

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique

Université Saad Dahlab Blida

N° D'ordre :



Faculté des sciences

Département d'informatique

Mémoire Présenté par :

ECHIKR Mohamed Amine

BELALIA Mouad

En vue d'obtenir le diplôme de master

Domaine : mathématique et informatique

Filière : Informatique

Spécialité : Informatique

Option : Ingénierie de logiciel

Thème : L'inférence probabiliste sur les ontologies floues

Soutenu le :

M : *Jadial. L*

Président

M : *cheurfa*

Examineur

Mme : Fareh.M

Promotrice

Mr : Riali.I

Encadreur

MA-004-505-1

Promotion 2016 / 2017

Dédicaces

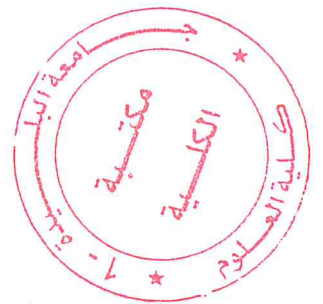
A :

Nos chers parents

Nos frères et sœurs

Toute notre famille

Tous nos amis



On dédie ce mémoire

*BELALIA Mouad
ECHIKR Mohamed Amine*

Remerciement

On remercie en premier dieu le tout puissant de nous avoir donné la chance, le courage et les moyens pour l'accomplissement de ce projet.

On remercie tout particulièrement Mme FAREH.M enseignante à l'université USDB, pour nous avoir aidés et épaulé durant toute cette année, de nous avoir guidé et soutenu dans la réalisation du projet, merci.

On remercie le doctorant Mr RIALI Ishak pour nous avoir aidé et suivi durant la réalisation de ce projet, merci.

On remercie tous les membres du jury d'avoir accepté d'examiner notre travail.

Au terme de ce mémoire, On tient à remercier toute personne ayant contribué à l'accomplissement de ce travail de près ou de loin et qui nous ont été bénéfique durant notre parcours pour la réalisation de ce travail.

Finalement, On remercie l'ensemble des enseignants du département d'informatique pour les efforts qu'ils fournissent pour notre réussite dans nos études.

BELALIA Mouad

ECHIKR Mohamed Amine

Résumé:

Pour une meilleure représentation de la connaissance vague et/ou imprécise plusieurs extensions des ontologies classiques ont été développées par l'intégration de la logique floue dans les ontologies classique. Parmi ces travaux l'OWL 2.0 floue (Fuzzy OWL) est une extension d'OWL 2.0, il a été développé pour permettre aux ontologies classiques de supporter et modéliser l'imprécision et la connaissance floue. De plus, plusieurs raisonneurs ont été développés pour faire l'inférence sur les ontologies floues telles que Fire [1], DeLorean [2], et le plus populaire FuzzyDL [3], cependant, ces derniers ne sont pas capable de faire l'inférence probabiliste sur les ontologies floue. Pour cette raison, l'objectif de ce travail est d'intégrer l'inférence probabiliste dans les ontologies floues.

Mots-clés : web sémantique, connaissance incertaine, logique floue, réseaux bayésiens floue, ontologies.

Abstract:

For a better representation of vague and/or imprecise knowledge several extensions of the classical ontology have been developed by integrating fuzzy logic into the classical ontology. Among these works the Fuzzy owl 2.0 is an extension of Owl 2.0, it has been developed to allow classical ontology to support and model vagueness and fuzzy knowledge. In addition, several researchers have been developed to make inferences about fuzzy ontology such as [1], [2], and the most popular [3], however, they are not able to make the probabilistic inference on fuzzy ontology. For this reason, the objective of this work is to integrate probabilistic inference into fuzzy ontology.

Key words: Semantic Web, uncertain knowledge, fuzzy logic, fuzzy Bayesian networks, ontology.

ملخص:

من أجل تمثيل أفضل للمعارف المبهمة و/أو غير الدقيقة ، تم تطوير عدة امتدادات للانطولوجيا من خلال دمج منطق

الغموض في الانطولوجيا الكلاسيكية (التقليدية) . من بين هذه الأعمال OWL2 الغامض هو امتداد ل OWL2 التقليدي ، وقد وضعت للسماح للانطولوجيا الكلاسيكية بدعم و نمذجة الغموض وعدم وضوح المعرفة .إضافة إلى ذلك ، تم تطوير عديد البرامج من المفكرين لإدخال الاستدلال حول الانطولوجيا الغامضة مثل [1] ، [2] ، والأكثر شعبية [3] ، ومع ذلك فإن هذه الأخيرة ليست قادرة على إدخال الاستدلال الاحتمالي في الانطولوجيا الغامضة. ولهذا السبب ، فإن الهدف من هذا العمل هو إدخال الاستدلال الاحتمالي على الانطولوجيا الغامضة.

الكلمات الرئيسية : الدلالة على شبكة الإنترنت ، المعرفة غير المؤكدة ، منطق الغموض ، شبكات بايز الغامضة، الانطولوجيا.

Sommaire :

1. Introduction générale	1
1.1. Contexte du travail	1
1.2. Problématique.....	2
1.3. Objectif.....	2

Chapitre 1 : Les ontologies formelles et les ontologies floues

1. Introduction	5
2. Les ontologies formelles	5
2.1. Définition	5
2.2. Les objectifs de l'ontologie	6
2.3. Type d'ontologie.....	6
2.4. Les applications d'ontologies	7
2.5. Construction d'une ontologie.....	7
3. L'incertitude dans le web sémantique.....	9
3.1. L'intervalle de vérité	10
3.2. Définition des concepts vagues	10
3.3. Fonctions d'appartenance pour les concepts vagues	11
4. Les ontologies floues.....	13
4.1. Définition	13
4.2. Les composants d'ontologies floues.....	14
4.3. Langage d'ontologie floue	15
4.4. L'outil de manipulation les ontologies	15
4.5. Représentation d'ontologies floues dans OWL 2.....	15
5. Conclusion.....	16

Chapitre 2 : Les réseaux bayésiens

1. Introduction	18
2. Réseaux bayésiens standard	18
2.1. Définition	18
2.2. Apprentissage dans les réseaux bayésiens.....	19

2.3. L'inférence dans les réseaux bayésiens	20
3. Réseaux bayésiens flous.....	21
3.1. Définition	22
3.2. Équation bayésienne floue.....	22
4. Les avantages des réseaux bayésiens	23
5. L'ontologie probabiliste	23
6. Intégration d'un réseau bayésien dans une ontologie	24
7. Conclusion.....	25

Chapitre 3 : Conception du système

1. Introduction	27
2. Schéma global de notre système	27
2.1. Ontologie floue.....	30
2.2. Extraction des composants de l'ontologie floue.....	31
2.3. Construction de réseau bayésien flou (structure et paramètres)	31
2.4. Méta-ontologie de RB-flou	33
2.5. Intégration de RB-flou dans l'ontologie floue.....	33
2.6. Ontologie floue et probabiliste	35
2.7. Requêtes	37
3. Conclusion.....	39

Chapitre 4 : Implémentation du système

1. Introduction	41
2. Outils de développement de notre système	41
3. La mise en œuvre du système	42
3.1. L'interface d'entrée	42
3.2. Interface chargement ontologie	43
3.3. Interface extraction des composantes de l'ontologie floue	44
3.4. Interface création du réseau bayésien floue.....	45
3.5. Apprentissage des paramètres et de la structure de RBF.....	46
4. Test.....	49

5. Conclusion.....	49
--------------------	----

Chapitre 5 : Test du système

1. Introduction	51
2. La création de l'ontologie floue	51
2.1. Motivation de l'étude de cas	51
2.2. Ontologie cardiovasculaire.....	51
2.3. La cardiovasculaires et les facteurs des risques	52
2.4. Les différents composants de l'ontologie floue	53
2.5. Les annotations de l'ontologie floue	54
3. Lancement du test	55
3.1. Chargement de l'ontologie flou cardiovasculaire	55
3.2. L'extraction des composants	56
3.3. Créer le réseau bayésien flou cardiovasculaire	57
4. Lancement des requêtes	61
5. Résultat de la création de l'ontologie probabiliste floue.....	61
5.1. Comparaison entre l'ontologie floue et l'ontologie probabiliste floue obtenue.....	62
6. Conclusion.....	66
Conclusion générale	68
Références bibliographiques	71

Liste des figures :

Figure 1 : Construction d'une ontologie opérationnelle [13].	8
Figure 2 : Les fonctions d'appartenances [15].	11
Figure 3 : Description sémantique de température [15].	12
Figure 4 : Fonction d'appartenance pour les termes du variable [15].	12
Figure 5 : Les pairs de la fonction d'appartenance $(x, \mu(x))$ [15].	12
Figure 6 : Exemple d'annotation fuzzOwl2.	16
Figure 7 : Exemple d'un réseau bayésien simple [27].	19
Figure 8 : Système global.	29
Figure 9 : FuzzyOwl2 annotations exemple.	30
Figure 10 : L'ontologie floue "Cardiovascular disease risk factors".	30
Figure 11 : Nœuds de réseau bayésien flou.	32
Figure 12 : Diagramme de classes de la méta-ontologie.	33
Figure 13 : Exemple des instances dans la méta-ontologie.	35
Figure 14 : Les classes méta-ontologie.	36
Figure 15 : Les classes d'ontologie "Cardiovascular disease risk factors".	36
Figure 16 : Les classes de la nouvelle ontologie.	36
Figure 17 : Exemple d'un réseau bayésien flou.	37
Figure 18 : Exemple d'état observé pour un nœud précis.	38
Figure 19 : Exemple d'inférence pour un nœud flou.	38
Figure 20 : Interface d'accueil.	42
Figure 21 : Interface chargement ontologie.	43
Figure 22 : Interface extraction des composantes de l'ontologie floue.	44
Figure 23 : Interface titre de réseau bayésien flou.	45
Figure 24 : Interface création des nœuds avec ses différents états.	46
Figure 25 : Interface création les relations entre les nœuds.	47
Figure 26 : Interface apprentissage des paramètres et structure.	48
Figure 27 : Interface de test.	49
Figure 28 : Vision générale sur l'ontologie floue cardiovasculaire.	52
Figure 29 : L'annotation de l'ontologie floue cardiovasculaire.	54
Figure 30 : Annotation d'un type de données avec sa fonction d'appartenance.	55
Figure 31 : Chargement de l'ontologie floue cardiovasculaire.	56
Figure 32 : Extraction des propriétés de données précises pour la classe patient.	57
Figure 33 : Donner un titre pour le réseau bayésien.	57
Figure 34 : La création des nœuds.	58
Figure 35 : La création des relations.	59

Figure 36 : Table de probabilité de Blood Pressure.	60
Figure 37 : Réseau bayésien flou cardiovasculaire.	60
Figure 38 : Lancement des requêtes.	61
Figure 39 : Exemple des différents composants de l'ontologie floue cardiovasculaire	62
Figure 40 : La hiérarchie des classes de l'ontologie floue cardiovasculaire.	62
Figure 41 : La hiérarchie des classes de l'ontologie floue probabiliste cardiovasculaire.	63
Figure 42 : La hiérarchie des propriétés d'objet de l'ontologie floue cardiovasculaire.	63
Figure 43 : La hiérarchie des propriétés d'objet de l'ontologie floue probabiliste cardiovasculaire.	64
Figure 44 : La hiérarchie des propriétés de donnée de l'ontologie floue cardiovasculaire.	64
Figure 45 : La hiérarchie des propriétés d'objet de l'ontologie floue probabiliste cardiovasculaire.	65
Figure 46 : Le réseau bayésien flou cardiovasculaire après le lancement du requête.	65

Liste des tableaux :

Tableau 1 : Les différents composants de méta-ontologie.	34
Tableau 2 : Les différents composants d'ontologie floue cardiovasculaire.	53
Tableau 3 : Les différents types de donnée de l'ontologie floue cardiovasculaire.	54
Tableau 4 : Les noeuds et ses états relatifs.	58

1. Introduction générale

1.1. Contexte du travail

"Le Web Sémantique", ce sont des termes qui reviennent de plus en plus souvent dans les propos des chercheurs spécialisés dans le Web en général et dans la recherche d'informations en particulier. Il est censé combler les lacunes du Web actuel en ajoutant une signification aux informations.

Le Web Sémantique est un Web sur lequel les chercheurs portent beaucoup d'espairs. Que ce soit dans le domaine de la recherche d'informations, Compétitive Intelligence, etc., il aura pour mission de donner une signification aux données et permettre aux machines d'analyser et de comprendre les informations qui circulent dedans.

L'une des caractéristiques de l'intelligence humaine est la capacité de raisonner avec une connaissance incertaine. Très souvent, il est impossible ou difficile d'obtenir une connaissance certaine. Dans la vie réelle, on se trouve fréquemment dans des situations de prise de décisions dans lesquelles on ne dispose que d'informations incertaines. On outre les ontologies ont été largement utilisées pour la modélisation et la représentation des connaissances dans divers domaines. Où une ontologie est une spécification formelle d'une conceptualisation partagée [4], cependant, les ontologies ne peuvent pas modéliser et représenter formellement la connaissance incertaine qui dut de l'incomplétude, l'aléatoire, l'imprécision...etc. Ceci est un inconvénient majeur pour les applications du Web sémantique, qui sont souvent nécessaires pour effectuer le raisonnement automatisé dans des environnements incertains et complexes. Pour cette raison une meilleure représentation et raisonnement avec l'incertitude dans les ontologies est devenu un sujet d'actualité. De plus, plusieurs chercheurs ont essayé de remédier aux défauts de l'ontologie classique par la proposition des extensions pour faire face à l'incertitude.

Cependant, la majorité des domaines qu'on cherche souvent à modéliser comprendre plusieurs types d'incertitudes à la fois, pour faire face à ce problème il paraît nécessaire de développer des solutions hybrides puissantes et capables de prendre en considération la richesse des connaissances en matière d'incertitude telle que le résultat obtenu peut être utilisée par la suite dans des applications d'aide à la décision qui doivent être capable de raisonner avec des connaissances incertaines.

1.2. Problématique

Une ontologie est une représentation qui regroupe un ensemble de concepts et relations décrivant un domaine particulier. Un de principaux défauts de l'ontologie est leur incapacité de représenter et raisonner sur l'incertitude. Dans les années précédentes, divers chercheurs Nottelmann [5], Costa [6], Heinsohn [7], Pool [8], etc. ont tenté de proposer des approches visant à intégrer l'incertitude dans les ontologies, qui est un axe de recherche intéressant. L'une des caractéristiques de réseau bayésien RB est leur capacité de représenter les informations incertaines sous la forme d'un modèle probabiliste. De plus, les réseaux bayésiens flous ont été proposés comme une solution hybride par l'intégration de l'imprécision dans les réseaux bayésiens standards pour bénéficier d'avantages de chaque formalisme. Mais, malheureusement cette représentation n'est pas formelle et les machines ne sont pas capables de la comprendre.

Et pour créer une ontologie floue qui supporte l'inférence probabiliste, on doit construire la structure de cette ontologie floue ainsi que ses tables de probabilité.

Donc :

- Comment fournir des mécanismes permettant de représenter les informations incertaines et d'inférer des connaissances implicites à partir des connaissances explicites stockées dans une ontologie floue en utilisant les réseaux bayésiens flous ?
- Comment représenter le réseau bayésien résultant d'une manière formelle?

1.3. Objectif

Pour répondre aux questions posées dans la problématique, l'objectif du système proposé dans ce projet est de prendre en considération plusieurs types d'incertitudes et /ou d'imperfections à la fois liées aux informations dans le contexte du web sémantique (des connaissances probabilistes et floues à la fois). Pour ce faire, nous allons basées sur les réseaux bayésiens flous RBF et les ontologies floues, donc nous allons transformer une ontologie floue (fuzzy OWL2) bien formalisée, qui comporte des informations floues codées en utilisant le langage fuzzy owl2, en une ontologie qu'elle peut supporter la représentation des connaissances probabiliste en un RBF pour faire l'inférence probabiliste sur les ontologies floues.

Introduction générale

La première partie :

Contient deux chapitres, elle présente le contexte du travail, elle a pour but de présenter les ontologies classiques et floues ainsi que quelque généralité sur les annotations du FuzzyOWL2, et les réseaux bayésiens standards et flous.

Chapitre 1- Les ontologies formelles et les ontologies floues: Ce chapitre propose une généralité sur les ontologies ainsi l'imprécision et l'incertitude pour démontré la façon de représenter les concepts vagues .Nous présentons les ontologies classique avec leurs caractéristiques, leurs constructions, ainsi que les domaines d'application. Nous allons décrire l'imprécision et l'incertitude, et nous allons introduire la notion des ontologies floues avec quelque généralité sur les annotations du FuzzyOWL2.

Chapitre 2- Les réseaux bayésiens : Ce chapitre a pour objectif de donner une vision sur les réseaux bayésiens standards et flous. Tout au long de ce chapitre, nous éclaircissons la notion des réseaux bayésiens standards et flous en présentant quelques travaux de l'intégration d'un réseau bayésien dans une ontologie.

La deuxième partie :

Après les généralités présentées dans la première partie. Cette partie est consacrée pour la conception et le développement du système.

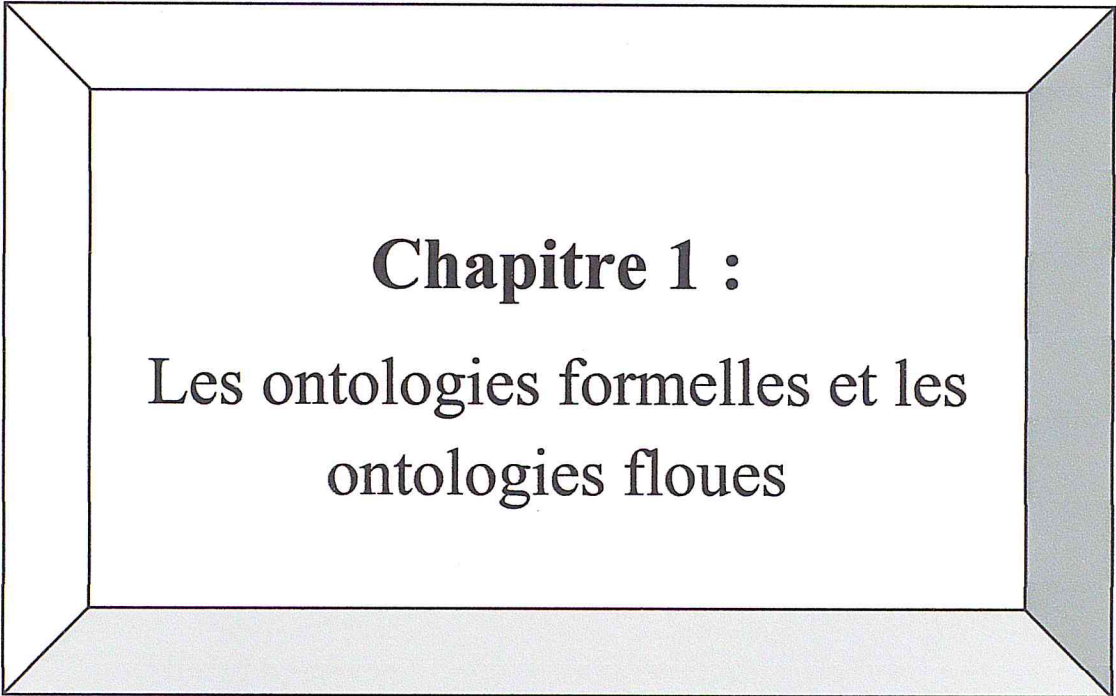
Cette partie met l'accent sur la conception de notre système de la modélisation de l'incertitude dans une ontologie OWL.

Chapitre 3 : La conception du système : Ce chapitre contient les étapes principales pour la construction d'une ontologie floue, et faire l'inférence probabiliste sur les ontologies floues, cela se fait à partir d'un schéma qui représente le système global de la modélisation et qui sera détaillé par la suite dans ce chapitre. Ce chapitre constitue un point de départ à l'implémentation.

Chapitre 4 : Implémentation du système: Ce chapitre est le résultat de la conception pour réaliser le système de la modélisation de l'ontologie floue, il s'agit de transformer les éléments décrits lors de la conception en éléments du langage cible. La fin de ce chapitre constitue un point de départ pour effectuer un test sur notre application.

Chapitre 5 : Test du système: expose un cas d'étude pour bien éclaircir notre approche. Nous prenons le domaine médical et plus précisément la spécialité cardiologique comme un champ d'étude.

La conclusion de ce mémoire synthétise les principales contributions de notre travail qui engendrent d'autres pistes de recherche liées au domaine de raisonnement sur les vagues concepts.



Chapitre 1 :

**Les ontologies formelles et les
ontologies floues**

1. Introduction

La définition des ontologies est héritée d'une tradition philosophique qui s'intéresse à la science de l'être, le terme Ontologie a été utilisé pour la première fois par les philosophes grecs dans une discipline qui a plus 2300 ans, aujourd'hui, elle signifie la « science des étant » c'est-à-dire l'ensemble des objets reconnus comme existants dans un domaine. Dans cette initiation nous allons donc aborder la notion d'ontologie et décrire les démarches connues pour leur construction.

2. Les ontologies formelles

2.1. Définition

Guarino et Giaretta [4] ont soulevé des préoccupations que le terme «**Ontologie**» a été utilisée de manière incohérente.

Le mot « ontologie » provient du domaine de la philosophie, qui sert à désigner une théorie basée sur l'étude de l'être, d'un côté, et il peut aussi être interprété comme l'ensemble de ce qui existe avec ses relations, restrictions, axiomes et vocabulaires dans le domaine de l'intelligence artificiel.

La définition d'ontologie la plus connue dans le domaine de l'intelligence artificielle (IA), est celle qui a été présentée par Gruber [9], « une ontologie est une spécification explicite d'une conceptualisation ». [10] a ajouté quelques précisions au concept de [9] et définit une ontologie comme « une spécification formelle et explicite d'une conceptualisation partagée » :

- **Formelle** : l'ontologie doit être traduite dans un langage compréhensible par la machine.
- **Explicite** : les définitions déclaratives de concepts utilisés et les contraintes quant à leurs usages sont explicitement définies.
- **Conceptualisation** : modèle abstrait d'un phénomène du monde pour lequel on a identifié les principaux concepts.
- **Partagée** : l'ontologie doit être consensuelle et acceptée par un groupe.

2.2. Les objectifs de l'ontologie

L'ontologie peut servir les objectifs de :

- **La communication (humains et organisations)** : Dans l'ontologie, il n'y a jamais deux termes ayant la même sémantique. Cette situation se produit souvent si l'on utilise un langage naturel pour la communication.
- **L'interopérabilité (machine et systèmes)** : l'ontologie sert à définir le format d'échange entre les systèmes.
- **L'ingénierie des systèmes** : l'ontologie peut servir divers aspects du développement des systèmes d'information. Elle assiste le processus de construction de la spécification des systèmes. Elle soutient aussi l'automatisation du processus de vérification de la fiabilité des systèmes.

2.3. Type d'ontologie

Une classification des ontologies a été proposée selon leurs niveaux de généralités [11]:

- **Les ontologies de haut niveau** : sont celles qui décrivent des concepts généraux (espace, temps, matière, objets, événements, actions, etc.) indépendants d'un problème ou d'un domaine d'application particulier.
- **Les ontologies de domaine et les ontologies de tâche** : expriment une conceptualisation qui les rend spécifiques à un domaine déterminé de la connaissance (comme la médecine ou les automobiles), ou une tâche, ou une activité générique (comme le diagnostic ou la vente), en spécialisant les concepts présentés dans les ontologies de haut niveau.
- **Les ontologies d'application** : contiennent les définitions requises pour modéliser la connaissance dans une application, par exemple identifier les maladies du cœur, d'après une ontologie du domaine de la cardiologie. Ces concepts correspondent souvent aux rôles joués par des entités de domaine tout en exécutant une certaine activité, comme l'unité remplaçable ou le composant disponible. On peut distinguer les ontologies selon le formalisme utilisé pour les exprimer :
- **Ontologie Informelle** : l'ontologie est exprimée en langage naturel. Cela peut permettre de rendre plus compréhensible l'ontologie pour l'utilisateur, mais cela peut rendre plus difficile la vérification de l'absence de redondances ou de contradiction

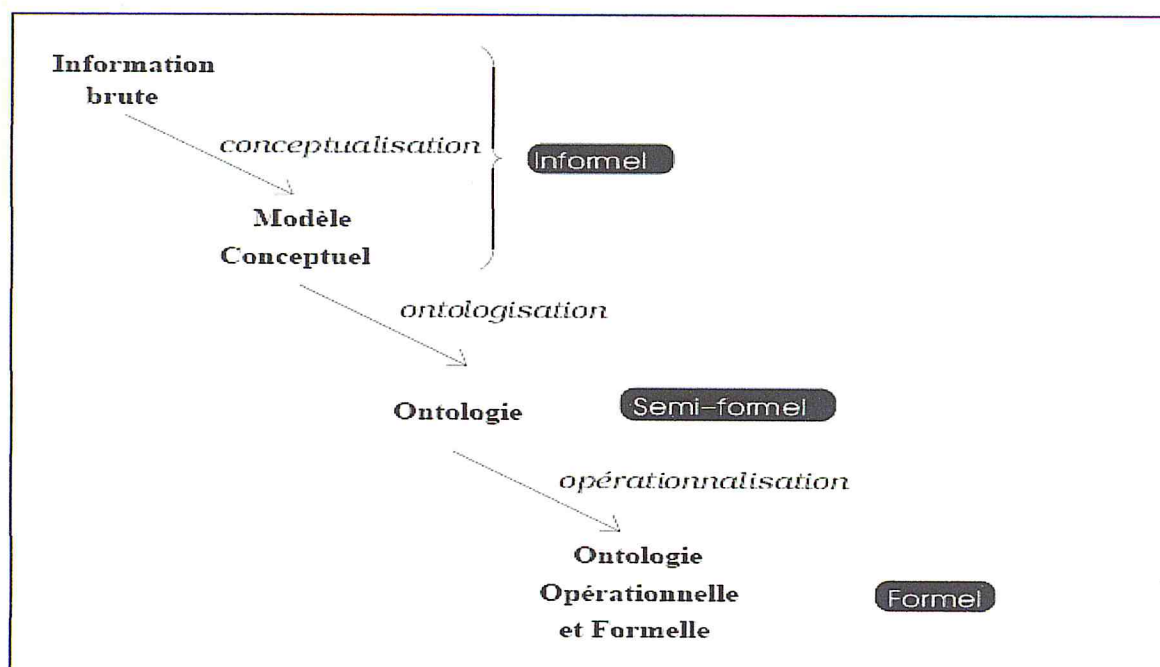


Figure 1 : Construction d'une ontologie opérationnelle [13].

La phase de construction peut être décomposée en 3 étapes : conceptualisation, ontologisation, et opérationnalisation (figure 1).

L'étape d'ontologisation peut être complétée d'une étape d'intégration au cours de laquelle une ou plusieurs ontologies vont être importées dans l'ontologie à construire.

- **La conceptualisation** : identification des connaissances contenues dans un corpus représentatif du domaine .ce travail doit être mène par un expert du domaine, assisté par un ingénieur de la connaissance.
- **L'ontologisation** : formalisation, autant que possible, du modèle conceptuel obtenu à l'étape précédente .ce travail doit être mène par l'ingénieur de la connaissance assisté de l'expert du domaine.
- **L'opérationnalisation** : transcription de l'ontologie dans un langage formel et opérationnel de représentation de connaissances. Ce travail doit être mène par l'ingénieur de la connaissance. Il est à noter que ce processus n'est pas linéaire et que de nombreux allers -retours sont nécessaires pour bâtir une ontologie opérationnelle adaptée aux besoins. Ce modèle de construction d'ontologie est ascendant, c'est -à-dire que l'on part des connaissances à représenter, pour obtenir à une représentation formelle .Une construction descendante est possible, qui consiste à choisir un modèle opérationnel de représentation, en fonction de l'objectif d'utilisation de l'ontologie, puis à instancier ce modèle avec les connaissances du domaine.

3. L'incertitude dans le web sémantique

Le terme «incertitude» dans le web sémantique est destiné à englober une variété d'aspects de la connaissance imparfaite, y compris l'incomplétude, l'indécidabilité, l'imprécision, l'ambiguïté, et d'autres, le terme "raisonnement d'incertitude", se réfère à toute la gamme des méthodes conçues pour représenter et raisonner avec la connaissance lorsque les valeurs de vérité booléennes sont inconnues, inconnaisables ou inapplicables.

Les approches couramment appliquées au raisonnement de l'incertitude comprennent la théorie des probabilités, la logique floue, et de nombreuses autres méthodes.

L'incertitude est une caractéristique intrinsèque de la plupart des tâches requises, et une réalisation complète du World Wide Web comme une source de données traitables qui exigent des formalismes capables de représenter et de raisonner sous incertitude.

- **Nature de l'incertitude :**

Ça concerne l'information sur la nature de l'incertitude, ça veut dire, si l'incertitude est inhérente au phénomène exprimée par la phrase, ou il est le résultat d'un manque de connaissances de l'agent (humain ou machine).

- ✓ *Aléatoire* : l'incertitude vient de l'évènement, l'incertitude est une propriété inhérente de l'évènement.
- ✓ *Épistémique* : l'incertitude est due à l'agent dont sa connaissance est limitée.

- **Types d'incertitude :**

- ✓ *Ambiguïté* : Les référents des termes dans une phrase sur l'évènement ne sont pas clairement spécifiés et donc il ne peut pas être déterminé si la phrase est satisfaite.
- ✓ *Empirique* : Une phrase sur un événement est soit satisfaite ou non satisfaite dans chaque monde, mais on ne sait pas dans quels mondes elle est satisfaite, cela peut être résolu par l'obtention des informations supplémentaires (par exemple, une expérience).
- ✓ *Le hasard* : Une phrase est une instance d'une classe pour laquelle il existe une loi statistique gouvernant si les instances sont satisfaites.
- ✓ *Imprécision* : Il n'y a pas une correspondance précise entre les termes de la phrase et les référents dans l'évènement.
- ✓ *Incohérence* : Il n'y a pas d'évènement qui satisferait la déclaration.

✓ *Incomplétude* : Les informations sur l'évènement sont incomplètes, certaines informations sont manquantes.

- **Modèles d'incertitude :**

Cette classe contient des informations sur les théories mathématiques pour les types d'incertitude. Les types spécifiques de théories comprennent, mais sans s'y limiter, ce qui suit:

- Probabilité.
- Ensembles flous.
- Fonctions de croyance.
- Combinaison de plusieurs modèles (hybrides) [14].

3.1. L'intervalle de vérité

Les systèmes flous appartiennent à la classe des "systèmes à base de connaissance". Leur but principal consiste à implémenter un savoir-faire humain ou des règles heuristiques sous forme d'un programme informatique. La logique floue fournit un formalisme mathématique pour réaliser ce but.

Malgré l'appui de la science sur la notion de mesure, il reste toujours des questions à répondre: comment représenter les valeurs non mesurables ? Comment représenter ce qui est incertain ou imprécis ? Comment représenter les termes du langage humain ? Les descriptions linguistiques d'un système sont souvent vagues. Si une donnée n'est pas connue précisément, elle peut être exprimée par un intervalle de confiance précis. Cet intervalle est un ensemble de valeurs possibles pour la donnée.

La logique floue est une technique de résolution de problèmes très puissants avec une large applicabilité dans le contrôle et la prise de décision. Elle est très utile lorsque des experts humains sont disponibles pour fournir des descriptions subjectives du comportement du système avec des termes en langage naturel. La logique floue est aussi sollicitée dans les situations où il y a de larges incertitudes et des variations inconnues dans les paramètres et la structure du système.

3.2. Définition des concepts vagues

L'ensemble de références d'un mot du langage naturel, s'appelle l'univers du discours. L'univers du discours d'un mot est un ensemble de termes qui évoquent le même concept mais à degrés différents. Il peut être fini ou non.

Le concept vague appelé dans la logique floue, variable linguistique, représente un état dans le système à régler. Chaque variable linguistique est caractérisée par un ensemble tel que :

$\{x, T(x), U, G, M\}$

- x : est le nom de la variable,
- $T(x)$: est l'ensemble des valeurs linguistiques que peut prendre x ,
- U : est l'univers du discours associé avec la valeur de base,
- G : est la règle syntaxique pour générer les valeurs linguistiques de x ,
- M : est la règle sémantique pour associer un sens à chaque valeur linguistique.

La variable linguistique x = température corporel, peut être définie avec un ensemble des termes :

$T(x) = \{\text{Hyperthermie, Normothermie, Hypothermie}\}$, qui forment son univers du discours $U = [33^\circ\text{C}, 42^\circ\text{C}]$. La variable de base est la température. Le terme Hypothermie représente une valeur linguistique. On peut l'interpréter, par exemple comme « la température corporelle, plus petite que 35°C ».

3.3. Fonctions d'appartenance pour les concepts vagues

Chaque sous-ensemble flou peut être représenté par sa fonction d'appartenance. Les fonctions d'appartenance peuvent théoriquement prendre n'importe quelle forme. Toutefois, elles sont souvent définies par des segments de droites, et dites « linéaires par morceaux ». (Très utilisées, car elles sont simples et comportent des zones où la notion est vraie, des zones où elle est fausse, ce qui simplifie le recueil d'expertise) figure (2).

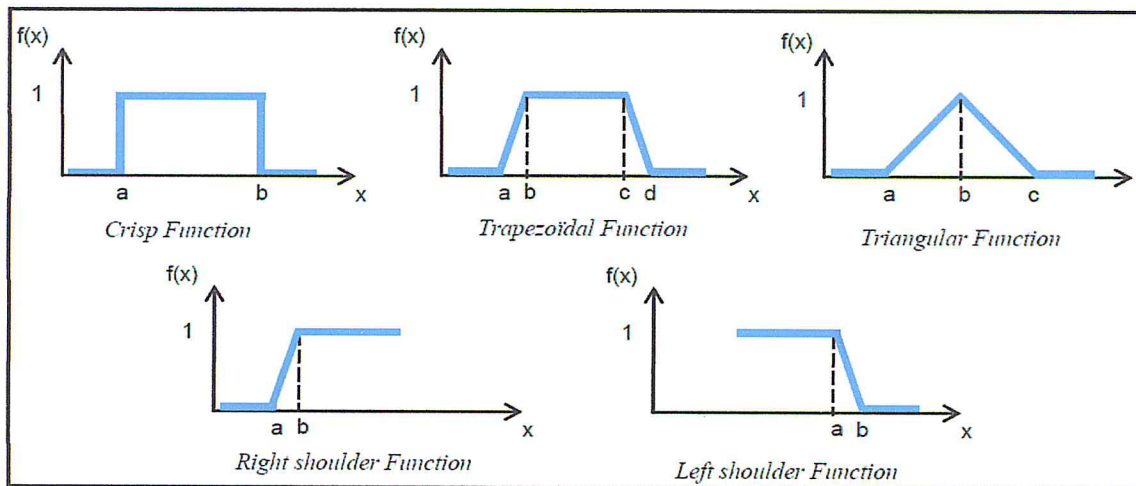


Figure 2 : Les fonctions d'appartenances [15].

Si nous revenons à notre exemple du variable linguistique « température corporel » qui définit l'ensemble des termes « hyperthermie », « normothermie » ou « hypothermie », nous pouvons représenter les termes selon la figure 3.

La représentation de cet intervalle est définie en utilisant les prédicats des domaines concrets, par le biais d'une subsumption du concept vague avec le rôle concret lié au prédicat et aux types de données.

4. Les ontologies floues

Les ontologies jouent un rôle clé dans le succès du web sémantique puisqu'elles sont le formalisme de représentation de connaissance pour spécifier un domaine et pour partager et réutiliser cette connaissance.

Les langages d'ontologie du web standard sont tout à fait réussis dans la définition des ontologies pour le web sémantique, leur formulation originale n'a pas adressé pour la représentation et la gestion de l'imprécision.

Beaucoup de recherches dans la théorie des ensembles flous et la logique ont abordé ce manque par intégration de théorie des ensembles flous et la logique dans la représentation d'ontologie elle-même et les langages de web sémantique.

4.1. Définition

Plusieurs définitions de l'ontologie floue ont été proposées dans la littérature, bien que nous pensions qu'aucune d'entre elles n'est assez générale. Une liste des définitions les plus pertinentes est incluse dans cette section, où la notation a été homogénéisée et dans certains cas résumée. Il est intéressant de noter, cependant, que certains auteurs ne prétendent pas proposer une définition universelle du terme.

- Une ontologie floue est une quadruple (C, R, H, A) , où C est un ensemble de concepts (possibles avec des attributs flous), R est un ensemble de relations floues, H est une hiérarchie de concept et A est un ensemble d'axiomes [16].
- Une ontologie floue est un quadruple (C, R, P, A) , où C est un ensemble de concepts, R est un ensemble de relations binaires entre deux individus, P est un ensemble de relations binaires entre deux concepts, et A est un ensemble d'axiomes floue [17].
- Une ontologie floue est un quadruple (C, R, P, I) , où C est un ensemble de concepts flous, R est un ensemble de relations binaires floues, P est un ensemble de liens appropriés flous, des concepts et I est un ensemble d'individus [18] [19].
- Une ontologie floue est une paire (C, R) , où C est un ensemble de concepts et R est un ensemble de relations floues entre les concepts [20].
- Une ontologie floue est un ensemble d'axiomes OWL floues [21]

Discussion : un premier groupe de définitions existantes essaye de formaliser la notion d'ontologie floue au moyen d'une énumération des éléments de l'ontologie, qui sont étendus, afin de soutenir la représentation des connaissances vagues.

Une ontologie floue est tout simplement une ontologie qui utilise la logique floue pour fournir une représentation naturelle de la connaissance imprécise et vague et facilite le raisonnement sur cette connaissance.

4.2. Les composants d'ontologies floues

Une ontologie floue comprend 6 composants : (1) les concepts précis, (2) les concepts flous, (3) les relations précises, (4) les relations floues, (5) les axiomes et (6) les instances. Les concepts et les relations précis gardent les mêmes définitions et jouent les mêmes rôles que dans une ontologie classique. Les concepts et les relations flous, définis selon la logique floue, servent à la représentation des éléments flous du domaine modélisé. Ils ont des répercussions sur l'expression des axiomes et des instances.

1. Les concepts flous

Un concept flou est un concept défini sur la base d'une valeur particulière d'une variable linguistique (ex. Classification-Age) relative à l'univers de discours, avec l'incertitude et l'imperfection.

2. Les relations floues

Une relation floue est une relation binaire (i.e., entre deux concepts d'un domaine) dont la sémantique est décrite par une valeur particulière d'une variable linguistique.

3. Les instances

Dans les ontologies floues, il existe deux types d'instances : (1) les instances des concepts et des relations précis et (2) les instances des concepts et des relations flous. L'appartenance d'une instance à un composant flou n'est pas sûre. Elle est déterminée par une variable probabiliste prenant ses valeurs entre 0 et 1, et déterminée par une fonction d'appartenance.

4. Les axiomes

Dans le cas d'ontologies floues, les axiomes servent aussi à exprimer les formules de calcul des fonctions d'appartenance.

4.3. Langage d'ontologie floue

Bien que plusieurs langages de définition d'ontologie pour le web sémantique aient apparus, le langage d'ontologie web OWL 2, une version mise à jour d'OWL 1 pour permettre plus d'expressivité, est devenu le langage d'ontologie web recommandée du W3C [22].

Pour créer une ontologie floue d'un OWL 2, des propriétés d'annotation d'OWL sont utilisées pour représenter les caractéristiques de l'ontologie floue qui ne peuvent pas être directement spécifiées dans l'OWL 2 [23].

4.4. L'outil de manipulation des ontologies

Des outils logiciels divers existent pour créer et utiliser des ontologies. Deux catégories très importantes pour des outils d'ontologie sont des éditeurs d'ontologie et des raisonneurs.

Un des éditeurs d'ontologie le plus largement utilisés est Protégé [23] développé à l'université de Stanford. Comme open source et facilement téléchargeable, Protégé a une grande base de support communautaire de l'académie, des utilisateurs gouvernementaux et d'entreprise. Il a été utilisé dans le développement d'applications basées sur la connaissance dans différents domaines comme la médecine, la biomédecine, l'éducation, la fabrication et le commerce électronique.

Il a de nombreux avantages comme le support du dernier langage d'ontologie web OWL 2 et les spécifications RDF du consortium World Wide Web. C'est très extensible par un plug-in et Java (API) pour construire des outils basés sur la connaissance et des applications. À cause de son extensibilité, les nombreux plug-ins existent pour la visualisation d'ontologie, Gestion de projet, génie logiciel et d'autres tâches de modélisation.

4.5. Représentation d'ontologies floues dans OWL 2

Au lieu d'utiliser l'une des propriétés d'annotation par défaut d'OWL 2, L'annotation `fuzzyLabel` peut être utilisée. En outre, pour chaque élément de l'ontologie, il peut y avoir au une annotation de ce type. Chaque annotation sera délimitée par une balise de début `<fuzzyOwl2>` et une balise de fin `</fuzzyOwl2>`, Avec un attribut `Fuzzy Type` spécifiant l'élément flou étant marqué.

- Types de données floues (Fuzzy datatypes) :

```
< fuzzyOwl2 fuzzyType = " datatype" >  
    < DATATYPE >  
</fuzzyOwl2 >
```

Avec : Les Types de données pourraient être

```
<DATATYPE > :=  
<Datatype type =" leftshoulder" a="<DOUBLE >" b="<DOUBLE >" /> |  
<Datatype type =" rightshoulder " a="<DOUBLE >" b="<DOUBLE >" /> |  
<Datatype type =" triangular " a="<DOUBLE >" b="<DOUBLE >" c="<DOUBLE >" /> |  
<Datatype type =" trapezoidal " a="<DOUBLE >" b="<DOUBLE >" c="<DOUBLE >"  
d="<DOUBLE >" />
```

Les analyseurs doivent vérifier les restrictions suivantes:

- $k1 \leq a \leq b \leq c \leq d \leq k2$ est vérifié.

Avec $k1$ et $k2$ sont les limites (max et min) (voir l'exemple figure 6)

$$k1 = 60.0 < a = 90.0 < b = 98.0 < c = 105.0 < d = 110.0 < k2 = 139.0$$

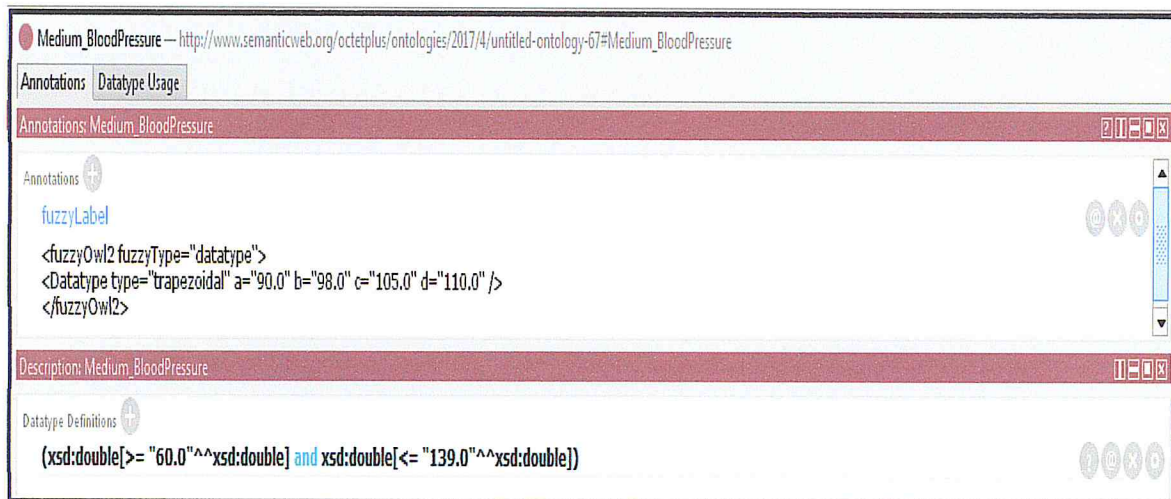
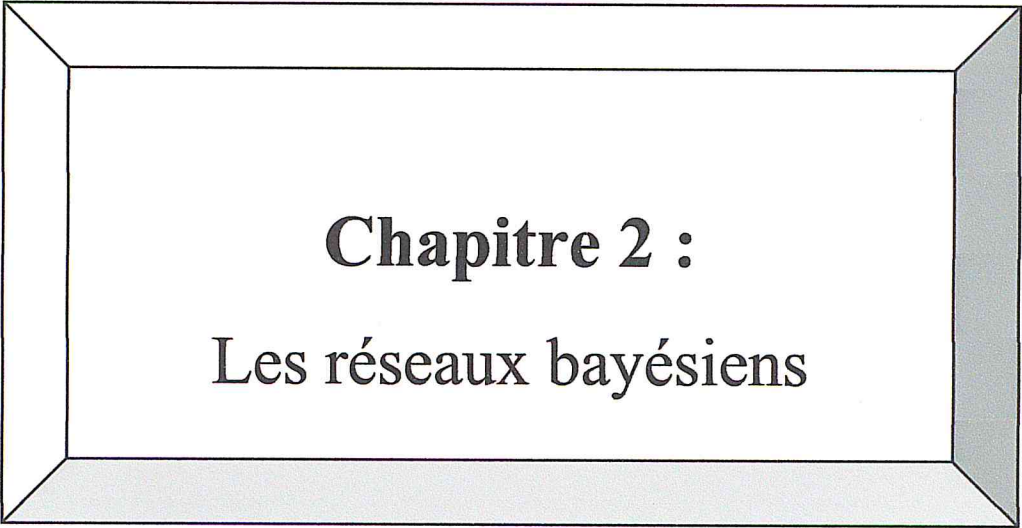


Figure 6 : Exemple d'annotation fuzzOwl2.

5. Conclusion

Dans ce chapitre nous avons présenté les ontologies formelles et les ontologies floues ainsi que les objectifs, les types, les applications, la construction, et les composants d'une ontologie. Par la suite nous avons détaillé l'incertitude dans le web sémantique.

Dans le chapitre suivant, nous allons expliquer les réseaux bayésiens flous et standard et comment peut-il être intégré dans les ontologies



Chapitre 2 :

Les réseaux bayésiens

1. Introduction

La représentation des connaissances et le raisonnement à partir de ces représentations a donné naissance à de nombreux modèles. Les modèles graphiques probabilistes, et plus précisément les réseaux bayésiens (RB), initiés par Judea Pearl dans les années 1980 [24].

Le développement théorique des réseaux bayésiens date des années 1970, mais à cette époque, le manque d'algorithmes efficaces et d'outils pratiques a empêché le développement d'applications. Aujourd'hui, grâce à des outils puissants et conviviaux, les réseaux bayésiens sont une technique en pleine expansion.

Le nom réseau bayésien provient du Bayes et plus précisément de son fameux Théorème, les réseaux bayésiens reposent sur un formalisme basé sur les théories des probabilités et des graphes.

2. Réseaux bayésiens standard

2.1. Définition

Un réseau bayésien est un graphe dans lequel les nœuds représentent des variables aléatoires, et les liens des influences entre variables. Le graphe est acyclique : il ne contient pas de boucle. Les flèches représentent des relations entre variables qui sont soit déterministes, soit probabilistes [25].

Un réseau bayésien est défini par :

- Un graphe orienté sans circuit (DAG) $G = (V, E)$, où V est l'ensemble des nœuds de G , et E l'ensemble des arcs de G .
- Un espace probabilisé fini.
- Ensemble des variables aléatoires associées aux nœuds du graphe et définies sur tel que :

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{Pa}(X_i)) \quad (1)$$

Où $\text{Pa}(X_i)$ est l'ensemble des causes (parents) de X_i dans le graphe G .

Un réseau bayésien est donc un graphe causal auquel on a associé une représentation probabiliste sous-jacente. Cette représentation permet de rendre quantitatifs les raisonnements sur les causalités que l'on peut faire à l'intérieur du graphe [26].

1. Apprentissage des paramètres

✓ À partir des données complètes

Dans le cas où toutes les variables sont observées, la méthode la plus simple et la plus utilisée est l'estimation statistique qui consiste à estimer la probabilité d'un événement par la fréquence d'apparition de l'événement dans la base de données.

✓ À partir des données incomplètes

Dans les applications pratiques, les bases de données sont très souvent incomplètes. Certaines variables ne sont observées que partiellement ou même jamais, que ce soit en raison d'une panne de capteurs, d'une variable mesurable seulement dans un contexte bien précis, d'une personne sondée ayant oublié de répondre à une question, etc. [28].

2. Apprentissage de la structure

L'objectif de l'apprentissage de la structure est de trouver une structure du graphe à partir des données disponibles et qui représente le mieux un problème. Une solution naïve pour trouver la meilleure structure d'un réseau bayésien, est de parcourir tous les graphes possibles, de leur associer un score, puis de choisir le graphe qui a le score le plus élevé. Cependant, le nombre de structures différentes pour un réseau bayésien de n nœuds est super exponentiel. Il est donc impossible d'effectuer un parcours exhaustif en un temps raisonnable. Pour cette raison, la plupart des méthodes d'apprentissage de structure utilisent une heuristique de recherche dans l'espace des graphes acycliques dirigés.

Les méthodes d'apprentissage de structure qui se basent sur un calcul de score maximisent le score de la structure G qui décrit le mieux les données D .

L'approche bayésienne de l'apprentissage de structure se résume à deux éléments essentiels :

La fonction d'évaluation associée à une structure, et la procédure de recherche de la structure visant à maximiser la fonction d'évaluation.

2.3. L'inférence dans les réseaux bayésiens

L'inférence dans un réseau bayésien concerne le calcul de la probabilité de n'importe quelle variable ou sous ensemble de variables à partir des autres variables observées. Il s'agit donc de déterminer les probabilités conditionnelles d'événements reliés par des relations d'influences.

Les algorithmes d'inférence dans les réseaux bayésiens se répartissent en deux groupes : algorithmes d'inférence exacte et algorithmes d'inférence approchée. Les algorithmes

d'inférence exacte exploitent les indépendances conditionnelles contenues dans le réseau pour calculer les probabilités a posteriori exactes [24], [23].

Concernant la deuxième catégorie d'algorithmes, les méthodes utilisées donnent des estimations approchées des probabilités à posteriori [29], [30].

1. Inférence exacte

La tâche de base de tout système d'inférence probabiliste consiste à calculer la distribution de probabilités a posteriori d'un ensemble de variables de requête, étant donné un événement observé autrement dite, une affectation de valeurs à un ensemble de variables d'observation.

X représente la variable de requête ; E représente l'ensemble de variable d'observation $E_1, \dots, E_m$, et E est un événement observé particulier ; Y les variables non observées $Y_1, \dots, Y_n$ (parfois nommées variables cachées). D'où l'ensemble complet de variables $X = \{X\} \cup E \cup Y$.

Une requête de type demande la distribution de probabilité à posteriori : $P(X|e)$ [31].

2. Inférence approchée

Etant donné que l'inférence exacte est impraticable dans de grands réseaux multiplement connectés, il est essentiel d'envisager des méthodes d'inférence approchée. Cette section décrit d'algorithmes d'échantillonnage aléatoire, également nommés algorithme de Monte-Carlo, qui fournissent des réponses approchées dont l'exactitude dépend du nombre d'échantillons générés. Ces dernières années, les algorithmes de Monte-Carlo ont été largement utilisés en informatique pour estimer des quantités difficiles à calculer exactement.

3. Réseaux bayésiens flous

Différents travaux de recherche ont récemment permis l'utilisation de la théorie des ensembles flous dans le cadre des réseaux bayésiens. Différentes approches ont été proposées, mais à ce jour aucune ne fait consensus. Fogelberg et al. [32] ont cherché à généraliser les réseaux bayésiens aux réseaux bayésiens flous qui remplacent les états discrets par états flous dans tout le réseau. Ceci contribue au renforcement de la robustesse du système.

Plus précisément le réseau bayésien flou est une extension de réseau bayésien standard telle qu'il fait la combinaison avec la théorie des ensembles flous.

3.1. Définition

Les réseaux bayésiens sont des outils très puissants et sont utilisés dans plusieurs situations et différents domaines.

Des ensembles flous peuvent représenter des données dans des termes linguistiques qui aident à améliorer la compréhensibilité. De plus, la fonction d'adhésion floue fournit un cadre pour représenter les degrés d'adhésion dans un ensemble.

Combiner ces deux idées peut être conceptuellement difficile parce que la signification d'une valeur d'adhésion floue et une probabilité est très différente, pourtant ils sont représentés de la même façon (un nombre réel dans le champ $[0,1]$).

3.2. Équation bayésienne floue

Cette méthode manipule la règle des bayes pour intégrer l'adhésion confuse dans le processus d'inférence.

Trois circonstances sont décrites dans lequel les valeurs d'adhésion floue peuvent être utilisées [33].

- La première utilise une variable floue comme une variable de question et une variable aléatoire craquante comme évidence.

$$P(\tilde{A}|B) = \frac{\sum_{i \in I} \mu_{\tilde{A}}(A_i) P(B|A_i) P(A_i)}{P(B)} \quad (2)$$

- La seconde utilise une variable de question craquante et une évidence floue.

$$P(A|\tilde{B}) = \frac{\sum_{i \in I} \mu_{\tilde{B}}(B_i) P(B_i|A) P(A)}{P(\tilde{B})} \quad (3)$$

- La méthode finale utilise une variable floue pour tous les deux l'évidence et la question.

$$P(\tilde{A}|\tilde{B}) = \frac{\sum_{i \in I} \sum_{j \in J} \mu_{\tilde{A}}(A_i) \mu_{\tilde{B}}(B_j) P(B_j|A_i) P(A_i)}{P(\tilde{B})} \quad (4)$$

Chacune de ces méthodes exige la marginalisation floue.

$$P(\tilde{B}) = \sum_{i \in I} \mu_{\tilde{B}}(B_i) P(B_i) \quad (5)$$

La méthode ci-dessus semble avoir un sens intuitif dans la combine des probabilités et l'appartenance des valeurs floue. Un inconvénient significatif du procédé est toutefois qu'il exige des changements à l'algorithme d'inférence sous-jacent, cela signifie qu'une inférence personnalisée doit être utilisée.

Les Équations 2 à 5, représente l'ensemble de tous les états pour une variable aléatoire donnée.

4. Les avantages des réseaux bayésiens

Un réseau bayésien peut être avantageusement substitué à différents modèles d'évaluation, de prévision, de diagnostic, d'aide à la décision ou de data mining. Pour les avantages des réseaux bayésien on peut citer :

- la possibilité de rassembler et de fusionner des connaissances de diverses natures dans un même modèle : données de retour d'expérience, expertise (exprimée sous forme de règles logiques, d'équations, ou de probabilités subjectives), observations.
- La convivialité : un réseau bayésien est un modèle graphique, compréhensible et manipulable par un non-spécialiste.
- La versatilité : on peut se servir d'un même modèle pour évaluer, prévoir, diagnostiquer, optimiser des décisions

5. L'ontologie probabiliste

Intuitivement, c'est une ontologie qui a des probabilités attaché aux certains de ses éléments, en général, les personnes confrontées au défi complexe de la représentation de l'incertitude dans des langages comme OWL ont tendance à commencer par écrire des probabilités sous forme d'annotations (par exemple, texte balisé décrivant quelques détails liés à un objet ou d'une propriété). Ceci est une solution palliative qui ne s'adresse qu'à une partie de l'information qui doit être représenté. Au cours des dernières décennies, des formalismes sémantiquement riches et informatiquement efficaces ont émergé pour représenter et raisonner avec la connaissance probabiliste. Annoter une ontologie standard avec des probabilités numériques ne suffit pas, vu que trop d'informations sont perdues à cause de l'absence d'un bon système de représentation qui capture les contraintes structurelles et les dépendances entre les probabilités. Une véritable ontologie probabiliste doit être capable de représenter correctement les nuances. Plus formellement:

Définition : Une ontologie probabiliste est une représentation de la connaissance formelle explicite qui exprime des connaissances sur un domaine d'application. Ceci comprend :

- ✓ Types d'entités qui existent dans le domaine;
- ✓ Propriétés de ces entités;
- ✓ Les relations entre les entités;
- ✓ Les processus et les événements qui se passent avec ces entités;
- ✓ Régularités statistiques qui caractérisent le domaine;
- ✓ Connaissances peu concluantes, ambiguës, incomplètes, peu fiables, et dissonantes liés aux entités du domaine ;
- ✓ Incertitude sur toutes les formes de connaissance citées ci-dessus ;

D'où, le terme entité se réfère à tout concept (réel ou fictif, concret ou abstrait) qui peut être décrit et raisonné sur selon le domaine d'application.

Les ontologies probabilistes sont utilisés dans le but de décrire en détail les connaissances sur un domaine et l'incertitude associée à ces connaissances d'une manière raisonnée, structurée et partageable, idéalement dans un format qui peut être lu et traité par un ordinateur. Ils élargissent également les possibilités d'ontologies standards en introduisant l'exigence d'une représentation adéquate des régularités statistiques et les preuves incertaines sur les entités dans un domaine d'application [34].

6. Intégration d'un réseau bayésien dans une ontologie

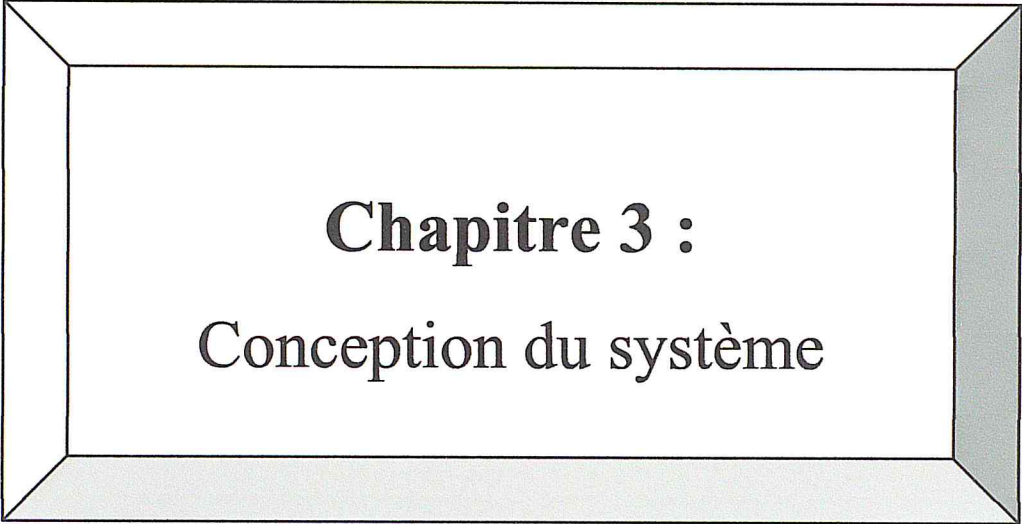
Divers chercheurs ont essayé de combiner les réseaux bayésien avec les ontologies afin de représenter et raisonner avec l'incertitude : [35], [36], [37] ont proposé des extensions du formalisme standard OWL qui sont BayesOWL, OntoBayes et PR-OWL. [38], [39], etc. Ont proposé des nouvelles extensions probabilistes de logiques de description probabilistes (LDP) qui sont une famille de langages de représentation de connaissances ontologiques probabilistes et qui s'adoptent bien au web.

Nous avons intégré le RB dans l'ontologie afin de remédier les défauts de chacune (le RB est incapable de représenter formellement les informations incertaines et l'ontologie est incapable de gérer l'incertitude) et de combiner les inférences bayésiennes (mise à jour de probabilités) avec l'inférence ontologique (vérification d'instances, etc.). Le résultat obtenu peut être utilisée par la suite dans des applications d'aide à la décision qui doivent être capable de raisonner avec des connaissances incertaines.

7. Conclusion

Dans ce chapitre nous avons présenté les réseaux bayésiens standard et flous ainsi que l'apprentissage de structure et des paramètres dans les réseaux bayésiens, par la suite nous avons cité les avantages des réseaux bayésiens. Et nous avons donné une définition de l'ontologie probabiliste et l'intégration d'un réseau bayésien dans une ontologie.

Dans le prochain chapitre, nous allons expliquer le concept de notre système global pour réaliser l'inférence probabiliste sur les ontologies floues.



Chapitre 3 :

Conception du système

1. Introduction

Nous avons vu dans le chapitre précédant un aperçu sur l'état de l'art à propos la représentation et le raisonnement avec la connaissance incertaine dans le web sémantique, de plus nous avons montré quelques lacunes des méthodes existant dans la littérature y compris le problème de la richesse d'incertitude dans un domaine particulier. Pour faire face à ce problème nous avons proposé une nouvelle solution pour la modélisation et le raisonnement avec la connaissance incertaine dans le web sémantique, notre solution basée sur la combinaison des ontologies floues avec les réseaux bayésiens flous. Pour développer notre système nous avons proposé un ensemble des règles de mapping pour construire un réseau bayésien flou à partir d'une ontologie floue.

2. Schéma global de notre système

Les ontologies ont été largement utilisées pour la modélisation et la représentation des connaissances dans divers domaines. Où une ontologie est une spécification formelle d'une conceptualisation partagée [4], cependant, les ontologies ne peuvent pas modéliser et représenter formellement la connaissance incertaine qui due de l'incomplétude, l'aléatoire, l'imprécision...etc. pour cette raison une meilleure représentation et raisonnement avec l'incertitude dans les ontologies est devenu un sujet d'actualité. De plus, plusieurs chercheurs ont essayé de remédier aux défauts de l'ontologie classique par la proposition des extensions pour faire face à l'incertitude, pour ce faire, deux importantes classes de langages ont été distingués, les ontologies probabilistes qui se basent généralement sur les réseaux bayésiens afin de faire face à l'aléatoire et les ontologies floues elles se basent généralement sur l'imprécision et la théorie de sous ensemble floue. Cependant, la majorité des domaines qu'on cherche souvent à modéliser comprennent plusieurs types d'incertitudes à la fois, pour faire face à ce problème il paraît nécessaire de développer des solutions hybrides puissantes et capables de prendre en considération la richesse des connaissances en matière d'incertitude.

De plus, les réseaux bayésiens flous ont été proposés comme une solution hybride par l'intégration de l'imprécision dans les réseaux bayésiens standards pour bénéficier d'avantages de chaque formalisme, où les réseaux bayésiens peuvent représenter des connaissances incertaines qui due à l'incomplétude, l'aléatoire...etc. (connaissance probabiliste), et ne peuvent pas manipuler la connaissance floue et ou imprécise. Par contre, la logique floue est jugé souhaitable pour représenter la connaissance vague et faire des inférences graduelles sur des connaissances imprécises, cependant la logique floue ne peut pas

représenter et faire des inférences sur des connaissances probabilistes. En outre, les réseaux bayésiens flous peuvent faire des raisonnements probabilistes basés sur des observations floues (i.e. si un nœud floue lors de l'inférence il peut prendre plusieurs états à la fois mais avec certains degrés d'appartenances).

Par conséquent, l'objectif du système proposé dans ce projet est de prendre en considération plusieurs types d'incertitudes et /ou d'imperfections à la fois liées aux informations dans le contexte du web sémantique (des connaissances probabilistes et floues à la fois). Pour ce faire, nous allons basées sur les réseaux bayésiens flous et les ontologies floues, donc nous allons transformer une ontologie floue (fuzzy OWL 2) bien formalisée, qui comporte des informations floues codées en utilisant le langage fuzzy owl2, en une ontologie qu'elle peut supporter la représentation des connaissances probabiliste en un RBF.

Le diagramme de la figure suivante donne une vision générale de notre travail et les différentes étapes du processus de transformation sont illustrées dans le schéma suivant :

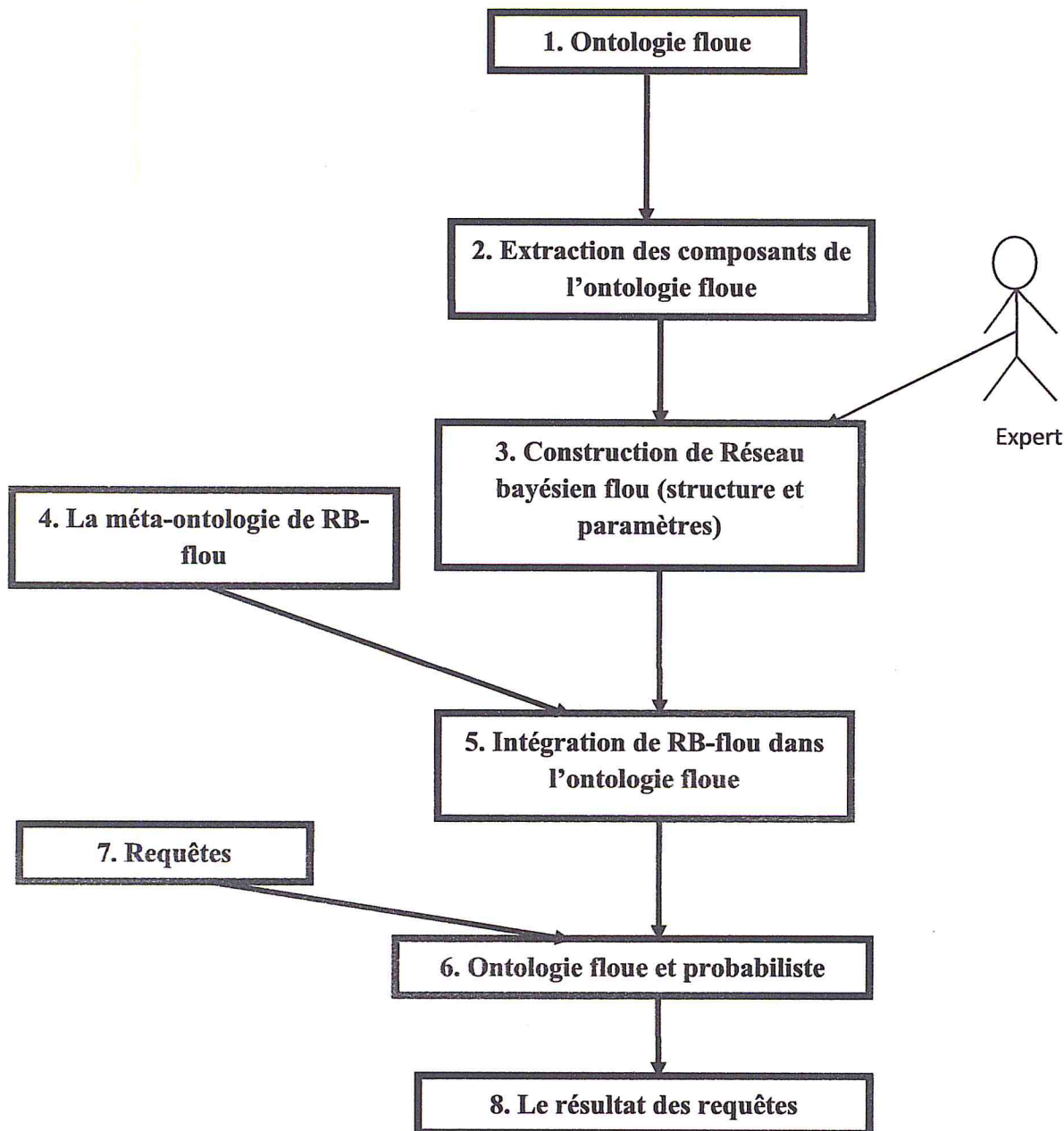


Figure 8 : Système global

- **Input** comme entré dans notre système, il faut une ontologie floue qu'elle représente la partie floue du domaine donné, et notre objectif de la rendre probabiliste pour qu'elle puisse représenter les connaissances probabilistes aussi, et donc augmenter son expressivité et faire face à l'incertitude dans des problèmes complexes et riches en termes d'incertitude.

2.1. Ontologie floue

C'est l'ontologie floue que nous voulons la rendre probabiliste. De plus, les ontologies floues sont des mécanismes utiles pour la représentation et le raisonnement avec des connaissances précises et floues.

Une ontologie floue c'est une ontologie classique avec des annotations "FuzzyLabel" en plus pour les composants flous, commence avec la balise <fuzzy Owl2> et se termine avec la balise </ fuzzy Owl2 > (figures 9, 10).

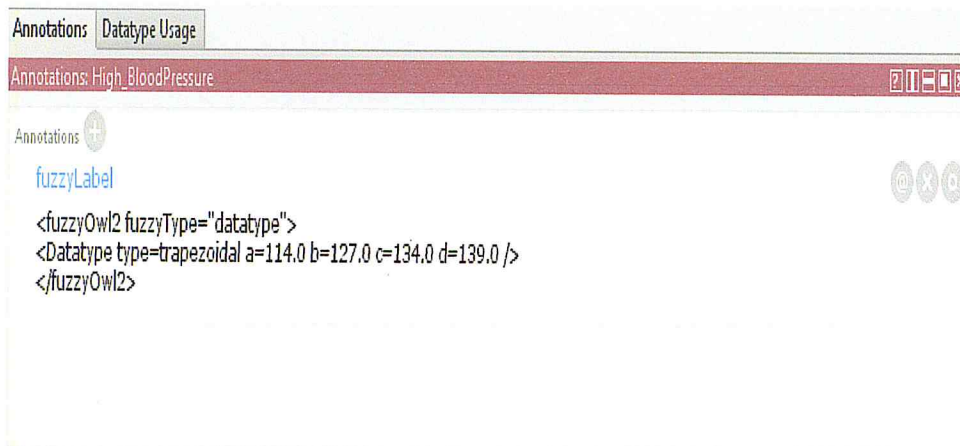


Figure 9 : FuzzyOwl2 annotations exemple.

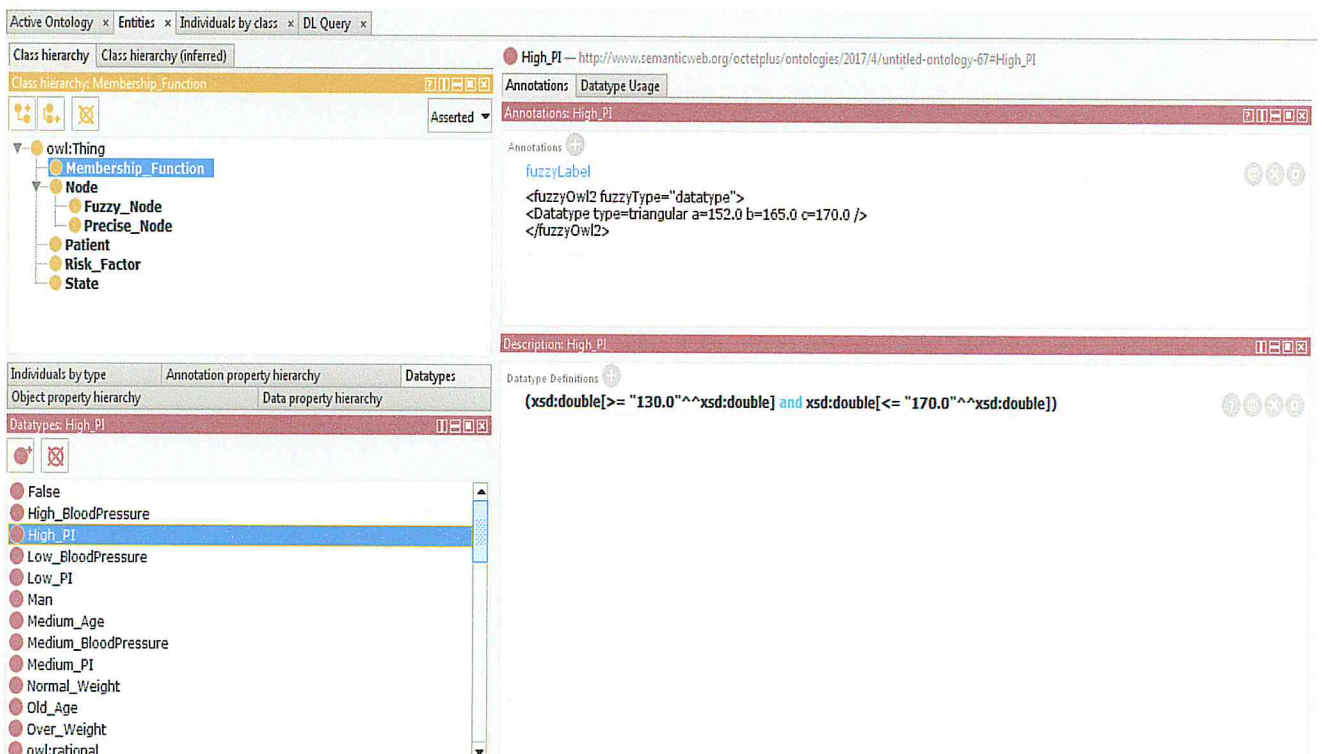


Figure 10 : L'ontologie floue "Cardiovascular disease risk factors".

2.2. Extraction des composants de l'ontologie floue

L'objectif principal de cette étape est d'extraire chaque composant de notre ontologie floue, fournir aux utilisateurs de notre système les ingrédients nécessaires pour qu'ils puissent construire un réseau bayésien flou à partir de l'ontologie floue, et donc représenter les connaissances probabilistes encapsulées dans l'ontologie floue.

On appelle une ontologie floue car il a des composantes floues, et on peut faire la différence entre une composante floue et précise par des annotations propriétés "FuzzyLabel" qu'il contient Fuzzy Owl2 littéral, donc on prépare tous les composants flous ou précis pour être utilisé dans l'étape suivante.

La similarité entre le réseau bayésien flou et l'ontologie floue est que les propriétés d'objets et les propriétés des données de l'ontologie floue représentent des nœuds dans le réseau bayésien et ce choix retourne à la granularité de l'ontologie, et les types des données qui représentent les états des nœuds dans le réseau bayésien parce qu'il joue le même rôle dans les deux cas.

2.3. Construction de réseau bayésien flou (structure et paramètres)

L'apprentissage des réseaux bayésiens doit répondre aux deux questions suivantes :

- Comment estimer les lois de probabilités conditionnelles ?
- Comment trouver la structure du réseau bayésien ?

Nous allons donc séparer le problème de l'apprentissage en deux parties :

- L'apprentissage de paramètres, où il faudra estimer des paramètres à partir des données observées de chaque nœud du réseau.
- L'apprentissage de la structure, où le but est de définir un graphe représentant la tâche à résoudre.

La construction de notre réseau bayésien se fait de manière semi-automatique avec l'aide d'expert humain, il ajoute les nœuds, les liens et les probabilités conditionnelles au réseau bayésien sur la base de la connaissance recueillie.

1. L'apprentissage de la structure

L'objectif de l'apprentissage de la structure est de trouver une structure du graphe à partir des données disponibles et qui représente le mieux un problème.

- **Créer des nœuds :**

Nous sélectionnons un nœud parmi les nœuds extraits (une propriété de données ou une propriété d'objet floue ou précis).

- **Attribuer les états au nœud choisit :**

Le moment où l'utilisateur crée un nœud, une liste des types de données est affichée (type de données précises si le nœud créé est précis et type de données flou si le nœud créé est flou), l'utilisateur attribue pour chaque nœud les états voulus.

- **Faire des relations entre les nœuds :**

L'utilisateur choisit tout simplement le nœud père et fils pour faire la relation entre les deux, et chaque nœud peut avoir plusieurs fils ou père.

2. L'apprentissage des paramètres

L'apprentissage des paramètres consiste à estimer les distributions des probabilités à partir des données disponibles, nous l'avons fait d'une manière semi-automatique avec l'aide d'expert humain.

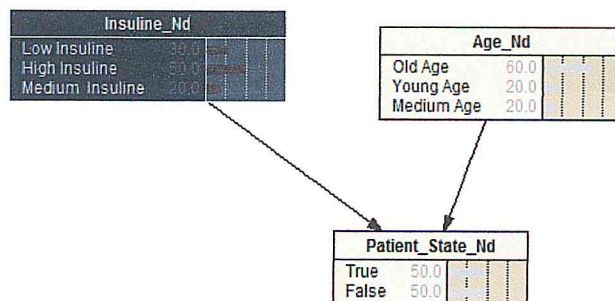


Figure 11 : Nœuds de réseau bayésien flou.

Cet exemple (figure 11) représente trois nœuds, deux flous (**Insuline_Nd**) et (**Age_Nd**) et le troisième précis (**Patient_State_Nd**).

- **Insuline_Nd** à trois états floue "Low Insuline, Médium Insuline et High Insuline".
- **Age_Nd** à trois états floue "Old Âge, Young Âge et MédiumÂge".
- **Patient_State_Nd** à deux états précis "True et False", chacune avec ses probabilités estimée par l'expert.

2.4. Méta-ontologie de RB-flou

Le rôle de cette ontologie dans notre système est de représenter les connaissances probabilistes d'une manière formelle, les connaissances probabilistes sont capturées à l'aide de réseaux bayésiens flous qui vont servir à répondre à des requêtes basées sur des connaissances probabilistes et floues à la fois, notre idée est de représenter le RBF construit à partir de l'ontologie floue dans une ontologie de haut niveau (méta-ontologie), pour ce faire, nous avons proposé un méta-modèle pour un réseau bayésien flou. Le diagramme de classe de la méta-ontologie est schématisé dans la figure 12.

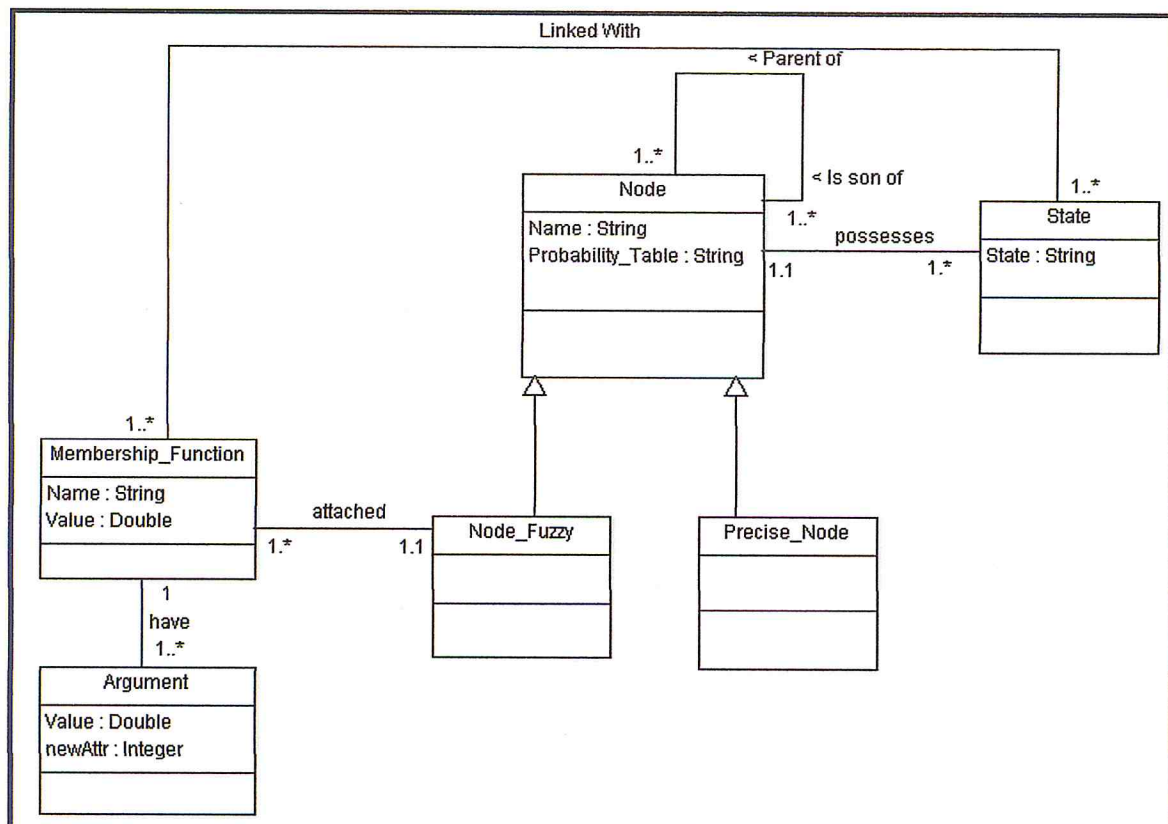


Figure 12 : Diagramme de classes de la méta-ontologie.

2.5. Intégration de RB-flou dans l'ontologie floue

Le diagramme de classe de la méta-ontologie est représenté sur la figure 12. Ses éléments sont les suivants :

- **Node** : super-classe de "Precise_Node" et "Fuzzy Node", a comme instance un nom et une table de probabilité et peut possède une ou plusieurs états. Chaque nœud peut avoir un ou plusieurs parents ou bien un ou plusieurs fils.
- **State** : l'ensemble des états des nœuds qui sont liés à un ou plusieurs fonctions d'appartenances.

- **Precise_Node** : les nœuds précis.
- **Fuzzy_Node** : représente les nœuds flous qui sont attachés à des fonctions d'appartenances.
- **Membership_Function** : représente la fonction d'appartenance de chaque nœud floue, a comme instance un nom, et chaque fonction d'appartenance peut avoir un ou plusieurs arguments.
- **Argument** : les arguments des fonctions d'appartenances.

Nous avons cité dans le tableau suivant les différents composants de méta-ontologie:

Les concepts	Les propriétés des données	Les propriétés d'objets
Node, State, Node_Fuzzy, Precise_Node, Membership_Function, Argument	Probability_Table	Attached, is_son_of, Parent_of, Possesses , Linked_With

Tableau 1 : Les différents composants de méta-ontologie.

En fait, les composants du réseau bayésien (nœuds, les états, ...etc.) construit à partir de l'ontologie floue seront représentés sous forme d'instances et des relations entre ces derniers dans la méta-ontologie (voir figure 13).

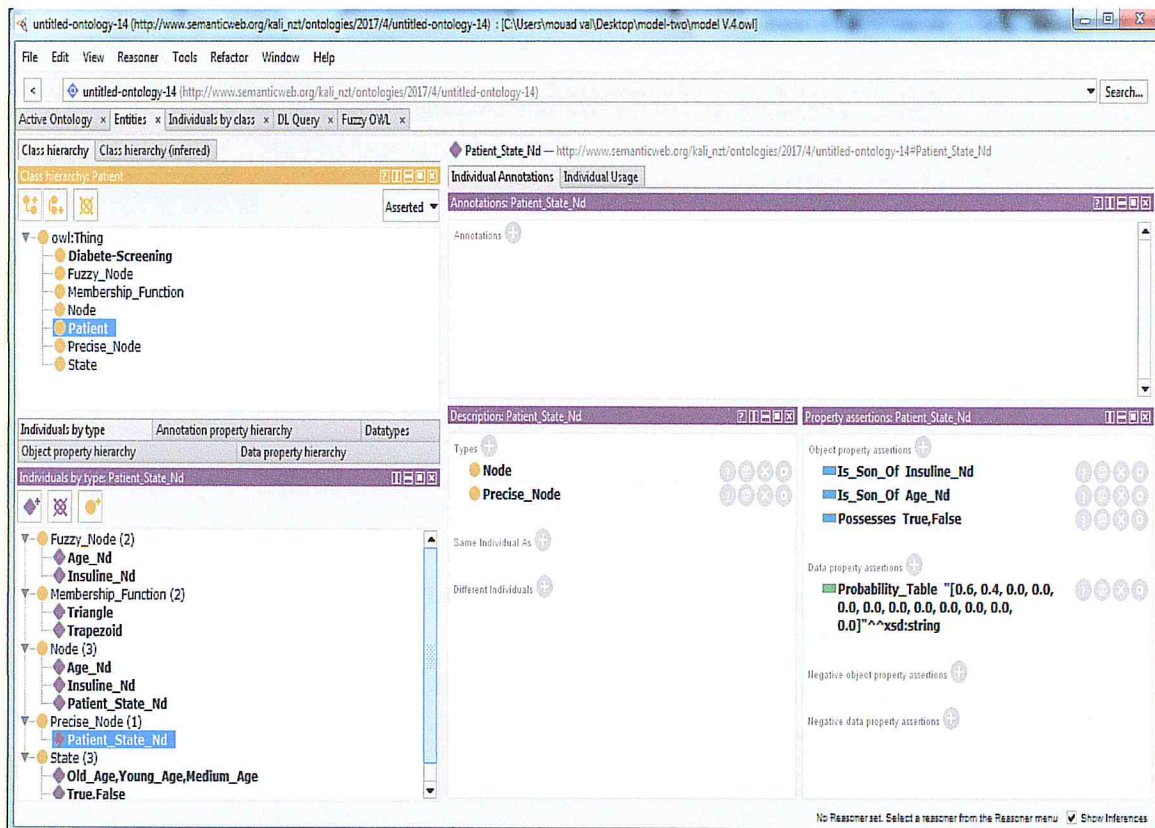


Figure 13 : Exemple des instances dans la méta-ontologie.

Cette figure représente l'exemple précédent (figure 11) dans la méta-ontologie, telle que la classe "Node" contient les deux classes floue et précise. Après chaque classe sera placé dans une autre classe, telle que la classe précise sera placée dans "fuzzy node"...etc.

2.6. Ontologie floue et probabiliste

L'ontologie probabiliste floue est une extension de l'ontologie floue par l'intégration de méta-ontologie de RBF dedans. En réalité, nous allons fusionner les deux ontologies. En conséquence, on obtient la nouvelle ontologie floue et probabiliste (voir la figure 16).

Cette ontologie est une combinaison entre méta-ontologie (voir figure 14) et notre ontologie floue (voir figure 15).

2.7. Requêtes

Comme avantage de l'ontologie floue et probabiliste c'est qu'elle peut représenter et faire des raisonnements avec plusieurs types de connaissances (connaissances déterministe, floue, probabiliste).

En outre, l'ontologie floue probabiliste peut répondre aux requêtes (faire de l'inférence) en utilisant l'inférence probabiliste floue, l'inférence floue, l'inférence classique comme dans les ontologies classiques avec les étapes suivantes :

- **Etape 1 :**

Supposons que : Age_Nd est un noeud flou, Tobacco_Use_Nd et Is_Diabetic_Nd sont des noeuds précis.

Exécuter un algorithme d'inférence (compiler le réseau, voir figure 17) :

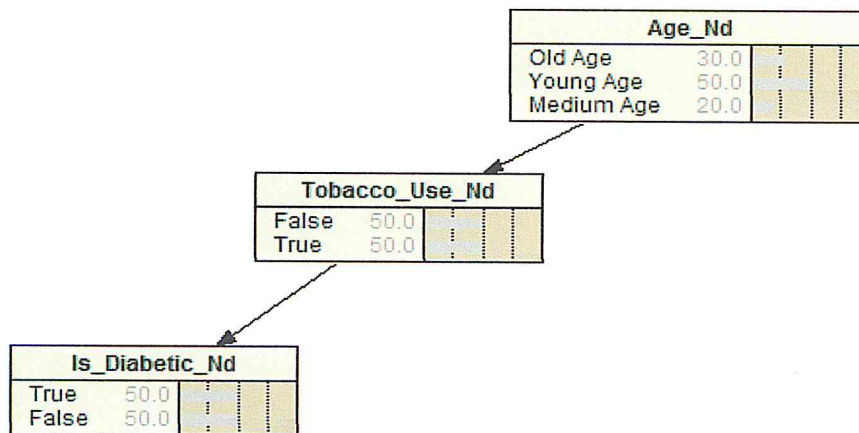


Figure 17 : Exemple d'un réseau bayésien flou.

- **Etape 2 :**

Supposons que pour le noeud Tobacco_Use_Nd l'évidence est dans l'état False alors nous doit le met à l'état False dans ce noeud, pour généraliser si un noeud observé est précis on met l'évidence à l'état observé (voir figure 18).

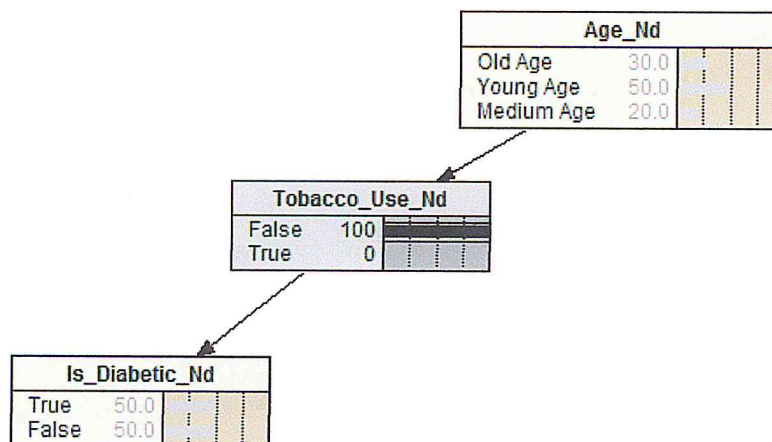


Figure 18 : Exemple d'état observé pour un nœud précis.

Si le nœud est flou alors l'utilisateur doit donner pour chaque nœud un paramètre m pour calculer la nouvelle probabilité (la normalisation) avec l'équation suivante :
 N =ensemble des nœuds observés = $\{n_1 \dots n_y\}$,

E = l'ensemble des états pour un nœud observé = $\{e_1 \dots e_z\}$, $\mu_i(m) =$

degré d'appartenance d'une état i ,

$P(i)$ = la probabilité d'un état i ,

$S = \sum_1^y (\mu_i(m) \times P(i))$,

La nouvelle probabilité normalisée = $P'(i) = \frac{\mu_i(m) \times P(i)}{S}$.

Après la compilation on aura (figure 19) :

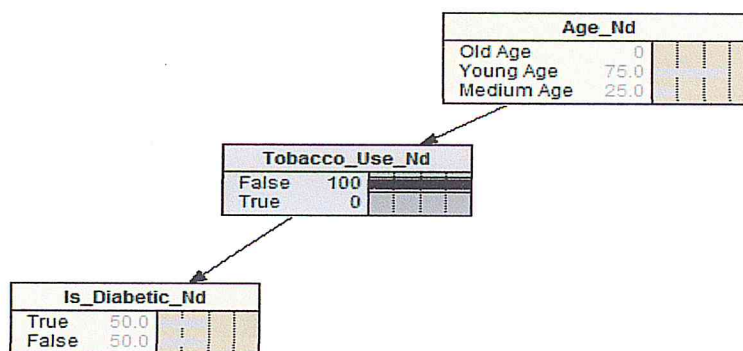


Figure 19 : Exemple d'inférence pour un nœud flou.

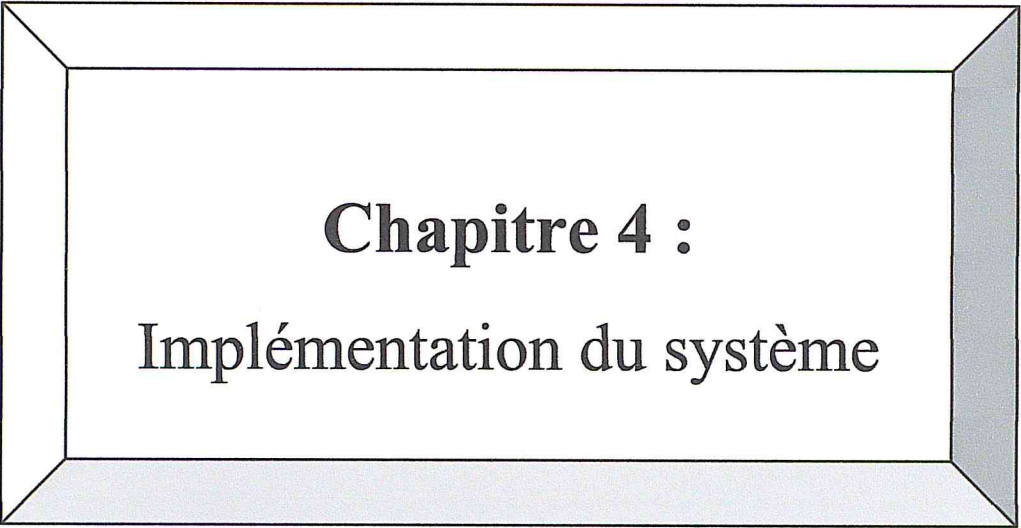
Le raisonneur FuzzyDL [40] pour les logiques de description floues est le plus ancien dans la littérature, qui est disponible publiquement, et sous une mise à jour constante. C'est peut-être le raisonnement le plus travaillé, soutenant certaines caractéristiques uniques. FuzzyDL

utilise des modificateurs de concepts (utilisant des fonctions triangulaires), des définitions explicites de concepts flous (au moyen de fonctions triangulaires, trapézoïdales, trapèze gauche et trapèze droit), de caractéristiques concrètes ou de types de données, qui peuvent avoir une valeur entière, un réel ou une chaîne, entre d'autres nouveaux constructeurs. Du point de vue du raisonnement, il est capable de calculer plusieurs requêtes.

Une autre caractéristique intéressante est que les degrés des axiomes flous peuvent être non seulement des constantes numériques, mais aussi des variables, ce qui permet de traiter des degrés de vérité inconnus.

3. Conclusion

Dans ce chapitre, Nous avons étudié les différentes étapes du processus de la construction d'une ontologie probabiliste. Avec l'explication de chaque étape de notre système à partir d'une ontologie floue se terminant par Les résultats de requêtes. Nous avons schématisé un diagramme nécessaire pour expliquer étape par étape le raisonnement de notre idée et comment fonctionne notre application .Ce chapitre constitue un point de départ à l'implémentation.



Chapitre 4 :

Implémentation du système

1. Introduction

Implémenter c'est « effectuer l'ensemble des opérations qui permettent de définir un projet et de le réaliser, de l'analyse du besoin à l'installation et la mise en service du système ou du produit ».

Dans ce chapitre nous allons décrire de façon visuelle l'implémentation de notre système, il s'agit de transformer les éléments décrits lors de la conception en éléments du langage cible en effectuant des captures d'écran des différentes interfaces du système.

2. Outils de développement de notre système

- **Eclipse IDE** Eclipse fournit des IDE et des plates-formes pour presque toutes les langues et l'architecture. Nous sommes reconnaissants pour nos IDE Java, C / C ++, JavaScript et PHP, construits sur des plates-formes extensibles pour créer des IDE de bureau, Web et cloud. Ces plates-formes offrent la plus vaste collection d'outils complémentaires disponibles pour les développeurs de logiciels [41].
- **Protégé** est une plate-forme gratuite et open source qui fournit une communauté d'utilisateurs croissante avec une suite d'outils pour construire des modèles de domaine et des applications basées sur le savoir avec des ontologies [42].
- **Apache Maven** est un outil de gestion de projet et de compréhension de logiciels. Basé sur le concept d'un modèle d'objet de projet (POM), Maven peut gérer la construction et la documentation d'un projet à partir d'une information centrale [43].
- **OWL API** est une API Java et une implémentation de référence pour la création, la manipulation et la sérialisation des ontologies d'Owl. La dernière version de l'API est axée sur OWL 2. L'API OWL est open source. [44].
- **FuzzyOwl2** Plus précisément, nous utilisons les propriétés d'annotation OWL 2 pour coder des ontologies floues. L'utilisation des propriétés d'annotation permet d'utiliser les éditeurs OWL 2 actuels (par exemple, Protégé) pour la représentation de l'ontologie floue. [45]
- **Netica API** : Développé depuis 1992 et commercialisé depuis 1995 par la société canadienne norsys (<http://www.norsys.com>) basé à Vancouver, le logiciel de réseaux bayésiens Netica est actuellement l'un des plus diffusés à l'échelle mondiale. Netica est utilisé pour le diagnostic, la précision ou la simulation dans les domaines de la finance, de l'environnement, de la médecine, de l'industrie et dans un grand nombre

d'application nécessitant de raisonner en univers incertain. Une version gratuite de logiciel entièrement fonctionnelle est téléchargeable sur le site internet de Norsys.

L'API Netica-J est une bibliothèque complète des classes Java pour travailler avec les réseaux bayésiens (également connu sous le nom des réseaux de croyances, des modèles graphiques ou des modèles causaux probabilistes). Elle contient des fonctions pour construire, apprentissage des données, modifier, transformer, test de performance, enregistrer et lire les réseaux, ainsi qu'un moteur d'inférence puissant [46].

3. La mise en œuvre du système

La première étape vers la construction d'une ontologie floue et probabiliste, est d'importer l'ontologie floue de format OWL, il faut importer une ontologie floue à chaque fois qu'on veut créer une nouvelle ontologie floue et probabiliste.

3.1. L'interface d'entrée

La première interface à afficher est la suivante :

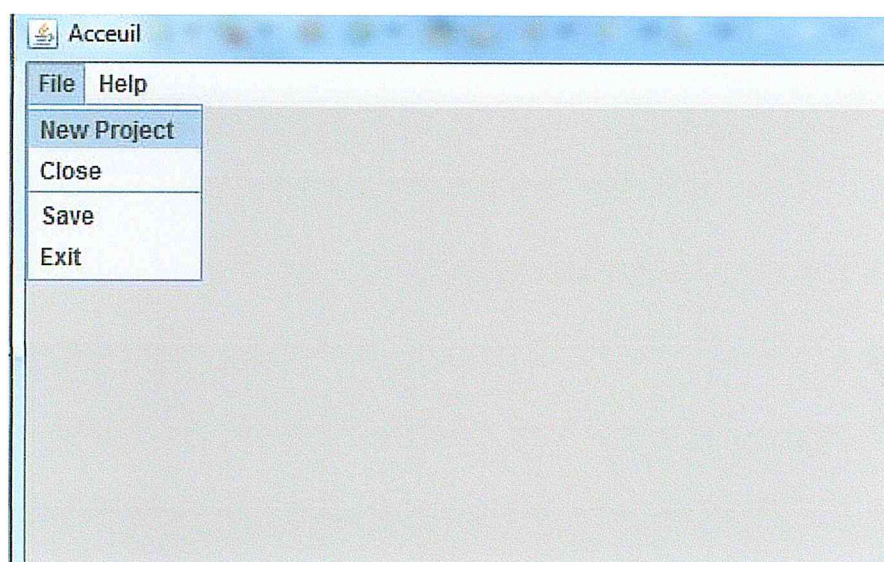


Figure 20 : Interface d'accueil.

C'est l'interface qui permet de créer un nouveau projet (FILE > New projet) pour passer au chargement de l'ontologie.

On doit d'abord importer l'ontologie floue de format OWL depuis un emplacement quelconque (voir figure 21), et puis, passer à l'extraction des différents composants de l'ontologie floue pour pouvoir accéder à l'interface de la création de l'ontologie floue et probabiliste.

3.2. Interface chargement ontologie

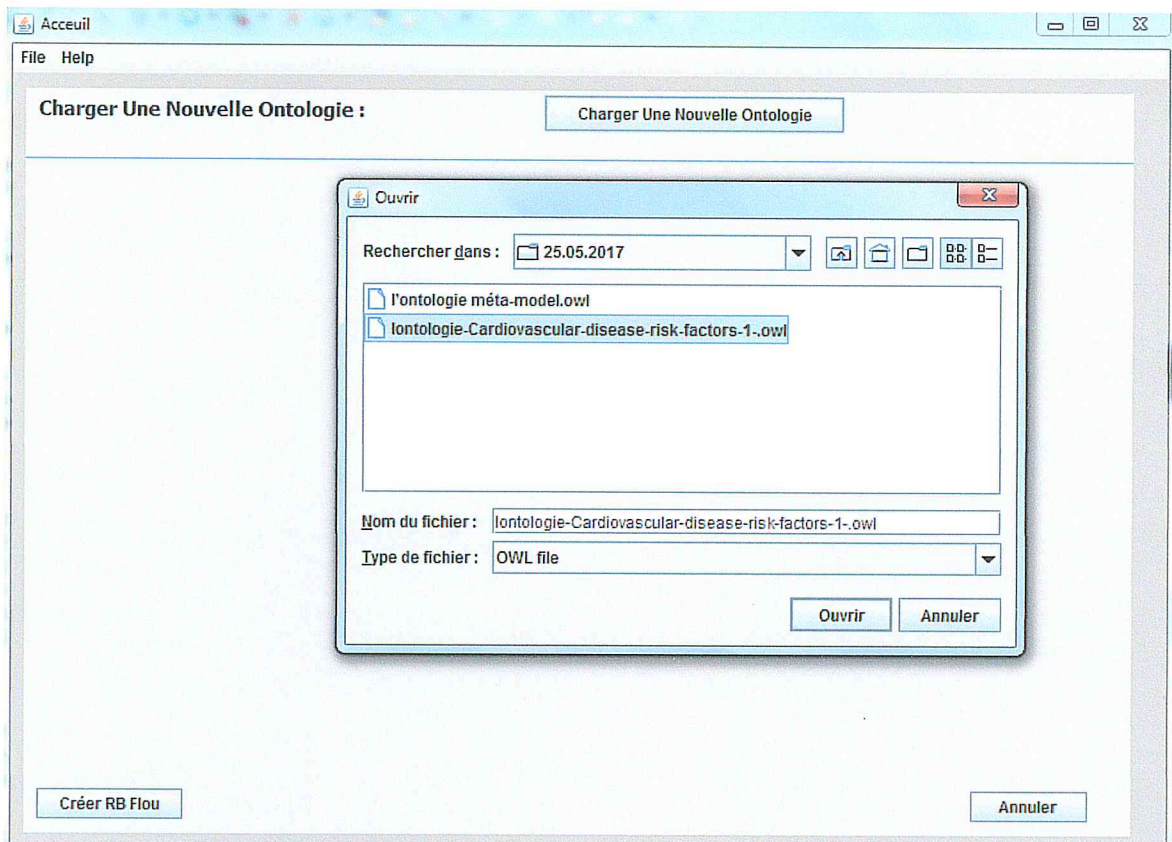


Figure 21 : Interface chargement ontologie.

Charger une nouvelle ontologie : elle importe le fichier de l'ontologie floue OWL en parcourant son chemin.

3.3. Interface extraction des composantes de l'ontologie floue

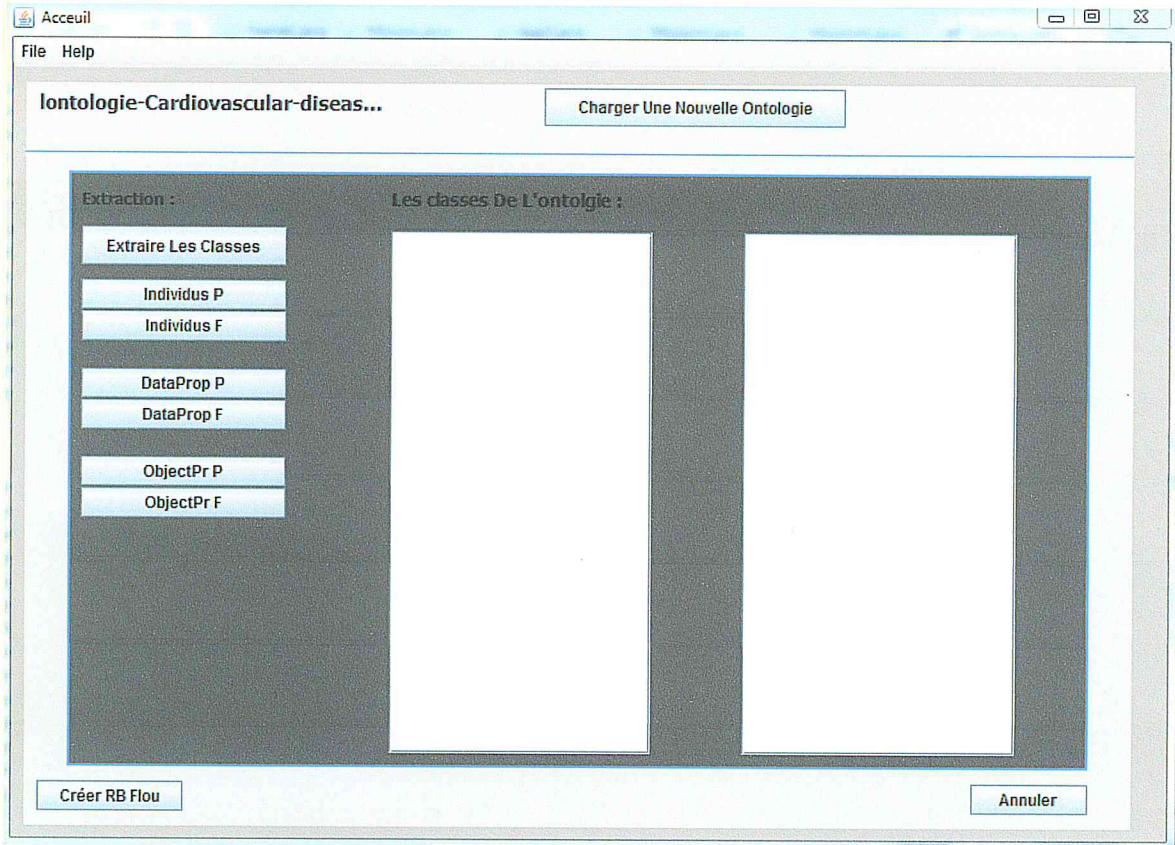


Figure 22 : Interface extraction des composantes de l'ontologie floue.

- **Extraire Les classes** : sa fonction est l'extraction des classes de l'ontologie OWL, pour les utiliser ultérieurement dans l'extraction des différents composants flous ou précis.
- **Individus P** : sa fonction est l'extraction des instances précises de l'ontologie floue OWL, il faut d'abord choisir une classe pour extraire ses instances précises.
- **Individus F** : sa fonction est l'extraction des instances floues de l'ontologie floue OWL, il faut d'abord choisir une classe pour extraire ses instances floues.
- **DataProp P** : sa fonction est l'extraction des propriétés des données précises de l'ontologie floue, il faut d'abord choisir une classe pour extraire ses propriétés de données Précises.
- **DataProp F** : sa fonction est l'extraction des propriétés des données floues de l'ontologie floue, il faut d'abord choisir une classe pour extraire ses propriétés de données floues.

- **ObjectProp P** : sa fonction est l'extraction des propriétés des objets précis de l'ontologie floue, il faut d'abord choisir une classe pour extraire ses propriétés d'objet Précis.
- **ObjectProp PF**: sa fonction est l'extraction des propriétés des objets flous de l'ontologie floue, il faut d'abord choisir une classe pour extraire ses propriétés d'objet floue.
- **Créer RB Flou**: après l'extraction des différents composants de l'ontologie floue ça nous donne l'accès à l'interface de la figure 23.

3.4. Interface création du réseau bayésien floue

Cette interface permet l'effectuation des différentes tâches de notre système : elle permet de créer un réseau bayésien flou (apprentissage de paramètres et de structure), une fois qu'on donne le nom de du réseau bayésien flou, on aura la main pour créer sa structure (les nœuds, les états des nœuds, et les relations entre les nœuds), et à la fin de la création de la structure on passe a l'apprentissage des paramètres qui consiste a estimer les distributions des probabilités à partir des données disponibles, et on a fait d'une manière semi-automatique avec l'aide d'experts humains .

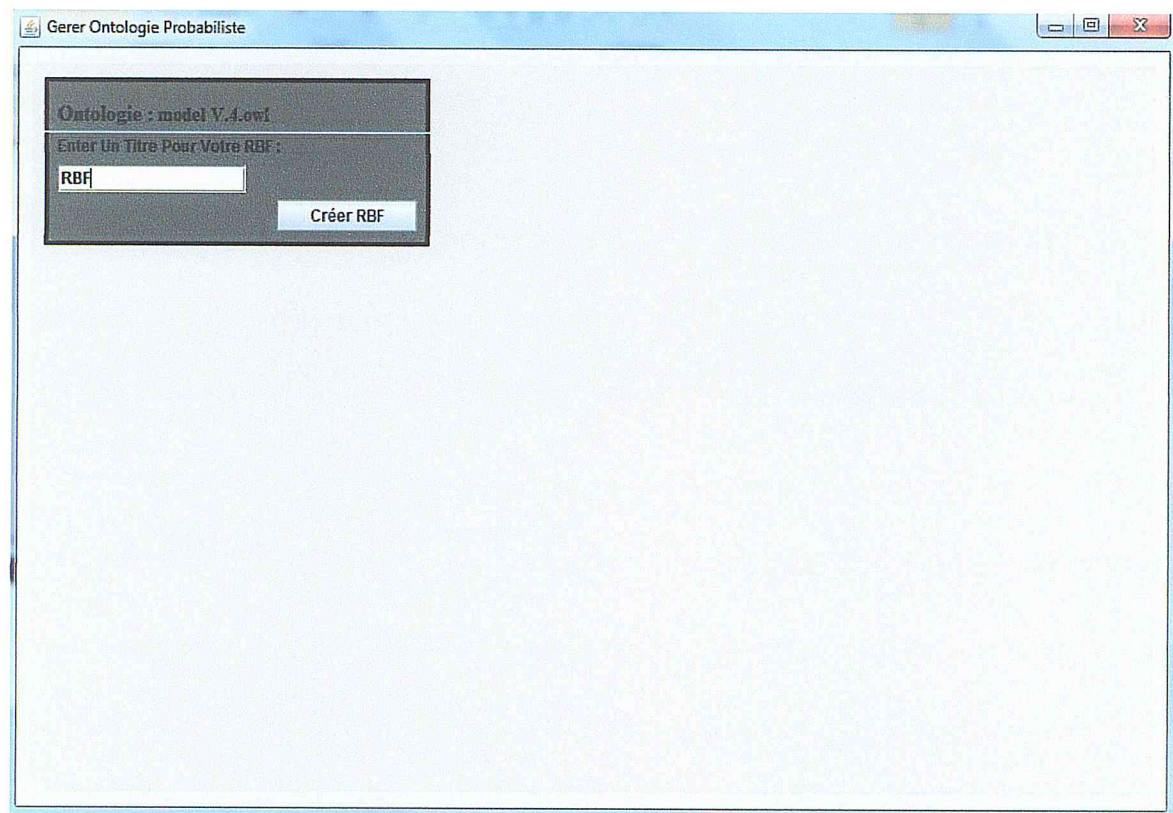


Figure 23 : Interface titre de réseau bayésien flou.

Premièrement il faut donner un titre au réseau bayésien flou a créé, pour faire passer à l'étape suivante :

3.5. Apprentissage des paramètres et de la structure de RBF

1. Apprentissage de la structure

- Création des nœuds avec ses différents états.

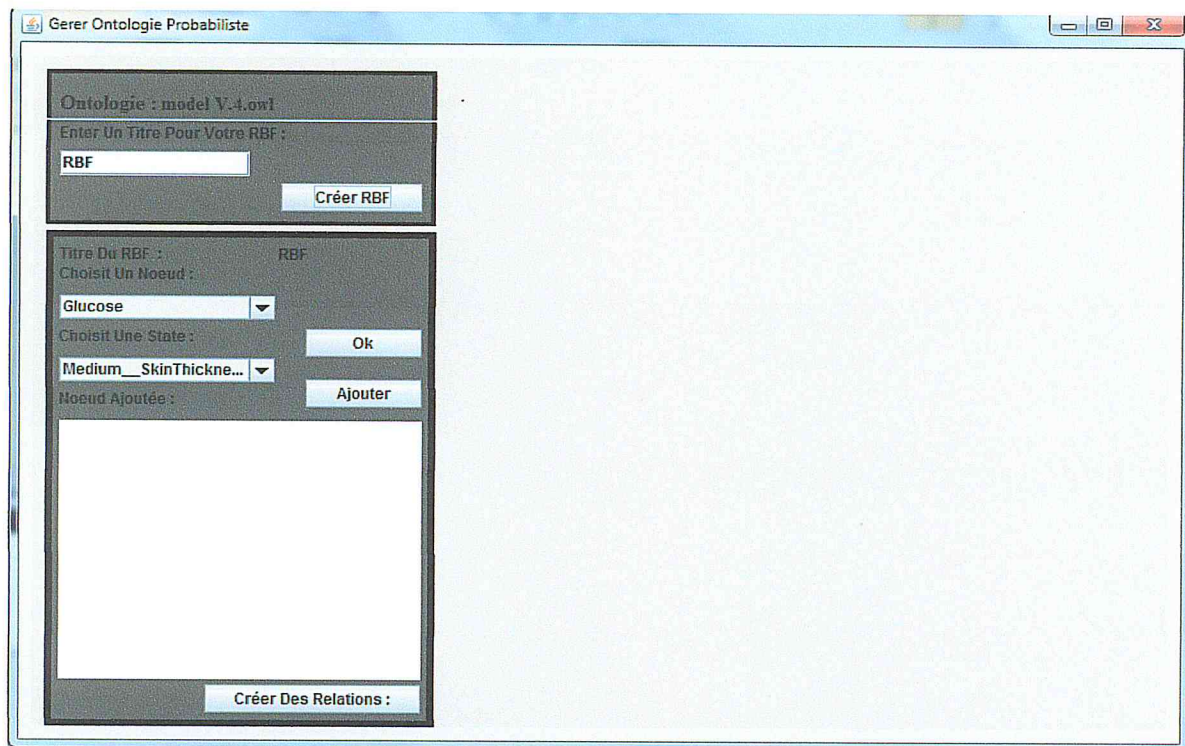


Figure 24 : Interface création des nœuds avec ses différents états.

Cette étape permet la création des nœuds (précis ou flous) avec ces paramètres, il faut d'abord choisir un nœud si le nœud choisit est un nœud flou, une liste des états flous est affichée, et à l'utilisateur de choisir les états voulus de ce nœud.

- **Ajouter** : après que l'utilisateur choisit les états voulus d'un nœud, le bouton ajouter permet d'ajouter un nœud avec ces états au réseau bayésien flou et afficher ce dernier dans une liste des nœuds créé pour passer à un autre nœud.

Une fois l'utilisateur crée tous les nœuds, le bouton (Créer des relations) permet de passer à l'étape suivante :

- Créer les relations entre les nœuds

Cette étape est la dernière étape pour l'apprentissage de la structure de réseau bayésien flou.

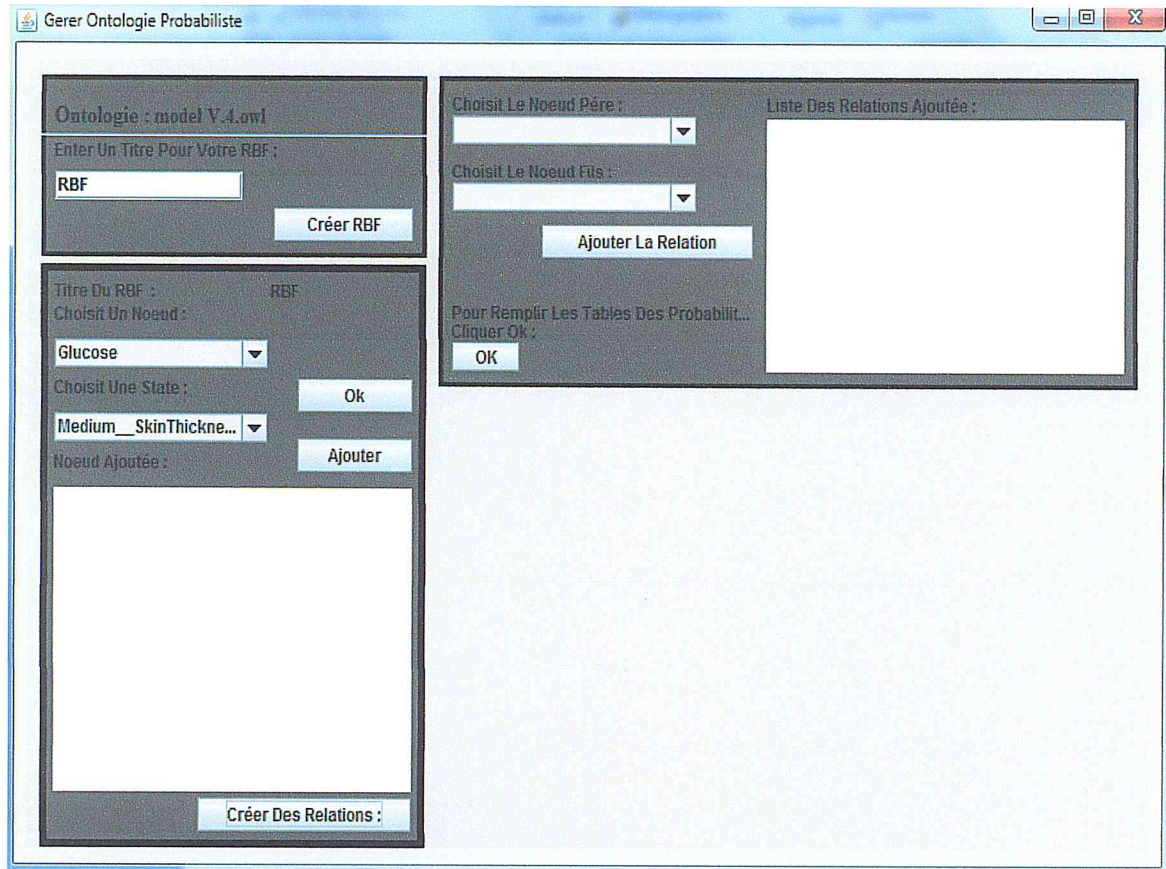


Figure 25 : Interface création les relations entre les nœuds.

- **Choisis le nœud père :** c'est une liste qui contient tous les nœuds créés dans l'étape précédente (floue et précise), pour que l'utilisateur choisisse le nœud comme un père d'un autre nœud.
- **Choisis le nœud fils :** c'est une liste qui contient tous les nœuds (flous et précis) sauf le nœud choisi comme un père, pour que l'utilisateur choisisse un nœud comme un fils du nœud père.
- **Ajouter relation :** permet d'ajouter une relation entre les deux nœuds choisis comme un père et fils dans le réseau bayésien flou, et affiche cette relation dans une liste.

Afin de terminer l'étape de l'apprentissage de structure de réseau bayésien flou, il faut passer à l'apprentissage des paramètres dans l'étape suivante.

2. Apprentissage des paramètres

Cette étape est la dernière étape dans la construction du réseau bayésien flou, elle consiste à estimer les distributions des probabilités à partir des données disponibles, d'une manière semi-automatique avec l'aide d'experts humains.

Figure 26 : Interface apprentissage des paramètres et structure.

Une fois l'utilisateur termine l'étape de l'apprentissage de structure, il passe a l'apprentissage des paramètres :

- **Table_Proba** : cette icône permet d'afficher la table de probabilité à remplir pour chaque nœud choisi par l'utilisateur.
- **Valider** : après le remplissage des probabilités de chaque nœud il faut cliquer sur valider pour valider les probabilités remplit et passe aux nœuds suivants.
- **Terminé** : après le remplissage et la validation de tous les tables des probabilités des nœuds, l'apprentissage des paramètres est terminé, donc il faut cliquer sur terminer pour enregistrer le réseau bayésien flou créé par l'utilisateur de format dne, et passer aux dernières interfaces pour faire le test.

4. Test

Dans cette dernière partie nous allons montrer l'interface qui concerne le test (voir figure 27).

The screenshot shows a graphical user interface for testing. It contains several input fields and buttons:

- A dropdown menu labeled "Choisit Un Noeud Pour Lance Votre Requ..." with "Family_History_Nd" selected. Below it is a file path "C:\Users\mopad\va\Desktop\..." and an "OK" button.
- A dropdown menu labeled "Choisit Une State:" with "False" selected and an "Ok" button.
- A text input field labeled "Entrer Votre Paramètre :" containing the value "26" and an "OK" button.
- A dropdown menu labeled "CDV_Nd" with an "OK" button.
- A large text area at the bottom displaying "Resultat : 0.798".

Figure 27 : Interface de test.

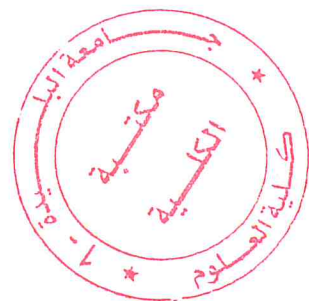
L'utilisateur doit choisir des nœuds évidences et définir ses observations (un paramètre si le nœud choisit est flou ou bien un état si le nœud choisit est précis), après il choisit le nœud requête pour calculer le résultat.

5. Conclusion

Dans ce chapitre nous avons montré l'implémentation de notre système de la modélisation de l'incertitude dans les ontologies floues avec des captures de notre application. La fin de l'implémentation va effectuer un point de départ pour faire un test sur notre application.

Chapitre 5 :

Test du système



1. Introduction

Dans ce dernier chapitre nous allons effectuer un test sur notre système implémenté.

La méthode que nous avons utilisée pour tester notre système c'est d'effectuer les étapes de notre modélisation pour faire l'inférence probabiliste sur les ontologies floues et créer une ontologie floue probabiliste utilisable pour soutenir le diagnostic d'un patient entrant dans une clinique avec des symptômes, s'il a récemment des facteurs de risque pour la maladie cardiovasculaire ou non .

On outre, notre système est générique, il est destiné à tous les domaines qui portent le flou tel que la chimie, biologie, vétérinaire,...etc. Il n'est pas spécifique pour un seul domaine.

2. La création de l'ontologie floue

Pour commencer notre test on a besoin d'une ontologie floue, et comme nous n'avons pas trouvé une ontologie floue prête à utiliser directement pour notre test, parce que la plupart sont payantes, alors nous avons choisi une petite ontologie déterministe qui concerne la médecine, et on a recyclé vers une ontologie floue.

2.1. Motivation de l'étude de cas

La médecine a été un des terrains d'expérimentation privilégiés de l'intelligence artificielle (IA) avec la construction de systèmes experts puis de systèmes à base de connaissances (SBC) en arrivant à l'ontologie et la logique de floue. Elle a été aussi le terrain des échecs de ces approches. Un échec provoqué par une mauvaise prise en compte de la représentation des concepts flous.

Dans ce contexte, nous allons entamer une étude de cas du domaine médical et plus précisément la spécialité cardiologique. Dans un tel domaine, l'incertitude est considérée importante pour la résolution d'un problème du diagnostic.

2.2. Ontologie cardiovasculaire

Les maladies cardiovasculaires désignent les pathologies qui touchent le cœur ou les vaisseaux sanguins. Selon un rapport de l'OMS paru en janvier 2015 [47], les maladies cardiovasculaires représentent 31% des décès et sont la première cause de mortalité mondiale. Si en Algérie la situation est en voie d'amélioration, les efforts à fournir pour parvenir à une diminution significative du nombre de décès sont encore importants. Car ces pathologies nécessitent une détection et une prise en charge adaptée.

2.3. La cardiovasculaires et les facteurs des risques

Les causes de ces événements sont multiples mais connues : nous parlons de facteurs de risque cardiovasculaire. Parmi ces facteurs de risque, nous identifions deux catégories principales : (i) les facteurs irréversibles (âge, sexe, hérédité) et (ii) les facteurs modifiables (tabagisme, surpoids, hypertension artérielle, diabète ou dyslipidémie, activité physique). Parmi ces facteurs de risque, nous identifions deux catégories aussi : (i) les facteurs précis (sexe, hérédité, tabagisme, diabète) et (ii) les facteurs flous (âge, surpoids, hypertension artérielle, activité physique).

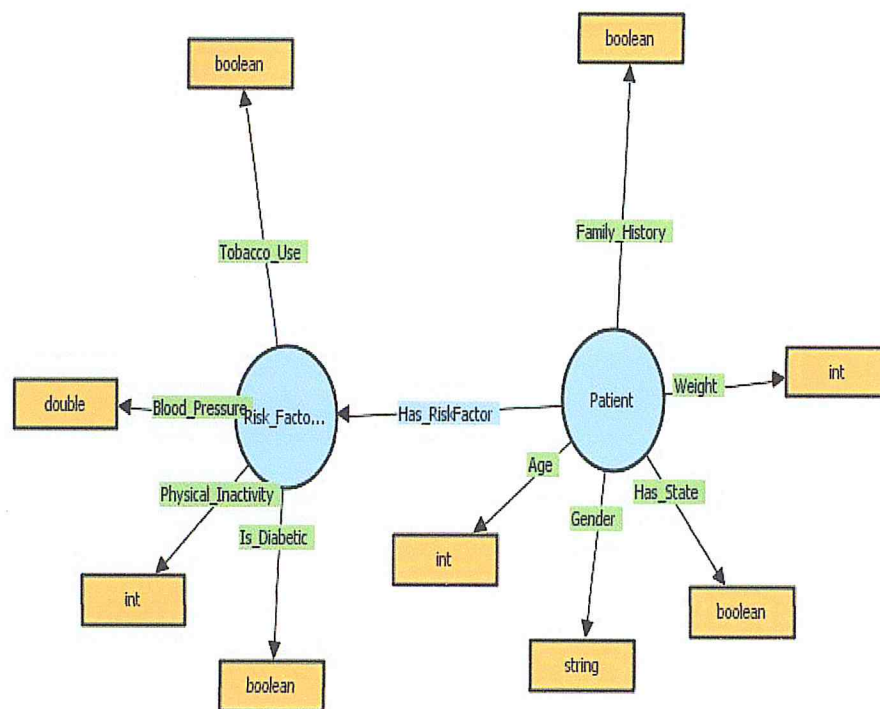


Figure 28 : Vision générale sur l'ontologie floue cardiovasculaire.

2.4. Les différents composants de l'ontologie floue

Classes		patient		Risk Factor	
Object Property	Floue				
	Précis			Has_Risk_Factor	
Data Property	Floue	Age : int Weight: int		Blood_Presure : double Phisical_Inactivity : int	
	Précis	Family_History : boolean Gender : String CDV: boolean		Is_Diabetic : boolean Tobacco_Use : boolean	
Data Type	Flou	Age	Adult_Age Young_Age Old_Age	BP	Low_Blood_Pressure Medium_Blood_Pressure High_Blood_Pressure
		Weight	Normal_Weight Under_Weight Over_Weight	PI	Low_PI Medium_PI High_PI
	Précis	CDV, Family_History	True Flase	Is_Diabetic Tobacco_Use	True Flase
		Gender	Man Woman		

Tableau 2 : Les différents composants d'ontologie floue cardiovasculaire.

PI: physical inactivity.

CVD: Cardiovascular disease.

BP: Blood Pressure.

Propriétés de données (Data Properties)	Type de données (Data Type)	fonction d'appartenance (Membership function)
Age	Young_Age	trapezoidal a=22 b=24 c=26 d=27
	Adult_Age	trapezoidal a=30 b=35 c=40 d=45
	Old_Age	trapezoidal a=50 b=60 c=70 d=80
Blood_Pressure	Low_Blood_Pressure	trapezoidal a=60 b=65 c=70 d=79
	Medium_Blood_Pressure	trapezoidal a=90 b=98 c=105 d=110
	High_Blood_Pressure	trapezoidal a=120 b=125 c=130 d=135
Physical_Inactivity	Low_PI	triangular a=130 b=135 c=140
	Medium_PI	triangular a=145 b=150 c=155
	High_PI	triangular a=160 b=160 c=170
Weight	Under_Weight	triangular a=44 b=49 c=52
	Normal_Weight	triangular a=53 b=60 c=70
	Over_Weight	triangular a=73 b=83 c=93
Has_State	True / False	
Is_Diabetic	True / False	
Family_History	True / False	
Tobacco_Use	True / False	
Gender	Man/Woman	

Tableau 3 : Les différents types de donnée de l'ontologie floue cardiovasculaire.

2.5. Les annotations de l'ontologie floue

Pour transformer une ontologie déterministe vers une ontologie floue, il faut annoter toutes les composantes floues par des annotations "Fuzzy Label".

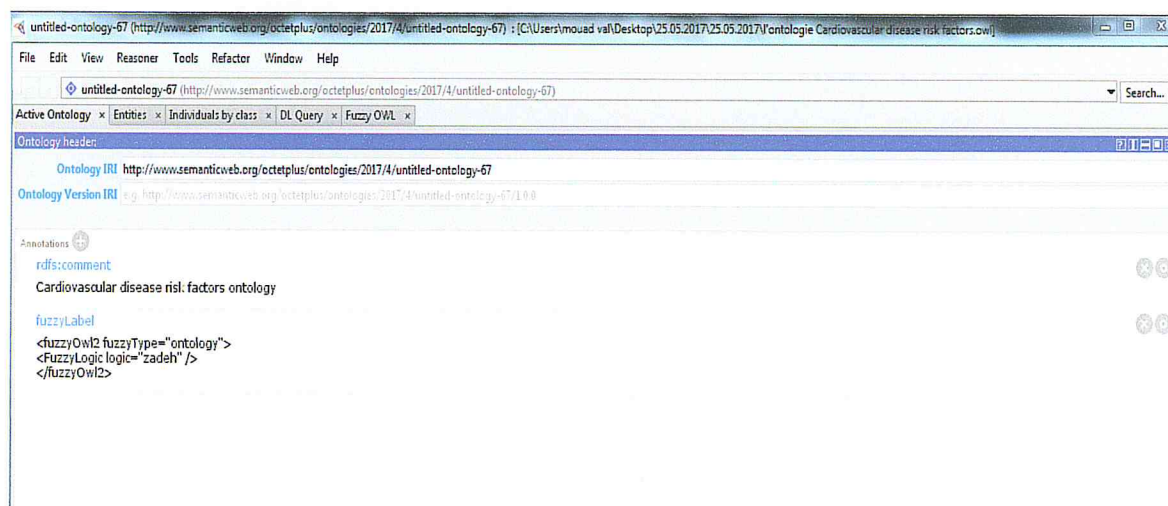


Figure 29 : L'annotation de l'ontologie floue cardiovasculaire.

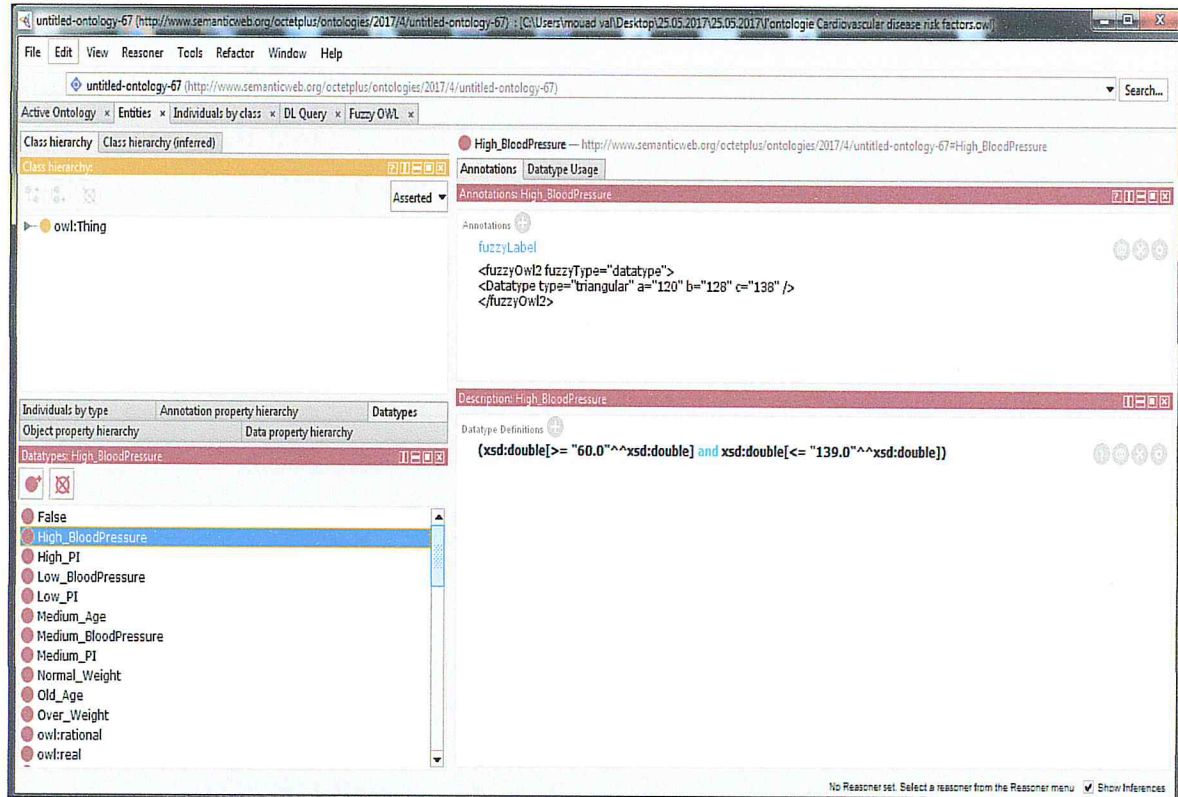


Figure 30 : Annotation d'un type de données avec sa fonction d'appartenance.

3. Lancement du test

Dans cette partie nous montrons les différentes étapes qu'on a suivies pour terminer notre test.

3.1. Chargement de l'ontologie flou cardiovasculaire

La première étape pour commencer notre test, est de télécharger notre ontologie floue de test cardiovasculaire.

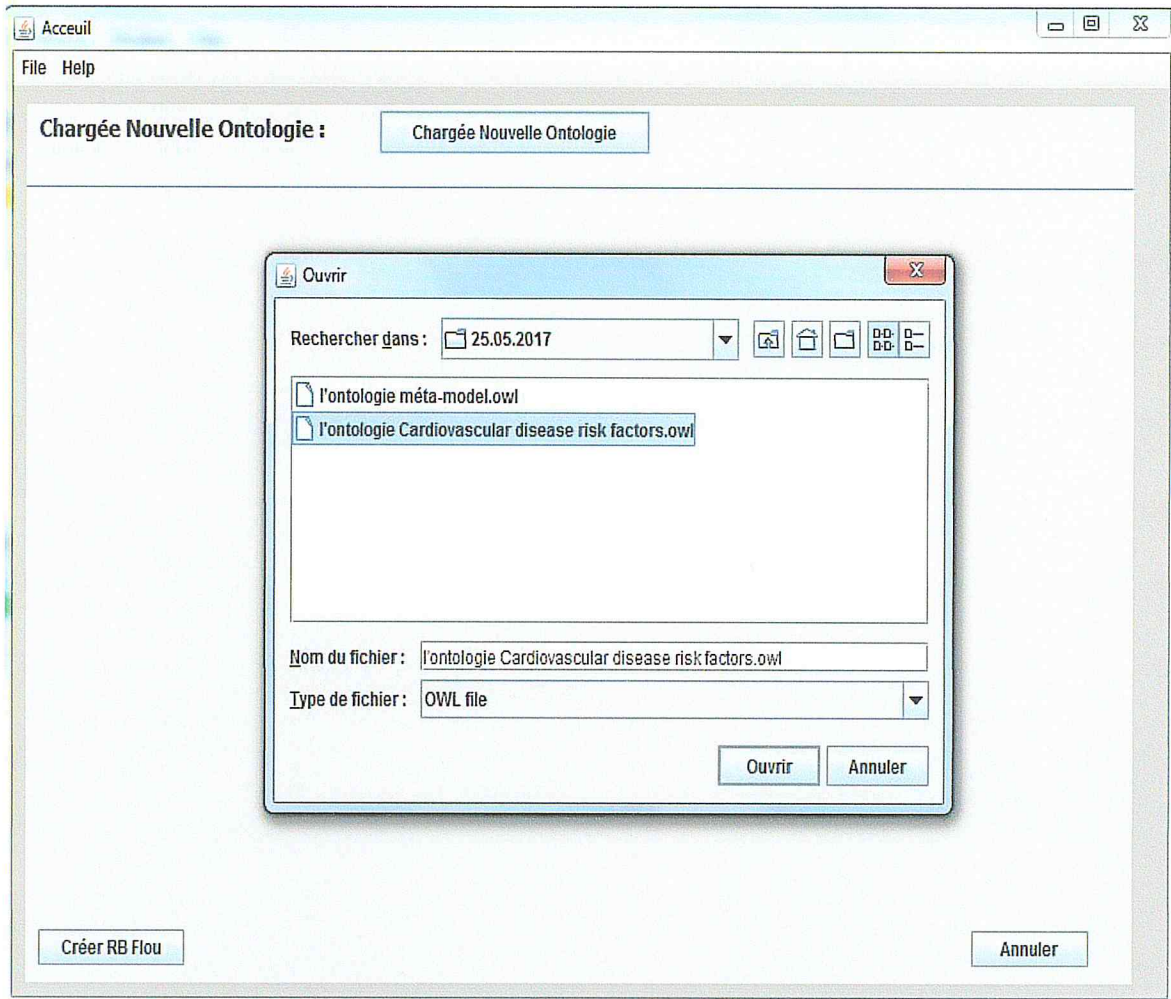


Figure 31 : Chargement de l'ontologie floue cardiovasculaire.

3.2. L'extraction des composants

Dans cette étape nous avons fait l'extraction des différentes composantes de notre ontologie floue cardiovasculaire.

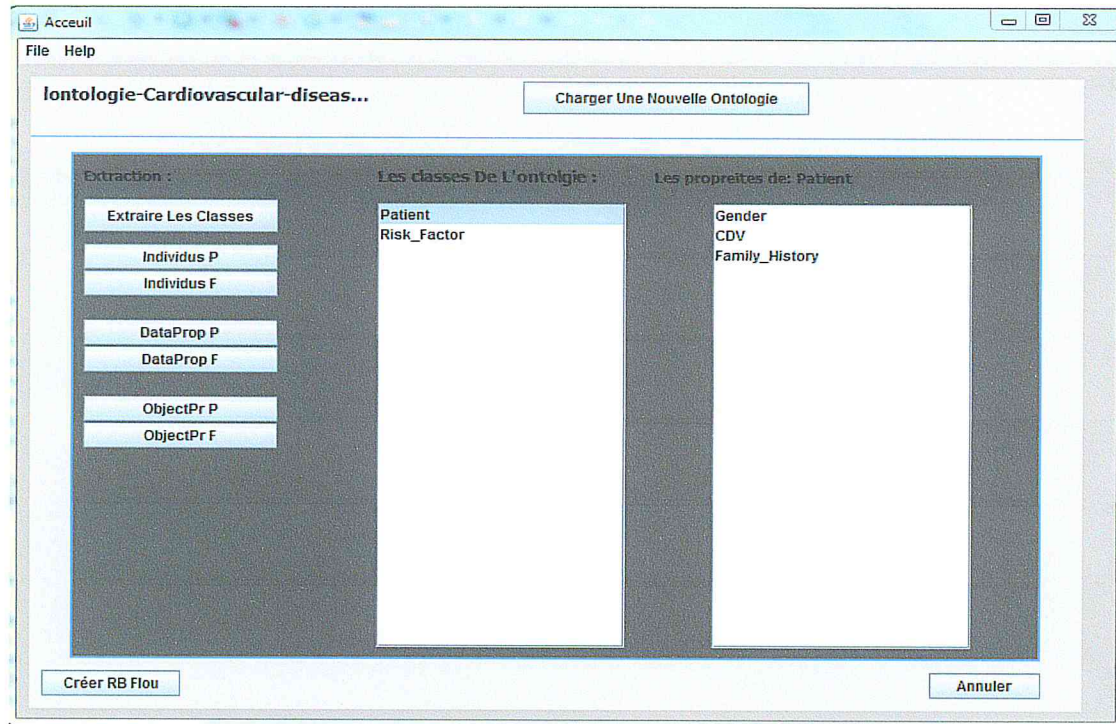


Figure 32 : Extraction des propriétés de données précises pour la classe patient.

3.3. Créer le réseau bayésien flou cardiovasculaire

Il faut tout d'abord donner un titre pour le réseau bayésien flou a créé comme elle montre la figure 33 :

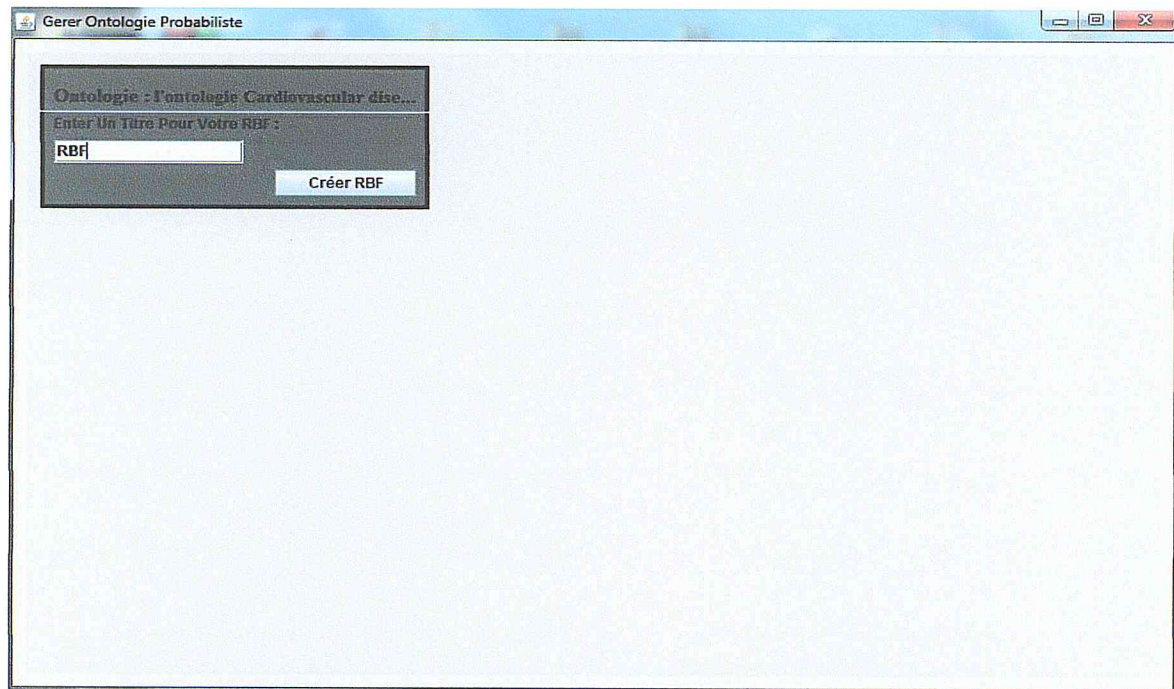


Figure 33 : Donner un titre pour le réseau bayésien.

A. Apprentissage de la structure de réseau bayésien flou

1. Création des nœuds

Dans cette étape il faut créer la structure de notre réseau bayésien flou, et pour cela il faut commencer par créer tous les nœuds, et donnée pour chaque nœud a créé les états relatifs à ce nœud, comme il montre le tableau suivant :

Nœud	États relatifs
Age	Adult / Young / Old
Blood_Pressure	High / Medium / Low
Phesical_Inactivity	High / Medium / Low
Weight	Normal / Under / Over
Is_Diabetic	True / False
Family_History	True / False
Tobacco_Use	True / False
CDV	True / False
Gender	String

Tableau 4 : Les noeuds et ses états relatifs.

CVD: Cardiovascular disease.

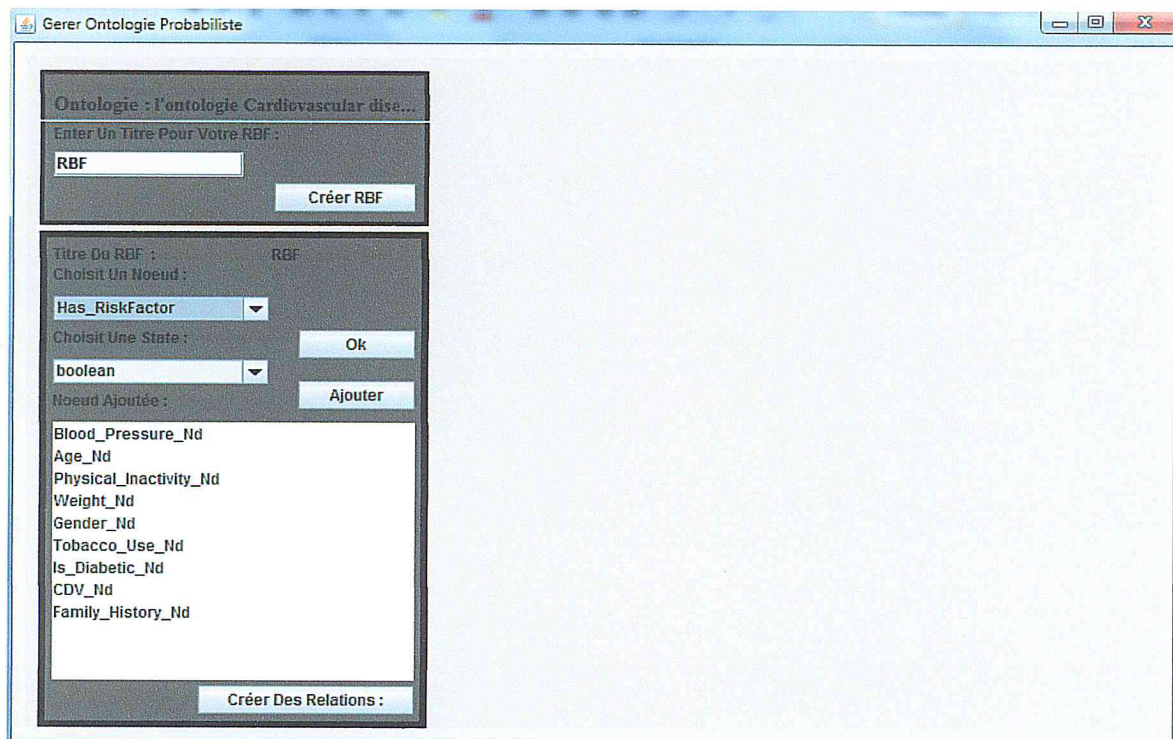


Figure 34 : La création des nœuds.

2. La création des relations

Après la création des nœuds, l'étape suivante sert à définir les relations entre les différents nœuds, comme il montre la figure suivante :

Figure 35 : La création des relations.

B. Apprentissage des paramètres de réseau bayésien flou

Après l'apprentissage de la structure, il reste que l'apprentissage des paramètres pour terminer la création de notre réseau bayésien flou, dans cette étape il suffit de compléter toutes les tables de probabilités comme il montre la figure suivante :

Gerer Ontologie Probabiliste

Ontologie : l'ontologie Cardiovascular dise...

Enter Un Titre Pour Votre RBF :

RBF

Créer RBF

Titre Du RBF : RBF

Choisit Un Noeud :

Has_RiskFactor

Choisit Une State :

boolean

Ok

Noeud Ajoutée :

Blood_Pressure_Nd
Age_Nd
Physical_Inactivity_Nd
Weight_Nd
Gender_Nd
Tobacco_Use_Nd
Is_Diabetic_Nd
CDV_Nd
Family_History_Nd

Créer Des Relations :

Choisit Le Noeud Père :

Family_History_Nd

Choisit Le Noeud Fils :

CDV_Nd

Ajouter La Relation

Pour Remplir Les Tables Des Probabil...
Cliquez Ok :

OK

Liste Des Relations Ajoutée :

Blood_Pressure_Nd -> CDV_Nd
Age_Nd -> CDV_Nd
Physical_Inactivity_Nd -> CDV_Nd
Weight_Nd -> CDV_Nd
Gender_Nd -> CDV_Nd
Tobacco_Use_Nd -> CDV_Nd
Is_Diabetic_Nd -> CDV_Nd
Family_History_Nd -> CDV_Nd

Choisit Un Noeud et Cliquez "Table_Proba" Pour Compléter Sont Table De Probabilité :

Blood_Pressure_Nd

Table_Proba

Compléter La Table De Probabilité. (Il faut la somme des colonnes égale a 1):

Low_BloodPressure	0.3
High_BloodPressure	0.5
Medium_BloodPressure	0.2

Cliquez Valider Afin De Compléter Chaque Table De Probabilité De Chaque Noeud :

Valider

Cliquez Terminé Pour Créer Un RBF a Netica, Et Passer Aux Dernier étape:

Terminé

Figure 36 : Table de probabilité de Blood Pressure.

Après avoir terminé de remplir toutes les tables de probabilité, notre réseau bayésien flou cardiovasculaire est prêt pour être utilisé dans l'étape suivante :

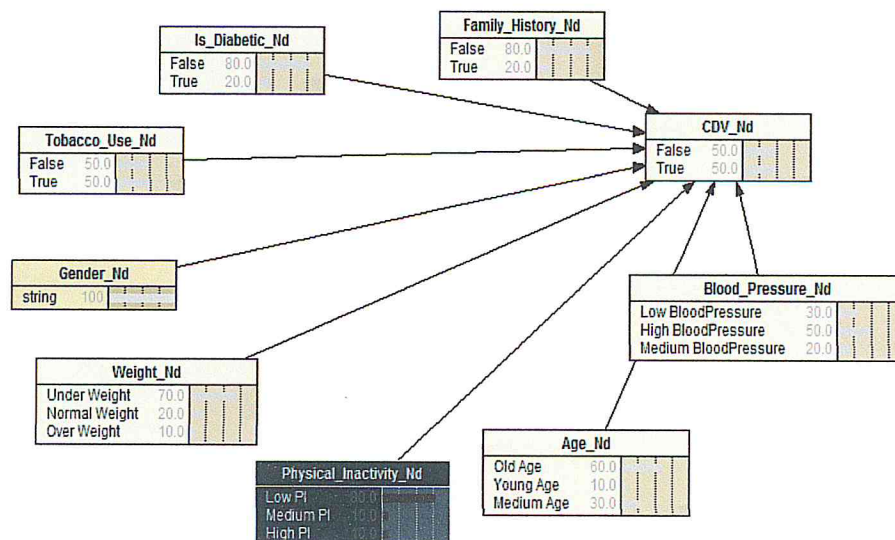


Figure 37 : Réseau byaésien flou cardiovasculaire.

4. Lancement des requêtes

Après l'estimation des distributions des probabilités, notre réseau bayésien flou cardiovasculaire est prêt pour être utilisé dans l'étape suivante :

The screenshot shows a graphical user interface with several input fields and buttons. At the top left, there's a panel titled "Choisir Un Nœud Pour Lance Votre Requ..." with a dropdown menu showing "Family_History_Nd" and an "OK" button. Below it is another panel titled "Choisir Une State :" with a dropdown menu showing "False" and an "Ok" button. To the right of these is a panel titled "Entrer Votre Paramètre :" with a text input field containing the number "26" and an "OK" button. At the bottom, there's a larger panel with a dropdown menu showing "CDV_Nd" and an "OK" button. Below this dropdown is a text box displaying the result: "Resultat : 0.798".

Figure 38 : Lancement des requêtes.

L'utilisateur doit choisir des nœuds évidences et définir ses observations (un paramètre si le nœud choisit est flou ou bien un état si le nœud choisit est précis), après il choisit le nœud requête pour calculer le résultat.

Le résultat obtenu pour notre test résume que le patient a un risque d'une maladie cardiovasculaire avec une certaine probabilité (i.e. nous avons calculé le résultat pour l'état "TRUE").

5. Résultat de la création de l'ontologie probabiliste floue

Après l'effectuation des étapes de la création de l'ontologie probabiliste floue cardiovasculaire nous obtenons un fichier OWL qui porte la méta-ontologie avec ces nouvelles instances de l'ontologie floue probabiliste cardiovasculaire que nous avons obtenues (figure 39).


```

151 <Class IRI="#Membership_Function"/>
152 <NamedIndividual IRI="#Trapezoid"/>
153 </ClassAssertion>
154 <ClassAssertion>
155 <Class IRI="#State"/>
156 <NamedIndividual IRI="#True,False"/>
157 </ClassAssertion>
158 <ObjectPropertyAssertion>
159 <ObjectProperty IRI="#Attached"/>
160 <NamedIndividual IRI="#Blood_Pressure_Nd"/>
161 <NamedIndividual IRI="#Trapezoid"/>
162 </ObjectPropertyAssertion>
163 <ObjectPropertyAssertion>
164 <ObjectProperty IRI="#Parent_Of"/>
165 <NamedIndividual IRI="#Blood_Pressure_Nd"/>
166 <NamedIndividual IRI="#CDV_Nd"/>
167 </ObjectPropertyAssertion>
168 <ObjectPropertyAssertion>
169 <ObjectProperty IRI="#Possesses"/>
170 <NamedIndividual IRI="#Blood_Pressure_Nd"/>
171 <NamedIndividual IRI="#Low_BloodPressure,High_BloodPressure,Medium_BloodPressure"/>
172 </ObjectPropertyAssertion>
173 <ObjectPropertyAssertion>
174 <ObjectProperty IRI="#Is_Son_Of"/>
175 <NamedIndividual IRI="#CDV_Nd"/>
176 <NamedIndividual IRI="#Tobacco_Use_Nd"/>
177 </ObjectPropertyAssertion>
178 <ObjectPropertyAssertion>
179 <ObjectProperty IRI="#Is_Son_Of"/>
180 <NamedIndividual IRI="#CDV_Nd"/>
181 <NamedIndividual IRI="#Blood_Pressure_Nd"/>
182 </ObjectPropertyAssertion>
183 <ObjectPropertyAssertion>
184 <ObjectProperty IRI="#Possesses"/>
185 <NamedIndividual IRI="#CDV_Nd"/>
186 <NamedIndividual IRI="#True,False"/>
187 </ObjectPropertyAssertion>
188 </ObjectPropertyAssertion>

```

Figure 39 : Exemple des différents composants de l'ontologie floue cardiovasculaire .

5.1. Comparaison entre l'ontologie floue et l'ontologie probabiliste floue obtenue

La hiérarchie des classes de l'ontologie floue cardiovasculaire :

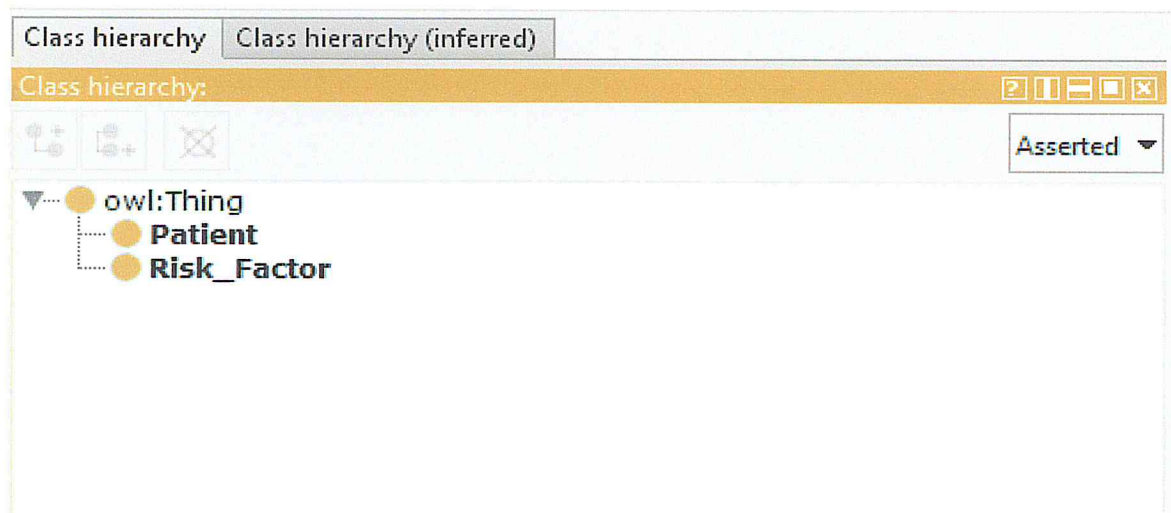


Figure 40 : La hiérarchie des classes de l'ontologie floue cardiovasculaire.

La hiérarchie des classes de l'ontologie floue probabiliste cardiovasculaire :

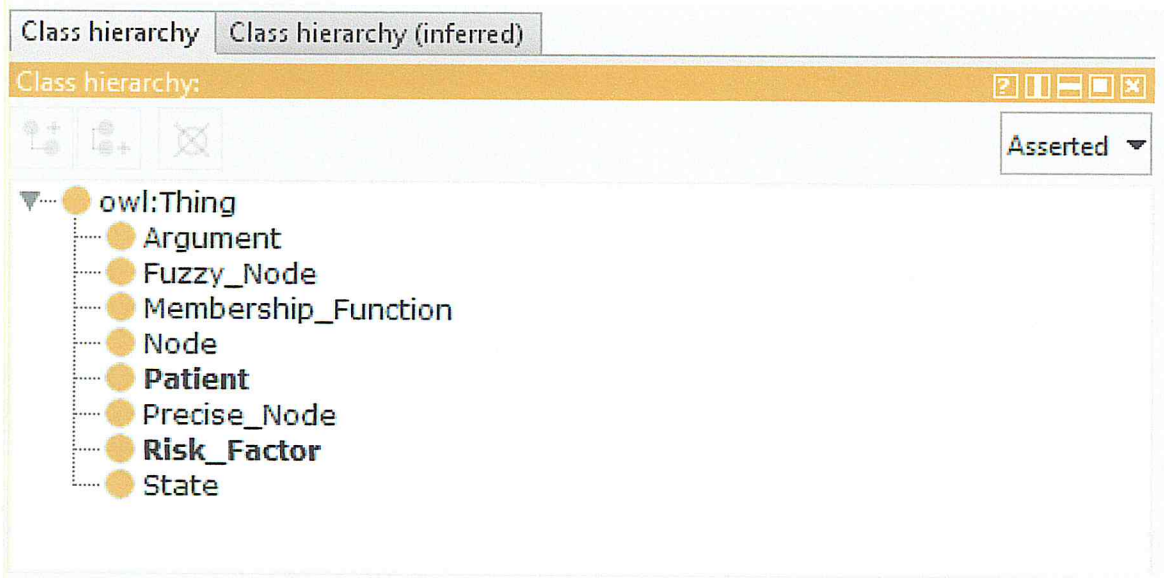


Figure 41 : La hiérarchie des classes de l'ontologie floue probabiliste cardiovasculaire.

La hiérarchie des propriétés d'objet de l'ontologie floue cardiovasculaire :

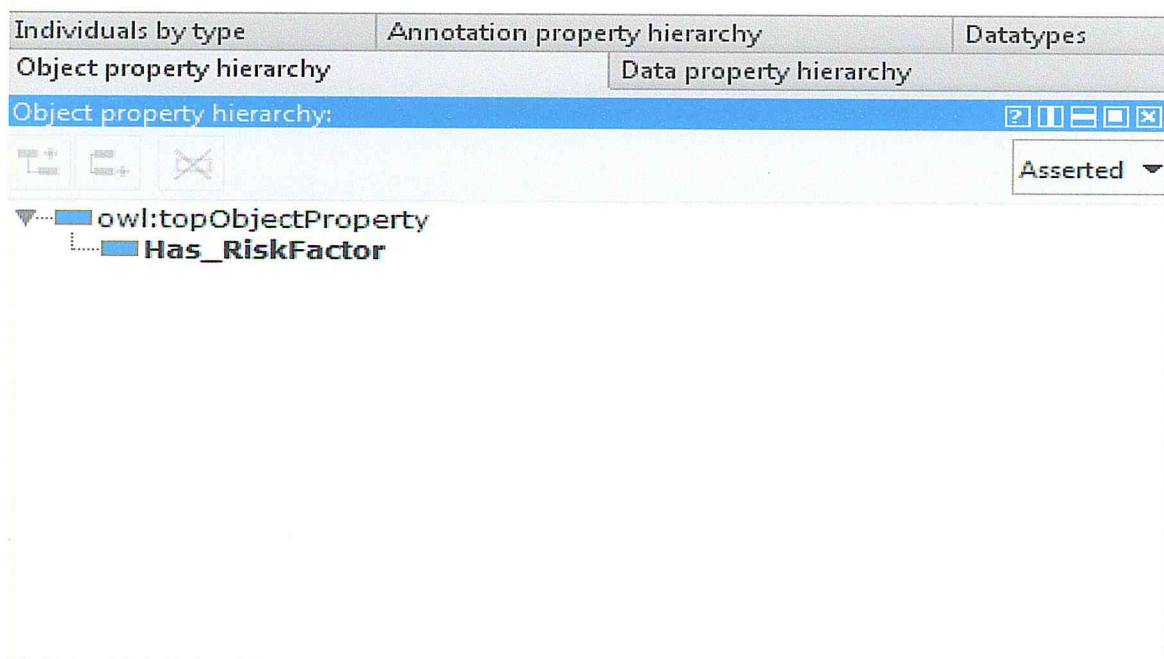


Figure 42 : La hiérarchie des propriétés d'objet de l'ontologie floue cardiovasculaire.

La hiérarchie des propriétés d'objet de l'ontologie floue probabiliste cardiovasculaire :

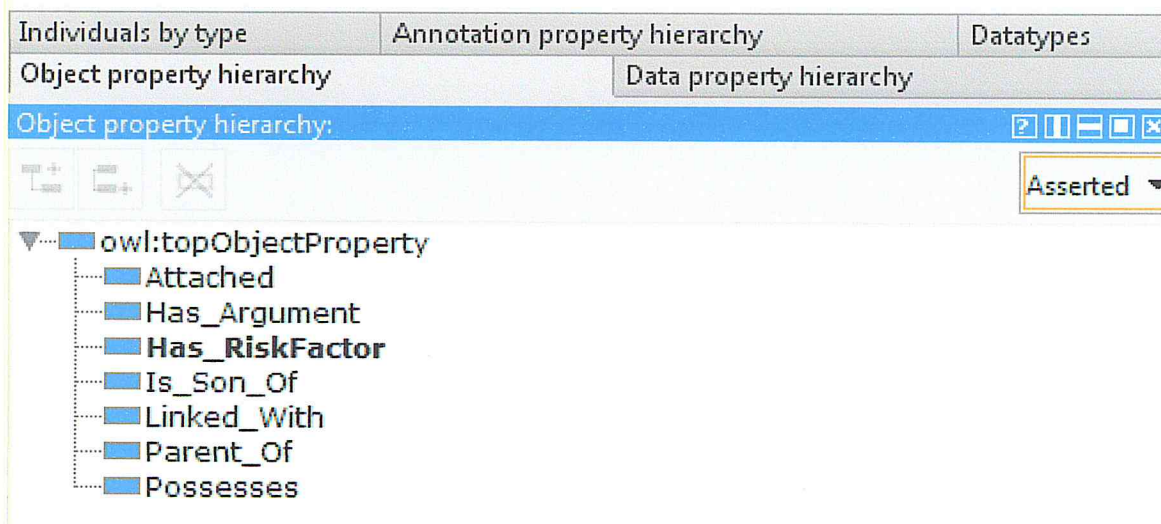


Figure 43 : La hiérarchie des propriétés d'objet de l'ontologie floue probabiliste cardiovasculaire.

La hiérarchie des propriétés de donnée de l'ontologie floue cardiovasculaire :

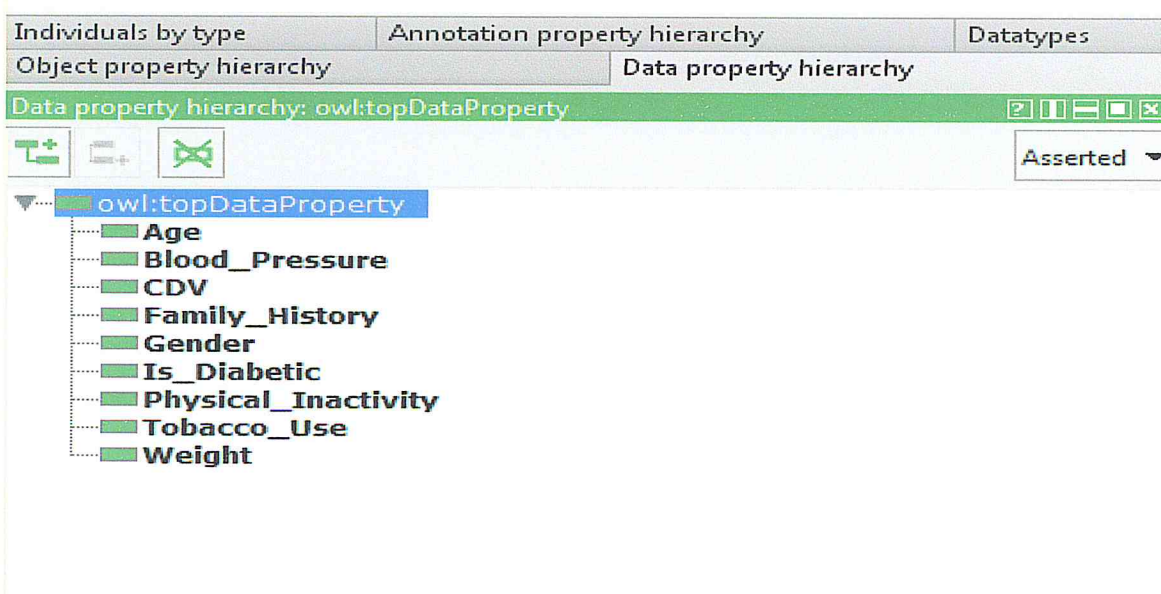


Figure 44 : La hiérarchie des propriétés de donnée de l'ontologie floue cardiovasculaire.

La hiérarchie des propriétés d'objet de l'ontologie floue probabiliste cardiovasculaire :

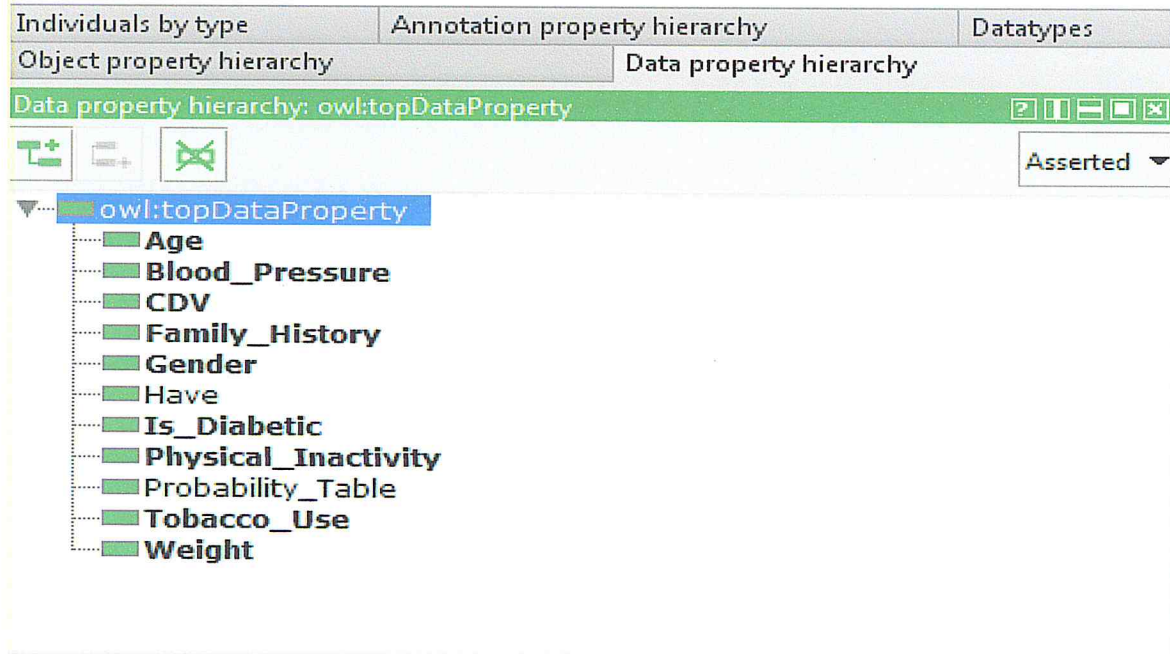


Figure 45 : La hiérarchie des propriétés d'objet de l'ontologie floue probabiliste cardiovasculaire.

Le réseau bayésien flou cardiovasculaire après le lancement du requête :

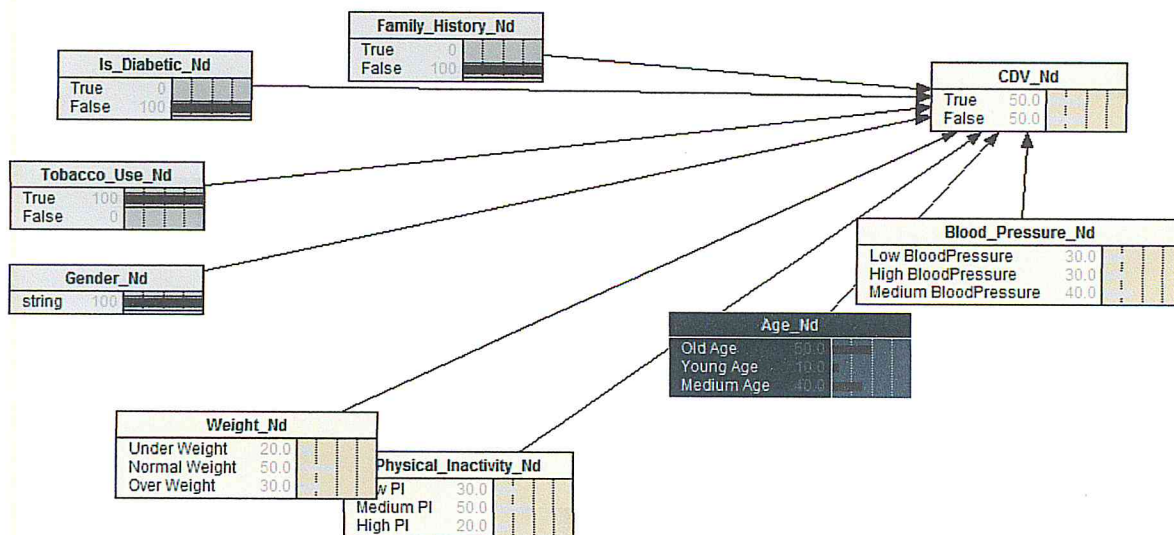
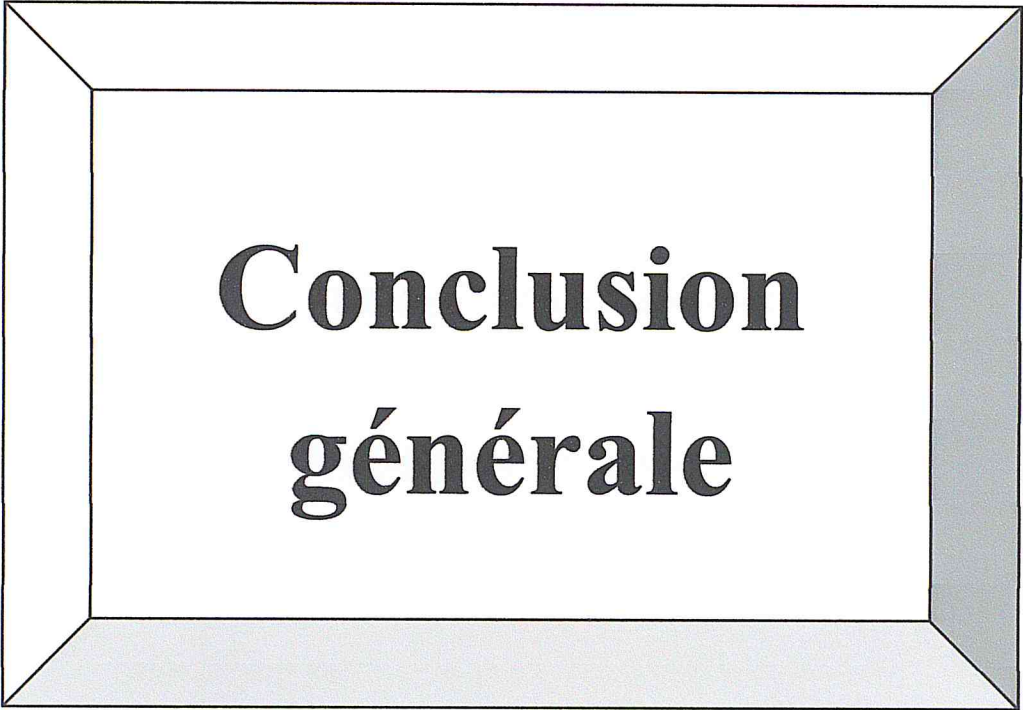


Figure 46 : Le réseau bayésien flou cardiovasculaire après le lancement du requête.

6. Conclusion

Ce chapitre a été consacré aux tests de notre système. En ce qui concerne les objectifs listés dans le chapitre d'introduction, le test permet de vérifier les résultats de l'implémentation en testant la construction du système et répondre au but de notre système qui est l'inférence probabiliste sur les ontologies floues.



Conclusion générale

Conclusion générale

Les ontologies sont une approche puissante pour permettre le partage des connaissances et de soutenir l'interopérabilité entre les personnes et les systèmes informatiques. Cependant, les ontologies traditionnelles (formelles) ne fournissent pas un soutien adéquat à l'incertitude.

Pour faire face à tels problème important. L'objectif du système proposé dans ce projet est de prendre en considération plusieurs types d'incertitudes et /ou d'imperfections à la fois liées aux informations dans le contexte du web sémantique (des connaissances probabilistes et floues à la fois). Pour ce faire, nous allons basées sur les réseaux bayésiens flous et les ontologies floues, donc nous allons transformer une ontologie floue (fuzzy OWL 2) bien formalisée, qui comporte des informations floues codées en utilisant le langage fuzzy owl2, en une ontologie qu'elle peut supporter la représentation des connaissances probabiliste en un RBF.

Il est prouvé que les réseaux bayésiens sont des méthodes fiables pour modéliser l'incertain et pour faire des prédictions en se basant sur des faits réels même en présence d'incertitudes autour de ces faits. Nous avons conclu que les réseaux bayésiens peuvent être appliqués dans le web sémantique afin de résoudre le problème de l'incertitude.

Nous avons proposé pour cette modélisation un processus qui porte des étapes :

- **L'importation de l'ontologie formelle OWL** : importer l'ontologie OWL depuis un emplacement quelconque.
- **Extraction des composants de l'ontologie floue** : L'objectif principal de cette étape est d'extraire chaque composant de notre ontologie floue.
- **Construction de réseau bayésien flou (structure et paramètres)** : La construction de notre réseau bayésien peut se faire d'une manière semi automatique avec l'aide d'experts humains.
- **Intégration de RB-flou dans l'ontologie floue** : Le rôle de cette étape dans notre système est de représenter les connaissances probabilistes d'une manière formelle.
- **Requêtes** : cette étape permet aux utilisateurs d'introduire ces observations.
- **Les résultats de requêtes** : Après l'exécution des requêtes on obtient les résultats voulus de l'inférence probabiliste sur les ontologies floues dont on a besoin pour traiter les données incertaines du domaine étudié.

Nous avons implémenté notre système qui est générique et nous l'avons testé avec une ontologie d'un domaine médicale, ainsi nous avons montré les différentes interfaces

Conclusion générale

nécessaires pour la construction de l'ontologie floue et faire l'inférence probabiliste sur ces ontologies.

**Références
bibliographiques**

Références bibliographiques

- [1] Stoilos, G., Simou, N., Stamou, G., Kollias, S.: Uncertainty and the Semantic Web. *IEEE Intelligent Systems* 21(5), 84–87 (2006).
- [2] Bobillo, F., Delgado, M., Gómez-Romero, J.: DeLorean: A reasoner for fuzzy OWL 1.1. In: *Proc. of the 4th International Workshop on Uncertainty Reasoning for the Semantic Web (URSW 2008)*. *CEUR Workshop Proceedings*, vol. 423 (2008) .
- [3] Bobillo, F., Straccia, U.: fuzzyDL: An expressive fuzzy Description Logic reasoner. In: *Proceedings of the 17th IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2008)*, pp. 923–930. *IEEE Computer Society*, Los Alamitos (2008).
- [4] Guarino, N., & Giaretta, P. Ontologies and Knowledge Bases. Towards a Terminological Clarification. In N. Mars (Ed.), *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing* (pp. 25–32). Amsterdam: IOS Press, 1995.
- [5] Nottelmann H. & Fuhr N. Adding probabilities and rules to OWL Lite subsets based on probabilistic Datalog, *Int. J. Uncertain. Fuzz.* pp 17–42. 2006.
- [6] da Costa P.C.G. et al. . da Costa P.C.G. et Laskey K.B. Laskey K.J., PR-OWL: A Bayesian ontology language for the Semantic Web, in: *Proceedings URSW*, pp. 23–33. 2005.
- [7] Heinsohn J. Probabilistic description logics, in: *Proceedings UAI-1994*, Morgan Kaufmann, pp. 311–318. 1994.
- [8] Pool M. & Aikin J. KEEPER and Protégé: An elicitation environment for Bayesian inference tools, in: *Proceedings of the Workshop on Protégé and Reasoning held at the IPC*. 2004.
- [9] Gruber, T.. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2), 199-220, 1993.
- [10] Borst, W. N. Construction of Engineering Ontologies. Center for Telematica and Information Technology, University of Twente, Enschede, NL, 1997.
- [11] Guarino, N. Formal Ontology and Information Systems. In *Proc. of Formal Ontology and Information Systems*, Trento, Italy. IOS Press, 1998. <http://www.loacnr.it/Papers/FOIS98.pdf>.
- [12] Jasper, R. and Uschold, M. A framework for understanding and classifying ontology applications. *IJCAI-99, Ontology Workshop*, Stockholm, 1999.
- [13] Fürst F. *L'ingénierie Ontologique (02-07)*. Nantes: Institut de Recherche en Informatique de Nantes. 2002.

- [14] Kenneth J. Laskey, et al., 20/02/2016. W3C Uncertainty Reasoning for the World Wide Web Incubator Group [en ligne], Adresse URL: <https://www.w3.org/2005/Incubator/urw3/group/draftReport.html>
- [15] Zongmin M. Ma, Jingwei Cheng, Haiong Wang, and Li Yan. "A Vague Description Logic SI". In: Proceedings of the 17th IEEE International Conference on Fuzzy Systems. Hong Kong, China: IEEE Computer Society, 2008.
- [16] Elie Sanchez and Takahiro Yamanoi. "Fuzzy Logic and the Semantic Web". In: Proceedings of the 10th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems. Perugia, Italy: 2004.
- [17] Jun Zhai, Lixin Shen, Zhou Zhou, and Yan Liang. "Fuzzy Ontology Model for Knowledge Management". In: Proceedings of the International Conference Proceedings on Intelligent Systems and Knowledge Engineering. Advances in Intelligent Systems Research. Chengdu, China: Atlantic, 2007.
- [18] Chingman Au Yeung and Ho fung Leung. "Ontology with Likelihood and Typicality of Objects in Concepts". In: Proceedings of the 25th International Conference on Conceptual Modeling. Volume 4 215. Tucson, AZ, USA: Springer-Verlag, 2006.
- [19] Yannick Loiseau, Henri Prade, and Mohand Boughanem. "Qualitative Pattern Matching with Linguistic Terms". In: AI Communications. 2004.
- [20] Lipika Dey and Muhammad Abulaish. "A Fuzzy Ontology-Based Text Mining System for Knowledge Acquisition, Ontology Enhancement and Query Answering from Bio-Medical Texts". In: Computational Intelligence in Bioinformatics. Wiley Inc., 2007.
- [21] Giorgos Stoilos, Giorgos Stamou, Vassilis Tzouvaras, Jeff Z. Pan, and Ian Horrocks. "Fuzzy OWL: Uncertainty and the Semantic Web". In: Proceedings of the 1st International Workshop on OWL: Experience and Directions. Volume 188. Gaiway, Ireland, 2005.
- [22] B. Motik, P. F. Patel-Schneider, and B. P. (editors). OWL 2 Web Ontology Language structural specification and functional-style syntax, 2009.
- [23] N. Noy, R. W. Ferguson, M A Musen, "The knowledge model of Protege-2000: Combining interoperability and flexibility," in Knowledge Engineering and Knowledge Management Methods, Models, and Tools, 2000, pp. 17 – 32.
- [24] Judea Pearl. Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann, 1988.
- [25] Cooper, G.F. (1990) The computational complexity of probabilistic inference using Bayesian belief networks. Artificial Intelligence, 42, 393-405 marc.bouissou.free.fr
- [26] P. Naïm, P-H. Willemin, P. Leray, O. Pourret, and A. Becker. Réseaux bayésiens. 423 pages, Eyrolles, Paris, 2007

- [27] P. Naïm, P-H. Wuillemin, P. Leray, O. Pourret, and A. Becker. Réseaux bayésiens. 440 pages, Eyrolles, Paris, 2007
- [28] Steffen L. Lauritzen et David J. Spiegelhalter. « Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems ». Journal of the Royal Statistical Society, Series B, 50 :157–224, 1988.
- [29] Frédéric Santos. L'algorithme EM : une courte présentation, CNRS, UMR 5199 PACEA, 2015.
- [30] Ross D. Shachter et Mark A. Peot. « Simulation Approaches to General Probabilistic Inference on Belief Networks ». Dans Proceedings of the Conference on Uncertainty in Artificial Intelligence, pages 221–234, Amsterdam, The Netherlands, 1990.
- [31] Robert Fung et Del B. Favero. « Backward Simulation in Bayesian Networks ». Dans Proceedings of the Conference on Uncertainty in Artificial Intelligence, pages 227–234, San Francisco, CA, USA, july 1994.
- [32] Christopher Fogelberg, Vasile Palade, and Phil Assheton. Belief propagation in fuzzy bayesian networks. In 1st International Workshop on Combinations of Intelligent Methods and Applications (CIMA) at ECAI'08, pages 19–24, 2008.
- [33] H. Tang and S. Liu, “Basic theory of fuzzy Bayesian networks and its application in machinery fault diagnosis,” in Proceedings of the Fourth International Conference on Fuzzy Systems and Knowledge Discovery - Volume 04, ser. FSKD '07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 132–137.
- [34] Paulo C. G. COSTA, Kathryn B. LASKEY. PR-OWL: A Framework for Probabilistic Ontologies The Volgenau School of Information Technology and Engineering, George Mason University, USA, 2006.
- [35] da Costa P.C.G. et al. da Costa P.C.G. et Laskey K.B. Laskey K.J., PR-OWL: A Bayesian ontology language for the Semantic Web, in: Proceedings URSW, pp. 23–33. 2005.
- [36] Yang Y. & Calmet J. Ontobayes: An ontology-driven uncertainty model. In: (IAWTIC'05). IEEE Computer Society (2005) 457–464. 2005.
- [37] Ding Z. & Peng Y. A probabilistic extension to ontology language OWL, in: Proceedings HICSS. 2004.
- [38] Koller D. & Levy A. P-CLASSIC: A tractable probabilistic description logic, in: Proceedings AAAI-1997, AAAI Press/MIT Press, pp. 390–397. 1997.
- [39] Fabio G. et al. Rodrigo B. Polastroa, Fabio G. Cozmana, Felipe I. Takiyamaa, and Kate C. Revoredob, Computing Inferences for Credal ALC Terminologies. 2011.

- [40] Bobillo Fernando, Miguel Delgado, and Juan Gómez-Romero. "A Crisp Representation for Fuzzy *SHOIN*. A with Fuzzy Nominais and General Concept Inclusions". In: Uncertainty Reasoning for the Semantic Web. Volume 5327. Springer Verlag, 2008.
- [41] <https://www.eclipse.org/>. (2017). [online] Available at: https://www.eclipse.org/#sec_ide [Accessed 3 Jun. 2017].
- [42] Research, S. (2017). protégé. [online] Protege.stanford.edu. Available at: <http://protege.stanford.edu/products.php> [Accessed 3 Jun. 2017].
- [43] Porter, B., Zyl, J. and Lamy, O. (2017). Maven – Welcome to Apache Maven. [online] Maven.apache.org. Available at: <https://maven.apache.org/> [Accessed 3 Jun. 2017].
- [44] Owlapi.sourceforge.net. (2017). OWL API. [online] Available at: <http://owlapi.sourceforge.net/> [Accessed 3 Jun. 2017].
- [45] Bobillo, F. (2017). FuzzyOWL2: Fuzzy Ontology Representation using OWL 2. [online] Umbertostraccia.it. Available at: <http://www.umbertostraccia.it/cs/software/FuzzyOWL/> [Accessed 3 Jun. 2017].
- [46] Patrick Naïm, Pierre-Henri Wuillemin, Philippe Leray, Olivier Pourret, Anna Becker. Réseaux Bayésiens, Eyrolles 2008.
- [47] World Health Organization. (2017). Cardiovascular diseases (CVDs). [online] Available at: <http://www.who.int/mediacentre/factsheets/fs317/en/> [Accessed 3 Jun. 2017].

