15, 14, 47, 46] consists in finding all individuals names i of $ABox$ that are instances of a given concept C (i.e., testing whether $\mathcal{KB} \models C(i)$, for $i \in ABox$).

A trivial instance retrieval algorithm, for a concept C , checks for every individual i of the $ABox$, if the knowledge base entails the assertion $C(i)$. Optimization techniques, tools and techniques from database and logic programming are also used for this problem [20, 51, 36, 65].

2.5.2.2 Variants of instance retrieval

Variants of instance retrieval [32, 78, 84] are reasonings which have the same kinds of inputs and outputs of a classical instance retrieval. The difference is that the inference process is modified.

Variants approaches are generally motivated by two main reasons:

- ***To improve the performance of reasoning or eliminate a constructor or axioms:*** an example of these approaches is the transformation of descriptions expressed in an expressive language into descriptions expressed in a less expressive language [70, 18, 71, 74, 75]. These approaches are called generally approximation.
- ***To expand query answers:*** the goal here is to find sound answers of a given query and also other answers close to the sound answers w.r.t. some constraints or measures [32, 78, 84, 83]. These approaches are generally called relaxations.

2.6 The DL Family

The DL family are the different DL languages. We presented in Section 2.3 some DL languages defined by the set of constructors and axioms they allow. Other extensions have been investigated such as the use of concrete domains, the temporal DL, the fuzzy DL, the probabilistic DL, the non-monotonic DL, etc.

According to the monotony property, the DL languages can be classified into two main families: monotonic DL and non-monotonic DL. We use both families in this thesis.

2.6.1 Monotonic DL

Reasoning in a knowledge representation formalism is monotonic [38] iff, for two knowledge bases $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ and $\mathcal{KB}' = (\mathcal{T}', \mathcal{A}')$ with $\mathcal{KB} \subset \mathcal{KB}'$ ($(\mathcal{T} \subseteq \mathcal{T}', \mathcal{A} \subset \mathcal{A}')$ or $(\mathcal{T} \subset \mathcal{T}', \mathcal{A} \subseteq \mathcal{A}')$ for DL) we have: for any axiom α , $\mathcal{KB} \models \alpha$ implies $\mathcal{KB}' \models \alpha$. It is nonmonotonic otherwise.

A Description Logic is monotonic if reasoning in this logic is monotonic.

Description logics with classical semantics (OWA) are monotonic and with CWA are nonmonotonic [54, 39, 38].

Example 2 Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be the knowledge base of Table 2.1.

We have $\mathcal{KB} \models \text{Person}(\text{bolt})$ but we have not an assertion $C(a)$ for that $\mathcal{KB} \cup \{C(a)\} \not\models \text{Person}(\text{bolt})$

Reasoning in the context of OWA approach is monotonic.

2.6.2 Non-monotonic DL

Non-monotonic description logics languages are languages where a reasoning could infer a knowledge which can be revised. The introduction of new knowledge can invalidate old inferred knowledge. These logics are useful for modeling systems where the response is necessary even though the information are incomplete.

Database theory was one of the earliest area on which the formalization of non-monotonic descriptions logic has been inspired, in particular with the closed world assumption. We use in this thesis the non-monotonic description logics in the context of the closed world assumption. For description logics, reasoning in the context of closed world assumption is non monotonic.

Example 3 Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be the knowledge base of Table 2.1.

We have $\mathcal{KB} \models_{gcwa} \neg \text{Man}(\text{federer})$ but $\mathcal{KB} \cup \{\text{Person}(\text{federer})\} \not\models_{gcwa} \neg \text{Man}(\text{federer})$

Reasoning in the context of GCWA approach is non monotonic.

2.7 Conclusion

In this chapter, we introduced the fundamental notions of description logics as our main contribution is based on it. Description logics structure the world in hierarchy and create relations between the objects of this world.

Among the fundamental notions introduced, the semantics of the closed world in description logics is presented. This semantic is not standard but it is well suited to the medical application domain and thus our work.

A description logics language is defined by its syntax (constructors and axioms) and its semantics. The choice of a language is guided by a need for expression but must comply with the constraints of decidability and complexity of the reasoning used for the language. The more the language is expressive, the more the reasoning is difficult. We have chosen to present in this chapter the syntax and the semantics of $\mathcal{ALCHOQ}(\neg)$ language as it will be used in the next chapter. $\mathcal{ALCHOQ}(\neg)$ is a very expressive language but decidable in the case of standard reasoning. We have chosen to define with precision a medical knowledge.

We will see in the next chapter the first step of our work, which is to use the description logics to represent the medical knowledge and images of the gastro-enterology and to develop a semantic gastroenterological images annotation and retrieval system.

A Description Logic Based Approach For Semantic Image Retrieval

Contents

3.1	Introduction	26
3.2	Image retrieval	26
3.2.1	Image retrieval techniques	26
3.2.2	Facets of image retrieval	28
3.2.3	Semantic image retrieval using DL	33
3.2.4	Medical image retrieval	35
3.3	Semantic gastroenterological image retrieval	36
3.3.1	Ontology	39
3.3.2	Annotation	42
3.3.3	Reasoning	45
3.3.4	Prototype and experiments	46
3.4	Conclusion	48

3.1 Introduction

Textual data are very limited for information representation. Therefore, multimedia technology, mainly images, are increasingly used. Image databases now represent large volume of information. Querying image databases is becoming a big challenge in computer science world.

To effectively manage and use these image databases, an image retrieval system is required. This is why it is a very active area since a long time. An image retrieval system manages access to several images, the images are represented and indexed according to the used approach and compared using various approaches to generated queries using specific techniques. In summary, an image retrieval system manage the image representation, the query construction and the selection of images.

The goal of this chapter is to give an overview of the image retrieval problem and propose a semantic gastroenterological images retrieval based on a new polyp ontology. For this purpose, this chapter is structured as follows: we start by the presentation of the two major image retrieval techniques proposed in the literature, then the facets of an image retrieval which are: query modalities, image annotation and image similarity. Afterwards, we present after the outline of the semantic image retrieval based on description logics and the medical images retrieval. Finally, we present the proposed approach including: the construction of an ontology, generation of image annotations and reasoning mechanisms, before concluding.

3.2 Image retrieval

3.2.1 Image retrieval techniques

The first image retrieval systems were based on a textual description of the image (keywords associated to images) [28, 77, 16], keywords are external source of information manually attached to images. The indexing process based on keywords is time consuming since it is manual, tricky, because the quality of the retrieval system depends largely on the relevance of the terms assigned to images and not standardized because the source of these keywords is generally not unique and depends on the person who associates keywords to images. In addition, a short list of keywords cannot completely cover the often rich semantics carried by an image.

The second generation of image retrieval systems are based on content. The content

based approach aims to directly extract information from the image itself to be able to classify it, this comes as an alternative to the textual approach but semantic problems related to the automatic processing of images are quickly noted. Techniques for solving these problems have been proposed in the literature [59, 35, 88, 80, 93, 26, 31].

We can classify the images retrieval approaches into two main currents: the current of content based image retrieval and the current of semantic image retrieval based on a knowledge representation formalism.

3.2.1.1 Content-based image retrieval

The principles of content based image retrieval (CBIR) approaches are to answer two questions [28]: how to mathematically describe an image ? and how to measure the similarity based on the abstract description ? These approaches usually apply statistics and machine learning techniques.

To describe images, most CBIR systems use characteristics automatically extracted from images such as color, texture and shape. These characteristics are used in different processes, such as similarity computation, model building, or even annotations.

One of the most known issues of content based approaches [28] is the gap between the extracted characteristics from the images to describe it which are low-level characteristics (description), and the human description of the image (image semantic) which is of high-level. In other words, the gap between the image description built automatically, and the human interpretation of the image. Indeed, the human uses more information and characteristics than the automated processes to interpret the image.

Liu et al. talk about this problem in [58]. They study and describe the different methods proposed which are classified in five major categories [58]:

- Using an ontology: to define high level objects and better interpret the extracted characteristics of low level.
- Using a machine learning methods: to predict from an input measure the value of an outcome measure of a process or describe the organization of the input measure.
- Using a user feedback: to try to learn the user's intentions to better understand their needs using an online processing.
- Generating a semantic template: to support high level image retrieval by using a templates to represent concepts calculated from a collection of reference images.

- Using a web context: for the web image retrieval. It consists to the utilisation of informations that can be structured as the url or HTML pages to improve the semantics of images.

3.2.1.2 Semantic-based image retrieval

In classical image retrieval approaches, images are described with a set of keywords [28], [77], [16]. The quality of this process depends largely on the relevance of these keywords, the keywords used in the user query, and the adequacy between them. Generally, this process performs a simple syntactical comparison between annotation keywords and user query keywords, and if the user does not use the same keywords, he may not get what he is looking for. Among the problems of these approaches is that the keywords may not cover the rich semantics carried by the image.

Semantic image retrieval approaches use generally a formalism of knowledge representation such as description logics or semantic networks [62]. The goal is to find a model of image representation. This representation has to be easily comparable and characterize the image in the best way. Semantic retrieval is based on the meaning of keywords (concept in this case), not its syntax. The comparison between concepts (generally referred to reasoning) is based on more advanced techniques than those of syntactic retrieval. It takes into account also the interaction between different concepts of the query to improve relevance compared to a syntactical retrieval.

Wang et al. [87] propose a comparison between ontology based image retrieval approaches and keywords based image retrieval approaches. The keywords based image retrieval is simple and easy to apply with an acceptable precision, and the ontology based image retrieval improves precision but requires a complete description of images.

3.2.2 Facets of image retrieval

To design an image retrieval system, we need to answer three key questions: how to built the query ? How do we represent an image ? And how to match between the query and the image representation (description) ?

These three questions are highly linked to each other and are considered as the three foundations of an image retrieval system. In the next sections of our document, we will have a brief overview of the possible answers to the above questions. The first and third questions concern the query modalities and the similarity computing respectively.

We are interested in this work to a semantic based retrieval, question two is treated in a semantic based techniques view, which usually represent the image by annotations. Other approaches, particularly in the case of CBIR systems use other techniques will not be mentioned here.

3.2.2.1 Query modalities

An important factor in an image retrieval system is the query modality. This modality define the query language supported by the system in terms of expressivity of generated user query. The most known query modalities of image retrieval system are [28]:

- *Keywords*: this modality is the most used. The query is presented in the form of keywords. The set of authorized keywords is not limited in general like the image retrieval systems on the web, but can be predefined, which will limit the expressivity of queries.
- *Free-Text*: for this modality, the user tries to define his need by sentences. These sentences are constructed freely, it may be questions, stories or other expressions.
- *Image*: like its name indicate, the introduced query by the user is an image and the system returns all images similar to the user image. This modality is very suitable for CBIR systems.
- *Graphics*: this modality is based on a graphical representation of the query, the user defines graphically his needs by drawing an image. The image can also be automatically generated by computer.
- *Composite*: it consists of a combination of the other modalities. In general, the user can choose a modality for define his query or define a parts of his query by using several modalities. The combination of modalities is interesting in the case of an interactive querying, the user can each time give more details about his query by changing the modality.

It is important to make the difference between an image retrieval technique based on keywords and query modality based on keywords. Query modalities are used as an input to the query construction process, which provides after, a query to the image retrieval process, regardless of the used technique.

In general, the query construction process is not necessary for syntactic retrieval techniques based on keywords, the set of keywords introduced by the user are used without pretreatment. By cons, it is very important for semantic image retrieval techniques to give some semantics to the query.

3.2.2.2 Images annotation

Image annotation is an image description process that facilitates access to images. It is mainly used by semantic based image retrieval techniques but can be also used by syntactic techniques. The difference is that the semantic techniques use a knowledge representation formalism to define an annotation, and a knowledge base to enrich and to better interpret the annotations.

To describe an image, we can associate to it several types of information. This informations are [16, 44]:

- *content-independent meta-data*: this information does not describe the content of images. They describe the context as the author of the image, the used camera, the image resolution, etc.;
- *visual content meta-data*: this information describes the visual content of images. It can be divided into two types:
 - *content-dependent meta-data*: describes the low level characteristics of images. It is generally information that can be automatically extracted such as color, shape, texture, etc.;
 - *content-descriptive meta-data*: describes the semantic of images content. This information represents the possible interpretations that a man can give to images.

The most used image annotation approaches are [44]:

1. **Free text descriptions**: free text is used to describe the image, the annotation is not formulated and its structure is not predefined. It is easy to make but difficult to exploit for obtaining an efficient system.
2. **Keywords**: a set of keywords freely chosen or selected from a predefined set of keywords is used to describe the image. These keywords can be used for the entire image or a segment of the image. The most difficult task for systems using this

type of annotation is the definition of the query modality which is not necessarily the keywords modality. It must be use in some cases a process that transforms the output of the query modality to a set of keywords.

3. **Based on ontologies:** an ontology is used to define the structure of the annotation and enrich its interpretation to improve the comparisons. The ontology concepts are used to annotate the images in general. The main problems of approaches using this type of annotation are: the formalization of the annotation, the definition of a query modality exploiting the knowledge of the ontology and the comparison process between a query and an annotation.

The first classification annotation approaches are based on the domain and structure of the annotation. We can also classify approaches depending on the generation process of the annotation. The generation can be manual, automatic or semi-automatic:

1. **Manual:** it is recommended for small images databases and usually for delicate applications which require maximum of accuracy such as medical imaging systems. Its advantage is that it is performed by human, therefore more accurate and very well suited for semantic techniques. Its limits is that it is very expensive and not possible to use for very large image databases.
2. **Automatic:** automatic annotation is the only solution in the case of very large image databases. It is generally used for CBIR systems and can also uses an ontology as proposed in [82]. Its advantage is that no human intervention is required and can be performed in the background but its limits is the lack of precision often and the difficulty of use in the case of a semantic technique.
3. **Semi-automatic:** it consists to the use of both types of previous annotations. The manual annotation can be used to review the automatic annotations or to complete them.

3.2.2.3 Images similarity

The similarity computing is a process used to retrieve the set of images answers. It defines an affinity relationship that can be quantified or not between a query and a image. The query is generated using the output of the query modality, it may be another image, text or a combination of both. The similarity computing also allows generally the ranking

of images answers even if other approaches use other parameters in addition to do the ranking [85, 91].

The similarity can be syntactic, semantic or hybrid. Syntactic similarity is based on a numerical comparison between the representations of the visual aspects of the images and the query. The result of this comparison is numerical and allows the ranking of the answers. By cons, semantic similarity is based on the comparison between the interpretation of the representations of visual aspects of images and the interpretation of the query, it generally uses a knowledge representation formalism and a knowledge base. This knowledge base is used to better interpret the representations of the images and the query for a better comparison. The result of the comparison is generally not numeric but boolean and does not allow necessarily the ranking of images answers. The hybrid similarity uses a combination of syntactic and semantic similarities. It uses the knowledge base to enrich and for a better comparison, and numeric computation for the ranking.

3.2.2.3.1 Syntactic similarity: The general principle of syntactic similarity is the computing of numerical distances between the query and a potential answers. The distance is defined according to the used model, it may be for example one or a set of values, numerically calculated using the representation of the query and the representation of a potential answer. The selection of answers is based on the values of distance, for example, by defining a distance threshold or the selection of N first answers after the ranking. We can distinguish two main approaches for computing the syntactic similarity:

Visual signature using: These approaches are mainly used in CBIR systems. The principle is to mathematically formulate the characteristics called signatures. This signatures are extracted from the images for using a techniques of distance computing to find one or a set of values representing the similarity.

Information retrieval models: The principle is simple, it consist to use text in form of keywords, tags, paragraphs or other forms to index images. The idea is to transform the image retrieval to an information retrieval to use techniques and models of information retrieval.

3.2.2.3.2 Semantic similarity: The similarity in this case is not a distance to compute usually but a process of comparing an elements of a given language. A knowledge representation formalism is used to formally define an annotation image and a query.

This formalism has a well-defined semantics for interpreting the annotations and queries. The management of interpretation is called reasoning. The reasoning is used to compare an annotation and a query. The comparison is usually logical and based on set theory, it can use a standard or non-standard reasoning.

Standard reasoning: These are classical standardized reasoning which belong to the standard definition of the language. These reasoning are generally mastered, necessary for the best exploitation of language and include on its technical tools such as the tool which allows to reasoning called reasoner. They are also the basis for defining other reasoning called non standard.

Non-standard reasoning: They consist to a non classical reasoning using numerical computing or probability theory often. They are often defined for specific applications and not available on technical tools such as reasoners. These reasoning are generally based on standard reasoning but can also use non-standard semantics.

3.2.3 Semantic image retrieval using DL

Description logics are increasingly used for formalized a knowledge area. They allow to well structure a domain, to define these elements and associate a well defined interpretation to each element. They also offer a set of reasonings to manipulate knowledge of the domain. These features are useful for image retrieval.

Description logics are mainly used in image retrieval to annotate images and to compare image annotations, i.e. retrieve images. Formally, it consists to define the syntax and semantics of an annotation, and define a reasoning to retrieve images. We must answer two main questions: what language to use and how to define an annotation image? and what reasoning to use ? The two issues are closely linked in the famous compromise between expressiveness and complexity.

Among the works in the domain, we can cite the work of Sciascio et al. [79] which propose a semantic indexing approach based on description logics for image retrieval. They use description logics and propose a formal language to structure the representation of an image and reduce the gap between low levels characteristics obtained by a vision algorithm and the complex structure of objects. They define syntactically an image by using concepts and roles. The composition of concepts and roles represent regions, simple shapes, complex shapes and geometric transformations. Each image is defined by

all of its regions which are connected parts of uniform color. A region is recognized as a form if there exist a geometric transformation that transform this form to this region. Geometric transformations describe the relative positions of the elements. The authors associate a semantic to their language for the interpretation of the definition of each image and propose two main reasonings to manage these interpretations. The recognition (classification) to decide if a form is recognized in an image and the subsumption to decide if a form is contained in another form. Other reasonings are defined using these two reasonings.

An experience of the utilisation of an ontology is proposed by Optitz et al. in [69]. An OWL EL ontology is used to annotate images and an instance retrieval reasoning is used to retrieve images. The paper shows that with a standard utilisation of semantic techniques, the image retrieval is improved. They identified before the experiment three main points that the utilisation of an ontology must improve compared to a keywords based techniques and they confirmed these points after the experiment.

The improvement of the recall¹ by a better management of synonyms. Indeed, for a keywords based retrieval, relevant images will not be retrieved if they are not annotated by the same vocabulary as the query, the utilisation of synonyms can degrade the value of the recall. With an ontology, the domain is formally defined, the image retrieval is based on the interpretation of keywords. The recall is improved because the synonyms have the same interpretation.

The precision² is improved also. A keyword depends on the context where it is used, a query can be misinterpreted when the interpretation is based on the syntax. Irrelevant images will be retrieved therefore a deterioration of the value of the precision but with the utilisation of an ontology, the query has a unique interpretation, the context is integrated in the query definition and the precision is improved.

The improvement of the construction of query. Indeed, with an interpreted language formally, the user can build more precise, well defined and expressive queries even with the used language which is not very expressive.

In the medical domain also, Hu et al. [49] propose a system to formally annotating medical images based on the description logic to aid in the diagnosis of breast cancer. This system is based on a domain ontology of breast cancer images. Their ontology is a multi levels model: graphic descriptors, images subjects and other medical terms. A

¹recall [60] is the number of relevant images returned divided by the number of all relevant images

²precision [60] is the number of relevant images returned divided by the number of images returned

last work can be cited is the work of Meghini et al. [62] which propose a framework based on fuzzy description logics to integrate the multidimensional aspects of multimedia information retrieval.

3.2.4 Medical image retrieval

Medical images are commonly targeted by image retrieval systems and medical domain is one of the main application cited by the CBIR systems [68, 81], but the most current access to medical images [68] are based on the patient's ID and other characteristics present in a DICOM file. DICOM (Digital Imaging and COmmunications in Medicine) [1] is a standard definition of medical images, informations of patients and other informations (concerning the examination for example) are stored with the image in the same file. These files are stored and managed by systems called PACS (Picture Archiving and Communication System).

Medical image retrieval systems are mainly used for teaching, research and diagnoses. For teaching, image retrieval can be used to illustrate a course, get acquainted with commonly cases and compare them. The images are used for the formation of physicians and also specialists. They are also used in research, for example, to analyse phenomena or propose a classification. The diagnosis aid is the most common, the most important but most difficult utilisation of medical image retrieval systems. The images are used mainly to compare cases and reuse other previously treated cases. To get an efficient utilisation of the system for the diagnosis, the interface should be very fast to use because the time factor is very important in the medical domain and rich to allow the user to create very precise queries. The answer of the system should be fast also and very relevant because the error can be fatal in this domain.

The various image retrieval techniques are used in the medical domain but with modifications. The CBIR systems for example are widely used but not with the same characteristics extracted from the image. The utilisation of color is not very suitable, several images do not contain colors and conditions of examinations such as coloring influence on color. Consequently, texture and shape are gaining importance and are much used.

Semantic medical image retrieval generally use medical ontologies. The purpose of medical ontologies is to provide a common vocabulary with formal definitions. These definitions can be used to represent images and medical assessments. It helps to standardize the interpretation of medical images for better treatment. We can distinguish two main approaches to developing medical ontologies [48]: general utilisation ones that are

independent of the application and others pointed for a particular application utilisation. UMLS Metathesaurus, SNOMED, MeSH and GALEN are examples of medical ontologies for a general utilisation.

The challenges of these approaches are: the tradeoff between the expressivity of image representations and reasoning performance in terms of relevance and computation time, adapted interfaces to medical domain and conditions of the examinations and the construction of ontology (reuse ? adapt ? create ? ...) and its maintenance for approaches based on ontologies.

The context of the medical domain and medical practices are very complicated and special, there is a gap between the theoretical approaches and the implemented systems. Several powerful techniques in other domain have not been tested, such as for example the non utilisation of user feedback [68] justified by his absence caused by the medical practices and the very limited time. Another example is the lack of investment in interfaces adapted to the working conditions [68]. The evaluation techniques and benchmarks are also lacking in the medical domain, although in 2004 a medical image retrieval tasks is added to ImageCLEF which is a part of the Cross Language Evaluation Forum (CLEF) [4]. In [67], The authors investigate the various interesting axes for medical image retrieval system evaluation.

For more information on medical practices and mainly propose a diagnosis aid for physicians, we propose and implement in this thesis a completed semantic gastroenterological image retrieval system [23, 24]. We built a polyp ontology based on standard classifications used in gastroenterology, propose a graphical interface to generate semantic queries and implement reasonings based on a slight extension of a standard reasonings used in description logic. We now present in detail all these parts.

3.3 Semantic gastroenterological image retrieval

In the medical domain, images are increasingly produced in large quantities. They are becoming increasingly important and necessary for some medical practices such as diagnosis, report writing and teaching. Management and access to this images become capital but also complex. Important decisions are based on these images. Like for the gastroenterology, where physicians for exploiting past experiences try to retrieve images of previously treated cases, similar to the treated case to decide whether they remove a type of tumor called a polyp or not. These polyps can develop a cancer. The decision to

remove a polyp or not is sometimes very complex because it can be very dangerous for the patient's health but can also generate unjustified costs. The error can be fatal in this domain and the time is very valuable, these physicians need an image retrieval system allowing to quickly, intuitively and efficiently generate queries and quickly retrieve very relevant images.

To help physicians, we propose in this work a semantic gastroenterological images annotation and retrieval system based on a new polyp ontology. We investigate the problems of ontology construction, image annotation using an ontology and semantic image retrieval. We proceed by constructing a polyp ontology which will be used by an image retrieval system. This ontology is the knowledge base of the system, it is used to define the representation of an image and the important elements of gastroenterology as standard classifications of polyps and for reasoning about all these definitions.

We use description logics as formalism to construct the polyp ontology, represent images and for reasoning about knowledge. The choice of description logics is justified by a set of criteria which can be summarized as:

- They allow to structure the world in hierarchy and create relationships between the objects of this world, which is well suited to medical domain for the formal definition of classifications, illnesses, symptoms, ... etc.
- They offer a wide range of languages with a variety of operators and a controlled reasonings but with a tradeoff between the language expressivity and the reasonings performances. We can always restrict the language expressivity to facilitate reasoning.
- A wide variety of technical tools as reasoners are complimentary. Experiments and development of systems are becoming less expensive and faster.
- Description logics are widely used for many years, we can reuse many results.
- Need to use an ontology to formalize medical knowledge, and separate the knowledge part from the processing part for a better exploitation, re-use, evolution and sharing of medical knowledge, knowing that the renowned language for the development of ontology, Ontology Web Language (OWL) [6], which is standardized by the W3C, is based on description logics.

As we have seen previously, description logics enable to describe a domain by individuals, concepts and relationships. For our domain, we represent images by individuals

which are instance of a concept image, images annotations by concepts descriptions and other medical terms by concepts. We present the polyp ontology and the conception details in section 3.3.1.

The main utilisation of the system is the semantic image retrieval, but it can also be used for other purposes such as teaching. It can be useful in the following cases:

- as an help for annotation: during a surgery, when he takes a photo, a surgeon may want to describe it even basically; in this context he can give his observations (color, shapes, ...) to the system which will display corresponding reference images (those described by a annotation semantically linked with the same observations), helping him annotating his image with the suited (and standard) concepts.
- as an help for a diagnosis: after having annotated the taken image, the surgeon may have to decide whether he removes or not the polyp. Then he can once again be helped by the system that would classify the image, according to its annotation, into various classifications (related to medical acts, pathologies, procedures,...). He can also be displayed images of the same kind of polyps that have been previously removed.
- as an help during medical education: learning a polyp classification can be eased by a system that is able to quickly relate a set of observations to the possible polyp classes it corresponds, and also to give the polyp classes that are not related to the observations by focusing on what is contradictory between the observation and the polyp description.

We distinguish three possible scenarios of utilisation:

- S1 semantic images retrieval: given an image annotation concept, the problem is to retrieve all images which can be annotated with this concept, in other words, they check all properties of this annotation. A property of an annotation is a part of its definition which represent an aspect of the annotated image, for example, the shape of the polyp or its color. This scenario is used to retrieve similar images and to help physicians in the diagnosis by the utilisation of images of previously treated cases.
- S2 exact classes retrieval: given an image annotation concept, and a standard gastroenterological classification concept, the problem is to retrieve all subclasses of

this annotation and are also subclass of the standard classification. This scenario can be used for the formation of physicians to different standard classifications.

S3 relaxed classes retrieval: given an image annotation concept, and a standard gastroenterological classification concept, the problem is to relax this annotation to retrieve these superclasses, which are also subclass of the standard classification. This scenario can be used as an help to the annotation or diagnosis.

From these three scenarios, we can create other interesting scenarios by combining these scenarios. For example, we can compare the standard classifications and find links between these classifications by the relaxed classes retrieval of images of a given annotation. This result is given by the sequence of scenarios S1 S3. We can also illustrate by images the definition and the different classes of a standard classification by the sequence of scenarios S1 S2. This combination is used for the formation of physicians.

The rest of the chapter is organized as follows. We present our choice of language, conception and formalization to construct the polyp ontology. Then, we explain how formally define and construct annotations or queries using the ontology and how to reason with this ontology and what reasoning used to answer queries. We present afterwards the prototype, some technical aspects and execution examples. We discuss at the end of our problems and choice compared to other approaches.

3.3.1 Ontology

Ontologies are designed to capture a shared knowledge and to solve the semantic heterogeneity between domains. Before the construction of an ontology, we must identify the important terms of the domain. These terms are represented by concepts, individuals and roles. We work about a domain of polyps, our ontology is an ontology of polyps and *Polyp* concept is the main concept of the ontology. Gastroenterological images are represented by individuals, instance of the *Image* concept. This *Image* concept represents a diagnostic, a site, a measurement or an observation. Observable concepts are atomic concepts, they are defined by the enumeration of their individual instance. For example, the *Color* concept is defined by a set of eight individuals: $Color \equiv \{white, blue, yellow, black, orange, pink, red, green\}$.

Part of the core hierarchy of the ontology is presented in Figure 3.1. This hierarchy is composed by five main parts: a part to describe the observable concepts, another part to describe some organs of the human anatomy, a part to describe the measurable concepts,

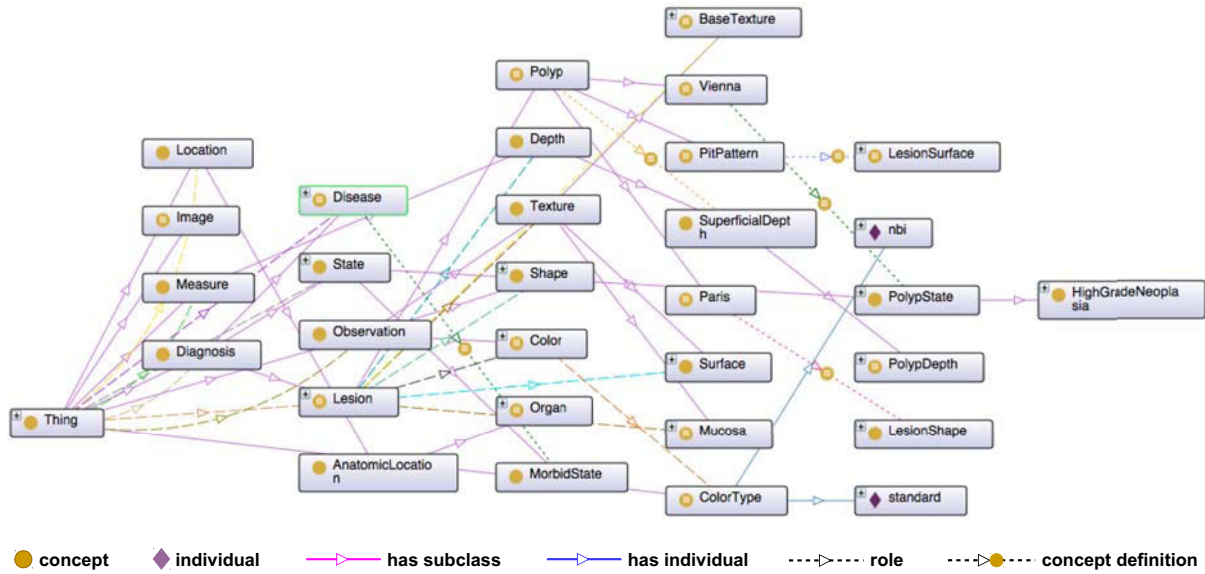


Figure 3.1: Part of the core hierarchy of the polyp ontology. Snapshot taken in Protégé 4.3 [10] with the OntoGraf plug-in.

a fourth part to describe the diagnosis and a last part to describe the images. Roles are used to define concepts and to connect these five parts. In particular, with the *Polyp* concept and the *Image* concept. The *Polyp* concept is the most important concept, it is defined by $Polyp \equiv Lesion \sqcap \exists depth.Superficial \sqcap \forall mayImply.Cancer$. It is a lesion which have at least a superficial depth and should be able to cause only cancer. Formally, it is an individual, instance of the *Lesion* concept and which are linked at least to an instance of the concept *Superficial* by the *depth* role and which are linked only to an instances of the *Cancer* concept by the *mayImply* role.

Roles are also organized in a hierarchy and use the same name of concepts to support the automatic processing of image annotation definitions.

The observable concepts part includes the concepts describing the observations of the physicians. These observations mainly concern the polyps, it consist of color, shape and texture. Some of these concepts are defined by listing their individual instance.

The human anatomy part represents some organs that may contain polyps. The representation is simple, each organ is an individual. We still have not found links between these organs and other concepts and roles of ontology.

The measurable concepts part defined measures concerning polyps obtained by medical equipment. We use roles on concrete domain in this part.

The image description part is composed by a concept of an image definition, concepts of an image annotation definition subsumed by the image concept and individuals image of previously treated cases, instance of image annotation concepts.

The diagnosis part contains the concepts defining a cancer diagnosis as polyps states, standard classifications used in gastroenterology ... etc.

The polyp ontology is used mainly for diagnosing cancer which is based on the analysis of polyps present in an organ. The principle is to clearly identify the type and state of the polyp by retrieving its standard classes used in gastroenterology. Formally speaking, it is consist to make a classification. We have integrated in the ontology the three most known standard classifications and widely used in gastroenterology. Namely, the Paris classification [72] that describes polyp shapes, the Pit-Pattern classification [55] that describes polyp surfaces and the Vienna classification [76] that describes pathological states associated to polyps. Each standard classification is represented by a hierarchy of concepts, the concept root is the name of the classification and the concepts children represent the classes of the classification. Each concept of the classification is defined using other concepts and roles. We have also integrated some concepts related to the domain of gastroenterology from the MST3.0 [5] that lists many gastroenterological concepts related to anatomy, observations and medical acts.

We use the $\mathcal{ALCHOQ}(\neg)$ language for the construction of the ontology. $\mathcal{ALCHOQ}(\neg)$ language is very expressive but decidable in the case of standard reasoning. The choice of $\mathcal{ALCHOQ}(\neg)$ language is justified by a need for constructors concepts and roles and reasonings performance to define medical classifications, describe the medical context and annotate medical images. Indeed, we need to define atomic concepts to generate more knowledge and be more specific about the description of a polyp. We remind that the main objective of using the ontology is to find the standard classes of the polyp. We chose $\mathcal{ALCHOQ}(\neg)$ to try to accurately define the standard gastroenterological classifications of polyps. For performance, our ontology is small and we use very simple adaptations of standard reasonings, the times answers are very reasonable, we did not have performance issues even though the language is very expressive. Statistically speaking, the ontology gathers 58 defined concepts, 23 roles and 188 individuals, among them 100 images and 88 various individuals (names or gastroenterological organs, colors, textures). These data are linked through 58 subsumption relations, 40 disjoint concepts axioms and about 200 individuals assertions. It is contained into a 476 KBytes files. This is a quite small ontology since it is still in its infancy.

3.3.2 Annotation

The annotation of images in our system is achieved manually by experts saw the importance of the made decision after an images retrieval. We use high levels information to describe the content of the image. Automatic extraction of low-level information is not used for the same reason that we have just mentioned and the information do not describe the image are also not used in the first version because we have not integrate information describing patients, context and physicians in the ontology and the query. We want to first test the first version of our system to see if the inclusion of such information contribute to improve the relevance of our results.

The used information for the annotation are concepts, individuals or roles of the polyp ontology described above, we use a knowledge base for the annotation, the text is not free. Note that, the process of generating annotations and the process of generating queries are based on the same principle with a slight difference. The query generation process is more general than the annotation generation process. It allows the construction of concept descriptions of other object and the construction of image concept descriptions like the annotation generation process. In addition to that, the annotation is performed by specialists but the interrogation can be performed by any user.

An annotation (or query) of an image is manually generated using an interactive interface. This interface represents graphically the polyp ontology and allows to navigate on concepts, roles and individuals of the ontology. According to his observation (or need for other users), the physician select most appropriate concepts, roles and individuals to generate an annotation (or query) of an image. The annotation generation process is fast and simple to use for satisfying the requirements of physicians. Other more complex interfaces are proposed in the literature including other information, such as the work of Xu et al. [92] which propose a new query interface by integrating the spatial distribution of concepts in the image. They propose a model for converting a concept map to a visual instance map, and a evaluation scheme for comparing the visual instance map with images. This system is not suitable for medical domain because images in general have the same spatial structure.

Formally, an annotation (or a query) of an image is a concept description A in conjunctive form defined as following:

$$\begin{aligned}
A \equiv \exists \text{represents.} (& \\
& \sqcap_{i=0}^I (\exists \text{diagnosis}_i . \text{Diagnosis}_i) \sqcap \\
& \sqcap_{j=0}^J (\exists \text{location}_j . \text{Location}_j) \sqcap \\
& \sqcap_{k=0}^K (\exists \text{measure}_k . \text{Measure}_k) \sqcap \\
& \sqcap_{l=0}^L (\exists \text{observation}_l . \text{Observation}_l) \sqcap \\
& \exists \text{length.} \text{Float} \sqcap \exists \text{width.} \text{Float})
\end{aligned}$$

Such as:

Diagnosis_i , Location_j , Measure_k , Observation_l are concepts descriptions and $\forall(i, j, k, l)$:

$$\text{Diagnosis}_i \sqsubseteq \text{Diagnosis}$$

$$\text{Location}_j \sqsubseteq \text{Location}$$

$$\text{Measure}_k \sqsubseteq \text{Measure}$$

$$\text{Observation}_l \sqsubseteq \text{Observation}$$

diagnosis_i , location_j , measure_k , observation_l are roles names and $\forall(i, j, k, l)$:

$$\text{diagnosis}_i \sqsubseteq \text{diagnosis}$$

$$\text{location}_j \sqsubseteq \text{location}$$

$$\text{measure}_k \sqsubseteq \text{measure}$$

$$\text{observation}_l \sqsubseteq \text{observation}$$

The annotation generation process is illustrated in figure 3.2. The user selects "stomach", "orange" and "haemorrhagic" from the ontology. It consists of three individuals from which we define using the concept constructor *oneOF* [42] three concepts: "*oneOF(stomach)*", "*oneOF(orange)*" and "*oneOF(haemorrhagic)*". The subsumers (belonging to the annotation concept definition) of these concepts will be determined in order to select the most appropriate roles for each concept. The subsumers are the concepts: *Location* (subsumer of *oneOF(stomach)*) and *Observation* (subsumer of *oneOF(orange)* and *oneOF(haemorrhagic)*). These subsumers allow to select from the definition Image concept the roles: location and observation (which will be used twice) to find the roles which to be used to define the annotation. Roles used to define the annotation satisfy two conditions: they are subsumed by the selected roles before, and their domain are the

selected concepts from the interface (or defined from a selection) or subsumers of these concepts. The roles location, color and mucous are found. We use all these results for define the user annotation as shown in the figure.

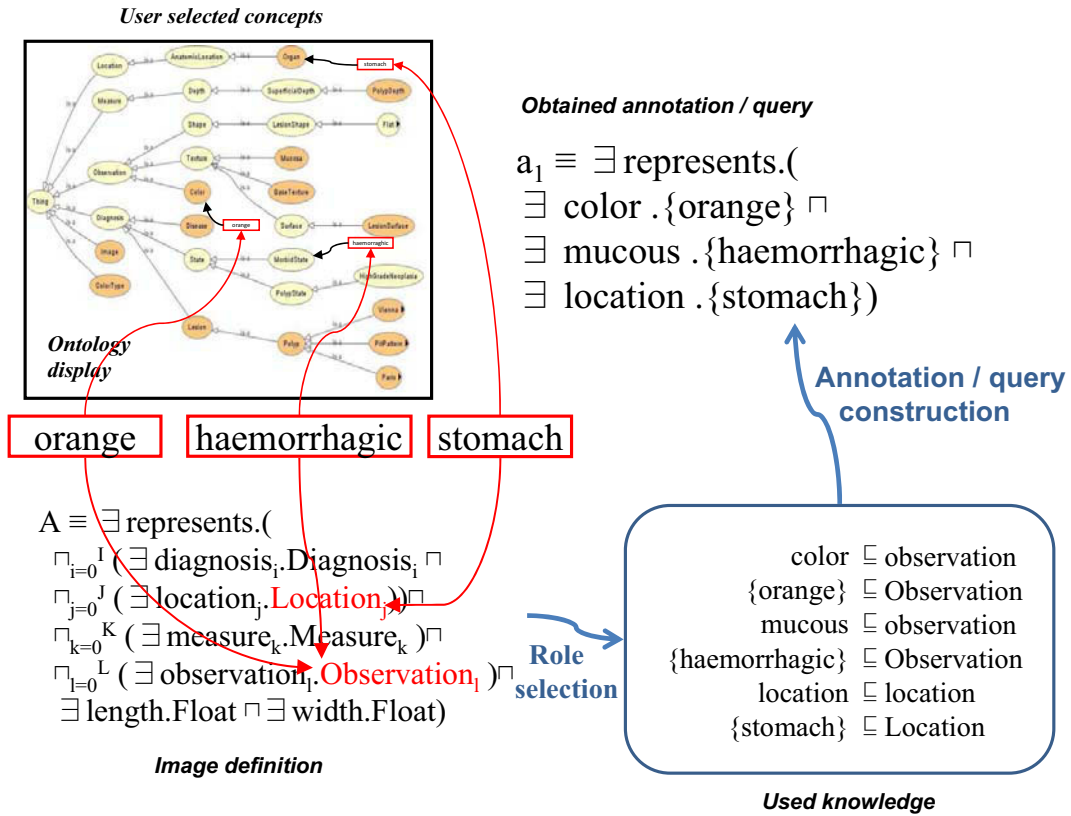


Figure 3.2: Annotation/query construction

The annotation generation process dependent and uses the definition of the Image concept. The same principle is used to generate other types of queries, the classes of polyps for example. The proposed interface is well suited to generate other types of queries and easily extensible for other needs.

Note that, the image annotation is generated by selecting concepts and individuals of the ontology. Roles and relationships between concepts and between individuals are generated automatically which restricts the expressivity of the query language. It is interesting to improve the query modality to allow the construction of other types of queries.

Reasoning	Input	Output	Using
Instance retrieval	TBox \mathcal{T} ABox \mathcal{A} Query C	All $i \in \mathcal{A}$ such that $\mathcal{T}, \mathcal{A} \models C(i)$	Retrieve similar images
Retrieval of local subsumeers	TBox \mathcal{T} Query A Classification concept D	All concepts $C \in \mathcal{T}$ such that $C \sqsubseteq A \sqcap D$	Retrieve standard classifications of a definition
Retrieval of local subsumers	TBox \mathcal{T} Query A Classification concept D	All concepts $C \in \mathcal{T}$ such that $A \sqsubseteq C \sqsubseteq D$	Retrieve relaxed standard classifications of a definition

Table 3.1: Implemented reasonings.

3.3.3 Reasoning

We represented a given image by an individual. This individual belongs to a general concept Image representing all the existing images. All images are associated with an image annotation. An image annotation is a concept description. The image associated with this annotation is an element of the interpretation. This description describes the contents and characteristics of the image. We want to find to find the annotation that best characterizes the image.

The correspondence between a query and a set of images is semantically computed in our system, the similarity is semantics. It is determined by reasoning. The reasoning is the logical management of knowledge to infer other knowledge.

We have implemented in our system three types of reasoning: a standard reasoning which is the instance retrieval and two other reasonings, a local subsumeers retrieval and a local subsumers retrieval which are a slight modifications of a standard reasoning which is subsumption. Table 3.1 shows the inputs, outputs and the utilisation of each implemented reasoning.

We use the instance retrieval to find an images results of a query, which can be used as an example of diagnosis when it is a previously treated cases or for the formation of new specialists. It is a *ABox* description logic reasoning and formally, it's consist to find from an *ABox* all individuals i which are instance of a concept C . The concept C represent in our system the user query. It is generated graphically as seen previously. Individuals i are image references, they are instances of one or more annotation concepts.

The retrieval of local subsumeers is used to find all the subclasses of the query and are also subclass of a standard classification used in gastroenterology. This result can be

used to better understand and analyze the definitions of the standard classifications used in gastroenterology. Formally, we find all the concepts C subsumed by the query A and a standard classification concept D , i.e. subsumed by the conjunction of the two concepts A and D . This is a *TBox* reasoning. The concepts C share all the properties defined in the query and in the standard classification, knowing that these concepts may have other properties that are not expressed in the query.

The last implemented reasoning is the retrieval of local subsumers used to find the classes of standard classifications used in gastroenterology which are superclasses of the query. It is also a *TBox* reasoning. We need to find all concepts C subsumed by a standard classification concept D and which subsume a query A . These concepts C are a classes used in gastroenterology, they share all properties defined in the classification concept D and part or all of the properties defined in the query. The query A is relaxed by the deletion of some of these properties until obtaining a standard class.

The tools which implement reasonings in description logic are called reasoners. Several reasonings implementing the standard reasonings are proposed as FaCT++ [2], Pellet [7], RacerPro [8] Hermit and [3]. We use Hermit for the implementation of our reasonings, it is one of the most effective reasoner up to our knowledge. It is based on the hypertableau algorithm [66] and most other reasoners are based on the tableau algorithm [15].

3.3.4 Prototype and experiments

The architecture of our application is presented in Figure 3.3. To implement the system we use the MVC model which is composed by three layers: the data layer to manage the data (images, ontology), the controller layer to manage the processing (query construction, reasoning) and the view layer to manage the interface and user interaction (query modality, answer display). The application is mainly composed by three modules: interface module, annotation module and reasoning module.

We use different technologies for the development of our application. Struts 2 framework for the implementation of the MVC pattern, Java Server Pages to create the interface, OWL API for the management of the ontology, Hermit reasoner [3] for the reasoning about the ontology and Prefuse [45] to create the interface of images annotation and query generation.

For the first scenario (semantic images retrieval), reasoning extend the images indexation. An image is indexed by an annotation concept but also by all concepts that can be inferred by the annotation concept. For example, if the query is the *ParisIII* class,

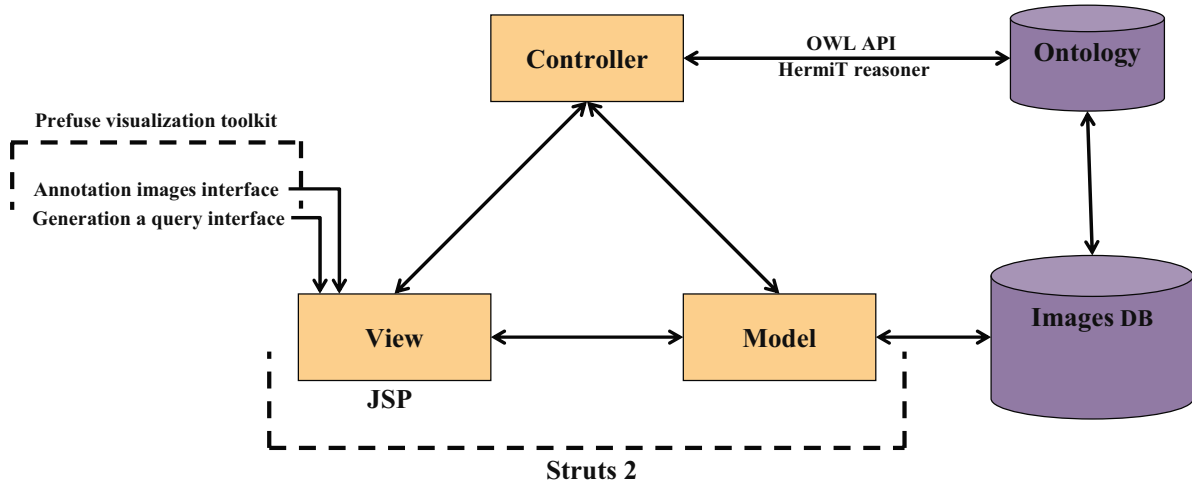


Figure 3.3: Application architecture

the images annotated by *superficial depth*, *cancer* and *ulcerated form* or *depthm2*, *big round surface*, *ulcerated form* and *dysplasia high grade state* are retrieved also.

In the second scenario (exact classes retrieval), reasoning is used for classification. The annotation concepts used to infer other concepts and information to find the class (standard classification) of the polyp. For example, a concept with a *flat depressed* form, *through submucosa* state, *papillary* surface and *m4* depth will infer other information as *depth is superficial*, *it is a lesion*, *this may cause cancer*, *it is a polyp* and from all that, it will deduce it belongs to the $ParisII_c$ class.

In the third scenario (relaxed classes retrieval), reasoning is used to find relaxed classes or classification (standard). The annotation concepts are used to infer other concepts and information in order to approach a class or classification. For example, if I ask to classify surfaces, after inference, the system returns the Pit Pattern classification and his hierarchy.

Let's see now an execution example. By browsing the ontology as represented in figure 3.1, the user may select the concepts *Lesion* and *totallyFlat* (which in fact is an individual). By selecting these concepts like keywords, the user expresses that he's looking for images where man can see some lesion that is totally flat. From these keywords, the prototype may build two semantic queries: the first will be built as the most general concept linking both *Lesion* and *totallyFlat*, and the second as the most general concept linking both *Lesion* and *totallyFlat* and being a subconcept of *Image*:

Query $\equiv Lesion \sqcap \forall observation.oneOf\{totallyFlat\}$ is the first query which can be as-

sociated to a classification name (e.g. Vienna) so as the system can run retrieval of local subsumees and instance retrieval reasonings.

Query $\equiv \exists \text{represents.}(\text{Lesion} \sqcap \forall \text{observation.oneOf}\{\text{totallyFlat}\})$ is the second query for which the system can retrieve all corresponding images (images that describe a totally flat lesion with other extra properties). On our current ontology, with a hundred of randomly generated images, the second query with instance retrieval reasoning shows that 11 images correspond to these criteria, and the first query, with retrieval of local subsumees and retrieval of local subsumers reasonings, shows that these criteria are present in the class Paris II B of the Paris classification, and in no other classification.

Semantic retrieval has brought more flexibility and precision. With ontology, the annotation vocabulary is unified. With a formal definition of concepts and reasoning techniques, the indexation of images is extended and the accuracy is improved. The quality of the image retrieval system depends directly on the quality of the image annotation and the quality of the query construction which depend on the expressivity of the used language, which is selected according to the compromise expressivity / performance. We propose a quick and simple method for the images annotation. The problem with this method is that the accuracy is low in some cases, if the doctor fails to well define the form of a polyp, for example.

3.4 Conclusion

We investigate in this first part of the thesis, how to implement an annotation, a semantic image retrieval and a classification system adapted to gastroenterologists. The proposed solution is based on a polyp ontology, a slight adaptation of the standard reasonings in description logic and a semi-automatic construction of queries and image annotation. Semantic retrieval has brought more flexibility and precision for image retrieval. With ontology, the vocabulary used for annotation is unified. With concepts structure and reasoning, indexing is extended, and with relationships between concepts and queries the annotation accuracy is improved which affects the result. The quality results depends directly on the quality of annotation and the quality of query generation. The quality of annotation and query depends on the used language expressivity, which is selected according to the compromise expressivity/decidability. We proposed a simple method for rapid annotation by graphic selection of concepts. The problem of this method is that the accuracy is low in some cases, if the physician fails to well defined the shape of a

polyp for example. Our goal is to develop a complete semantic system then define an interesting problem and detailing it.

It is clear that the system is still in its infancy, we randomly generate the images for the first tests, practice validation is required. Performance is acceptable but the ontology is to evolve by specialists and the interface of image annotation construction too. We were interested in the second part of the thesis to improve reasoning. Two main issues motivated our approach:

- Ranking: some physicians do not detail their queries, the number of retrieved images is great and naturally the instance retrieval do not distinguish between instances of a given concept. How to present images is a problem, we must rank the results by relevance.
- Relax the query: in the opposite of the first problem other physicians are very greedy and construct queries with many properties. With a standard instance retrieval, the set of retrieved images is empty. We must relax the query by sacrificing some properties to have a non-empty result.

We propose a new relaxed instances retrieval reasoning to solve both problems. The advantage of our approach is that for a given result the sacrificed properties are not the same for every individual.

We investigate in the second part of the thesis a new approach for computing relaxed answers of ontological queries based on a notion of an edit distance of a given individual w.r.t. a given query. Such a distance is computed by counting the number of elementary operations needed to be applied to an ABox in order to make a given individual a correct answer to a given query. The considered elementary operations are adding to or removing from an ABox, assertions of the form of an atomic concept (or its negation) and/or atomic roles.

\mathcal{W} Relaxation : A Relaxed Answer For Ontological Queries

Contents

4.1	Introduction	52
4.2	Using \mathcal{W}Relaxation with a medical images ontology	53
4.3	Motivation and main ideas	54
4.4	A new relaxation based on the ABox	56
4.4.1	\mathcal{W} Relaxation by considering separately the individuals	57
4.4.2	\mathcal{W} Relaxation by considering a set of individuals	62
4.5	Comparison w.r.t. existing reasonings	64
4.5.1	ABox abduction	65
4.5.2	Other reasonings	67
4.6	Conclusion	73

4.1 Introduction

We proposed in the previous chapter a semantic system for querying, annotation and retrieval of medical images based on description logics. We will now work on the reasonings for improving the images retrieval research part of the system.

The instance retrieval reasoning determines the individuals of the ABox which are instances of a concept C w.r.t. a knowledge base \mathcal{KB} . If an individual a is an instance of a concept C w.r.t. a knowledge base \mathcal{KB} , all models of \mathcal{KB} are also models of $C(a)$. If only one model of \mathcal{KB} is not model of $C(a)$, it is sufficient to infer that a is not an instance of C . When a is not instance of C two cases are possible: no model of \mathcal{KB} is a model of $C(a)$ or some models of \mathcal{KB} are not models of $C(a)$. For some applications, the models of \mathcal{KB} which are not models of $C(a)$ result from a poor modelization or a lack of knowledge. A slight modification of knowledge base might be sufficient for a to become instance of C without having a large effect on the other knowledge. For our application for example, physicians can forget to annotate a color c_1 of a polyp for an image i_1 , an instance retrieval with the criterion c_1 on the color cannot find the image i_1 even if all the other criteria are satisfied, but an update of the knowledge base by adding the assertion forgotten by the physicians regarding the color of the polyp allows to retrieve i_1 . This observation motivates our work, which consist to create a more flexible reasoning than instance retrieval. This flexibility allows to retrieve more instances of a concept, which is necessary for a very specific concept when the application needs a large number of answers.

It is difficult to formally define what is the flexibility for an instance retrieval reasoning, but the main idea is to increase the number of concepts instances according to defined criteria. Classical approaches for relaxed instance retrieval work mainly at the terminological level [84, 83, 18, 74, 70, 17], such approaches compute in general the subsumants concept without the exploitation of the ABox. For these approaches, the result of the relaxation of a concept is the set of instances of another concept, but the instances of a concept do not convey necessarily the same knowledge. In our approach, we propose to deal separately with each instance.

The TBox and the Abox are complementary, the TBox defines the domain organization and the ABox defines the individuals of this domain. The combination of these two types of knowledge is necessary to define an interesting flexibility for the instance retrieval. The exploitation of ABox knowledge allows for example to have an information on the

distribution of individuals on the structure of the domain. We propose in this chapter a new reasoning mechanism to compute relaxed answers of ontological queries which work at the ABox level. This new mechanism is based on a notion of an edit distance of a given individual w.r.t. a given query. Such a distance is computed by counting the number of elementary operations that need to be applied to an ABox in order to make a given individual a correct answer to a given query. The considered elementary operations are add to or remove from an ABox, assertions on atomic concept, on a negation of atomic concept or on atomic role. This reasoning is investigated in the context of the Open World Assumption (OWA) and the Generalized Closed World Assumption (GCWA) approaches.

The main motivation of our work comes from the application proposed in the previous chapter where the constructed knowledge bases include incomplete and/or incorrect human annotations.

4.2 Using \mathcal{W} Relaxation with a medical images ontology

The main motivation of our approach comes from our application of semantic retrieval of medical images that we have presented in the previous chapter. We remember that the large part of the assertions are used for the images annotation. Three main scenarios are justified, in a first step the utilisation of the relaxation of the answers, and in a second step, the insertion and deletion of assertions:

- **Answers need and limited time:** the query of physicians are in general very detailed when they use the images retrieval for a decision analysis. It is important to find an answer in this case, given the importance of the decision to take. Note that, physicians have a very limited time, the time of interrogation system must be minimal, they cannot rewrite a query to make it less detailed, a result must be returned for each query.
- **Incomplete annotations:** as we have already said, the physicians have a very limited time, they quickly pass the images annotation step, the annotation is not very well detailed and they can forget a very important characteristics. The knowledge base becomes incomplete, assertions used for the images annotation are missing in the ABox, adding these assertions improve the precision and makes the knowledge base complete.

- Incorrect annotations: like any manual annotation, two physicians can annotate differently an image, they may not agree on the classification of the polyp for example. The context may also influence the annotation, and induce to an incorrect annotation. The ABox will contain not valid assertions which can be detected if the knowledge base becomes inconsistent but can also not generate an inconsistent knowledge base. The deletion of these assertions becomes necessary to not retrieve the images annotated with these assertions by a wrong queries.

The standard utilisation of the \mathcal{W} Relaxation with a medical images ontology is the improvement of the semantic images retrieval, by the introduction of a distance computed between a query and a potential answer. This distance allows the parameterization of the precision. The \mathcal{W} Relaxation is used as an image retrieval reasoning without the update of the annotations or with the update of the annotations after the validation of an expert. The \mathcal{W} Relaxation can also be used to enrich and/or to correct an existing images annotations database. For example, a physicians can perform queries for which the result is mastered (medical classifications for example), to validate the insertion or deletion of annotations of images that are at a distance greater than zero from the query, but who should check the query.

The \mathcal{W} Relaxation is interesting for the parameterization of the precision and for the evolution of the ABox, but the medical ontologies are in general large and built using a very expressive languages. The \mathcal{W} Relaxation is a very costly reasoning, using it for medical ontologies is still a challenge.

4.3 Motivation and main ideas

Consider the knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ of Table 4.1 which is a part of our polyp ontology seen in the previous chapter. The *ABox* \mathcal{A} stores observations about endoscopic images of a physician while the *TBox* \mathcal{T} defines some associated medical concepts. A **polyp** may be described by several features, such as **shape**, **color**, **classification** (e.g., **Paris** is a polyp classification). Given such a knowledge base, one may be interested, for example, on computing the instances of the concept *Image* or of the concepts *Query₁* and *Query₂* of the knowledge base \mathcal{KB} . In our example, there are three individuals in the ABox \mathcal{A} , namely *image₁*, *image₂* and *image₃*, which are instances of the concept *Image* (can be inferred from *histology(image₁, h₁)*, *class(image₂, p₂)* and *color(image₃, clear)* respectively) while the concepts *Query₁* and *Query₂* do not have any instance in \mathcal{A} .

\mathcal{T}	
<i>Image</i>	$\equiv \exists \text{represent.}((\exists \text{diagnosis.} \text{Diagnosis}) \sqcup (\exists \text{location.} \text{Location}) \sqcup (\exists \text{measure.} \text{Measure}) \sqcup (\exists \text{observation.} \text{Observation}))$
<i>Lesion</i>	$\equiv \exists \text{growOn.} \text{Mucosa} \sqcap \exists \text{state.} \{\text{lesionnel}\}$
<i>Polyp</i>	$\equiv \text{Lesion} \sqcap \exists \text{depth.} \text{Superficial} \sqcap \forall \text{mayImply.} \text{Cancer}$
<i>ParisI</i>	$\equiv \text{Polyp} \sqcap \exists \text{shape.} \{\text{raised}\}$
<i>ParisII</i>	$\equiv \text{Polyp} \sqcap \exists \text{shape.} \text{Flat}$
<i>ParisIII</i>	$\equiv \text{Polyp} \sqcap \exists \text{shape.} \{\text{ulcerated}\}$
<i>LesionShape</i>	$\equiv \{\text{ulcerated, raised}\} \sqcup \text{Flat}$
<i>Histology</i>	$\equiv \{h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8\}$
<i>Color</i>	$\equiv \{\text{yellow, pink, red, black, brown, clear}\}$
<i>ParisI</i> \sqcap <i>ParisIII</i>	$\equiv \perp$
<i>Lesion</i>	$\sqsubseteq \text{Diagnosis}$
<i>Histology</i>	$\sqsubseteq \text{Diagnosis}$
<i>Color</i>	$\sqsubseteq \text{Observation}$
<i>class</i>	$\sqsubseteq \text{diagnosis}$
<i>color</i>	$\sqsubseteq \text{observation}$
<i>histology</i>	$\sqsubseteq \text{diagnosis}$
<i>Query₁</i>	$\equiv \exists \text{represent.}(\exists \text{class.} \text{Paris1} \sqcap \leq 1 \text{ class.} \text{Thing})$
<i>Query₂</i>	$\equiv \exists \text{represent.}(\exists \text{color.} \{\text{pink, red}\} \sqcap \exists \text{class.} \text{Paris3} \sqcap \exists \text{histology.} \{h_6\})$

\mathcal{A}
<i>Paris2</i> (p_1), <i>Paris2</i> (p_2), <i>Paris3</i> (p_2), <i>class</i> (<i>image₁</i> , p_1), <i>class</i> (<i>image₂</i> , p_2) <i>color</i> (<i>image₁</i> , <i>clear</i>), <i>color</i> (<i>image₂</i> , <i>red</i>), <i>color</i> (<i>image₃</i> , <i>clear</i>), <i>histology</i> (<i>image₁</i> , h_1)

Table 4.1: A knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ of a medical images system.

Query₁ and *Query₂* may represent user queries and, in this case, the instance retrieval problem can be viewed as a query answering problem (i.e., the problem of computing the answers of a given query Q). In this chapter, we are interested by the problem of computing relaxed answers to a given query. To achieve this task, we propose a new reasoning mechanism, called \mathcal{WR} Relaxation, which is more flexible than the classical instance retrieval mechanism. In particular, we relax the constraints that must be satisfied by a given individual in order to be considered as an instance of a given concept. For this purpose, we define a notion of a distance of a given individual w.r.t. a given query using elementary operations performed over the ABox.

More precisely, we consider the following elementary operations over an ABox: add to and remove from, an assertion on atomic concept, a negation of an atomic concept or an atomic role. Back to the example, by adding *Paris1*(p_1), the individual *image₁*

becomes relaxed answer of $Query_1$. Hence, $image_1$ is a relaxed answer to the query $Query_1$ with a distance equal to 1 (i.e., only one elementary operation is needed). On the other side, adding $Paris1(p_2)$ to the ABox makes the knowledge base inconsistent. Therefore, $image_2$ does not belong to any relaxation of $Query_1$ computed only by adding new assertions to the ABox.

We investigate \mathcal{W} Relaxation both in the context of the Open World Assumption (OWA) and the Generalized Closed World Assumption (GCWA) approaches. OWA is a widely used approach in the context of description logics while the use of GCWA is motivated by our application context. The considered polyp images retrieval application provides assistance to surgeons to decide whether or not polyp must be removed. An inadequate decision in this context may generate unjustified costs or be dangerous for the patient's health. Therefore, accurate annotation of images is essential. GCWA turned out to be an interesting approach to support validation of image annotation by young physicians and forces expert physicians to be very precise in their annotations.

Note that, in the next part of the thesis, we use the term atomic assertions to denote assertions on atomic concept, a negation of an atomic concept or an atomic role.

4.4 A new relaxation based on the ABox

As explained before, the general idea behind \mathcal{W} Relaxation is to determine the minimum knowledge to add to or to remove from an ABox so that a given individual can be considered as an instance of a given concept. Such a *quantity* of knowledge is defined by counting the elementary operations performed over the ABox. We consider the following elementary operations over an ABox: add to or remove from an atomic assertions. We define below three types of \mathcal{W} Relaxation based on the type of authorized elementary operations:

- The \mathcal{W}^+ Relaxation that allows only insertion of new assertions.
- The \mathcal{W}^- Relaxation that allows only deletion of assertions.
- The \mathcal{W} Relaxation which is more general and allows both insertion and deletion of assertions.

Note that, in the context of the GCWA, the elementary operations are performed before the computation of the generalized closed world assumption set.

Assertions on atomic concept, on a negation of atomic concept or on atomic role are used because they represent the minimal knowledge to add to or to remove from an ABox. For an acyclic terminology, concepts definitions can be unfolded into descriptions made with atomic concepts. In this case, adding a non atomic concept assertion is equivalent for these languages to adding a set of atomic assertions. Therefore we need to normalize the computing of the number of adding or removing assertions which represents the distance of the relaxation reasoning.

In our notation, ‘ \mathcal{W} ’ denotes the number of actions (insertion or deletions) performed. When $\mathcal{W} = 0$, the \mathcal{W} Relaxation coincides with the instance retrieval problem.

4.4.1 \mathcal{W} Relaxation by considering separately the individuals

We define in this first part the \mathcal{W} Relaxation and explain how an individual is integrated into the instances of a concept and then we discuss the limits of considering separately the individuals.

4.4.1.1 \mathcal{W}^+ Relaxation: relaxation by adding assertions

This relaxation computes the minimum set of assertions to be added to an ABox so that a given individual becomes relaxed answer of a given concept. Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a consistent knowledge base, C be a concept and \mathcal{W} be an integer. We define the function $InsR_{\mathcal{W}}^+(C, \mathcal{KB})$ which returns the instances of a given concept C w.r.t. a knowledge base obtained by adding at most \mathcal{W} assertions to the ABox \mathcal{A} .

Definition 4.4.1

$InsR_{\mathcal{W}}^+(C, \mathcal{KB}) = \{a \in Ind(\mathcal{A}) \mid \exists \mathcal{A}' \text{ such that:}$

- i. $\mathcal{A} \subseteq \mathcal{A}'$
- ii. $\mathcal{A}' \setminus \mathcal{A}$ is a set of atomic assertions such that $|\mathcal{A}' \setminus \mathcal{A}| \leq \mathcal{W}$
- iii. $\mathcal{KB}' = (\mathcal{T}, \mathcal{A}')$ is consistent
- iv. $a \in InsR(C, \mathcal{KB}')$ }

Note that, the function $InsR_{\mathcal{W}}^+(C, \mathcal{KB})$ and all the relaxation functions that we will define can be used with a third parameter representing the context used, $InsR_{\mathcal{W}}^+(C, \mathcal{KB}, OWA)$ for the OWA and $InsR_{\mathcal{W}}^+(C, \mathcal{KB}, GCWA)$ for the GCWA. We

use the function without the third parameter to indicate that the two contexts are taken into account.

Note also that, a \mathcal{W}^+ Relaxation of \mathcal{KB} is the knowledge base $\mathcal{KB}^{\mathcal{W}^+}$ obtained by adding to the ABox of \mathcal{KB} , at most \mathcal{W} atomic assertions.

Consider now the case of monotonic DLs. The following property is straightforward from the definition of monotonicity.

Lemma 4.4.2 *Let \mathcal{KB} be a monotonic description logic based knowledge base and let I be an interpretation of \mathcal{KB} .*

Then, $\forall I, I \models \mathcal{KB}^{\mathcal{W}^+} \Rightarrow I \models \mathcal{KB}$.

Hence, in the case of monotonic DLs, \mathcal{W}^+ Relaxation decreases the number of models of a given knowledge base. Regarding the relation between OWA and GCWA, we exhibit the following first result.

Corollary 4.4.3 *Let \mathcal{KB} be a knowledge base and let C and a be respectively a concept and an individual.*

Then, $a \in \text{Ins}R_{\mathcal{W}^+}^+(C, \mathcal{KB}, \text{OWA}) \Rightarrow a \in \text{Ins}R_{\mathcal{W}^+}^+(C, \mathcal{KB}, \text{GCWA})$.

Example 4 *Let a DL knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$.*

\mathcal{T}	
C	$\equiv (A \sqcap B) \sqcup \forall R.P$
D	$\equiv B \sqcup Q$
E	$\equiv \geq 1R$
F	$\equiv = 2R.P \sqcap \exists R.\exists S.A$
G	$\equiv A \sqcap \exists R.(B \sqcap Q)$
H	$\equiv A \sqcap (\leq 3R.B) \sqcap (\exists R.(B \sqcap Q))$
I	$\equiv \neg A \sqcap B \sqcap \leq 1R$
K	$\equiv \forall S.G \sqcap \leq 3R$

\mathcal{A}
$A(a), \neg A(b), B(f), \neg B(c), D(c), F(f), G(d), Q(f), R(a, b), R(b, d),$ $R(b, f), R(e, b), R(e, c), R(e, d), R(e, f), S(a, d), S(b, a), S(f, b)$

In the case of an OWA :

The individual “ a ” is not an instance of “ G ”, but if we add the “ $R(a, f)$ ” assertion, then “ a ” will be instance of “ G ”. Hence, $a \in \text{Ins}R_1^+(G, \mathcal{KB}, \text{OWA})$, i.e., “ a ” is a relaxed answer of “ G ” with distance 1.

The individuals “a” and “c” cannot be relaxed answers of “I” (i.e., $a \notin \text{InsR}_{\mathcal{W}}^+(I, \mathcal{KB}, OWA) \forall \mathcal{W}$) because the assertions “ $\neg A(a)$ ” and “ $B(c)$ ” cannot be added, since there will had an inconsistency with “ $A(a)$ ” and “ $\neg B(c)$ ”, knowing that $I \sqsubseteq \neg A \sqcap B$. The individual “d” cannot be relaxed answer of “I” because the concepts “G” and “I” are disjoint.

No individuals can be relaxed answer of “K” because we assume the open world. We cannot guarantee that for a given individual all his “S” role assertions are with an individuals of “G” and that the number of relations is less than or equal to 3 without explicit declaration. The assertions $\forall S.G(a)$ and $\leq 3R(a)$ cannot be constructed from atomic assertions.

In the case of OWA, assertions using some constructors such as “ \forall ” and “ $\leq n$ ” cannot be constructed from atomic assertions.

In the case of a GCWA :

If we add “ $R(a, f)$ ” assertion in this case, “a” will be relaxed answer of “G” and of “H”, with distance 1.

Hence, “a” is from one operation to the instances of “G” and the instances of “H”, therefore $a \in \text{InsR}_1^+(G, \mathcal{KB}, GCWA) \cap \text{InsR}_1^+(H, \mathcal{KB}, GCWA)$.

The individual “b” will be relaxed answer of “K” if we add the assertion “ $R(a, f)$ ”.

Example 4 illustrates one possible case of the difference between \mathcal{W}^+ Relaxation in the GCWA and in the OWA. We can note that for some concepts the \mathcal{W}^+ Relaxation does not exist in the OWA but exists in the GCWA, concepts built by “ \forall ” constructor for example. We can also note that the number of added assertions in the GCWA is less than or equal to the number of added assertions in the OWA.

4.4.1.2 \mathcal{W}^- Relaxation: relaxation by deleting assertions

In this type of relaxation, we are interested by the minimum set of assertions to be deleted from an ABox so that a given individual become relaxed answer of a given concept. Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a consistent knowledge base, C be a concept and let \mathcal{W} be an integer. We define $\text{InsR}_{\mathcal{W}}^-(C, \mathcal{KB})$ as a function which returns the instances of a concept C w.r.t. a knowledge base \mathcal{KB} obtained by deleting at most \mathcal{W} assertions from the ABox.

Definition 4.4.4

$\text{InsR}_{\mathcal{W}}^-(C, \mathcal{KB}) = \{a \in \text{Ind}(\mathcal{A}) \mid \exists \mathcal{A}' \text{ such that:}$

- i. $\mathcal{A}' \subseteq \mathcal{A}$

ii. $\mathcal{A} \setminus \mathcal{A}'$ is a set of atomic assertions such that $|\mathcal{A} \setminus \mathcal{A}'| \leq \mathcal{W}$

iii. $a \in \text{InsR}(C, \mathcal{KB}') \mid \mathcal{KB}' = (\mathcal{T}, \mathcal{A}') \}$

Note that, \mathcal{W}^- Relaxation of \mathcal{KB} is the knowledge base $\mathcal{KB}^{\mathcal{W}^-}$ obtained by deleting atomic assertions.

The following lemma shows that, in the case of monotonic DLs, \mathcal{W}^- Relaxation does not extend the set of instances of a concept.

Lemma 4.4.5 *Let \mathcal{KB} be a monotonic description logic based knowledge base. Then, $\forall C \in \text{Conc}(\mathcal{T}), \forall \mathcal{W} > 0, \text{InsR}_{\mathcal{W}^-}(C, \mathcal{KB}) \subseteq \text{InsR}(C, \mathcal{KB})$.*

Hence, \mathcal{W}^- Relaxation is useless in the case of monotonic DL.

Example 5 *Let the knowledge base of the example 4.*

In the case of a OWA :

The relaxation is not possible in this case (see the lemma 4.4.5).

In the case of a GCWA :

If we delete the assertion “ $R(b, d)$ ” or “ $R(b, f)$ ”, “ b ” will be relaxed answer of “ F ”.

The individual “ b ” cannot become relaxed answer of “ G ” and of “ H ” because we have $\neg A(b)$ and $G \sqsubseteq H \sqsubseteq A$.

4.4.1.3 \mathcal{W} Relaxation: relaxation by adding/deleting assertions

This type of relaxation deals with the minimum set of assertions to be added to and/or deleted from the ABox so that a given individual become relaxed answer of a given concept. Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a consistent knowledge base, C be a concept and let \mathcal{W} be an integer. We define $\text{InsR}_{\mathcal{W}}(C, \mathcal{KB})$ as a function that return the instances of a concept C w.r.t. a knowledge base \mathcal{KB} obtained by adding and/or deleting at most \mathcal{W} assertions.

Definition 4.4.6

$\text{InsR}_{\mathcal{W}}(C, \mathcal{KB}) = \{a \in \text{Ind}(\mathcal{A}) \mid \exists \mathcal{A}' \text{ such that:}$

i. $(\mathcal{A}' \setminus \mathcal{A}) \cup (\mathcal{A} \setminus \mathcal{A}')$ is a set of atomic assertions such that $|\mathcal{A}' \setminus \mathcal{A} \cup (\mathcal{A} \setminus \mathcal{A}')| \leq \mathcal{W}$

ii. $\mathcal{KB}' = (\mathcal{T}, \mathcal{A}')$ is consistent

iii. $a \in \text{InsR}(C, \mathcal{KB}') \}$

Note that, \mathcal{W} Relaxation of \mathcal{KB} is the knowledge base $\mathcal{KB}^{\mathcal{W}}$ obtained from \mathcal{KB} by adding and/or deleting atomic assertions.

Corollary 4.4.7 *Let \mathcal{KB} be a knowledge base and let C be a concept and a an individual. Then, $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB}, \text{OWA}) \Rightarrow a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB}, \text{GCWA})$.*

Example 6 *Let us consider the knowledge base of the example 4.*

In the case of a OWA :

The individual “b” will be relaxed answer of “G” if we delete the assertion “ $\neg A(b)$ ” and we add the assertion “ $A(b)$ ”, therefore $b \in \text{InsR}_2(G, \mathcal{KB}, \text{OWA})$.

In the case of a GCWA :

The individual “f” will be relaxed answer of “K” if we delete the assertion “ $\neg A(b)$ ” and we add the assertion “ $A(b)$ ”, therefore $f \in \text{InsR}_2(K, \mathcal{KB}, \text{GCWA})$.

If we add “ $A(e)$ ” assertion and we delete “ $R(e, b)$ ”, “ $R(e, c)$ ” or “ $R(e, d)$ ” assertions, “e” will be relaxed answer of “H”, therefore $e \in \text{InsR}_2(H, \mathcal{KB}, \text{GCWA})$.

The previous definitions are provided on the perspective of a single individual. In the case of a relaxation of a set of individuals, possible interactions between the assertions required by distinct individuals are not considered in the definition. An example of such interactions is given in Example 7. Two consistent ABoxes (respectively, \mathcal{A}_1 and \mathcal{A}_2) could be generated to enable respectively the individuals a or b to become an instance of C . However, it is not possible to generate a unique ABox where both a and b are instances of C .

Example 7 *Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be the following knowledge base using the OWA.*

$\mathcal{T} = \{C \equiv 1R.A \sqcap \neg A\}$ and $\mathcal{A} = \{R(a, b), R(b, a)\}$.

$(\mathcal{T}, \mathcal{A}) \not\models C(a)$ and $(\mathcal{T}, \mathcal{A}) \not\models C(b)$

Let $\mathcal{A}_1 = \mathcal{A} \cup \{A(b), \neg A(a)\}$ and $\mathcal{A}_2 = \mathcal{A} \cup \{A(a), \neg A(b)\}$

$(\mathcal{T}, \mathcal{A}_1) \models C(a)$ and $(\mathcal{T}, \mathcal{A}_2) \models C(b)$.

Hence, $a \in \text{InsR}_2^+(C, \mathcal{KB}, \text{OWA})$ and $b \in \text{InsR}_2^+(C, \mathcal{KB}, \text{OWA})$ but it is not possible to build \mathcal{A}_3 such that $(\mathcal{T}, \mathcal{A}_3) \models C(a) \wedge C(b)$ and $\{a, b\} \in \text{InsR}_2^+(C, \mathcal{KB}, \text{OWA})$.

In the Example 8 below, the generation of a unique ABox such that the individuals a and b become relaxed answers of D is possible while however the number of added assertions for b is not minimal.

Example 8 Consider the previous knowledge base augmented with the following concept: $D \equiv C \sqcup (B_1 \sqcap B_2 \sqcap B_3)$. Here, we need two assertions $A(b)$ and $\neg A(a)$ so that a become relaxed answer of D and also two assertions $A(a)$ and $\neg A(b)$ so that b become relaxed answer of D . But we need five assertions $R(a, b)$, $R(b, a)$, $A(b)$, $\neg A(a)$, $B_1(b)$, $B_2(b)$ and $B_3(b)$ so that a and b become together relaxed answers of D .

4.4.2 \mathcal{W} Relaxation by considering a set of individuals

We have defined in the first part of the chapter how each individual is integrated in the relaxation of a concept by the application of elementary operations. The elementary operations which allow to integrate all individuals of the relaxation must be consistent between them and keep the consistency of the knowledge base. From a practical point of view, it is not interesting to identify a set of individuals who may not belong to the same consistent knowledge base.

We redefine now the three types of relaxations to take into account the interaction between the elementary operations which allow to integrate each individual. The objective is to built a single consistent ABox which allows to integrate a set of individuals in the relaxation of a given concept.

4.4.2.1 \mathcal{W}^+ Relaxation: relaxation by adding assertions

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a consistent knowledge base, C be a concept, and \mathcal{W} and N be two integers. We define $RxdIR_{\mathcal{W}}^+(C, \mathcal{KB})$ as an extension of the \mathcal{W}^+ Relaxation function to take into consideration the interactions between the relaxation of several individuals.

Definition 4.4.8

The set $RxdIR_{\mathcal{W}}^+(C, \mathcal{KB})$ of relaxed answers of C is defined as follows:

$$RxdIR_{\mathcal{W}}^+(C, \mathcal{KB}) = \{a \in Ind(\mathcal{A}) \mid (\mathcal{T}, \mathcal{A}') \models C(a)\}$$

Such $\exists \mathcal{A}'$ which satisfies:

- i. $\mathcal{A} \subseteq \mathcal{A}'$
- ii. $\mathcal{A}' \setminus \mathcal{A}$ is a set of atomic assertions
- iii. $\mathcal{KB}' = (\mathcal{T}, \mathcal{A}')$ is consistent
- iv. $Dist^+(C, \mathcal{KB}, \mathcal{A}') \leq \mathcal{W}$

where $Dist^+(C, \mathcal{KB}, \mathcal{A}')$ is a distance defined between two ABoxes over the same TBox. Such a distance is application dependent and could be calculated in different ways:

- *Global distance*, defined at a global level, i.e. it is calculated at the *ABox* level, as the total number of added assertions (i.e., equal to $|\mathcal{A}' \setminus \mathcal{A}|$). This distance is useful when the goal is to minimize the overall amount of knowledge added to an ABox.
- *Local distance*, defined at a local level, i.e. calculated at the individuals level. Let $indDist(a) = minimal(|assAdd|)$ with $assAdd \subseteq \mathcal{A}' \setminus \mathcal{A}$ and $a \in InsR(C, (\mathcal{T}, \mathcal{A} \cup assAdd))$, we give two examples of local distances:
 - *Average distance*: calculated as the average number of added assertions such that the individuals become relaxed answers of C . It is equal to $average(indDist(a)) \forall a \in (RxdIR_{\mathcal{W}}^+(C, \mathcal{KB}) \setminus InsR(C, \mathcal{KB}))$. This distance is used when the goal is to minimize the average added knowledge for each individuals.
 - *Maximal distance*: calculated as the largest number of added assertions such that the individuals become relaxed answers of C . It is equal to $maximal(indDist(a)) \forall a \in (RxdIR_{\mathcal{W}}^+(C, \mathcal{KB}) \setminus InsR(C, \mathcal{KB}))$. This distance is used when the goal is to fix a maximal distance beyond which an individual cannot be considered as a relevant answer to a given query.

It is worth noting that, a \mathcal{W}^+ Relaxation for individual focuses on the relation between an individual and a concept. In the case of an \mathcal{W}^+ Relaxation of a set of individuals, the focus is on the relation between a set of individuals and a concept.

4.4.2.2 \mathcal{W}^- Relaxation: relaxation by deleting assertions

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a consistent knowledge base, C a concept and let \mathcal{W} and N be two integers. We define $RxdIR_{\mathcal{W}}^-(C, \mathcal{KB})$ as the function that extends \mathcal{W}^- Relaxation to take into consideration the interactions between the relaxation of several individuals.

Definition 4.4.9

The set $RxdIR_{\mathcal{W}}^-(C, \mathcal{KB})$ of relaxed answers of C is defined as follows:

$$RxdIR_{\mathcal{W}}^-(C, \mathcal{KB}) = \{a \in Ind(\mathcal{A}) \mid (\mathcal{T}, \mathcal{A}') \models C(a)\}$$

Such $\exists \mathcal{A}'$ which satisfies:

- i. $\mathcal{A}' \subseteq \mathcal{A}$
- ii. $\mathcal{A} \setminus \mathcal{A}'$ is a set of atomic assertions
- iii. $Dist^-(C, \mathcal{KB}, \mathcal{A}') \leq \mathcal{W}$

where $Dist^-(C, \mathcal{KB}, \mathcal{A}')$ is a distance between two ABoxes over a same TBox. As previously, such a distance is application dependent and can be calculated in different ways as discussed for the case of $Dist^+(C, \mathcal{KB}, \mathcal{A}')$ (Section 4.4.2.1).

4.4.2.3 \mathcal{W} Relaxation: relaxation by adding/deleting assertions

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a consistent knowledge base, C be a concept, and let \mathcal{W} and N be two integers. We define $RxdIR_{\mathcal{W}}(C, \mathcal{KB})$ as an extension of \mathcal{W} Relaxation to take into consideration the interactions between the relaxation of several individuals.

Definition 4.4.10

The set $RxdIR_{\mathcal{W}}(C, \mathcal{KB})$ of relaxed answers of C is defined as follows:

$$RxdIR_{\mathcal{W}}(C, \mathcal{KB}) = \{a \in Ind(\mathcal{A}) \mid (\mathcal{T}, \mathcal{A}') \models C(a)\}$$

Such $\exists \mathcal{A}'$ which satisfies:

- i. $(\mathcal{A}' \setminus \mathcal{A}) \cup (\mathcal{A} \setminus \mathcal{A}')$ is a set of atomic assertions
- ii. $\mathcal{KB}' = (\mathcal{T}, \mathcal{A}')$ is consistent
- iii. $Dist(C, \mathcal{KB}, \mathcal{A}') \leq \mathcal{W}$

where $Dist(C, \mathcal{KB}, \mathcal{A}')$ is a distance between two ABoxes over the same TBox. It can be calculated as in the case of $Dist^+(C, \mathcal{KB}, \mathcal{A}')$ (Section 4.4.2.1) by taking into consideration the addition and deletion of assertions.

4.5 Comparison w.r.t. existing reasonings

We investigated in this chapter a new approach for computing relaxed answers of ontological queries. The proposed approach concern several aspects of description logic reasoning as the instance retrieval, the approximation, the relaxation reasoning and the dynamic aspects of the ABox. In the following subsection, we review related work.

4.5.1 ABox abduction

Contrary to the deduction, the abduction [86] is interested in finding the causes of a set of facts. The abduction problem is defined for description logics for the TBox [27, 43] and also for the ABox [30, 53]. It consists for the TBox (resp. ABox) to identify the set of TBox axioms (resp. ABox assertions) need to be added to a DL knowledge base to allow a TBox axioms (resp. ABox assertions) to be entailed. We are interested in this work to the abduction problem defined for the ABox, because it is similar to one of the problems that we are trying to address in this thesis, i.e. the problem of adding assertions to an ABox to entail other assertions. We can refer to the definition proposed in [53] for the ABox abduction:

Definition 4.5.1 (ABox abduction problem) *Let \mathcal{L}_K and \mathcal{L}_Q be DLs, $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ a knowledge base in \mathcal{L}_K and Φ a set of ABox assertions in \mathcal{L}_Q , denoted as the abductive query. We call the tuple (\mathcal{KB}, Φ) an ABox abduction problem iff $\mathcal{KB} \not\models \Phi$ and $\mathcal{KB} \cup \Phi \not\models \perp$.*

Definition 4.5.2 (ABox abduction solution) *Let \mathcal{L}_S be a DL and A a set of ABox assertions in \mathcal{L}_S . A is a (plain) solution to abductive problem (\mathcal{KB}, Φ) iff $\mathcal{KB} \cup A \models \Phi$. It is consistent iff $\mathcal{KB} \cup A \not\models \perp$, relevant iff $A \not\models \Phi$ or minimal iff there is no solution B to (\mathcal{KB}, Φ) that is minimal with respect to A .*

B is minimal with respect to A iff there exists a renaming $\rho : N_I^(B) \mapsto N_I^*(A)$, where $N_I^*(B)$ and $N_I^*(A)$ are the sets of individual names from A and B that do not occur in \mathcal{KB} , such that $A \models \rho B$, but for every renaming $\varrho : N_I^*(A) \mapsto N_I^*(B)$ it holds that $B \not\models \varrho A$.*

The three criteria, consistency, relevance and minimality are often used to limit the number of possible solutions to a ABox abduction problem which is in general infinite. The choice of these criteria is widely adopted in the literature [53, 12, 73]. This choice is justified by:

- The consistency: for the practical problems, a solution is interesting if it is consistent with the knowledge base. It must not contradict the knowledge already acquired which is in general non-editable.
- The relevance: the ABox abduction problem is to identify the assertions A to add to a knowledge base for that other assertions Φ are entailed. An obvious solution to this problem is to add Φ to the knowledge base but this solution trivializes the problem instead of really solving it because it is independent of the knowledge base used.

- The minimality: this criterion consist on adding the minimum of necessary assertions to the knowledge base in order to resolve the problem, without adding unnecessary information to the knowledge base. It is necessary to minimize the change of the overall structure of the knowledge base to do not generate unnecessary knowledge or even false.

Klarman et al. [53] introduce a formal computational framework for ABox abduction in the DL \mathcal{ALC} . They propose two novel reasoning for solving this problem. The two introduced reasoning are variants of resolution and tableaux algorithms for the first-order logic. The general idea is to transform the inputs of an abduction problem (\mathcal{KB}, Φ) into first order logic and try to construct a refutation proof for $\mathcal{KB} \models \Phi$ by proving that $\mathcal{KB} \cup \neg\Phi \vdash \perp$. *Model-theoretic* and *proof-theoretic* aspects of abductive reasoning are the basis for resolution. For *model-theoretic*, the resolution can be considered as to try to find a formula that is unsatisfiable in all models of the knowledge base in which the adductive query is not satisfied (eliminate all non interesting models). For *proof-theoretic* and for a goal-oriented strategy for example, the resolution use a back chaining to try to derive from the abductive query intermediate results that are connected and so on.

For an ABox abduction, the goal is to identify a set of assertions to be added to an ABox so that a given set of assertions known in advance becomes entailed by the knowledge base, i.e., the input is an assertions set and the output is also an assertions set. For relaxations that we proposed, the goal is also to identify a set of assertions to be added to (in the case of \mathcal{W}^+ Relaxation), to be removed from (in the case of \mathcal{W}^- Relaxation) or to be added to and to be removed from (in the case of \mathcal{W} Relaxation) an ABox, the output is an assertions set but the input is not an assertions set, because the set of individuals which may be relaxed is not known (the number of possible combination is exponential). Let C and a an input of a \mathcal{W}^+ Relaxation problem, respectively a concept and an individual, we have that the problem of existence of \mathcal{W} for a \mathcal{W}^+ Relaxation and the problem of finding \mathcal{W} for a \mathcal{W}^+ Relaxation can be reduced in polynomial time to an *ABox abduction problem* as following:

- $\Phi = C(a)$.
- Find a consistent and minimal Abox abduction solution A .
- Compute \mathcal{W} , the number of all atomic assertions.

Note that, this reduction is only possible if the used language allows to transform the assertions of the *Abox* into atomic assertions.

4.5.2 Other reasonings

4.5.2.1 Instance retrieval reasoning

The instance retrieval is a standard reasoning for retrieving the instances of a concept. It consists in finding all individuals names i of the ABox that are instances of a given concept C . It is a costly reasoning for some applications, when the ABox represents a database with a large number of individuals.

The instance retrieval problem is investigated since the first description logics systems [64], the first algorithms reduce [15, 64] the problem to a satisfiability problem but the complexity of the problem resulted in the proposition of several optimization techniques [15, 64]. The reduction of the problem to the datalog programs to exploit the techniques of the theory of databases is also proposed in recent approaches [19, 65, 50].

4.5.2.1.1 Reduction to satisfiability problems A trivial solution to the problem of the instance retrieval of a concept is to check for each individual in the ABox if it is instance of the concept. It consists to translate an instance retrieval problem to a set of instance checking problems. Then each instance checking problem is transformed to an ABox satisfiability problem. The limits of this approach is that it is very costly, it consists to resolve $|ABox|$ problems of ABox satisfiability.

4.5.2.1.2 Optimization techniques The reduction of an instance retrieval problem to satisfiability problems requires the definition of a satisfiability problem for each individual. This can be very costly for expressive languages. A range of optimization techniques are defined [15] to improve performance. In general, the principle of these techniques is aimed at reducing the size of the search space which is often very large. This search space is represented in general by a tree, which allows to reusing a few search techniques for a tree.

There are several techniques for satisfiability optimizations, we can cite [15]: the semantic branching search, the local simplification, the dependency directed backtracking, the heuristic guided search and the caching satisfiability

4.5.2.1.3 Reduction to Datalog problems The general interest of the reduction to Datalog approaches [19, 20, 65, 50] is to reduce the expressivity gap between the database query language and the description logics as ontology language, to transfer

the techniques, evidences and concepts of the database theory to description logics theory. The general principle of these approaches is that the ABox can be considered as an extensional database and the TBox as a set of integrity constraints or a functional dependencies.

Calì et al. [19, 20] introduce a family of expressive extension of datalog to reduce the gap between the semantic web and the databases. They defined a special class of tuple-generating dependency (TGD) so that the boolean conjunctive queries are decidable and of polynomial complexity in general. They have also defined another class of TGD, a negative constraints and equality generating dependencies to define two extensions of Datalog: Datalog_{\pm}^0 and Datalog_{\pm}^1 .

Motik and Sattler [65] developed an algorithm for reducing a DL knowledge base to a disjunctive datalog program and compared a reasoning approaches for querying large description logic Aboxes. They concluded that the translation to datalog is interesting when the TBox is simple and the ABox is large. The authors proved that their extension is strictly more expressive than a few languages of the DL-Lite family and proposed translation functions which are used to rewrite LD-Lite systems in their datalog extension.

In [50], Hustadt et al. translated a slight restriction of description logic \mathcal{SHIQ} (\mathcal{SHIQ}^-) to disjunctive datalog programs to reuse optimization techniques from deductive databases for resolving the problem of consistency of Aboxes, instance checking and query answering.

4.5.2.2 Approximation

A description logic system is defined in general by the definition of three components: the language, the knowledge base and the reasoning. An approximation approach may be defined for one or several components. Groot et al. [40] defined three types of approximations approaches depending on the used component:

- *Language Weakening*: It consists in a rewriting of the knowledge base in a less expressive language by the elimination of a constructor or an axiom. By reducing the expressivity, the reasoning becomes less costly.
- *Knowledge Compilation*: It is a pre-processing performed on the knowledge base before the reasoning. It can be the normalization of the knowledge base or the insertion of entailed knowledge to avoid recalculating it each time.

- *Approximate Deduction*: It consists on the approximation of the logical inference to make it more flexible. In general, the approximated inferences are sound but incomplete, or complete but unsound.

Approximation approaches are generally motivated by three main goals: improving the performance of reasoning, eliminating a constructor or axioms or expanding query answers. We are interested in this work on the expansion query answering approximation.

4.5.2.2.1 Performance improvements The performance improvements approaches [18, 74, 75, 70, 17] approximate generally the syntax of a language by the elimination of constructors and axioms for translate the language into another less expressive language. They are used in general to improve the performance of a system, make decidable undecidable problems or restrict the language used in a system.

The authors of [18] defined the approximation of a concept as its rewriting in another logic less expressive in optimizing the subsumption between this rewriting and the original concept. They investigated the mapping of \mathcal{ALC} to \mathcal{ALE} and demonstrate that this mapping always exists. They defined the upper approximation as the minimal subsuming concept and also introduce a syntactic difference operator to calculate the accuracy of the approximation. The main objective of this work is to improve the performance of reasoning despite that the departure language is not very expressive. More precisely, the result of the approximation of a concept C is given by the instances of all his subsumers D that satisfy $C \sqsubseteq E \sqsubseteq D \Rightarrow D \equiv E$. This result does not necessarily extend the instance set of the concept. The subsumption constraint is very restrictive and the approximation of the concept in this case may be equivalent to the instance retrieval mechanism. The proposed approach does not explore solutions beyond the first subsumers, while this is possible in our case by increasing the value of the parameter \mathcal{W} .

In [74, 75] the authors proposed an approximation approach for the instance retrieval that preserves the soundness of the result for performing *ABox* reasoning. The approximation and the reasoning can be performed in PTime. The principle of approximation is to transform an expressive ontology language (DL \mathcal{RO}) to a less expressive ontology language (DL \mathcal{EL}^{++}) and a complement table to maintain the complementary relations between concept names.

In this work, the approximation is for the completeness of the result. The result is a subset of the interpretation of the concept. The authors aim is to improve the performance of reasoning which can be performed in PTime.

The co-authors of [70] proposed an approximation of OWL-DL language to DL-Lite language and of [17] from OWL 2 to DL-LiteA.

4.5.2.2.2 Query answering expansion It consist in general on the definition of a new non-standard semantic or on a new operator of logical inference. These approaches are used to make flexible the instance retrieval reasoning. They increase in general the number of instances of concepts.

Ecke et al. [32] proposed a new algorithm for relaxed instances query answering. This algorithm use a concept similarity measures to relax the classical instance retrieval reasoning but it is independent of the chosen concept similarity measures approach. The general principle of the algorithm is very simple, to relax instances of a concept C they compute instances of all concepts D_i such as the similarity value of the pair (C, D_i) is greater than the degree of relaxation. The proposed approach can be used with any similarity measure but the quality of relaxed instances depends on the quality of the used similarity measure. The authors discuss also two major aspects of the relaxed instance query answering: the maximum degree of relaxation and taking into account the choice of the part of the query that can be relaxed.

In [84, 83] the author introduces two non-standard interpretations to define the approximation of concepts subsumption. They describe the degree of similarity in terms of the used sub vocabulary. The two non-standard interpretations are equivalent to the standard interpretation in a subset S of the concept set. The first non-standard interpretation is a lower interpretation. It interprets the concepts that do not belong to S by \emptyset , i.e. for a lower interpretation I , $(A^I \cup \neg A^I) = \emptyset$ if $A \notin S$. This interpretation reduces the satisfiability of formulas (add non existing clash). The problem of subsumption is complete but not sound. The second non-standard interpretation is an upper interpretation. It interprets the concepts that do not belong to S by Δ^I , i.e. for a upper interpretation I , $(A^I \cap \neg A^I) = \Delta^I$ if $A \notin S$. This interpretation increases the satisfiability of formulas. The problem of subsumption is sound but not complete.

The aim of this work is to propose a new subsumption axiom more flexible than conventional subsumption axiom. This flexibility is configurable and depends on the selected subset of concepts. This work can be used to approximate concepts by defining a new flexible equivalence axiom. This equivalence axiom between two concepts C and D is equivalent to the subsumption defined by the authors between C and D and between D and C . The advantage of the approach is that it does not depend on any language. Its

limits are the low accuracy of the results which depends on the selected sub vocabulary and the incoherence of results in some cases.

4.5.2.3 ABox revision/update/erasure

These approaches investigate the dynamic aspect of a knowledge base. The dynamic aspects of a knowledge related to our work are those which concern the Abox, therefore, we present only the ABoxes work. These work concerns two main problems:

- ABox update and erasure problem: the update/erasure of a knowledge base is an old problem in logic and in database [89, 33, 90]. For the ABox, the problem is the definition of transformation occurred on the knowledge base after adding and/or deleting assertions.
- ABox revision problem: the evolution or the revision of knowledge is a manual procedure. Its integration in the knowledge based systems is necessary for some applications. Abox revision problem maintain the consistency of the ABox and integrate a new knowledge.

Liu et al. [57] use an update \mathcal{U} which is a simple consistent ABox to define the update of a interpretation with \mathcal{U} :

Definition 4.5.3 (Interpretation Update) *Let \mathcal{U} be an update and $\mathcal{I}, \mathcal{I}'$ interpretations such that $\Delta^{\mathcal{I}} = \Delta^{\mathcal{I}'}$. Then \mathcal{I}' is the result of updating \mathcal{I} with \mathcal{U} , $\mathcal{I}^{\mathcal{U}}$, if for all concept names C and roles names r :*

$$C^{\mathcal{I}'} = (C^{\mathcal{I}} \cup \{a^{\mathcal{I}} \mid C(a) \in \mathcal{U}\}) \setminus \{a^{\mathcal{I}} \mid \neg C(a) \in \mathcal{U}\}$$

$$r^{\mathcal{I}'} = (r^{\mathcal{I}} \cup \{(a^{\mathcal{I}}, b^{\mathcal{I}}) \mid r(a, b) \in \mathcal{U}\}) \setminus \{(a^{\mathcal{I}}, b^{\mathcal{I}}) \mid \neg r(a, b) \in \mathcal{U}\}$$

Definition 4.5.4 (ABox Update) *Let \mathcal{A} be an ABox and \mathcal{U} an update. An ABox \mathcal{A}' is the result of updating \mathcal{A} with \mathcal{U} if $M(\mathcal{A}') = \{\mathcal{I}^{\mathcal{U}} \mid \mathcal{I} \in M(\mathcal{A})\}$*

De Giacomo et al. [29] use the same principle and define also the erasure of an interpretation. We resume its definition and we use the same format of the first definition for having:

Definition 4.5.5 (Interpretation Erasure) *Let \mathcal{U} be an update and $\mathcal{I}, \mathcal{I}'$ interpretations such that $\Delta^{\mathcal{I}} = \Delta^{\mathcal{I}'}$. Then \mathcal{I}' is the result of erasing \mathcal{I} with \mathcal{U} , $\mathcal{I}_{\mathcal{U}}$, if for all concept names C and roles names r :*

$$C^{\mathcal{I}'} = (C^{\mathcal{I}} \cup \{a^{\mathcal{I}} \mid \neg C(a) \in \mathcal{U}\}) \setminus \{a^{\mathcal{I}} \mid C(a) \in \mathcal{U}\}$$

$$r^{\mathcal{I}'} = (r^{\mathcal{I}} \cup \{(a^{\mathcal{I}}, b^{\mathcal{I}}) \mid \neg r(a, b) \in \mathcal{U}\}) \setminus \{(a^{\mathcal{I}}, b^{\mathcal{I}}) \mid r(a, b) \in \mathcal{U}\}$$

Definition 4.5.6 (ABox Erasure) Let \mathcal{A} be an ABox and \mathcal{U} an update and $M(\mathcal{A})$ the set of all models of the ABox \mathcal{A} . An ABox \mathcal{A}' is the result of erasing \mathcal{A} with \mathcal{U} if $M(\mathcal{A}') = \{\mathcal{I}_{\mathcal{U}} \mid \mathcal{I} \in M(\mathcal{A})\}$

We can also define a standard ABox revision problem by:

Definition 4.5.7 (ABox revision problem) Let \mathcal{L}_K and \mathcal{L}_Q be DLs, $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ a knowledge base in \mathcal{L}_K and \mathcal{A}^* an ABox in \mathcal{L}_Q . We call the tuple $(\mathcal{KB}, \mathcal{A}^*)$ an ABox revision problem iff \mathcal{A}^* is consistent w.r.t. \mathcal{T} .

Definition 4.5.8 (ABox revision solution) Let \mathcal{L}_S be a DL and \mathcal{A}' an ABox in \mathcal{L}_S . \mathcal{A}' is a solution to revision problem $(\mathcal{KB}, \mathcal{A}^*)$ iff $\mathcal{KB}' = (\mathcal{T}, \mathcal{A}')$ is consistent and $\mathcal{KB}' \models \mathcal{A}^*$.

A revision operator [25]:

- *must preserve the consistency of knowledge bases;*
- *must entail the new information and preserve the protected part;*
- *should not change the original knowledge base if there is no conflict;*
- *should be independent of the syntactical forms of knowledge bases;*
- *should guarantee a minimal change.*

The result of the revision of a DL knowledge base \mathcal{KB} must be independent of the syntax of \mathcal{KB} . It can be carried out by two operations: adding and/or deleting knowledge. For both cases, the result must be a consistent knowledge base \mathcal{KB}' [56]. Two main approaches to define the distances between \mathcal{KB} and \mathcal{KB}' are distinguished by the definition of minimal change [56, 25]:

- **Model based approaches:** the result of a revision operator is defined by a set of models. The semantics of minimal change is the distance between the model of the new knowledge base and the model of the initial knowledge base. They are independent of the syntax of the knowledge base but only works for limited forms of LD-Lite family. One of these basic problems is to characterize the necessary language to capture the results of all models.

- Formula based approaches: the result is defined in terms of formulas. They can be used for more expressive DL as *SHOIN* but depend on the syntax of the knowledge base. The semantics of minimal change is reflected in the minimality of formulas which will be changed. One of these basic problems is that the formula is not unique in general.

The set of models corresponding to the update/erasure of a knowledge base K may not be expressible in the language of \mathcal{KB} [57]. De Giacomo et al. [29] investigate this problem and define the sound approximation and the maximal approximation of a terminology to introduce the maximal approximation of an ABox update/erasure to characterize the ABoxes which capture the better the update. They prove that this maximal approximation is unique and present a polynomial algorithm to calculate it for $DL\text{-Lite}_F$.

In [56], the authors investigated the problem of ABox update/erasure for the knowledge base which verify that the new knowledge base can be expressed in the same language as the initial knowledge base. They propose an algorithm to calculate a knowledge base after an insertion of assertions and another algorithm to calculate a knowledge base after a deletion of assertions for $DL\text{-Lite}_A$.

Gao et al. [34] proposed an operator for the $DL\text{-Lite}$ ABox revision. This operator deletes an assertion of each minimal inconsistent sub set from the initial ABox \mathcal{A} up to found the revised ABox \mathcal{A}' .

We have presented in this section different approaches which investigate in the dynamic aspect of Aboxes. We have presented how to integrate a new knowledge with maintaining the consistency of the knowledge base and how to characterize a knowledge base after adding and/or deleting assertions. These aspects are interesting and related to our work. We also add and delete assertions to integrate new knowledge but the problem changes, the sets of assertions to add and/or delete are initially unknown and the new knowledge to integrate is unknown also. The definition of the interpretation update changes also.

4.6 Conclusion

In this chapter, we investigated a new approach to relax answers of ontological queries that take into consideration the knowledge in an ABox. The main idea is to determine the minimum knowledge to add to or delete from the ABox so that a relaxed answer become an exact answer. The benefit of such an approach lies in the exploitation of the

knowledge that is hidden in the data (in addition to the terminology) in order to compute relaxed answers and the parameterization of the relaxation.

The \mathcal{W} Relaxation and its principle can be used in many applications. Naturally, as an instance retrieval reasoning or for the evolution of an ABox, but also for other applications less obvious as the computing of membership degree of an individual i in a concept C . This value could be used to compute a distance between concepts or individuals. The distance between two individuals can be defined as the number of common concept or role assertions or the minimum or maximum number of different concept or role assertions. The distance between two concepts can be defined as the average distance between all individuals of the two concepts or the minimum or maximum distance between two individuals of the two concepts or by another distance methods. These distances can be used for indexation or to investigate problems based on a similarity matching for semantic retrieval.

In our work, we used assertions of atomic concepts (or its negation) and roles. It will be interesting to extend our approach to assertions of defined concepts and roles and propose to normalize the computation of the distance in this context. This will extend the covered language constructed from adding/deletion assertions especially in the case of OWA for some constructors such as " \forall " and " $\leq n$ ".

We will explore in the next chapter a decision and optimization problems. The \mathcal{W} Relaxation is a very costly reasoning, we investigate these problems with the $\mathcal{ELU}_{\perp}^{(-)}$ language and we present after, the challenges for more expressive languages.

\mathcal{W} Relaxation: Algorithms And Results

Contents

5.1	Introduction	76
5.2	Decision Problems	76
5.3	Optimisation Problems	77
5.3.1	Computing minimal \mathcal{W} for \mathcal{W}^+ Relaxation	77
5.3.2	Computing minimal \mathcal{W} for \mathcal{W}^- Relaxation	82
5.3.3	Computing minimal \mathcal{W} for \mathcal{W} Relaxation	83
5.3.4	Complexity	87
5.4	Optimisation problems related to \mathcal{W}Relaxation of set of individuals	88
5.5	Conclusion	89

5.1 Introduction

We proposed in the previous chapter a new relaxation reasoning for instance retrieval. The novelty is that the reasoning is based on the ABox instead of the TBox used by classical approaches.

Many problems related to \mathcal{W} Relaxation are worth to explore, such as computing the set of relaxed answers for a given ontological query, computing the minimal distance \mathcal{W} for a given individual w.r.t. a query Q or the existence of relaxed answers for a given query. We present in this chapter the results concerning some interesting decision problems and related optimization problems w.r.t. \mathcal{W} Relaxation in the context of the OWA and the GCWA. All the problems related to \mathcal{W} Relaxation are studied by considering the language of Table 5.1. We use this language because all types of assertions in this language can be constructed using atomic assertions.

Concept constructors	Name
\perp	Universal concept
\top	Bottom concept
A	Atomic concept
$\neg A$	Atomic negation
$C \sqcap D$	Intersection
$C \sqcup D$	Union
$\exists R.C$	Qualified existential quantification
$\geq nR.C$	Qualified at-least restrictions

Table 5.1: Used concept constructors

5.2 Decision Problems

We consider first, the following existence problem in the case of \mathcal{W}^+ Relaxation:

Problem 5.2.1 (Exist $\mathcal{W}^+(C, \mathcal{KB}, a)$) Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and a be an individual of the ABox \mathcal{A} . The problem Exist $\mathcal{W}^+(C, \mathcal{KB}, a)$ is the problem of deciding whether there exists an integer \mathcal{W} such that $a \in \text{Ins}R_{\mathcal{W}}^+(C, \mathcal{KB})$.

The theorem below provides a full characterization of the Exist $\mathcal{W}^+(C, \mathcal{KB}, a)$ problem in both OWA and GCWA.

Theorem 5.2.2 *Let \mathcal{L} be the language of Table 5.1, let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a \mathcal{L} -knowledge base and C be a \mathcal{L} -concept definition. Let a be an individual of the $ABox$ \mathcal{A} . Then:*

$\exists \mathcal{W}$ such as $a \in \text{InsR}_{\mathcal{W}}^+(C, \mathcal{KB}) \Leftrightarrow \mathcal{A} \cup \{C(a)\}$ is consistent w.r.t. \mathcal{T} .

The proof of the theorem is given in the appendix. Theorem 5.2.2 reduces $\text{ExistW}^+(C, \mathcal{KB}, a)$ to a known problem of checking the consistency of an $ABox$ w.r.t. a $TBox$.

We consider now, the existence problem in the general case of \mathcal{W} Relaxation:

Problem 5.2.3 (Exist $\mathcal{W}(C, \mathcal{KB}, a)$) *Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and a be an individual of the $ABox$ \mathcal{A} . The problem $\text{ExistW}(C, \mathcal{KB}, a)$ is the problem of deciding whether there exists an integer \mathcal{W} such that $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB})$.*

We give below a full characterization of the problem $\text{ExistW}(C, \mathcal{KB}, a)$ in both OWA and GCWA.

Theorem 5.2.4 *Let \mathcal{L} be the language of Table 5.1, let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a \mathcal{L} -knowledge base and C be a \mathcal{L} -concept definition. Let a be an individual of the $ABox$ \mathcal{A} and let \mathcal{A}' the set of all atomic assertions of \mathcal{A} . Then:*

$\exists \mathcal{W}$ such as $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB}) \Leftrightarrow (\mathcal{A} \setminus \mathcal{A}') \cup \{C(a)\}$ is consistent w.r.t. \mathcal{T} .

The proof of the theorem is given in the appendix. Theorem 5.2.4 reduces also $\text{ExistW}(C, \mathcal{KB}, a)$ to a known problem of checking the consistency of an $ABox$ w.r.t. a $TBox$.

5.3 Optimisation Problems

We study in this section three optimisation problems, related to computation of the minimal distance \mathcal{W} , needed to make a given individual a instance of a given concept C w.r.t. the two semantics considered in this thesis.

5.3.1 Computing minimal \mathcal{W} for \mathcal{W}^+ Relaxation

We are interested in this section in the problem of computing the minimal distance \mathcal{W} in the context of \mathcal{W}^+ Relaxation. This problem can be stated formally as follows:

Problem 5.3.1 (Comput $\mathcal{W}^+(C, \mathcal{KB}, a)$) Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and a be an individual. The problem $\text{Comput}\mathcal{W}^+(C, \mathcal{KB}, a)$ is the problem of computing minimal \mathcal{W} such that $a \in \text{Ins}R_{\mathcal{W}}^+(C, \mathcal{KB})$.

To solve $\text{Comput}\mathcal{W}^+(C, \mathcal{KB}, a)$ problem, we propose to transform the assertion $C(a)$ to a set of atomic assertions to identify the atomic assertions which are not entailed by \mathcal{KB} , i.e., the atomic assertions to add. The transformation can generate several sets of atomic assertions. The solution is hence to identify the set defining the minimal number of atomic assertions to add.

First, the knowledge base is unfolded. This task consists in recursively replacing all non-atomic concept name C (resp. non atomic assertion $C(a)$) defined in \mathcal{T} (resp. \mathcal{A}) by $C \equiv D$ with D (resp. $D(x)$).

Second, we proceed in two steps: construction of the minimal atomic assertions sets and computation of the minimal distance \mathcal{W}^+ . These two steps are detailed below.

Step 1: Construction of the Minimal Atomic Assertions Sets (MAAS)

In this step, we construct the minimum sets of atomic assertions representing the possible decomposition of $C(a)$. Algorithm 1 computes these sets. It consists mainly in the application of the transformation rules of Table 5.2 to decompose assertions. The first rule allows to detect entailments and the other three rules allow to eliminate the operators \sqcup , \exists and \sqcap respectively.

Algorithm 1 takes as input an assertion $C(a)$ and produces as output the possible sets required to make a instance of C .

Algorithm 1: Computing Minimal Atomic Assertions Sets MAAS.

Input: C : a concept, a : an individual, $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$: a knowledge base

Output: $MAAS(\mathcal{KB}, C, a)$: sets of atomic assertions

```

1 begin
2    $MAAS(\mathcal{KB}, C, a) \leftarrow \{C(a)\};$ 
3   while a rule of Table 5.2 can be applied to a set  $S \in MAAS(\mathcal{KB}, C, a)$  do
4     Apply an inference rule ;
5     if no inference rule can be applied then
6       Apply a decomposition rule ;
7   return  $MAAS(\mathcal{KB}, C, a)$  ;
```

Inference rule	
\rightarrow_{\sqcup} rule	condition $(D \sqcup E)(x) \in \mathcal{A}$ and $E(x) \in S$ and D not contains " \exists " and $(S \cup \{\neg D(x)\} \setminus \{E(x)\}) \notin MAAS(\mathcal{KB}, C, a)$ action $MAAS(\mathcal{KB}, C, a) = MAAS(\mathcal{KB}, C, a) \cup \{S \cup \{\neg D(x)\} \setminus \{E(x)\}\}$
Decomposition rules	
\rightarrow_{\sqcup} rule	condition S contains $(D_1 \sqcup D_2)(x)$ action $MAAS(\mathcal{KB}, C, a) = MAAS(\mathcal{KB}, C, a) \cup (\{S \cup \{D_1(x)\} \setminus \{(D_1 \sqcup D_2)(x)\}\} \cup (\{S \cup \{D_2(x)\} \setminus \{(D_1 \sqcup D_2)(x)\}\}) \setminus \{S\}$
\rightarrow_{\exists} rule	condition S contains $(\exists R.D)(x)$ action $MAAS(\mathcal{KB}, C, a) = MAAS(\mathcal{KB}, C, a) \cup (\{S \cup \{R(x, z), D(z)\} \setminus \{(\exists R.D)(x)\}\} \cup_{y \in ind(\mathcal{A})} (\{S \cup \{R(x, y), D(y)\} \setminus \{(\exists R.D)(x)\}\}) \setminus \{S\}$ such that $z \notin ind(\mathcal{A})$ is an individual.
\rightarrow_{\sqcap} rule	condition S contains $(D_1 \sqcap D_2)(x)$ action $S = S \cup \{D_1(x), D_2(x)\} \setminus \{(D_1 \sqcap D_2)(x)\}$

Table 5.2: Transformation rules of MAAS computing algorithm.

Example 9 Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be the following knowledge base.

$\mathcal{T} = \{C \equiv (A \sqcap B) \sqcup \neg B\}$ and $\mathcal{A} = \{a\}$.

The application to $C(a)$ of the " \rightarrow_{\sqcup} rule" and the " \rightarrow_{\sqcap} rule" respectively enables to generate the two following sets: $set_1 = \{A(a), B(a)\}$ and $set_2 = \{\neg B(a)\}$.

Hence, we have $MAAS(\mathcal{KB}, C, a) = \{set_1, set_2\}$. One of the sets set_1 or set_2 is required to make a instance of C ,

The following lemma ensures the correctness of Algorithm 1.

Lemma 5.3.2 Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and a be an individual. Then, we have $\mathcal{KB} \models C(a) \Leftrightarrow \exists S \in MAAS(\mathcal{KB}, C, a) \mid \mathcal{KB} \models S$.

The proof of the lemma is given on the appendix.

The MAAS computed by Algorithm 1, is used in the second step described below, to compute the minimal distance \mathcal{W}^+ .

Step 2: Computation of the Minimal distance \mathcal{W}^+

To compute the minimal distance \mathcal{W}^+ , we proceed in two steps: first, we remove from the MAAS computed by Algorithm 1 the sets which are inconsistent with the knowledge

base. The obtained set noted $MAAS^+$ is computed as follows:

$$MAAS^+(\mathcal{KB}, C, a) = \{S \in MAAS(\mathcal{KB}, C, a) \mid \mathcal{KB} \cup S \not\models \perp\}.$$

Each element of $MAAS^+$ represents a candidate set of atomic assertions that can be added to \mathcal{KB} to make the individual a instance of C . However, some elements of $MAAS^+$ contain redundancy, that is to say they include atomic assertions which can be deduced from the knowledge base. Therefore, these sets are not minimal w.r.t. the number of atomic assertions required to make a instance of C .

So, we define the function Res^+ , that enables to remove redundant elements from a set S as follows:

$$\forall S \in MAAS^+(\mathcal{KB}, C, a), Res^+(\mathcal{KB}, S) = \{P \in S \mid \mathcal{KB} \not\models P\}.$$

As a consequence, the function Res^+ enables to transform any set elements of $MAAS^+$ to a minimal set of atomic assertions that can be added to \mathcal{KB} to make a instance of C .

The cardinality of each set $Res^+(\mathcal{KB}, S)$, for $S \in MAAS^+(\mathcal{KB}, C, a)$, provides a distance \mathcal{W}^+ for the \mathcal{W}^+ Relaxation of an individual a w.r.t. a concept C .

Hence, a minimal distance \mathcal{W}^+ can be computed for these sets as stated by the following lemma:

Lemma 5.3.3

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base, C be a concept and a be an individual.

The smallest $\mathcal{W} \mid a \in InsR_{\mathcal{W}}^+(C, \mathcal{KB})$ is equal to $Min(| Res^+(\mathcal{KB}, S) |)$, $\forall S \in MAAS^+(\mathcal{KB}, C, a)$.

Example 10 Let a DL knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ of Table 5.3. We illustrate the different steps of the proposed approach to compute minimal \mathcal{W} such that $a \in InsR_{\mathcal{W}}^+(C, \mathcal{KB})$ in the context of OWA.

\mathcal{T}		\mathcal{A}	
C	$\equiv E \sqcup F$	A	$\{a, c\}$
D	$\equiv A \sqcap B$	$\neg A$	$\{b\}$
E	$\equiv D \sqcap \exists R.(A \sqcup B)$	$\neg B$	$\{b, c\}$
F	$\equiv \exists S.B$	R	$\{(a, b), (c, a)\}$
		$B \sqcup F$	$\{a\}$

Table 5.3: Example of knowledge base.

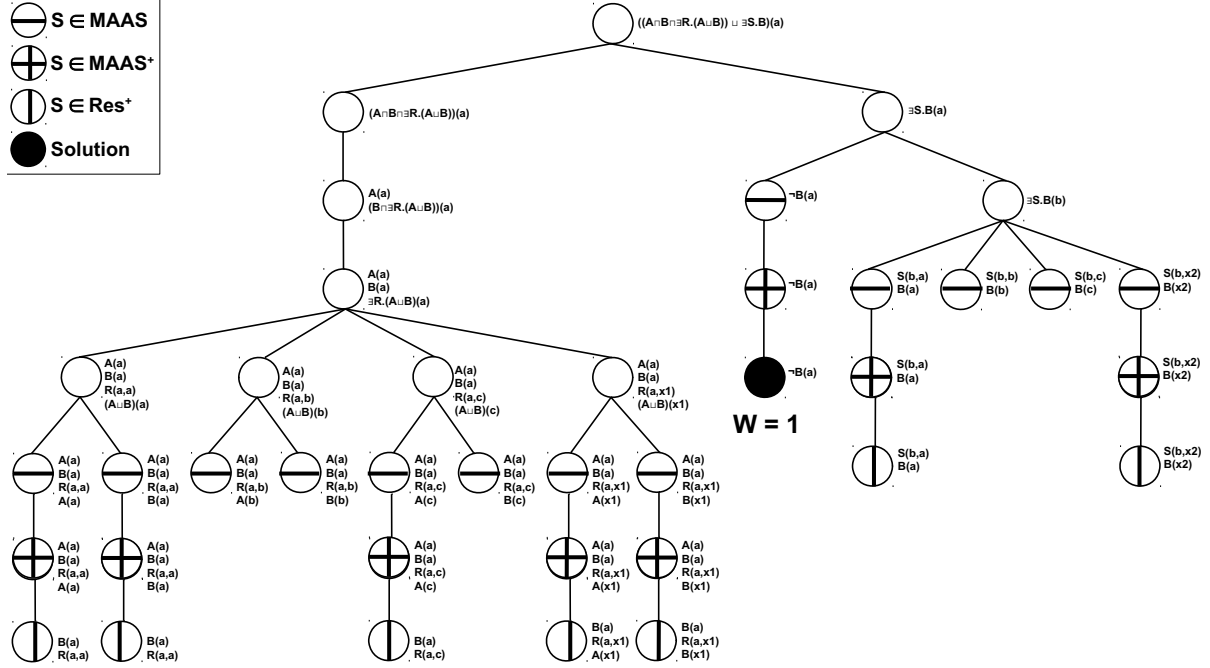


Figure 5.1: Example of computing minimal \mathcal{W} for \mathcal{W}^+ Relaxation

Figure 5.1 shows the computation of the minimal distance \mathcal{W} in the context of \mathcal{W}^+ Relaxation. We start with the root of the tree labelled with the assertion $C(a)$. Some edges correspond to the application of the transformation rules while other edges correspond to the computation of MAAS^+ and Res^+ .

For example, we start with the root labelled with $(A \cap B \cap \exists R.(A \cup B) \cup \exists S.B)(a)$ and apply the “ \rightarrow_{\cup} ” rule to obtain two children nodes labelled respectively with $(A \cap B \cap \exists R.(A \cup B))(a)$ and $\exists S.B(a)$.

A node make labelled with an element $S \in \text{MAAS}(\mathcal{KB}, C, a)$ when the application of the transformation rules is not possible. As for the two nodes N_1 and N_2 labelled respectively with $S_1 = \{A(a), B(a), R(a, a), B(a)\}$ and $S_2 = \{A(a), B(a), R(a, b), A(b)\}$.

S_1 is consistent with \mathcal{KB} , then $S_1 \in \text{MAAS}^+(\mathcal{KB}, C, a)$ and S_2 is inconsistent with \mathcal{KB} , then $S_2 \notin \text{MAAS}^+(\mathcal{KB}, C, a)$.

Finally, the redundant elements of S_1 is removed. We obtain a candidate solution labelled with $S_3 = \{B(a), R(a, a)\}$.

The leaf nodes are candidate solutions. The optimal solutions are given by the filled leave node.

5.3.2 Computing minimal \mathcal{W} for \mathcal{W}^- Relaxation

We consider now the problem of computation the minimal \mathcal{W} in the context of \mathcal{W}^- Relaxation. This problem can be stated formally as follows:

Problem 5.3.4 (Comput $\mathcal{W}^-(C, \mathcal{KB}, a)$) *Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and a be an individual. The problem $\text{Comput}\mathcal{W}^-(C, \mathcal{KB}, a)$ is the problem of computing minimal \mathcal{W} such that $a \in \text{Ins}R_{\mathcal{W}}^-(C, \mathcal{KB})$.*

As in the previous case, to solve this problem, we propose to transform the assertion $C(a)$ to a set of atomic assertions to identify the atomic assertions which are inconsistent with \mathcal{KB} , i.e., the atomic assertions to delete. The transformation can generate several candidate sets. The solution is hence to identify the set defining the minimal number of atomic assertions to delete.

We proceed in two steps: construction of the minimal atomic assertions sets and computation of the minimal distance \mathcal{W}^- .

Step 1: Construction of the Minimal Atomic Assertions Sets (MAAS)

The $MAAS(\mathcal{KB}, C, a)$ is computed as previously (Algorithm 1). It is used in the second step described below to identify the set of atomic assertions which are inconsistent \mathcal{KB} and compute the minimal distance \mathcal{W}^- .

Step 2: Computation of the Minimal distance \mathcal{W}^-

To compute the minimal distance \mathcal{W}^- , we proceed in two steps: first we define the function Res_{\perp} , that enables to identify each atomic assertion of a set S which is inconsistent with the $ABox$ \mathcal{A} as follows:

$$\forall S \in MAAS(\mathcal{KB}, C, a), Res_{\perp}(\mathcal{KB}, S) = \{P \in \mathcal{A} \mid \neg P \in S \text{ and } S \text{ is consistent w.r.t. } \mathcal{T}\}.$$

Each set $Res_{\perp}(\mathcal{KB}, S)$ represents a candidate set of atomic assertions that can be removed from \mathcal{KB} to make the individual a instance of C . However, the removal of this set must make a instance of C , to become a solution.

So, we define the function Res_{\perp}^- , that enables to identify the elements S of $MAAS(\mathcal{KB}, C, a)$, such that the removal of $Res_{\perp}(\mathcal{KB}, S)$ from the $ABox$ make a instance of C as follows:

$$\forall S \in MAAS(\mathcal{KB}, C, a), Res_{\perp}^-(\mathcal{KB}, C, a) = \{Res_{\perp}(\mathcal{KB}, S) \mid \mathcal{KB} \setminus Res_{\perp}(\mathcal{KB}, S) \models C(a)\}.$$

As a consequence, the function Res_{\perp}^- enables to identify from any set elements of $MAAS$ a minimal set of atomic assertions that can be removed from \mathcal{KB} to make a instance of C .

The cardinality of each set $Res_{\perp}^-(\mathcal{KB}, C, a)$, for $S \in MAAS(\mathcal{KB}, C, a)$, provides the distance \mathcal{W}^- for the \mathcal{W}^- -Relaxation of an individual a w.r.t. a concept C .

Hence, a minimal distance \mathcal{W}^- can be computed from these sets as stated by the following lemma:

Lemma 5.3.5

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base, C be a concept and a be an individual.

The smallest $\mathcal{W}^- \mid a \in InsR_{\mathcal{W}^-}^-(C, \mathcal{KB})$ is equal to $Min(| Res_{\perp}^-(\mathcal{KB}, C, a) |)$.

Example 11 *Let the DL knowledge base \mathcal{KB} of Example 10 and a concept $G \equiv \exists R. \neg A \sqcap (A \sqcup B)$. We illustrate the different steps of the proposed approach to compute minimal \mathcal{W}^- such that $c \in InsR_{\mathcal{W}^-}^-(G, \mathcal{KB})$ in the context of GCWA.*

Figure 5.2 shows the computation of the minimal distance \mathcal{W}^- in the context of \mathcal{W}^- -Relaxation. We start with the root of the tree labelled with the assertion $G(c)$. Some edges correspond to the application of the transformation rules while other edges correspond to the computation of Res_{\perp} and Res_{\perp}^- .

For example, let the node labelled with $\{\exists R. \neg A(c), (A \sqcup B)(c)\}$ and apply the “ $\rightarrow \exists$ rule” to obtain four children nodes labelled respectively with $R(c, a), \neg A(a), (A \sqcup B)(c)$, $R(c, b), \neg A(b), (A \sqcup B)(c)$, $R(c, c), \neg A(c), (A \sqcup B)(c)$ and $R(c, x1), \neg A(x1), (A \sqcup B)(c)$.

The two nodes N_1 and N_2 labelled respectively with $S_1 = \{R(c, a), \neg A(a), A(c)\}$ and $S_2 = \{R(c, a), \neg A(a), B(c)\}$ are labelled with elements of $MAAS(\mathcal{KB}, G, c)$.

Then, we identify atomic assertions of each element of $MAAS(\mathcal{KB}, G, c)$, which is inconsistent with \mathcal{KB} . $S'_1 = \{A(a)\}$ for S_1 and $S'_2 = \{A(a), \neg B(c)\}$ for S_2 .

Finally, candidate solutions are identified. S'_1 is a solution but S'_2 is not a solution.

The leaf nodes are candidate solutions. The optimal solution are given by the filled leave node.

5.3.3 Computing minimal \mathcal{W}^- for \mathcal{W}^- Relaxation

We consider now the general problem of computing the minimal \mathcal{W}^- in the context of \mathcal{W}^- Relaxation. This problem can be stated formally as follows:

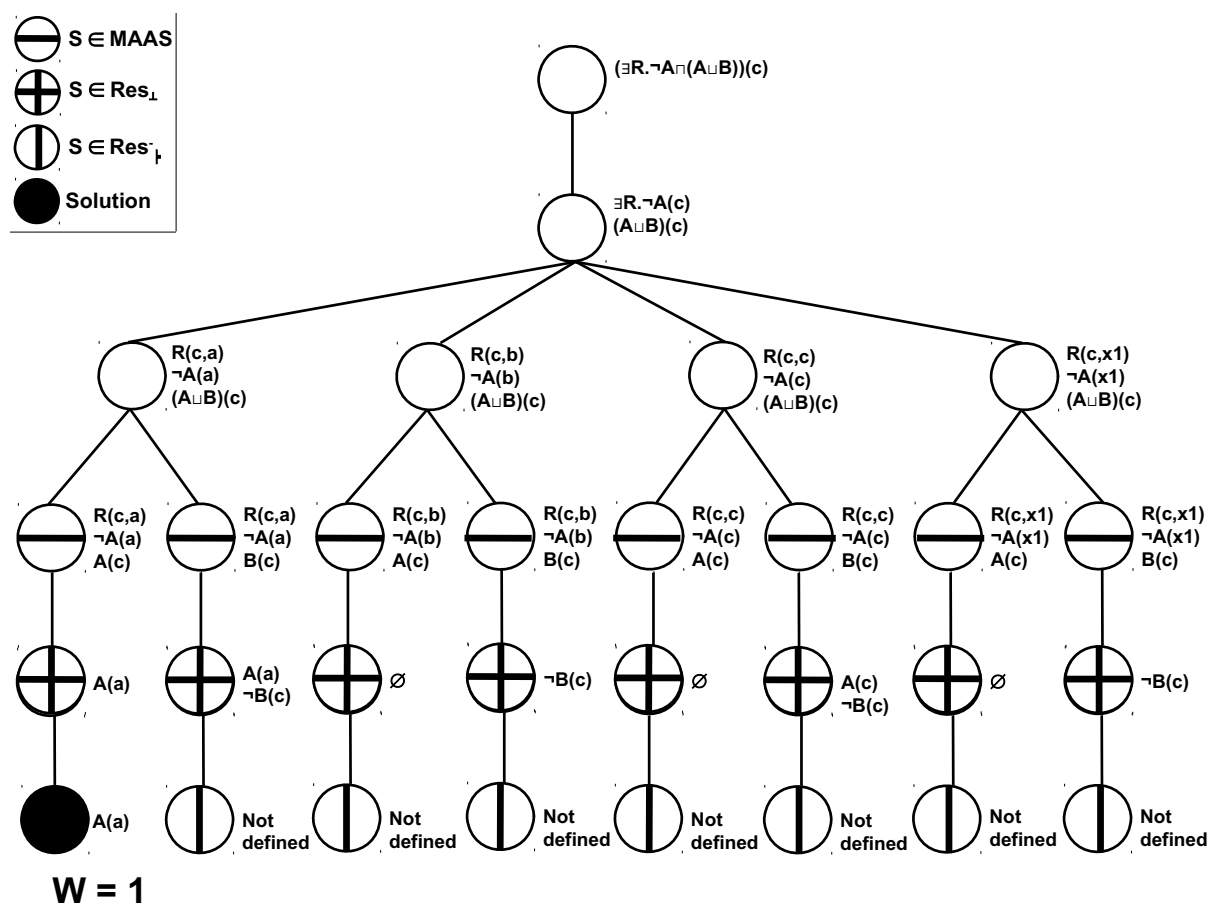


Figure 5.2: Example of computing minimal \mathcal{W} for \mathcal{W}^- Relaxation

Problem 5.3.6 (Comput $\mathcal{W}(C, \mathcal{KB}, a)$) Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and a be an individual. The problem $\text{Comput}\mathcal{W}(C, \mathcal{KB}, a)$ is the problem of computing minimal \mathcal{W} such that $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB})$.

As in the previous case, we propose to transform the assertion $C(a)$ into a set of atomic assertions to identify the atomic assertions which cannot be deduced from \mathcal{KB} and the atomic assertions which are inconsistent with \mathcal{KB} , i.e., the atomic assertions to add and the atomic assertions to remove. The transformation can generate several candidate sets. The solution is hence to identify the set defining the minimal number of atomic assertions to add and to remove.

We proceed in two steps: construction of the minimal atomic assertions sets and computation of the minimal distance \mathcal{W} .

Step 1: Construction of the Minimal Atomic Assertions Sets ($MAAS$)

The $MAAS(\mathcal{KB}, C, a)$ is computed as previously (Algorithm 1). It is used in the second step described below to identify the set of atomic assertions which are not deduced from \mathcal{KB} and the set of atomic assertions which are inconsistent with \mathcal{KB} .

Step 2: Computation of the Minimal distance \mathcal{W}

To compute the minimal distance \mathcal{W} , we proceed in two steps: first we define the same function Res_{\perp} of the previous problem.

Now, each set $Res_{\perp}(\mathcal{KB}, S)$ represents a candidate set of atomic assertions that can be removed from \mathcal{KB} . It remains now to compute the set of atomic assertions to be added.

So, we define the function Res_{\models} , that enables to identify for each element $S \in MAAS(\mathcal{KB}, C, a)$, the set of atomic assertions which are not deduced from \mathcal{KB} as follows: $\forall S \in MAAS(\mathcal{KB}, C, a), Res_{\models}(\mathcal{KB}, S) = \{P \in S \mid \mathcal{KB} \setminus Res_{\perp}(\mathcal{KB}, S) \not\models P\}$

As a consequence, the functions Res_{\models} and Res_{\perp} enable us to identify from any set elements of $MAAS$ a minimal set of atomic assertions that can be respectively added to and removed from \mathcal{KB} to make a instance of C .

The cardinality of each set $(Res_{\models}(\mathcal{KB}, S) \cup Res_{\perp}(\mathcal{KB}, S))$, with $S \in MAAS(\mathcal{KB}, C, a)$ provides the distance \mathcal{W} for the \mathcal{W} Relaxation of an individual a w.r.t. a concept C .

Hence, a minimal distance \mathcal{W} can be computed for these sets as stated by the following lemma:

Lemma 5.3.7

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base, C be a concept and a be an individual.

The smallest $\mathcal{W} \mid a \in \text{Ins}R_{\mathcal{W}}(C, \mathcal{KB})$ is equal to $\text{Min}(| \text{Res}_{\models}(\mathcal{KB}, S) \cup \text{Res}_{\perp}(\mathcal{KB}, S) |)$, $\forall S \in \text{MAAS}(\mathcal{KB}, C, a)$.

Example 12 Let us consider the DL knowledge base \mathcal{KB} of Example 10 and a concept $G \equiv \exists R. \neg A \sqcap (A \sqcup B)$. We illustrate the different steps of the proposed approach to compute minimal \mathcal{W} such that $c \in \text{Ins}R_{\mathcal{W}}(G, \mathcal{KB})$ in the context of OWA.

Figure 5.3 shows the computation of the minimal distance \mathcal{W} in the context of

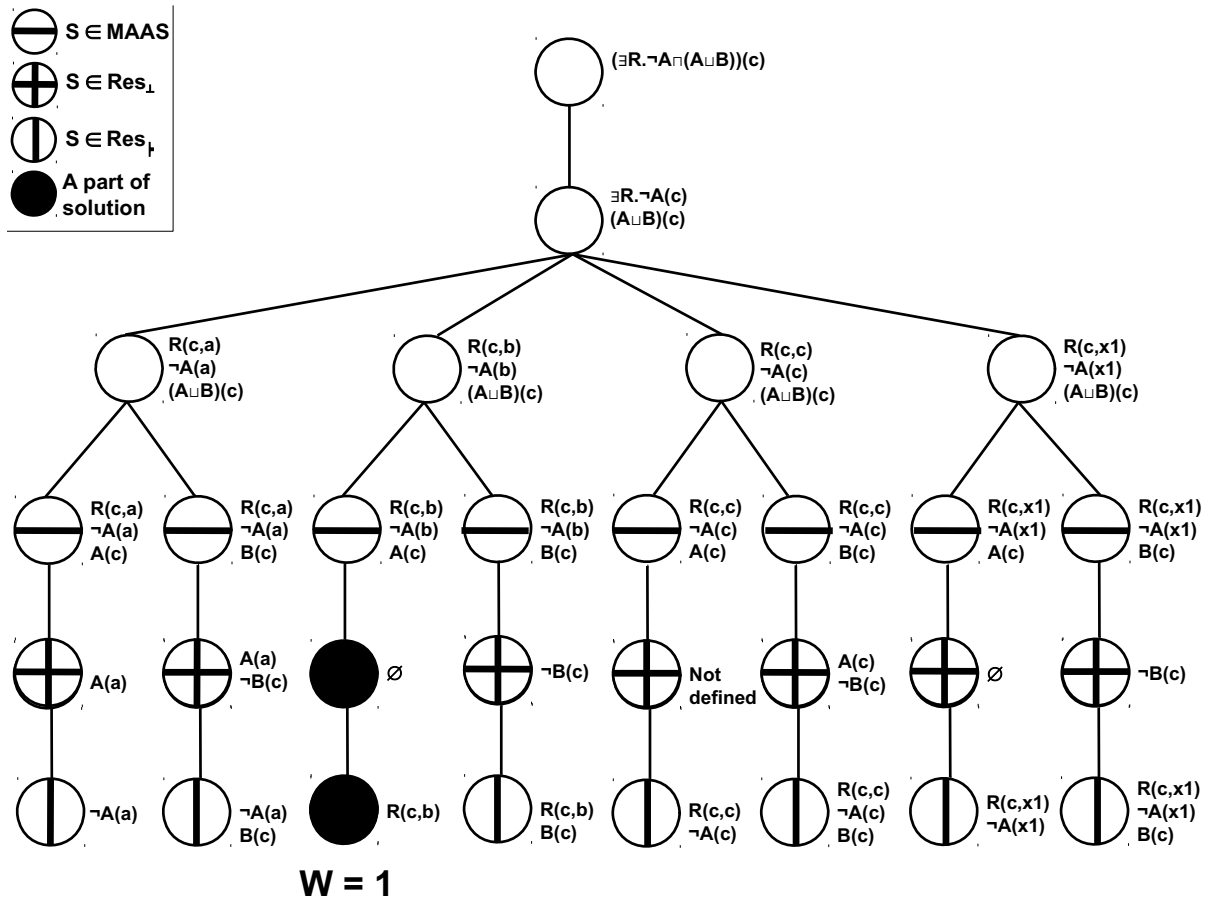


Figure 5.3: Example of computing minimal \mathcal{W} for \mathcal{W} Relaxation

\mathcal{W} Relaxation. We start with the root of the tree labelled with the assertion $G(c)$. Some edges correspond to the application of the transformation rules while other edges correspond to the computation of Res_{\perp} and Res_{\models} .

For example, we start with the root labelled with $\{(\exists R. \neg A \sqcap (A \sqcup B))(c)\}$ and apply the

“ \rightarrow_{\sqcap} rule” to obtain one child node labelled with $\{\exists R.\neg A(c), (A \sqcup B)(c)\}$.

The two nodes N_1 and N_2 labelled respectively with $S_1 = \{R(c, a), \neg A(a), B(c)\}$ and $S_2 = \{R(c, b), \neg A(b), A(c)\}$ are labelled with elements of $MAAS(\mathcal{KB}, G, c)$.

Then, atomic assertions to add to (to remove from) the ABox are identified. For S_1 , the atomic assertions to add are $\{\neg A(a), B(c)\}$ and the atomic assertions to remove are $\{A(a), \neg B(c)\}$.

The leaf nodes are candidate solutions. The optimal solution are given by the filled leaves nodes.

5.3.4 Complexity

The following lemma gives the complexity of $\text{Comput}\mathcal{W}(C, \mathcal{KB}, a)$.

Lemma 5.3.8 *The problem $\text{Comput}\mathcal{W}(C, \mathcal{KB}, a)$ is NP-hard.*

Proof of this lemma is given by the reduction of the boolean satisfiability problem (SAT) explained below.

Reduction of SAT problem to $\text{Exist}\mathcal{W}(C, \mathcal{KB}, a)$ problem

Let a boolean formula E , defined with propositional variables, constants (true or false), and \wedge, \vee, \neg operators.

The SAT problem is to decide if there are some assignments to variables of E such that E is true [41].

The SAT problem can be reduced to the $\text{Exist}\mathcal{W}(C, \mathcal{KB}, a)$ problem as follows:

- Each literal “ p ” is represented by an atomic concept “ A_p ”.
- The boolean formula is transformed into a concept C by replacing each literal “ p ” by “ A_p ”, each literal “ $\neg p$ ” by “ $\neg A_p$ ”, “ \wedge ” by “ \sqcap ” and “ \vee ” by “ \sqcup ”.
- We start with an empty ABox and we seek if an individual i belongs to a \mathcal{W} Relaxation of C .
- The SAT problem will be solved if \mathcal{W} exists.

Hence, $\text{Exist}\mathcal{W}(C, \mathcal{KB}, a)$ is at least as hard to SAT and $\text{Comput}\mathcal{W}(C, \mathcal{KB}, a)$ is NP-hard (if a decision problem is NP-complete, then the optimization problem is NP-hard).

Example 13 Let a SAT problem P_1 : let a propositional formula $E \Leftrightarrow (p \vee \neg q) \wedge (\neg p \vee \neg q) \wedge (p \vee q)$, decide if there is an assignment to p and q such that E is true.

P_1 can be reduced to an $\text{Exist}\mathcal{W}(C, \mathcal{KB}, a)$ problem P_2 as follows :

P_2 : p and q are represented by atomic concepts A and B respectively, $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$, $\mathcal{T} = \{A, B, C\}$, $\mathcal{A} = \emptyset$, $\text{Ind}(\mathcal{A}) = \{a\}$, $C \equiv (A \sqcup \neg B) \sqcap (\neg A \sqcup \neg B) \sqcap (A \sqcup B)$. The problem is to decide if $\exists \mathcal{W}$ such that $a \in \text{Ins}R_{\mathcal{W}}^+(C, \mathcal{KB})$.

For $\mathcal{A} = \mathcal{A} \cup \{A(a), \neg B(a)\}$, a is instance of C therefore $\exists \mathcal{W} = 2$ and $a \in \text{Ins}R_2^+(C, \mathcal{KB})$. So, there exists an assignment $p = \text{true}$ and $q = \text{false}$ such that E is true.

5.4 Optimisation problems related to \mathcal{W} Relaxation of set of individuals

We study in this section two optimisation problems, related to computation of relaxed answers by considering a set of individuals w.r.t. the two semantics considered in the thesis. We consider first, the problem in the case of \mathcal{W}^+ Relaxation.

Problem 5.4.1 (RlxComput $\mathcal{W}^+(C, \mathcal{KB})$) Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{RlxComput}\mathcal{W}^+(C, \mathcal{KB})$ is the problem of computing $\text{RxdIR}_{\mathcal{W}}^+(C, \mathcal{KB})$, the set of relaxed instances of the concept C , by adding atomic assertions, with a distance \mathcal{W} .

The lemma below enables to use the algorithm of computing \mathcal{W} for a \mathcal{W}^+ Relaxation to construct a solution. The constructed solutions (element of $MAAS^+$) for each individual can be combined w.r.t. criteria used by a given distance function.

Lemma 5.4.2 Let \mathcal{L} be the language of Table 5.1 and let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a \mathcal{L} -knowledge base, C be a \mathcal{L} -concept definition, a, b be individuals of the ABox \mathcal{A} and let $\mathcal{W}_1, \mathcal{W}_2, \mathcal{W}_3$ be integers.

Then, $a \in \text{Ins}R_{\mathcal{W}_1}^+(C, \mathcal{KB})$, $b \in \text{Ins}R_{\mathcal{W}_2}^+(C, \mathcal{KB}) \Rightarrow \exists \mathcal{W}_3$ such as $\{a, b\} \subseteq \text{RxdIR}_{\mathcal{W}_3}^+(C, \mathcal{KB})$.

We consider now, the optimisation problem in the general case of \mathcal{W} Relaxation:

Problem 5.4.3 (RlxComput $\mathcal{W}(C, \mathcal{KB})$) Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{RlxComput}\mathcal{W}(C, \mathcal{KB})$ is the problem

of computing $RxdIR_{\mathcal{W}}(C, \mathcal{KB})$, the set of relaxed instances of the concept C , by adding and removing atomic assertions, with a distance \mathcal{W} .

relaxation instances of the concept C by adding and removing assertions with a distance \mathcal{W} , $RxdIR_{\mathcal{W}}(C, \mathcal{KB})$.

The lemma below enables to use the algorithm of computing \mathcal{W} for a \mathcal{W} Relaxation to construct a solution.

Lemma 5.4.4 *Let \mathcal{L} be the language of Table 5.1 and let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a \mathcal{L} -knowledge base, C be a \mathcal{L} -concept definition, a, b be individuals of the ABox \mathcal{A} and let $\mathcal{W}_1, \mathcal{W}_2, \mathcal{W}_3$ be integers.*

Then, $a \in InsR_{\mathcal{W}_1}(C, \mathcal{KB})$, $b \in InsR_{\mathcal{W}_2}(C, \mathcal{KB}) \Rightarrow \exists \mathcal{W}_3$ such as $\{a, b\} \subseteq RxdIR_{\mathcal{W}_3}(C, \mathcal{KB})$.

5.5 Conclusion

In this section, we have defined some of the \mathcal{W} Relaxation problems. These problems are: decision problems related to deciding the existence of \mathcal{W} needed to make a given individual a instance of a given concept C , and optimisation problems, related to computation of the minimal distance \mathcal{W} , needed to make a given individual a instance of a given concept C , and related to computation of relaxed answers by considering a set of individuals, in the case of the three \mathcal{W} Relaxation.

Conclusions and Future Work

In this thesis, two main problems are investigated: a semantic gastroenterological image retrieval and a relaxed answers for ontological queries.

We proposed and implemented a semantic gastroenterological images annotation and retrieval system based on a new polyp ontology to help physicians, to decide whether they remove a polyp or not. The system includes:

- *Ontology*: we use description logics as formalism to define the polyp ontology. The knowledge collected about polyps is used for diagnosing cancer which is based on the analysis of polyps present in an organ. The principle is to clearly identify the type and the state of the polyp by retrieving its standard classes used in gastroenterology.
- *Annotation*: An annotation (or query) of an image is manually generated using an interactive interface. We use semantic information to describe the content of the image, i.e., concepts, individuals or roles of the polyp ontology.
- *Reasoning*: we have implemented three types of reasoning: a standard reasoning which is the instance retrieval and two other reasonings, a local subsumees retrieval and a local subsumers retrieval which are a slight modifications of the standard subsumption reasoning.

We proposed also a new approach for computing relaxed answers of ontological queries based on a notion of an edit distance of a given individual w.r.t. a given query. Such a distance is computed by counting the number of elementary operations needed to be applied to an ABox in order to make a given individual a correct answer to a given query. The considered elementary operations are added to or removed from an ABox, assertions of the form of an atomic concept (or its negation) and/or atomic roles. The main goal is to increase the number of concepts instances according to some defined criteria. In particular, we investigate:

- *WRelaxation by integrating individuals one by one*: The general idea is to determine the minimum knowledge to add to or to remove from an ABox so that a given individual can be considered as an instance of a given concept. Such a quantity of knowledge is defined by counting the elementary operations performed over the ABox. The individuals are integrated into the instances of a concept one by one.
- *WRelaxation by integrating a set of individuals*: After having presented the limits of the integration of individuals one by one, we redefine the WRelaxation by taking into account the interaction between the elementary operations which allow to integrate each individual. The objective is to form a single consistent ABox which allows for the integration of a set of individuals in the relaxation of a given concept.
- *Algorithms for decision problems*: we explore a decision problems related to deciding the existence of \mathcal{W} needed to make a given individual a instance of a given concept C , and we reduce these problems to known problems of checking the consistency of an ABox w.r.t. a TBox.
- *Algorithms for optimisation problems*: we study some optimisation problems, related to computation of the minimal distance \mathcal{W} , needed to make a given individual a instance of a given concept C , and related to computation of relaxed answers by considering a set of individuals, in the case of the three WRelaxation. We propose three algorithms inspired from tableau algorithm to solve some optimisation problems, related to computation of the minimal distance \mathcal{W} .
- *Application of WRelaxation*: we present the utilisation of WRelaxation with a medical images ontology to improve the semantic images retrieval and to enrich and/or to correct an existing images annotations database.

Future Work

We want to study two extension of our reasoning: to support relaxation constraints and also the generalization of approach to non elementary operations.

1. *Distance extension*: The distance between an individual and a query is computed by counting the number of elementary operations needed to be applied to an ABox in order to make the individual instance of the query. We defined the elementary

operations as the adding to or the removing from an ABox, of atomic assertions. For the open world assumption, the types of assertions which can be constructed with these operations are limited. The assertion $\forall R.A(a)$ for example, cannot be constructed using atomic assertions. It becomes necessary to extend the elementary operations to be able to construct other types of assertions for expressive languages.

2. ***Constraints extension:*** The main objective of our reasoning is to make flexible, the instance retrieval reasoning for increasing the set of answers. This flexibility is implemented by the adding to or the removing from an ABox of atomic assertions. Which is not practical when answers must have some properties.

Consider the following example, let a query $Q \equiv Polyp \sqcap \exists color.\{red\} \sqcap \exists shape.\{flat\}$, to retrieve the red polyps with flat shape. If we want to relax the query, it is clear that a physician who wants to decide whether he remove a polyp or not seeks to have of polyps as results. It is not interesting to retrieve for example, a red car with a flat shape but rather red polyps or polyps with a flat shape.

We add the notion of prohibited operations to our definition of relaxation. An insertion or a deletion assertion operation is prohibited if it cannot be used to relax an individual.

List of Figures

3.1	Part of the core hierarchy of the polyp ontology. Snapshot taken in Protégé 4.3 [10] with the OntoGraf plug-in.	40
3.2	Annotation/query construction	44
3.3	Application architecture	47
5.1	Example of computing minimal \mathcal{W} for \mathcal{W}^+ Relaxation	81
5.2	Example of computing minimal \mathcal{W} for \mathcal{W}^- Relaxation	84
5.3	Example of computing minimal \mathcal{W} for \mathcal{W} Relaxation	86

List of Tables

2.1	Example of DL based knowledge base.	9
2.2	Concept constructors of $\mathcal{ALCHOQ}(\neg)$	12
2.3	Individual axioms of $\mathcal{ALCHOQ}(\neg)$	13
2.4	Semantics of $\mathcal{ALCHOQ}(\neg)$	15
3.1	Implemented reasonings.	45
4.1	A knowledge base $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ of a medical images system.	55
5.1	Used concept constructors	76
5.2	Transformation rules of $MAAS$ computing algorithm.	79
5.3	Example of knowledge base.	80

Bibliography

- [1] Dicom, digital imaging and communications in medicine, 2011. <http://dicom.nema.org/>.
- [2] Fact++, the fact++ owl-dl and (partial) owl2-dl reasoner, 2011. <http://code.google.com/p/factplusplus/>.
- [3] Hermit, the hermit owl2 reasoner., 2011. <http://www.hermit-reasoner.com/>.
- [4] Imageclef, the clef cross language image retrieval track, 2011. <http://www.imageclef.org/>.
- [5] Mst, minimal standard terminology for gastrointestinal endoscopy, 2011. <http://www.worldendo.org/mst.html>.
- [6] Owl, the web ontology language, 2011. <http://www.w3.org/TR/owl-features/>.
- [7] Pellet, the pellet owl2 reasoner., 2011. <http://clarkparsia.com/pellet/>.
- [8] Racerpro, the racerpro owl/rdf reasoner., 2011. <http://www.racer-systems.com/>.
- [9] Rdf, the resource description framework, 2011. <http://www.w3.org/RDF/>.
- [10] Protégé, the open source ontology editor and knowledge-base framework., 2014. <http://protege.stanford.edu>.
- [11] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995.
- [12] Atocha Aliseda-Llera. *Seeking Explanations: Abduction in Logic, Philosophy of Science and Artificial Intelligence*. PhD thesis, Stanford, CA, USA, 1998. UMI Order No. GAX98-10072.
- [13] Grigoris Antoniou and Mary-Anne Williams. *Nonmonotonic reasoning*. MIT Press, 1997.

-
- [14] Franz Baader. The instance problem and the most specific concept in the description logic EL w.r.t. terminological cycles with descriptive semantics. In *KI 2003: Advances in Artificial Intelligence, 26th Annual German Conference on AI, KI 2003, Hamburg, Germany, September 15-18, 2003, Proceedings*, pages 64–78, 2003.
- [15] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications (2nd Edition)*. Cambridge University Press, 2007.
- [16] Alberto Del Bimbo. *Visual information retrieval*. Morgan Kaufmann, 1999.
- [17] Elena Botoeva, Diego Calvanese, and Mariano Rodriguez-Muro. Expressive approximations in *DL-Lite* ontologies. In Darina Dicheva and Danail Dochev, editors, *Artificial Intelligence: Methodology, Systems, and Applications, 14th International Conference, AIMS A 2010, Varna, Bulgaria, September 8-10, 2010. Proceedings*, volume 6304 of *Lecture Notes in Computer Science*, pages 21–31. Springer, 2010.
- [18] Sebastian Brandt, Ralf Küsters, and Anni-Yasmin Turhan. Approximation and difference in description logics. In *Proceedings of the Eight International Conference on Principles and Knowledge Representation and Reasoning (KR-02), Toulouse, France, April 22-25, 2002*, pages 203–214, 2002.
- [19] Andrea Cali, Georg Gottlob, and Thomas Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. In Jan Paredaens and Jianwen Su, editors, *Proceedings of the Twenty-Eighth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2009, June 19 - July 1, 2009, Providence, Rhode Island, USA*, pages 77–86. ACM, 2009.
- [20] Andrea Cali, Georg Gottlob, and Thomas Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. *J. Web Sem.*, 14:57–83, 2012.
- [21] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. DL-lite: Tractable description logics for ontologies. In *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA*, pages 602–607, 2005.

-
- [22] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The *dl-lite* family. *J. Autom. Reasoning*, 39(3):385–429, 2007.
- [23] Yahia Chabane, Laurent d’Orazio, Le Gruenwald, Baraa Mohamad, and Christophe Rey. Medical data management in the SYSEO project. *SIGMOD Record*, 42(3):48–53, 2013.
- [24] Yahia Chabane and Christophe Rey. Semantic gastroenterological images annotation and retrieval - reasoning with a polyp ontology. In *KEOD 2013 - Proceedings of the International Conference on Knowledge Engineering and Ontology Development, Vilamoura, Algarve, Portugal, 19-22 September, 2013*, pages 293–300, 2013.
- [25] Liang Chang, Uli Sattler, and Tianlong Gu. An abox revision algorithm for the description logic *el_bot*. In Meghyn Bienvenu, Magdalena Ortiz, Riccardo Rosati, and Mantas Simkus, editors, *Informal Proceedings of the 27th International Workshop on Description Logics, Vienna, Austria, July 17-20, 2014.*, volume 1193 of *CEUR Workshop Proceedings*, pages 459–470. CEUR-WS.org, 2014.
- [26] Shyi-Chyi Cheng, Tzu-Chuan Chou, Chao-Lung Yang, and Hung-Yi Chang. A semantic learning for content-based image retrieval using analytical hierarchy process. *Expert Syst. Appl.*, 28(3):495–505, 2005.
- [27] Simona Colucci, Tommaso Di Noia, Eugenio Di Sciascio, Francesco M. Donini, and Marina Mongiello. Concept abduction and contraction in description logics. In *Proceedings of the 2003 International Workshop on Description Logics (DL2003), Rome, Italy September 5-7, 2003*, 2003.
- [28] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Ze Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40(2), 2008.
- [29] Giuseppe De Giacomo, Maurizio Lenzerini, Antonella Poggi, and Riccardo Rosati. On the approximation of instance level update and erasure in description logics. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, pages 403–408. AAAI Press, 2007.

-
- [30] Jianfeng Du, Guilin Qi, Yi-Dong Shen, and Jeff Z. Pan. Towards practical abox abduction in large OWL DL ontologies. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco, California, USA, August 7-11, 2011*, 2011.
- [31] John P. Eakins. Towards intelligent image retrieval. *Pattern Recognition*, 35(1):3–14, 2002.
- [32] Andreas Ecke, Rafael Peñaloza, and Anni-Yasmin Turhan. Answering instance queries relaxed by concept similarity. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014*, 2014.
- [33] Ronald Fagin, Gabriel M. Kuper, Jeffrey D. Ullman, and Moshe Y. Vardi. Updating logical databases. *Advances in Computing Research*, 3:1–18, 1986.
- [34] Sibe Gao, Guilin Qi, and Haofen Wang. A new operator for abox revision in dl-lite. In Jörg Hoffmann and Bart Selman, editors, *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada*. AAAI Press, 2012.
- [35] Arnab Ghoshal, Pavel Ircing, and Sanjeev Khudanpur. Hidden markov models for automatic annotation and content-based retrieval of images and video. In *SIGIR 2005: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, August 15-19, 2005*, pages 544–551, 2005.
- [36] Georg Gottlob, Giorgio Orsi, and Andreas Pieris. Ontological queries: Rewriting and optimization. In *Proceedings of the 27th International Conference on Data Engineering, ICDE 2011, April 11-16, 2011, Hannover, Germany*, pages 2–13, 2011.
- [37] Oliver Gries. Generalized closed world reasoning in description logics with extended domain closure. In *Description Logics*, 2009.
- [38] Stephan Grimm. *Semantic Matchmaking with Nonmonotonic Description Logics*, volume 1 of *Studies on the Semantic Web*. IOS Press, 2009.

-
- [39] Stephan Grimm, Boris Motik, and Chris Preist. Matching semantic service descriptions with local closed-world reasoning. In *The Semantic Web: Research and Applications, 3rd European Semantic Web Conference, ESWC 2006, Budva, Montenegro, June 11-14, 2006, Proceedings*, pages 575–589, 2006.
- [40] Perry Groot, Heiner Stuckenschmidt, and Holger Wache. Approximating description logic classification for semantic web reasoning. In *The Semantic Web: Research and Applications, Second European Semantic Web Conference, ESWC 2005, Heraklion, Crete, Greece, May 29 - June 1, 2005, Proceedings*, pages 318–332, 2005.
- [41] Jun Gu, Paul W. Purdom, John Franco, and Benjamin W. Wah. Algorithms for the satisfiability (SAT) problem: A survey. In *Satisfiability Problem: Theory and Applications, Proceedings of a DIMACS Workshop, Piscataway, New Jersey, USA, March 11-13, 1996*, pages 19–152, 1996.
- [42] Mohand-Said Hacid, Alain Léger, Christophe Rey, and Farouk Toumani. Dynamic discovery of e-services. In *18èmes Journées Bases de Données Avancées, BDA '02, 21-25 octobre 2002, Evry, Actes (Informal Proceedings)*., 2002.
- [43] Ken Halland, Arina Britz, and Szymon Klarman. Tbox abduction in ALC using a DL tableau. In *Informal Proceedings of the 27th International Workshop on Description Logics, Vienna, Austria, July 17-20, 2014.*, pages 556–566, 2014.
- [44] Allan Hanbury. A survey of methods for image annotation. *J. Vis. Lang. Comput.*, 19(5):617–627, 2008.
- [45] Jeffrey Heer, Stuart K. Card, and James A. Landay. prefuse: a toolkit for interactive information visualization. In *Proceedings of the 2005 Conference on Human Factors in Computing Systems, CHI 2005, Portland, Oregon, USA, April 2-7, 2005*, pages 421–430, 2005.
- [46] Ian Horrocks, Lei Li, Daniele Turi, and Sean Bechhofer. The instance store: DL reasoning with large numbers of individuals. In *Proceedings of the 2004 International Workshop on Description Logics (DL2004), Whistler, British Columbia, Canada, June 6-8, 2004*, 2004.
- [47] Ian Horrocks, Ulrike Sattler, and Stephan Tobies. Reasoning with individuals for the description logic SHIQ. In *Automated Deduction - CADE-17, 17th International*

- Conference on Automated Deduction, Pittsburgh, PA, USA, June 17-20, 2000, Proceedings*, pages 482–496, 2000.
- [48] Bo Hu, Srinandan Dasmahapatra, David Dupplaw, Paul H. Lewis, and Nigel Shadbolt. Reflections on a medical ontology. *International Journal of Man-Machine Studies*, 65(7):569–582, 2007.
- [49] Bo Hu, Srinandan Dasmahapatra, Paul H. Lewis, and Nigel Shadbolt. Ontology-based medical image annotation with description logics. In *15th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2003), 3-5 November 2003, Sacramento, California, USA*, page 77, 2003.
- [50] Ullrich Hustadt, Boris Motik, and Ulrike Sattler. Reducing shiq-description logic to disjunctive datalog programs. In Didier Dubois, Christopher A. Welty, and Mary-Anne Williams, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR2004), Whistler, Canada, June 2-5, 2004*, pages 152–162. AAAI Press, 2004.
- [51] Ullrich Hustadt, Boris Motik, and Ulrike Sattler. Reasoning in description logics by a reduction to disjunctive datalog. *J. Autom. Reasoning*, 39(3):351–384, 2007.
- [52] Mohammed Lamine Kherfi, Djemel Ziou, and Alan Bernardi. Image retrieval from the world wide web: Issues, techniques, and systems. *ACM Comput. Surv.*, 36(1):35–67, 2004.
- [53] Szymon Klarman, Ulle Endriss, and Stefan Schlobach. Abox abduction in the description logic *ALC*. *J. Autom. Reasoning*, 46(1):43–80, 2011.
- [54] Matthias Knorr, José Júlio Alferes, and Pascal Hitzler. Local closed world reasoning with description logics under the well-founded semantics. *Artif. Intell.*, 175(9-10):1528–1554, 2011.
- [55] S Kudo, S Hirota, T Nakajima, S Hosobe, H Kusaka, T Kobayashi, M Himori, and A Yagyuu. Colorectal tumours and pit pattern. *J Clin Pathol*, 47(10):880–5, 1994.
- [56] Maurizio Lenzerini and Domenico Fabio Savo. On the evolution of the instance level of dl-lite knowledge bases. In Riccardo Rosati, Sebastian Rudolph, and Michael

- Zakharyashev, editors, *Proceedings of the 24th International Workshop on Description Logics (DL 2011), Barcelona, Spain, July 13-16, 2011*, volume 745 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2011.
- [57] Hongkai Liu, Carsten Lutz, Maja Milicic, and Frank Wolter. Updating description logic aboxes. In Patrick Doherty, John Mylopoulos, and Christopher A. Welty, editors, *Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning, Lake District of the United Kingdom, June 2-5, 2006*, pages 46–56. AAAI Press, 2006.
- [58] Ying Liu, Dengsheng Zhang, Guojun Lu, and Wei-Ying Ma. A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, 40(1):262–282, 2007.
- [59] Jiebo Luo, Andreas E. Savakis, and Amit Singhal. A bayesian network-based framework for semantic image understanding. *Pattern Recognition*, 38(6):919–934, 2005.
- [60] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.
- [61] Drew V. McDermott and Jon Doyle. Non-monotonic logic I. *Artif. Intell.*, 13(1-2):41–72, 1980.
- [62] Carlo Meghini, Fabrizio Sebastiani, and Umberto Straccia. A model of multimedia information retrieval. *J. ACM*, 48(5):909–970, 2001.
- [63] Jack Minker. On indefinite databases and the closed world assumption. In *6th Conference on Automated Deduction, New York, USA, June 7-9, 1982, Proceedings*, pages 292–308, 1982.
- [64] Ravi Bhushan Mishra and Sandeep Kumar. Semantic web reasoners and languages. *Artif. Intell. Rev.*, 35(4):339–368, 2011.
- [65] Boris Motik and Ulrike Sattler. A comparison of reasoning techniques for querying large description logic aboxes. In Miki Hermann and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning, 13th International Conference, LPAR 2006, Phnom Penh, Cambodia, November 13-17, 2006, Proceedings*, volume 4246 of *Lecture Notes in Computer Science*, pages 227–241. Springer, 2006.

-
- [66] Boris Motik, Rob Shearer, and Ian Horrocks. Hypertableau reasoning for description logics. *J. Artif. Intell. Res. (JAIR)*, 36:165–228, 2009.
- [67] Henning Müller, Paul D. Clough, William R. Hersh, Thomas Deselaers, Thomas Martin Lehmann, and Antoine Geissbühler. Evaluation axes for medical image retrieval systems: the imageclef experience. In *Proceedings of the 13th ACM International Conference on Multimedia, Singapore, November 6-11, 2005*, pages 1014–1022, 2005.
- [68] Henning Müller, Nicolas Michoux, David Bandon, and Antoine Geissbühler. A review of content-based image retrieval systems in medical applications - clinical benefits and future directions. *I. J. Medical Informatics*, 73(1):1–23, 2004.
- [69] Jasmin Opitz, Bijan Parsia, and Ulrike Sattler. Using ontologies for medical image retrieval - an experiment. In *Proceedings of the 5th International Workshop on OWL: Experiences and Directions (OWLED 2009), Chantilly, VA, United States, October 23-24, 2009*, 2009.
- [70] Jeff Z. Pan and Edward Thomas. Approximating OWL-DL ontologies. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, pages 1434–1439, 2007.
- [71] Jeff Z. Pan, Edward Thomas, and Yuting Zhao. Completeness guaranteed approximations for OWL-DL query answering. In *Proceedings of the 22nd International Workshop on Description Logics (DL 2009), Oxford, UK, July 27-30, 2009*, 2009.
- [72] Paris. The paris endoscopic classification of superficial neoplastic lesions: esophagus, stomach, and colon: November 30 to december 1, 2002. *Gastrointestinal Endoscopy*, 58(6 Suppl):3–43, 2003.
- [73] Gabriele Paul. Approaches to abductive reasoning: an overview. *Artif. Intell. Rev.*, 7(2):109–152, 1993.
- [74] Yuan Ren, Jeff Z. Pan, and Yuting Zhao. Soundness preserving approximation for tbox reasoning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*, 2010.
- [75] Yuan Ren, Jeff Z. Pan, and Yuting Zhao. Towards soundness preserving approximation for abox reasoning of OWL2. In *Proceedings of the 23rd International Workshop on Description Logics (DL 2010), Waterloo, Ontario, Canada, May 4-7, 2010*, 2010.

-
- [76] Schlemper RJ, Riddell RH, Kato Y, Borchard F, and al. The vienna classification of gastrointestinal epithelial neoplasia. *Gut.*, 47:251–255, 2000.
- [77] Yong Rui, Thomas S. Huang, and Shih-Fu Chang. Image retrieval: Current techniques, promising directions, and open issues. *J. Visual Communication and Image Representation*, 10(1):39–62, 1999.
- [78] Marco Schaerf and Marco Cadoli. Tractable reasoning via approximation. *Artif. Intell.*, 74(2):249–310, 1995.
- [79] Eugenio Di Sciascio, Francesco M. Donini, and Marina Mongiello. Structured knowledge representation for image retrieval. *J. Artif. Intell. Res. (JAIR)*, 16:209–257, 2002.
- [80] Amin Shah-Hosseini and Gerald M. Knapp. Semantic image retrieval based on probabilistic latent semantic analysis. In *Proceedings of the 14th ACM International Conference on Multimedia, Santa Barbara, CA, USA, October 23-27, 2006*, pages 703–706, 2006.
- [81] Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(12):1349–1380, 2000.
- [82] Munirathnam Srikanth, Joshua Varner, Mitchell Bowden, and Dan I. Moldovan. Exploiting ontologies for automatic image annotation. In *SIGIR 2005: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Salvador, Brazil, August 15-19, 2005*, pages 552–558, 2005.
- [83] Heiner Stuckenschmidt. Approximate subsumption for complex description logics. In *Proceedings of the 2007 International Workshop on Description Logics (DL2007), Brixen-Bressanone, near Bozen-Bolzano, Italy, 8-10 June, 2007*, 2007.
- [84] Heiner Stuckenschmidt. Partial matchmaking using approximate subsumption. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, pages 1459–1464, 2007.
- [85] Michele Trevisiol, Luca Chiarandini, Luca Maria Aiello, and Alejandro Jaimes. Image ranking based on user browsing behavior. In *The 35th International ACM SIGIR*

- conference on research and development in Information Retrieval, SIGIR '12, Portland, OR, USA, August 12-16, 2012*, pages 445–454, 2012.
- [86] Douglas N. Walton. *Abductive reasoning*. Univ. of Alabama Press, 2004.
- [87] Huan Wang, Song Liu, and Liang-Tien Chia. Does ontology help in image retrieval?: a comparison between keyword, text ontology and multi-modality ontology approaches. In *Proceedings of the 14th ACM International Conference on Multimedia, Santa Barbara, CA, USA, October 23-27, 2006*, pages 109–112, 2006.
- [88] Tao Wang, Yong Rui, Shi-Min Hu, and Jia-Guang Sun. Adaptive tree similarity learning for image retrieval. *Multimedia Syst.*, 9(2):131–143, 2003.
- [89] Marianne Winslett. A model-theoretic approach to updating logical databases. In Avi Silberschatz, editor, *Proceedings of the Fifth ACM SIGACT-SIGMOD Symposium on Principles of Database Systems, March 24-26, 1986, Cambridge, Massachusetts, USA*, pages 224–234. ACM, 1986.
- [90] Marianne Winslett. A framework for comparison of update semantics. In Chris Edmondson-Yurkanan and Mihalis Yannakakis, editors, *Proceedings of the Seventh ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, March 21-23, 1988, Austin, Texas, USA*, pages 315–324. ACM, 1988.
- [91] Bin Xu, Jiajun Bu, Chun Chen, Deng Cai, Xiaofei He, Wei Liu, and Jiebo Luo. Efficient manifold ranking for image retrieval. In *Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, Beijing, China, July 25-29, 2011*, pages 525–534, 2011.
- [92] Hao Xu, Jingdong Wang, Xian-Sheng Hua, and Shipeng Li. Image search by concept map. In *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010*, pages 275–282, 2010.
- [93] Ruofei Zhang and Zhongfei (Mark) Zhang. FAST: toward more effective and efficient image retrieval. *Multimedia Syst.*, 10(6):529–543, 2005.
- [94] Evgeny Zolin. Description logic complexity navigator, 2011. <http://www.cs.man.ac.uk/~ezolin/dl/>.

A.1 Proofs

A.1.1 Proof of Theorem 5.2.2

For $\exists \mathcal{W}$ such that $a \in \text{InsR}_{\mathcal{W}}^+(C, \mathcal{KB}) \Rightarrow \mathcal{A} \cup \{C(a)\}$ is consistent w.r.t. \mathcal{T} :

By definition of the \mathcal{W}^+ Relaxation, if $a \in \text{InsR}_{\mathcal{W}}^+(C, \mathcal{KB})$ then :

- $\exists \mathcal{A}''$, $\mathcal{A}^{W+} = \mathcal{A} \cup \mathcal{A}''$.
- $\mathcal{KB}^{W+} = (\mathcal{T}, \mathcal{A}^{W+}) \models C(a)$.
- \mathcal{KB}^{W+} is consistent.

$\mathcal{A}^{W+} \cup C(a)$ is consistent w.r.t. $\mathcal{T} \Rightarrow \mathcal{A} \cup \mathcal{A}'' \cup C(a)$ is consistent w.r.t. $\mathcal{T} \Rightarrow \mathcal{A} \cup C(a)$ is consistent w.r.t. \mathcal{T} .

Therefore : $\exists \mathcal{W}$ such as $a \in \text{InsR}_{\mathcal{W}}^+(C, \mathcal{KB}) \Rightarrow \mathcal{A} \cup \{C(a)\}$ is consistent w.r.t. \mathcal{T} .

For $\mathcal{A} \cup \{C(a)\}$ is consistent w.r.t. $\mathcal{T} \Rightarrow \exists \mathcal{W}$ such that $a \in \text{InsR}_{\mathcal{W}}^+(C, \mathcal{KB})$

Let \mathcal{L} be the language supporting the concepts constructors of Table 5.1 and let C be a \mathcal{L} -concept definition and an individual a . We can transform the assertion $C(a)$ into an equivalent set of assertions on atomic concepts, on negation of atomic concepts and on atomic roles. This set may not be unique.

The transformation of the assertion $C(a)$ is a finished set, let S be one of the decomposition sets :

- S contains only assertions on atomic concepts, on negation of atomic concepts and on atomic roles.
- $\mathcal{A} \cup S$ is consistent w.r.t. \mathcal{T} .
- $(\mathcal{T}, \mathcal{A} \cup S) \models C(a)$.

There exists then at least one set that satisfies the constraints of the \mathcal{W}^+ Relaxation. Therefore : $\mathcal{A} \cup \{C(a)\}$ is consistent w.r.t. $\mathcal{T} \Rightarrow \exists \mathcal{W}$ such as $a \in \text{InsR}_{\mathcal{W}}^+(C, \mathcal{KB})$

A.1.2 Proof of Theorem 5.2.4

For $\exists \mathcal{W}$ such as $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB}) \Rightarrow (\mathcal{A} \setminus \mathcal{A}') \cup \{C(a)\}$ is consistent w.r.t. \mathcal{T}

By definition of the \mathcal{W} Relaxation, if $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB})$ then :

- $\exists \mathcal{A}'', \mathcal{A}^{\mathcal{W}} = (\mathcal{A} \setminus \mathcal{A}'') \cup (\mathcal{A}'' \setminus \mathcal{A})$.
- $\mathcal{KB}^{\mathcal{W}} = (\mathcal{T}, \mathcal{A}^{\mathcal{W}}) \models C(a)$.
- $\mathcal{KB}^{\mathcal{W}}$ is consistent.

$\mathcal{A}^{\mathcal{W}} \cup C(a)$ is consistent w.r.t. $\mathcal{T} \Rightarrow (\mathcal{A} \setminus \mathcal{A}'') \cup (\mathcal{A}'' \setminus \mathcal{A}) \cup C(a)$ is consistent w.r.t. \mathcal{T}
 $\Rightarrow (\mathcal{A} \setminus \mathcal{A}'') \cup C(a)$ is consistent w.r.t. \mathcal{T}
 $\Rightarrow (\mathcal{A} \setminus \mathcal{A}') \cup C(a)$ is consistent w.r.t. \mathcal{T} (\mathcal{A}' is the set of assertions on atomic concepts, on negation of atomic concepts and on atomic roles in \mathcal{A} such that $\mathcal{A}'' \subset \mathcal{A}'$)
 Therefore : $\exists \mathcal{W}$ such as $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB}) \Rightarrow (\mathcal{A} \setminus \mathcal{A}') \cup \{C(a)\}$ is consistent w.r.t. \mathcal{T} .

For $(\mathcal{A} \setminus \mathcal{A}') \cup \{C(a)\}$ is consistent w.r.t. $\mathcal{T} \Rightarrow \exists \mathcal{W}$ such as $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB})$

The transformation of the assertion $C(a)$ is a finished set, let S be one of the decomposition sets :

- $S \cup \mathcal{A}'$ contains only assertions on atomic concepts, on negation of atomic concepts and on atomic roles.
- $(\mathcal{A} \setminus \mathcal{A}') \cup S$ is consistent w.r.t. \mathcal{T} .
- $(\mathcal{T}, (\mathcal{A} \setminus \mathcal{A}') \cup S) \models C(a)$.

There exists then at least one set that satisfies the constraints of the \mathcal{W} Relaxation. Therefore : $(\mathcal{A} \setminus \mathcal{A}') \cup \{C(a)\}$ is consistent w.r.t. $\mathcal{T} \Rightarrow \exists \mathcal{W}$ such as $a \in \text{InsR}_{\mathcal{W}}(C, \mathcal{KB})$.

A.1.3 Proof of Lemma 5.3.2

For $\mathcal{K} \models C(a) \Rightarrow \exists S \in \text{MAAS}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$

At the beginning, $\text{MAAS}(\mathcal{K}, C, a) = \{\{C(a)\}\}$ therefore $\mathcal{K} \models C(a) \Rightarrow \exists S \in$

$MAAS(\mathcal{K}, C, a) \mid \mathcal{K} \models S$

Now, we must prove that this property is valid for each rule and the application of rules is finite.

Let $MAAS_n(\mathcal{K}, C, a)$ and $MAAS_{n+1}(\mathcal{K}, C, a)$ two steps of the construction of minimal atomic assertions sets of $C(a)$, such that $MAAS_{n+1}(\mathcal{K}, C, a)$ is the result of rule application on $MAAS_n(\mathcal{K}, C, a)$.

We assume that $\exists S \in MAAS_n(\mathcal{K}, C, a) \mid \mathcal{K} \models S$ and we prove that $\exists S \in MAAS_{n+1}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$ for all applied rule.

\rightarrow_{\sqcap} rule:

$MAAS_n(\mathcal{K}, C, a) = \{S_1, \dots, S_n\}$ and $MAAS_{n+1}(\mathcal{K}, C, a) = \{S_1, \dots, S_n \cup \{D_1(x), D_2(x)\} \setminus \{(D_1 \sqcap D_2)(x)\}\}$

Suppose that $\nexists S \in MAAS_{n+1}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$, that is to say $\mathcal{K} \models S_n$ and $\mathcal{K} \not\models S_n \cup \{D_1(x), D_2(x)\} \setminus \{(D_1 \sqcap D_2)(x)\}$

So the assumption is false, therefore $\exists S \in MAAS_n(\mathcal{K}, C, a) \mid \mathcal{K} \models S \Rightarrow \exists S \in MAAS_{n+1}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$

\rightarrow_{\sqcup} rule:

$MAAS_n(\mathcal{K}, C, a) = \{S_1, \dots, S_n\}$ and $MAAS_{n+1}(\mathcal{K}, C, a) = \{S_1, \dots, S_n \cup \{D_1(x)\} \setminus \{(D_1 \sqcup D_2)(x)\}, S_n \cup \{D_2(x)\} \setminus \{(D_1 \sqcup D_2)(x)\}\}$

Suppose that $\nexists S \in MAAS_{n+1}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$, that is to say $\mathcal{K} \models S_n$ and $\mathcal{K} \not\models S_n \cup \{D_1(x)\} \setminus \{(D_1 \sqcup D_2)(x)\} \cup S_n \cup \{D_2(x)\} \setminus \{(D_1 \sqcup D_2)(x)\}$

So the assumption is false, therefore $\exists S \in MAAS_n(\mathcal{K}, C, a) \mid \mathcal{K} \models S \Rightarrow \exists S \in MAAS_{n+1}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$

\rightarrow_{\exists} rule:

$MAAS_n(\mathcal{K}, C, a) = \{S_1, \dots, S_n\}$ and $\forall m \geq 0, y_m \in ind(\mathcal{A}), \forall z \notin ind(\mathcal{A}), MAAS_{n+1}(\mathcal{K}, C, a) = \{S_1, \dots, S_n \cup \{R(x, z), D(z)\} \setminus \{(\exists R.D)(x)\}, S_n \cup \{R(x, y_1), D(y_1)\} \setminus \{(\exists R.D)(x)\}, \dots, S_n \cup \{R(x, y_m), D(y_m)\} \setminus \{(\exists R.D)(x)\}\}$

$MAAS_n(\mathcal{K}, C, a) \subset MAAS_{n+1}(\mathcal{K}, C, a)$, therefore $\exists S \in MAAS_n(\mathcal{K}, C, a) \mid \mathcal{K} \models S \Rightarrow \exists S \in MAAS_{n+1}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$

\rightarrow_{\models} rule:

$MAAS_n(\mathcal{K}, C, a) = \{S_1, \dots, S_n\}$ and $MAAS_{n+1}(\mathcal{K}, C, a) = \{S_1, \dots, S_n, S_n \cup \{\neg D(x)\} \setminus \{E(x)\}\}$

Suppose that $\nexists S \in MAAS_{n+1}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$, that is to say $\mathcal{K} \models S_n$ and $\mathcal{K} \not\models S_n \cup \{R(x, z), D(z)\} \setminus \{(\exists R.D)(x)\} \cup_{\forall m \geq 0} \{R(x, y_m), D(y_m)\} \setminus \{(\exists R.D)(x)\}$

So the assumption is false, therefore $\exists S \in MAAS_n(\mathcal{K}, C, a) \mid \mathcal{K} \models S \Rightarrow \exists S \in MAAS_{n+1}(\mathcal{K}, C, a) \mid \mathcal{K} \models S$

Note that, \rightarrow_{\exists} rule is the only rule that can add new individuals. The total number of new individuals is finite because it is bounded by the number of existential restrictions of C and the number of individuals of the ABox \mathcal{A} .

The decomposition rules are applied to an assertion $C(a)$ and the results are always assertions on a sub-descriptions of C . The number of decomposition rule applications per individual is finite. For the inference rules also, the number of applications per individual is finite because an inference rule cannot be applied a second time to an individual if it has been applied to the individual on an ascendant node.

The number of individuals is finite and the number of rule applications per individual is finite also so the algorithm always compute $MAAS(\mathcal{K}, C, a)$ in a finite time.

For $\exists S \in MAAS(\mathcal{K}, C, a) \mid \mathcal{K} \models S \Rightarrow \mathcal{K} \models C(a)$

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A} \cup MAAS(\mathcal{K}, C, a))$ a knowledge on which no rule is applicable and let C, a respectively the concept and the individual of the problem.

We must prove that $\mathcal{KB} \models C(a)$

We define a canonical interpretation $\mathcal{I}_{\mathcal{A}}$ such that:

1. $\Delta^{\mathcal{I}_{\mathcal{A}}}$ of $\mathcal{I}_{\mathcal{A}}$ consists of all the individual names occurring in \mathcal{A} .
2. $\forall C, C^{\mathcal{I}_{\mathcal{A}}} = \{x \mid C(x) \in \mathcal{A}\}$
3. $\forall R, R^{\mathcal{I}_{\mathcal{A}}} = \{(x, y) \mid r(x, y) \in \mathcal{A}\}$

\mathcal{A} is consistent then $\mathcal{KB} \models C(a)$

A.2 Other problems definition

Problem A.2.1 (Exist $\mathcal{W}^+(C, \mathcal{KB})$) : Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem Exist $\mathcal{W}^+(C, \mathcal{KB})$ is the problem of deciding whether $InsR_{\mathcal{W}}^+(C, \mathcal{KB})$ is not empty.

Problem A.2.2 ($\text{Exist}\mathcal{W}^-(C, \mathcal{KB})$) : Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{Exist}\mathcal{W}^-(C, \mathcal{KB})$ is the problem of deciding whether $\text{InsR}_{\mathcal{W}}^-(C, \mathcal{KB}, \text{GCWA})$ is not empty.

Problem A.2.3 ($\text{Exist}\mathcal{W}(C, \mathcal{KB})$) : Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{Exist}\mathcal{W}(C, \mathcal{KB})$ is the problem of deciding whether $\text{InsR}_{\mathcal{W}}(C, \mathcal{KB})$ is not empty.

Problem A.2.4 ($\text{NoEmpty}\mathcal{W}^+(C, \mathcal{KB})$) Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{NoEmpty}\mathcal{W}^+(C, \mathcal{KB})$ is the problem of deciding whether there exists an integer \mathcal{W} such that $\text{RxdIR}_{\mathcal{W}}^+(C, \mathcal{KB})$ is not empty.

Problem A.2.5 ($\text{NoEmpty}\mathcal{W}^-(C, \mathcal{KB})$) Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{NoEmpty}\mathcal{W}^-(C, \mathcal{KB})$ is the problem of deciding whether there exists an integer \mathcal{W} such that $\text{RxdIR}_{\mathcal{W}}^-(C, \mathcal{KB}, \text{GCWA})$ is not empty.

Problem A.2.6 ($\text{NoEmpty}\mathcal{W}(C, \mathcal{KB})$) Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{NoEmpty}\mathcal{W}(C, \mathcal{KB})$ is the problem of deciding whether there exists an integer \mathcal{W} such that $\text{RxdIR}_{\mathcal{W}}(C, \mathcal{KB})$ is not empty.

A.2.1 Optimisation problems

Problem A.2.7 ($\text{RlxNoEmpty}\mathcal{W}^+(C, \mathcal{KB})$) : Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{RlxNoEmpty}\mathcal{W}^+(C, \mathcal{KB})$ is the problem of computing minimal \mathcal{W} for that the relaxation instances of the concept C by adding assertions $\text{RxdIR}_{\mathcal{W}}^+(C, \mathcal{KB})$ is not empty.

Problem A.2.8 ($\text{RlxNoEmpty}\mathcal{W}^-(C, \mathcal{KB})$) : Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{RlxNoEmpty}\mathcal{W}^-(C, \mathcal{KB})$ is the problem of computing minimal \mathcal{W} for that the relaxation instances of the concept C by removing assertions $\text{RxdIR}_{\mathcal{W}}^+(C, \mathcal{KB}, \text{GCWA})$ is not empty.

Problem A.2.9 ($\text{RlxNoEmpty}\mathcal{W}(C, \mathcal{KB})$) : Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a knowledge base and let C be a concept and \mathcal{W} be an integer. The problem $\text{RlxNoEmpty}\mathcal{W}(C, \mathcal{KB})$ is the problem of computing minimal \mathcal{W} for that the relaxation instances of the concept C by adding and removing assertions $\text{RxdIR}_{\mathcal{W}}^+(C, \mathcal{KB})$ is not empty.