

UNIVERSITÉ SAAD DAHLEB DE BLIDA1

Faculté des sciences

Département d'informatique



MEMOIRE DE MASTER

En Informatique

Option : Ingénierie Des Logiciels

THÈME :

**Vers un framework générique pour
l'intégration de la connaissance
probabiliste dans des ontologies
floues.**

Réalisé par
AOUFI Sara
BOUDAA Radia

Encadré par
Dr. I. RIALI
Dr. M. FAREH

Présidente Mezzi
Examineur Ferfera

Septembre 2022

Remerciements

Ce travail est le résultat de nos années de travail acharné, de patience et de courage. Premièrement, nous remercions Dieu de nous avoir donné la santé, la force, le courage, et la volonté de terminer ce mémoire.

Nous exprimons tous nos remerciements et notre respect à notre promoteur **M. RIALI** et à notre Co-promoteur **Mme. FAREH**, d'avoir accepté de nous encadrer et pour leurs orientations durant notre travail.

Nous remercions également tous les membres de jury de nous avoir honorés de leur présence et d'avoir accepté d'évaluer notre travail.

Enfin, nous adressons nos plus sincères remerciements à nos familles et tous nos proches et amis qui nous ont aidés, soutenus et encouragés tout au long de la réalisation de ce travail.

Résumé

Les technologies du Web sémantique, en particulier les ontologies, ont reçu une attention particulière ces dernières années pour leur capacité à faciliter la gestion et le partage des connaissances, et à faciliter l'interopérabilité sémantique entre les systèmes d'information. Cependant, les ontologies classiques sont considérées comme inefficaces lorsqu'il existe une incertitude dans le domaine à traiter.

Dans ce contexte, les ontologies probabilistes et les ontologies basées sur la logique floue ont été développées pour permettre la modélisation et le raisonnement avec l'incertitude.

De ce fait, l'objectif de notre travail est de concevoir un système qui utilise à la fois des ontologies probabilistes et floues pour mieux représenter sémantiquement les connaissances incertaines. Une des solutions pour traiter les deux types d'incertitudes (l'aléatoire et l'impression) simultanément est d'enrichir la sémantique d'une ontologie floue par des connaissances probabilistes en utilisant un réseau bayésien flou.

Mots clés :

Web Sémantique, Connaissance Incertaine, Ontologie Floue, Réseau Bayésien Flou.

Abstract

Semantic Web technologies, in particular ontologies, have received much attention in recent years for their ability to facilitate knowledge management and sharing, and to facilitate semantic interoperability between information systems. However, classical ontologies are considered inefficient when there is uncertainty in the domain to be treated.

In this context, probabilistic ontologies and ontologies based on fuzzy logic have been developed to allow modeling and reasoning with uncertainty.

Therefore, the objective of our work is to design a system that uses both probabilistic and fuzzy ontologies to better semantically represent uncertain knowledge. One of the solutions to deal with both types of uncertainties (randomness and impression) simultaneously is to enrich the semantics of a fuzzy ontology with probabilistic knowledge using a fuzzy Bayesian network.

Key words :

Semantic Web, Uncertain Knowledge, Fuzzy Ontology, Fuzzy Bayesian Network.

الملخص :

حظيت تقنيات الويب الدلالية ، ولا سيما الأنطولوجيا ، باهتمام كبير في السنوات الأخيرة لقدرتها على تسهيل إدارة المعرفة وتبادلها ، وتسهيل التشغيل البيئي الدلالي بين أنظمة المعلومات. ومع ذلك ، تعتبر الأنطولوجيا الكلاسيكية غير فعالة عندما يكون هناك عدم يقين في المجال المراد معالجته.

في هذا السياق ، تم تطوير الأنطولوجيا الاحتمالية والأنطولوجيا القائمة على المنطق الضبابي للسماح بالتمنجة والتفكير مع عدم اليقين.

لذلك ، فإن الهدف من عملنا هو تصميم نظام يستخدم كل من الأنطولوجيا الاحتمالية والغامضة لتمثيل المعرفة غير المؤكدة بشكل أفضل. أحد الحلول للتعامل مع كلا النوعين من أوجه عدم اليقين في وقت واحد هو إثراء دلالات الأنطولوجيا الغامضة بالمعرفة الاحتمالية باستخدام شبكة بايزية غامضة.

الكلمات المفتاحية :

الويب الدلالي ، المعرفة غير المؤكدة ، الأنطولوجيا الضبابية ، شبكة بايزية الضبابية.

Table des matières

Table des figures	viii
Liste des tableaux	x
Introduction générale	1
1 ONTOLOGIES CLASSIQUES ET ONTOLOGIES FLOUES	4
1.1 Introduction	4
1.2 Ontologies classiques	4
1.2.1 Définition	4
1.2.2 Composants d'une ontologie	5
1.2.3 Les avantages des ontologies	6
1.2.4 Cycle de vie d'une ontologie	6
1.2.5 Langages de représentation des ontologies	7
1.2.6 La représentation des connaissances à l'aide des logiques de description	8
1.2.7 Les limites des ontologies	8
1.3 Incertitude dans les ontologies	8
1.3.1 Définition	8
1.3.2 Classification de l'incertitude par le W3C	9
1.4 La logique floue	10
1.4.1 Définition	10
1.4.2 La Différence entre la logique booléenne et la logique floue	10
1.4.3 Concepts et Fondements théoriques	11
1.5 Les ontologies floues	14
1.5.1 Définition	14
1.5.2 Les composants d'une ontologie floue	14
1.6 Méthode de développement des ontologies floues	17
1.7 Conclusion	18

2	RÉSEAU BAYÉSIEN	19
2.1	Introduction	19
2.2	Réseaux Bayésiens	19
2.2.1	Définition	19
2.2.2	Construction d'un réseau bayésien	20
2.2.3	Inférence dans les réseaux bayésiens	21
2.3	Réseaux Bayésiens flous	22
2.4	Exemple d'un réseau bayésien flou	22
2.4.1	Méthode de pondération	24
2.4.2	Distribution probabiliste floue	25
2.4.3	Méthode de l'évidence virtuelle	26
2.5	L'ontologie probabiliste	26
2.5.1	Définition	27
2.5.2	Les langages des ontologies probabilistes	27
2.6	Travaux connexes	29
2.7	Analyse et critique	30
2.8	Conclusion	31
3	CONCEPTION DU SYSTÈME	32
3.1	Introduction	32
3.2	Motivation	32
3.3	Dataset	33
3.4	Construction de l'ontologie floue pour la maladie cardiovasculaire	34
3.5	Aperçu de notre Système	39
3.5.1	Charger l'ontologie floue	39
3.5.2	Extraction des composantes	40
3.5.3	Construction de Réseau Bayésien Flou	41
3.5.4	Intégration des connaissances probabilistes dans l'ontologie floue	43
3.5.5	Ontologie floue et probabiliste	46
3.6	Conclusion	47
4	IMPLEMENTATION ET TEST DU SYSTÈME	48
4.1	Introduction	48
4.2	Implémentation de Notre Système	48
4.2.1	Outil de développement	48
4.2.1.1	JAVA	48
4.2.1.2	Protégé	49
4.2.1.3	Netbeans	49
4.2.1.4	OWL API	50
4.2.1.5	SMILE	50

4.2.1.6	GeNie	50
4.2.1.7	FUZZY OWL 2	51
4.2.2	Les interfaces graphiques	52
4.2.2.1	L'interface d'accueil	52
4.2.2.2	L'interface extraction des composants de l'ontologie floue	53
4.2.2.3	L'interface créer réseau bayésien flou	54
4.3	Validation du système	56
4.3.1	Mesures d'évaluation	56
4.3.2	Résultats obtenus	57
4.4	Conclusion	61
	Conclusion générale	62
	Bibliographie	63

Table des figures

1.1	Cycle de vie d'une ontologie [1].	6
1.2	Le niveau supérieur de l'ontologie d'incertitude développée par URW3-XG [2].	9
1.3	(A) Logique booléenne (B) Logique floue.	11
1.4	Variable linguistique "Age".	12
1.5	Représentation du sous-ensemble flou Z.	13
1.6	La différence entre l'ensemble classique et l'ensemble flou graphiquement.	13
1.7	Fonction d'appartenance du concept flou <Employé-Jeune>.	15
1.8	Exemple d'une ontologie floue.	16
1.9	La fonction d'appartenance de l'ensemble flou A <Compétence-Maitrisée> et B <Compétence-Parfaitement-Maitrisée >.	16
2.1	Réseau bayésien simple.	23
2.2	Fonctions d'appartenance pour le réseau.	24
2.3	Le Réseau Bayésien avec un nœud de l'évidence virtuelle.	26
2.4	Composition d'un MTheory de PR-OWL [3].	28
3.1	L'ontologie floue de la maladie cardiovasculaire.	38
3.2	Les classes, les propriétés d'objets, les propriétés de données de l'ontologie floue.	38
3.3	Schéma global du système.	39
3.4	Exemple d'une annotation "Fuzzy Label".	40
3.5	Les composantes de l'ontologie floue.	41
3.6	La structure du réseau bayésien.	42
3.7	Le réseau bayésien après l'apprentissage des paramètres.	43
3.8	Méta-ontologie UOFBN.	44
3.9	Les composants de UOFBN.	45
3.10	Exemple des instances dans le méta-ontologie.	46
3.11	Les instances de la classe PriorTab.	46
3.12	La hiérarchie des classes de l'ontologie floue probabiliste.	47
4.1	La page de départ de protégé.	49

4.2	La page d'accueil de Netbeans.	50
4.3	la page principale de GeNie.	51
4.4	La fenêtre de Fuzzy OWL 2.	51
4.5	L'interface d'accueil.	52
4.6	Choisir le fichier OWL.	53
4.7	L'interface d'extraction des composants de l'ontologie floue.	54
4.8	L'interface de création de réseau bayésien flou.	55
4.9	Les classes, les propriétés d'objets, les propriétés de données de l'ontologie floue probabiliste.	56
4.10	la table de confusion.	57
4.11	Histogramme de l'accuracy.	59
4.12	Histogramme de rappel.	60
4.13	Histogramme de précision.	61

Liste des tableaux

2.1	Tableau de probabilité conditionnelle pour le nœud de court-circuit de la résistance.	23
2.2	Tableau de probabilité conditionnelle pour le nœud de Test de Courant.	23
2.3	Tableau de comparaison entre les travaux existants.	30
3.1	Significations, unités de mesure et intervalles de chaque caractéristique du dataset. .	34
3.2	Les informations floues et précises.	36
3.3	Les concepts flous.	37
4.1	Tableau de comparaison des 'Accuracy' (%) obtenues par chaque classifieur.	58
4.2	Tableau de comparaison des 'Rappels' (%) obtenues par chaque classifieur.	59
4.3	Tableau de comparaison des 'Précisions' (%) obtenues par chaque classifieur.	60

Introduction générale

Contexte

Les technologies du Web sémantique notamment les ontologies ont reçu une importance très remarquable durant les dernières années, elles sont capables de faciliter la gestion et le partage des connaissances et de promouvoir l'interopérabilité sémantique entre les systèmes d'information. Néanmoins, les ontologies classiques sont considérées inefficaces quand le domaine à traiter présente de l'incertitude.

Dans ce contexte, les ontologies probabilistes et celles basées sur la logique floue ont été développées pour permettre la modélisation et le raisonnement avec l'incertitude.

Ces ontologies ont démontré leur efficacité dans de multiples domaines tels que l'aide à la décision, domaine médical, etc. Alors, notre étude de l'état de l'art dans les domaines de la gestion de l'incertitude et des ontologies mettent en évidence que les connaissances probabilistes et les connaissances floues sont traitées séparément dans les ontologies.

Cependant, dans les problèmes du monde réel, ces deux types de connaissances se produisent simultanément et sont souvent interconnectés et complémentaires.

Pour faire face à ce problème, il semble nécessaire de développer des systèmes qui utilisent les ontologies probabilistes et les ontologies floues simultanément pour mieux quantifier et représenter sémantiquement la connaissance incertaine. Afin de bien présenter notre travail, nous avons réalisé une étude de cas portant sur la maladie cardiovasculaire.

Problématique

En raison des limites des l'ontologies classiques, qui ne peuvent traiter les informations ambiguës, l'ontologie floue s'est imposée dans tous les domaines. L'ontologie probabiliste, quant à elle, représente les connaissances probabilistes construites sur des réseaux bayésiens. Nous observons que la majorité des solutions proposées traitent les connaissances probabilistes et floues de manière indépendante. Or, dans le monde réel, ces deux types de connaissances sont fréquemment complémentaires.

Cela conduit à poser les questions suivantes :

1. Comment traiter simultanément les connaissances probabilistes et les connaissances floues, en utilisant l'ontologie floue et le réseau bayésien flou ?
2. Comment intégrer les connaissances probabilistes dans l'ontologie floue ?

Objectif

L'objectif de notre projet est de créer un système qui utilise simultanément des ontologies probabilistes et floues pour améliorer la représentation sémantique des connaissances incertaines. Une façon de traiter les deux types d'incertitude en même temps est d'utiliser un réseau bayésien flou pour ajouter des connaissances probabilistes à la sémantique d'une ontologie floue. Nos objectifs portent sur :

- La modélisation de la connaissance probabiliste sous forme d'un RB en se basant sur les composants de l'ontologie floue.
- La modélisation du RB sémantiquement à l'aide d'un méta-modèle.
- L'intégration du RB construit dans l'ontologie floue pour enrichir sa sémantique.

Organisation du mémoire

Afin d'atteindre le but de notre travail, l'organisation de notre mémoire sera comme suit :

La première partie : Se compose de deux chapitres qui présentent le contexte du travail, les ontologies classiques et floues, et les réseaux bayésiens classiques et flous.

- **Chapitre 1 : ONTOLOGIES CLASSIQUES ET ONTOLOGIES FLOUES :** Ce chapitre propose des informations sur les ontologies classiques et les ontologies floues avec leur définition et composantes, l'incertitude dans les ontologies et la logique floue.
- **Chapitre 2 : RÉSEAU BAYÉSIEN :** Ce chapitre donne un aperçu sur les réseaux bayésiens classiques et flous, les ontologies probabilistes et les travaux connexes.

La deuxième partie : En suivant les grandes lignes décrites dans la première partie. La conception et le développement du système font l'objet de cette partie.

- **Chapitre 3 : CONCEPTION DU SYSTÈME :** Ce chapitre décrit la motivation de notre système et notre étude de cas. Puis les étapes de la construction de notre ontologie floue et enfin le fonctionnement détaillé de notre système par un schéma global.
- **Chapitre 4 : IMPLÉMENTATION ET TEST DU SYSTÈME :** Dans ce chapitre, nous allons présenter le résultat de la conception pour la réalisation du système, et des captures d'écrans des interfaces pour le présenter.

Ce mémoire sera clôturé par une conclusion générale reprenant les principaux points abordés.

ONTOLOGIES CLASSIQUES ET ONTOLOGIES FLOUES

1.1 Introduction

Les ontologies font partie de la norme W3C pour le web sémantique, dans laquelle elles sont utilisées pour spécifier des vocabulaires conceptuels standards pour échanger des données entre systèmes, fournir des bases de connaissances réutilisables et faciliter l'interopérabilité entre plusieurs systèmes et bases de données hétérogènes. Cependant, les ontologies classiques ne permettent pas de traiter les informations vagues que l'on trouve couramment dans de nombreux domaines d'application. Une solution réalisable est d'importer la capacité floue pour étendre les ontologies classiques.

Dans ce chapitre nous allons présenter la notion d'ontologie, ses composants, leur cycle de vie et leurs avantages. Par ailleurs, nous présentons certaines de ses limites pour la représentation de la connaissance incertaine et nous introduisons la notion d'incertitude. Ensuite nous décrivons la logique floue et les ontologies floues.

1.2 Ontologies classiques

Dans cette section, nous présenterons l'ontologie classique avec ses différents composants, ses avantages, son cycle de vie, ses langages de représentation, et la limite par rapport à la représentation de la connaissance incertaine.

1.2.1 Définition

L'ontologie est une branche de la philosophie qui concerne l'étude de l'être en tant qu'être et de ses propriétés. Parmi plusieurs définitions proposées qui expliquent l'ontologie dans le contexte de l'informatique, nous citons la suivante : « *Une ontologie est une spécification formelle et explicite d'une conceptualisation partagée* ». Cette définition est issue de deux définitions

largement adoptées, fournies par Tom Gruber et Willem Borst. A notre avis la définition préétablie est importante car elle contient les aspects clés suivants des ontologies [4] :

spécification formelle : l'ontologie doit être traduite dans un langage interprétable par une machine.

explicite : les concepts, les relations et les autres éléments de l'ontologie sont définis de manière déclarative.

conceptualisation : un modèle abstrait composé de concepts liés à la description du monde réel.

partagée : cela signifie qu'une ontologie capture la connaissance consensuelle, c'est à dire qu'elle n'est pas réservée à quelque individus, mais partagée par un groupe ou une communauté d'acteurs.

1.2.2 Composants d'une ontologie

Les connaissances traduites par une ontologie sont à véhiculer à l'aide des éléments suivants [5] : Concepts, Relations, Fonctions, Axiomes, Instances.

- **Les concepts** : Un concept peut représenter un objet, une notion, une idée. Un concept peut être divisé en trois parties : un terme (ou plusieurs), une notion et un ensemble d'objets. Le terme est un élément lexical qui permet d'exprimer le concept en langue naturelle, il peut admettre des synonymes. La notion également appelée intension du concept, contient la sémantique du concept, exprimée en termes de propriétés et attributs, et de contraintes. L'ensemble d'objets appelé extension du concept, regroupe les objets manipulés à travers le concept ; ces objets sont appelés instances du concept.
- **Les relations** : Une relation permet de lier des instances de concepts, ou des concepts génériques. Elles sont caractérisées par un terme (voire plusieurs) et une signature qui précise le nombre d'instances de concepts que la relation lie, leurs types et l'ordre des concepts, c'est-à-dire la façon dont la relation doit être lue. Par exemple, la relation « écrit » lie une instance du concept « personne » et une instance du concept « texte », dans cet ordre.
- **Les fonctions** : Les fonctions constituent des cas particuliers de relations, dans laquelle un élément de la relation, le nième (extrant) est défini en fonction des n-1 éléments précédents (intrants).
- **Les axiomes** : Les axiomes constituent des assertions acceptées comme vraies à propos des abstractions du domaine traduit par l'ontologie. Interviennent dans la définition des concepts ou des relations, dans l'inférence de nouvelles informations, etc.
- **Les instances** : Les instances constituent la définition extensionnelle de l'ontologie, ces objets véhiculent les connaissances (statiques, factuelles) à propos du domaine du problème.

1.2.3 Les avantages des ontologies

Parmi les avantages des ontologies :

- Une compréhension partagée de la structure de l'information entre les personnes ou les fabricants de logiciels.
- Assurer l'interopérabilité entre les systèmes, permettant l'échange de connaissances entre les systèmes.
- Permettre la réutilisation des connaissances du domaine : créer et maintenir des bases de connaissances réutilisables.
- Une aide à la communication entre agents humains et aussi entre organisations.
- Ils peuvent permettre des inférences.
- Amélioration des recherches.

1.2.4 Cycle de vie d'une ontologie

Les ontologies doivent être considérées comme des objets techniques évolutifs et possédants un cycle de vie qui nécessite d'être précisé [1].

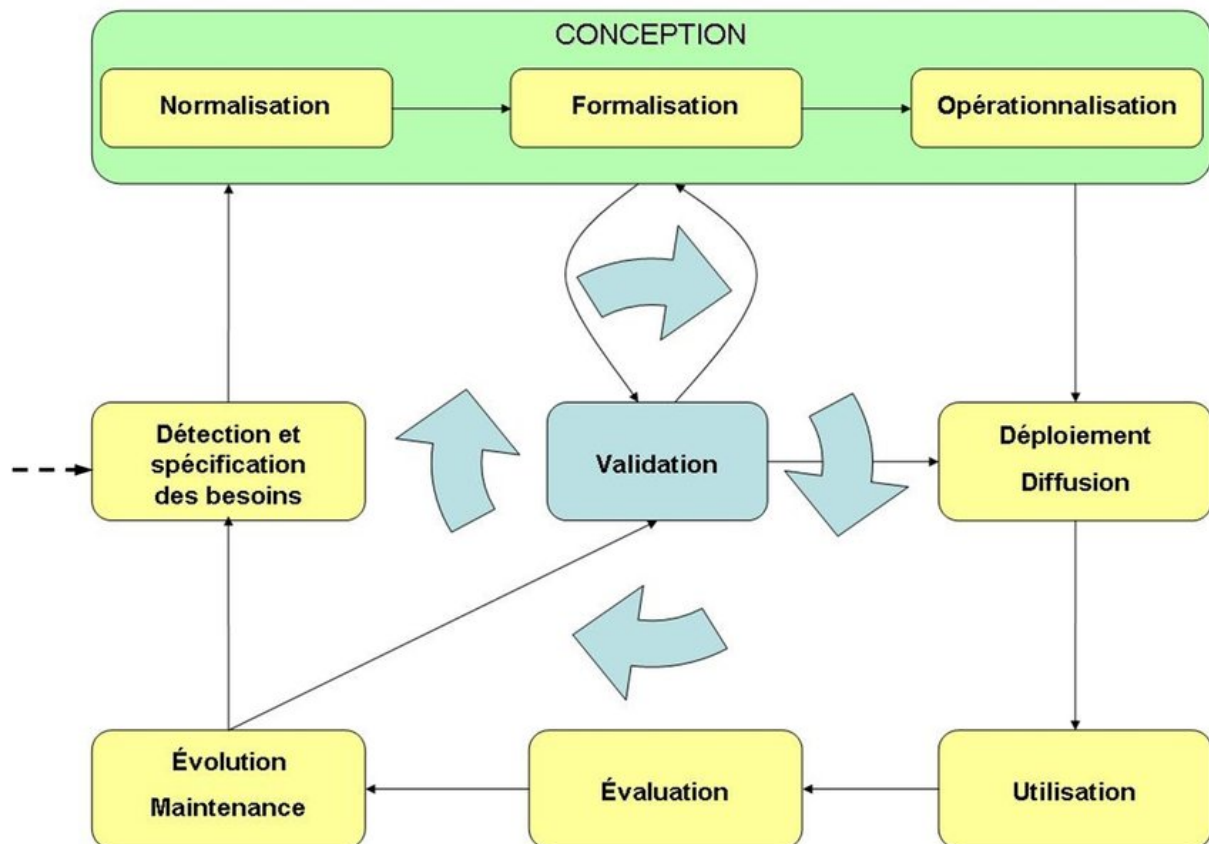


FIGURE 1.1 – Cycle de vie d'une ontologie [1].

Un cycle de vie, voir la figure 1.1, il comporte une étape initiale de détection et de spécification des besoins, qui permet notamment de circonscrire précisément le domaine de connaissances, une étape de conception qui se subdivise en trois phases (Normalisation, Formalisation, Opérationnalisation), une étape de déploiement et de diffusion, une étape d'utilisation et d'évaluation. Après chaque utilisation significative l'ontologie et les besoins doivent être réévalués et l'ontologie peut être étendue et si nécessaire en partie reconstruite. Enfin une sixième étape consacrée à l'évolution et à la maintenance du modèle. La validation du modèle est au centre du processus et se fait de manière itérative.

1.2.5 Langages de représentation des ontologies

— RDF Schéma :

RDF Schéma (ou RDFS) est un langage extensible permettant la représentation des connaissances et la structuration des documents écrits en RDF. Il utilise un vocabulaire pour représenter des ressources RDF. Il détermine une sémantique de généralisation/hiéarchisation formée de concepts et de leurs propriétés, et permet de définir des triplets de graphes RDF. Il caractérise une des bases du Web sémantique puisqu'il permet de représenter des concepts liés à d'autres concepts définis et partagés sur le Web. par la suite nous présentons RDF.

RDF : RDF (Resource Description Framework) a été développé par W3C, il caractérise un modèle standard permettant l'échange de données sur le Web. Il assure également la structuration des ressources ainsi que leurs métadonnées. Principalement, il permet de faire l'annotation sémantique des documents non structurés à travers un modèle de graphe. En plus, ce langage est lié directement au Web sémantique en utilisant la syntaxe XML (eXtended Markup Language). Il est utilisé pour exprimer formellement les liaisons entre les ressources sous la forme d'un triplet <Sujet, Prédicat, Objet>. Le sujet et l'objet définissent les ressources à décrire, et le prédicat représente une propriété applicable à une ressource. Les composants de ces triplets peuvent être des URIs (Universal Resource Identifiers), des variables ou des littéraux.

— OWL :

OWL (Web Ontology Language) est un langage universel standardisé par le W3C et basé sur la syntaxe RDF/XML. La particularité de ce langage par rapport à RDF et RDFS réside dans sa capacité à insérer des outils de comparaison de propriétés et de classes, ainsi que des relations sémantiques entre ces classes, telles que l'équivalence, la disjonction, la cardinalité, la symétrie, etc. En se basant sur le niveau d'expressivité et en fonction des besoins de représentation ou d'inférence, trois sous-langages ont été proposés par W3C avec une expressivité croissante, OWL-Lite, OWL DL et OWL Full.

1.2.6 La représentation des connaissances à l'aide des logiques de description

Les logiques de description (LD) sont une famille de langages de représentation de connaissance qui peuvent être utilisés pour représenter les connaissances terminologiques d'un domaine d'application d'une manière formelle et structurée. Le nom de « logique de description » se rapporte, d'une part à la description des concepts utilisés pour décrire un domaine et d'autre part à la sémantique fondée sur la logique qui peut être donnée par une transcription en logique des prédicats du premier ordre (LPO) [6].

1.2.7 Les limites des ontologies

L'une des principales limites des formalismes ontologiques classiques est l'absence de support cohérent pour l'incertitude et l'imprécision. En fait, les ontologies classiques actuellement conçues ne contiennent que des concepts et des relations qui décrivent des faits affirmés sur le monde. Cela signifie que l'ontologie elle-même n'est pas incertaine par nature, mais présente plutôt un modèle a priori du monde, qui doit être considéré comme vrai par ses utilisateurs.

Il n'existe pas actuellement de mécanisme standard pour la description et l'échange interopérable d'informations incertaines sur des ontologies et encore moins de standard pour raisonner sur ce type d'information. Cependant, de nombreux chercheurs tentent actuellement d'améliorer les capacités ontologiques dans ce domaine.

En fait, l'exigence de représentation a conduit le World Wide Web Consortium (W3C) à créer le groupe d'incubation Uncertainty Reasoning for the World Wide Web en 2007. Il a remis son rapport final en mars 2008, qui exposait principalement les différents défis du raisonnement et les divers cas d'utilisation de la nécessité de la représentation de l'incertitude dans le Web sémantique [7].

1.3 Incertitude dans les ontologies

Avec l'émergence d'Internet et du Web en particulier, l'accès à l'information est de moins en moins restreint. Cependant, la fiabilité de ces informations n'est pas toujours garantie. Dans cette section, nous allons introduire la notion d'incertitude.

1.3.1 Définition

Le terme "incertitude" tel que le définit le dictionnaire français Larousse désigne une information qui n'est pas établie avec exactitude, qui peut ou non se produire et qui peut être de nature vague. Le but sera alors d'attribuer une valeur de probabilité ou une possibilité à une proposition pour la rendre vraie ou fausse.

1.3.2 Classification de l'incertitude par le W3C

URW3-XG (Uncertainty Reasoning for the World Wide Web Incubator Group) [2] a développé une ontologie simple pour l'incertitude. De plus, cette ontologie a été utilisée pour classer les cas d'utilisation développés par ce groupe dans le but d'obtenir une couverture relativement complète des fonctionnalités liées au raisonnement d'incertitude sur les informations disponibles sur le World Wide Web.

Selon l'ontologie, comme il est illustré dans la figure 1.2, l'incertitude est associée à des phrases qui font des assertions sur le monde, et qui sont affirmées par des agents (humains ou informatiques) [2].

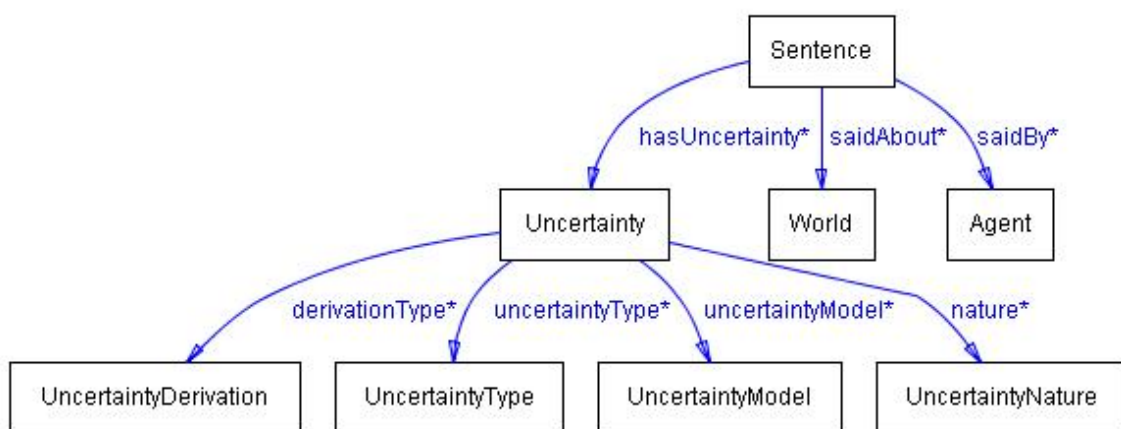


FIGURE 1.2 – Le niveau supérieur de l'ontologie d'incertitude développée par URW3-XG [2].

La dérivation de l'incertitude peut être objective (via un processus formel et répétable) ou subjective (jugement ou supposition).

Les types d'incertitudes comprennent :

- **Ambiguïté** : Les référents des termes dans une phrase sur le monde ne sont pas clairement spécifiés et il est donc impossible de déterminer si la phrase est satisfaite.
- **Empirique** : une phrase concernant un événement est satisfaite ou non satisfaite dans chaque monde, mais on ne sait pas dans quels mondes elle est satisfaite, cela peut être résolu en obtenant des informations supplémentaires (par exemple, une expérience).
- **Aléatoire** : La phrase est une instance d'une classe pour laquelle il existe une loi statistique déterminant si les instances sont satisfaites.
- **Imprécision** : il n'y a pas de correspondance précise entre les termes de la phrase et les référents dans le monde.
- **Incohérence** : il n'existe pas de monde qui pourrait satisfaire l'énoncé.

- **Incomplétude** : les informations sur le monde sont incomplètes, certaines informations sont manquantes.

Les modèles d'incertitude comprennent les probabilités, la logique floue, les fonctions de croyance, les ensembles rugueux et d'autres modèles mathématiques pour le raisonnement dans l'incertitude. La nature de l'incertitude peut être aléatoire (inhérente au phénomène) ou épistémique (due à la connaissance limitée de l'agent).

1.4 La logique floue

Dans cette section, nous allons introduire la logique floue, la différence entre la logique booléenne et la logique floue et ses concepts et fondations théoriques.

1.4.1 Définition

Pour le terme logique floue, plusieurs définitions ont été envisagées. Par la suite, nous fournirons quelques définitions :

Définition de Zadeh : La logique floue s'intéresse aux fondements du raisonnement par approximation, qui est considéré comme limité. Son objectif est d'imiter le mode de raisonnement imprécis, qui joue un rôle clé dans la capacité exceptionnelle des humains à prendre des décisions face à l'incertitude [8].

Zadeh est un chercheur de l'Université de Berkeley en Californie. Il a introduit la notion de sous-ensemble flou pour répondre aux problèmes auxquels sont confrontés de nombreux systèmes complexes, qui doivent traiter des informations qui sont de nature imparfaite, son concept de base est de graduer l'appartenance à un ensemble, c'est un moyen efficace pour prendre en compte l'imprécision dans la connaissance et de formaliser le processus de raisonnement humain [9].

Définition de Timothy J. Ross : La logique floue est un mécanisme permettant de formaliser la capacité humaine à raisonner de manière inexacte. Ce type de pensée illustre la capacité humaine à raisonner et à évaluer face à l'incertitude. Toutes les vérités en logique floue sont partielles ou approximatives. En ce sens, ce raisonnement est également connu sous le nom de raisonnement interpolatif, car la capacité de la logique floue à contenir des vérités partielles représente l'acte d'interpolation entre les extrêmes binaires du vrai et du faux [10].

1.4.2 La Différence entre la logique booléenne et la logique floue

La différence fondamentale entre la logique booléenne (Crisp) et la logique floue est la façon dont le degré d'appartenance à différents groupes est défini et considéré. La logique booléenne se limite aux valeurs 0 et 1, ou vrai ou faux, ce qui signifie qu'une instance appartient ou n'appartient pas à un ensemble. Ceci est illustré à la figure 1.3 (A). Tant que la valeur se situe à gauche de la valeur définie, elle sera évaluée comme 0 (faux) et dès qu'elle a une valeur plus

élevée que la valeur définie, elle basculera du côté vrai (1). Avec la logique floue, figure 1.4 (B), la valeur d'appartenance se déplace dans un continuum entre 0 et 1. Ce qui signifie qu'elle peut à la fois appartenir et ne pas appartenir à l'ensemble en même temps. Cela donne plus d'options lors de la modélisation de données imprécises, en permettant d'indiquer que, même si la valeur d'appartenance n'est pas égale à 1, il présente encore des similitudes avec l'ensemble et a donc, par exemple, la valeur 0,8 [11].

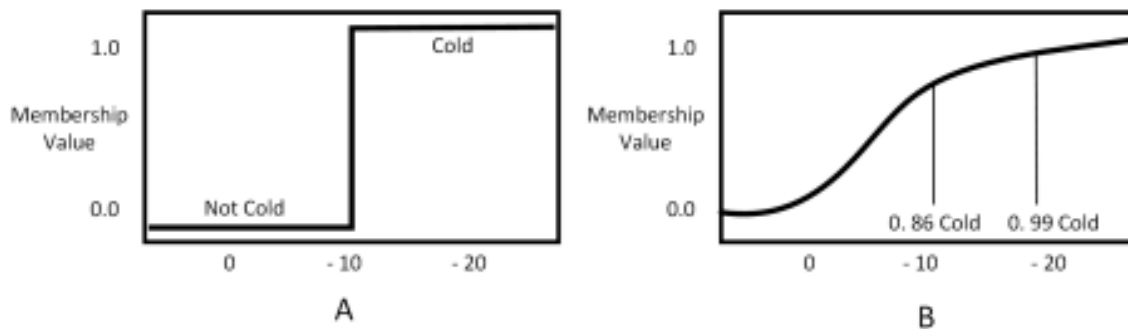


FIGURE 1.3 – (A) Logique booléenne (B) Logique floue.

1.4.3 Concepts et Fondements théoriques

On y retrouve les différents concepts et fondements théoriques de la logique floue.

1. Les Variables Linguistiques :

Une variable linguistique est une variable dont les valeurs sont des mots ou des phrases dans une langue naturelle ou artificielle. Par exemple, l'âge est une variable linguistique si ses valeurs sont linguistiques plutôt que numériques, c'est-à-dire jeune, pas jeune, très jeune, assez jeune, vieux, pas très vieux et pas très jeune, etc. plutôt que 20, 21, 22, 23 [12].

En termes plus spécifiques une variable linguistique est caractérisée par un quintuple $\langle x, T(x), U, G, M \rangle$ où :

- x : est le nom de la variable.
- $T(x)$: est l'ensemble des valeurs linguistique que peut prendre x .
- U : est l'univers du discours associé avec la valeur de base.
- G : est une règle syntaxique qui génère les termes de $T(x)$.
- M : est une règle sémantique qui associe à chaque valeur linguistique x .

On prend l'Age, son ensemble de terme $T(\text{Age})$:

$T(\text{Age}) = (\text{jeune}, \text{moyen-âge}, \text{vieux}, \text{très vieux})$

Ou chaque terme dans $T(\text{Age})$ est caractérisé par un ensemble flou dans l'univers de

discours $U = [0,100]$. Pour l'âge inférieur ou égal à 20 ans intervient la variable linguistique (jeune), et pour l'âge supérieur ou égal à 75 ans intervient la variable linguistique (très vieux). Ces deux termes peuvent être considérés comme des ensembles flous dont les fonctions d'appartenance sont représentées dans cette figure 1.4.

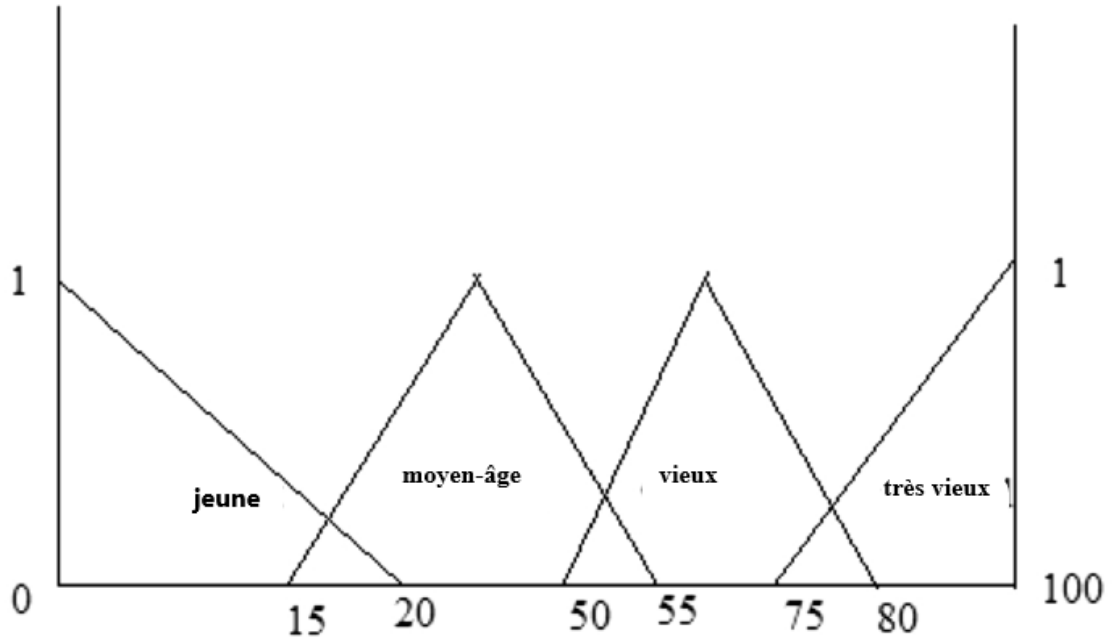


FIGURE 1.4 – Variable linguistique "Age".

2. Ensembles flous :

En logique floue, un ensemble flou contient plusieurs valeurs. Un ensemble Z défini sur un univers du discours X , est caractérisé par une fonction d'appartenance $U_2(X)$ qui prend des valeurs dans l'intervalle $[0,1]$, tel que $\forall x \in X, U_2(X) \in [0,1]$. $U_2(X)$ représente le degré d'appartenance de x à l'ensemble flou Z que nous appelons aussi valeur de vérité. Un ensemble flou est représenté par un ensemble de paires ordonnées tel que :

$$Z = \{x, U_2(X) : x \in X\}$$

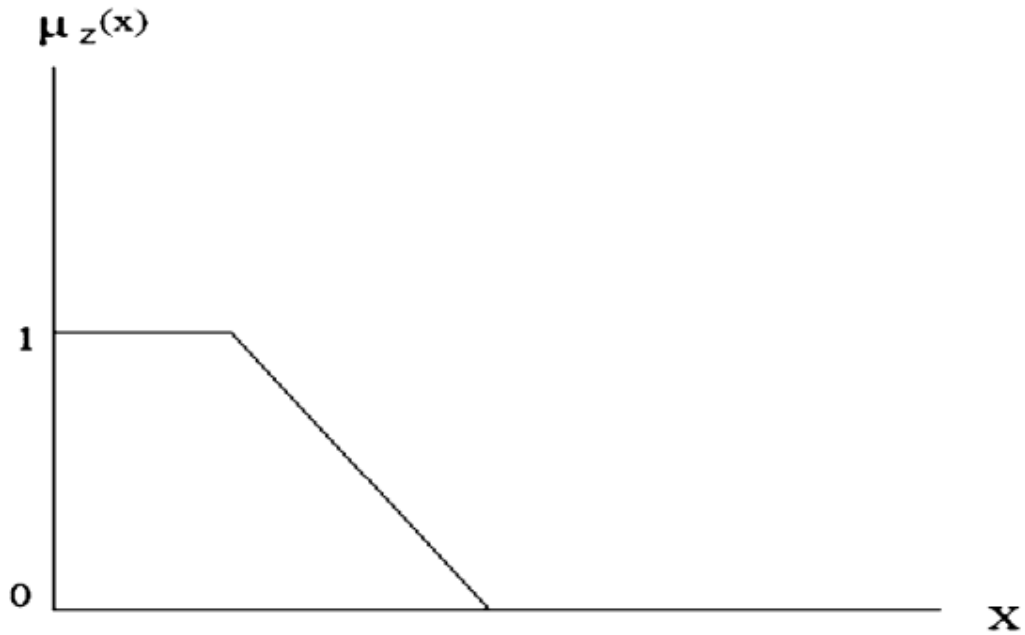


FIGURE 1.5 – Représentation du sous-ensemble flou Z.

La fonction d'appartenance $U_2(X)$ est définie comme une fonction :

$$U_2(X) : X \rightarrow 0, 1$$

$$U_2(X) = \begin{cases} 1, & x \text{ est fortement situé dans } X. \\ (0, 1), & x \text{ est partiellement compris dans } X. \\ 0, & x \text{ n'est pas dans } X. \end{cases}$$

La figure 1.6 montre où se situe la différence entre l'ensemble classique et l'ensemble flou graphiquement.

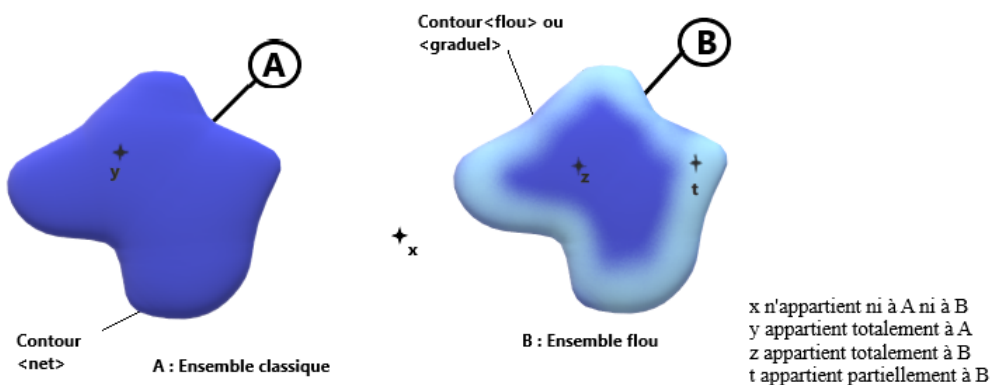


FIGURE 1.6 – La différence entre l'ensemble classique et l'ensemble flou graphiquement.

1.5 Les ontologies floues

Dans cette section, on introduit une ontologie floue avec ces composants.

1.5.1 Définition

Différents chercheurs présentent différentes définitions de l'ontologie floue. La plupart de ces définitions dépendent de l'application et visent à rendre flous certains éléments de l'ontologie en fonction des besoins de l'application [13].

Dans [14], [15], Samani et Shamsfard définissent l'ontologie floue comme une 4-Tuple :
 $O_F = (C, P^C, R, A)$ tel que :

- C est un ensemble de concepts.
- P^C est un ensemble de propriétés de l'entité qui peut être représenté par un 8-Tuples $P^C(c, p, v_p, g_p, n_p, q_p, h_p, f)$ où :
 - c est un concept de l'ontologie.
 - p est le nom de la propriété.
 - v_p est un ensemble de valeurs de la propriété.
 - g_p est un ensemble de fonctions d'appartenance attribuées aux membres de v_p .
 - n_p est un ensemble de degrés d'appartenance assignés à v_p .
 - q_p modélise le modificateur linguistique (qui est facultatif).
 - h_p est un ensemble de fonctions d'appartenance assignées à chaque modificateur.
 - f est une facette de restriction telle que le type ou la cardinalité. Le type peut être {Integer, float, etc}. Cardinalité définit les limites supérieure et inférieure du nombre de valeurs de la propriété.
- R est un ensemble de relation entre concepts.
- A est un ensemble d'axiomes.

1.5.2 Les composants d'une ontologie floue

Une ontologie floue comprend :

- Des concepts précis (exacts) et des concepts flous,
- Des relations précises (exactes) et des relations floues,
- Des axiomes,
- Des instances.

Dans une ontologie traditionnelle, les concepts et relations précis ont les mêmes définitions et jouent les mêmes rôles. Les concepts et relations flous, définis selon la logique floue, sont utilisés pour représenter les éléments flous du domaine modélisé. Ils ont un impact sur l'expression des axiomes et des instances [16].

— **Les concepts flous :**

Un concept flou est considéré comme un ensemble flou. Ses instances possèdent ainsi des degrés d'appartenance prenant leur valeur dans l'intervalle $[0,1]$. Ces degrés sont calculés conformément aux formules de calcul relatives à la fonction d'appartenance définie pour le concept [17].

Exemple :

Ahmed est un Employé. Il a comme âge 36 ans. On peut alors le considérer comme un « Employé-Jeune » avec un degré d'appartenance de 0.9, calculé selon la formule de calcul de la fonction d'appartenance trapézoïdale définie précédemment.

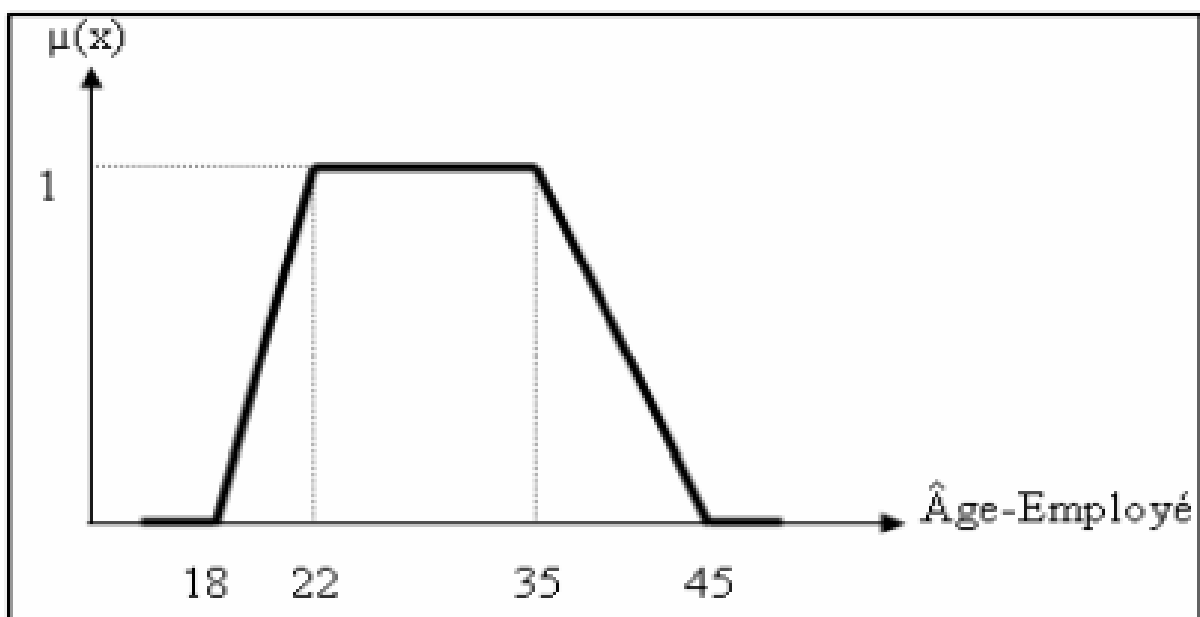


FIGURE 1.7 – Fonction d'appartenance du concept flou <Employé-Jeune>.

— **Les relations flous :**

Une relation floue est aussi définie comme un ensemble flou. Ses instances possèdent ainsi des degrés d'appartenance prenant leur valeur dans l'intervalle $[0,1]$. Ces degrés sont calculés conformément aux formules de calcul relatives à la fonction d'appartenance définie pour la relation.

Exemple :

La relation floue « Acquiert », de la figure 1.8, relie les deux concepts précis « Employé » et

« Compétence-Acquise ». Elle dépend de la valeur de la propriété « Niveau-Acquisition ». Elle a quatre sous-ensembles flous : Compétence-Inexistante, Compétence-Peu-Maîtrisée, Compétence-Maîtrisée et Compétence-Parfaitement-Maîtrisée. La figure 1.9 montre les fonctions d'appartenance des deux derniers ensembles.

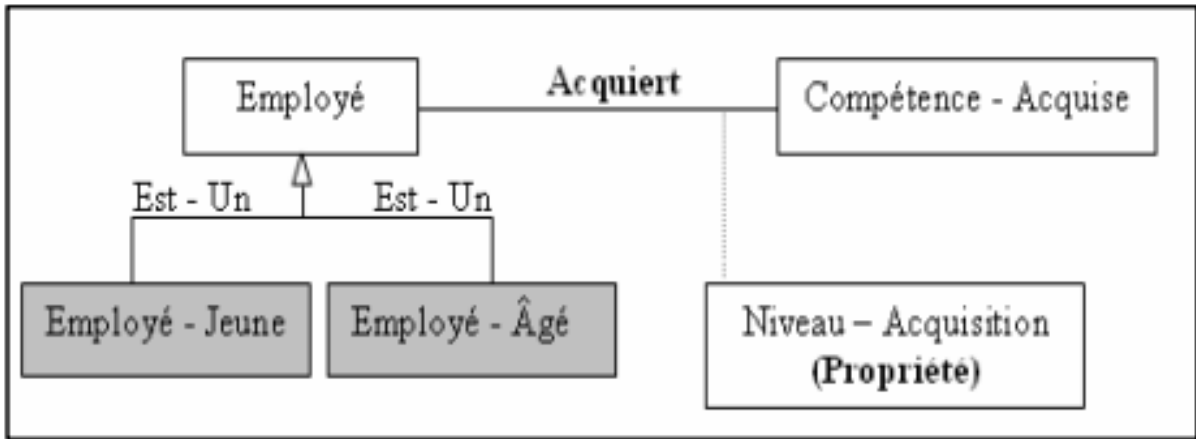


FIGURE 1.8 – Exemple d'une ontologie floue.

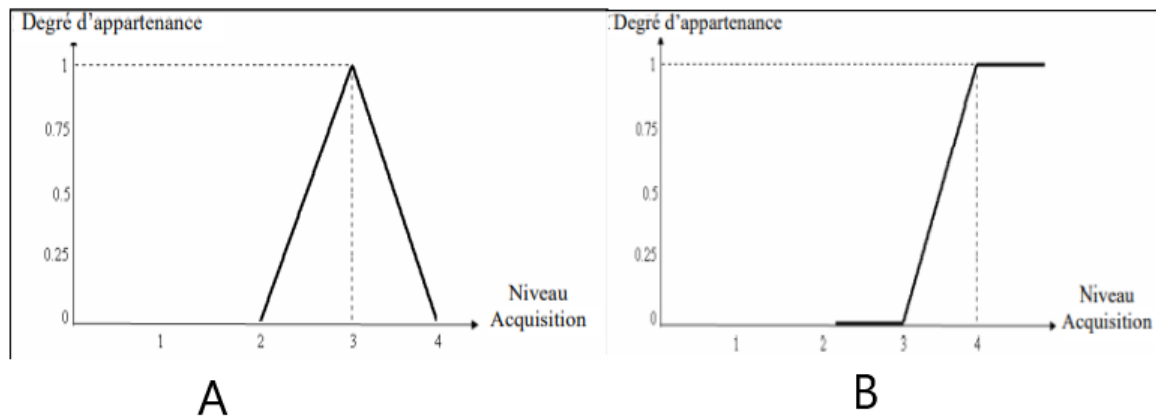


FIGURE 1.9 – La fonction d'appartenance de l'ensemble flou A <Compétence-Maitrisée> et B <Compétence-Parfaitement-Maitrisée >.

— **Les axiomes :**

A part leurs utilisations classiques, les axiomes servent à exprimer les formules de calcul des fonctions d'appartenance. Par exemple, nous devons déclarer, pour le schéma ontologique de la figure 1.7, des axiomes pour définir les formules de calcul des degrés d'appartenance des différents concepts flous :

Axiomes pour le concept flou « Employé-Jeune » :

- $\text{Degré_Appartenance} = (\text{Age} - 18) / (22 - 18)$ si $\text{Age} \in [18; 22]$;
 - $\text{Degré_Appartenance} = 1$ si $\text{Age} \in [22; 35]$;
 - $\text{Degré_Appartenance} = (45 - \text{Age}) / (45 - 35)$ si $\text{Age} \in [35; 45]$;
 - $\text{Degré_Appartenance} = 0$ ailleurs.
- **Les instances :**
- Dans les ontologies floues, nous trouvons deux types d'instances : les instances des concepts et des relations précises et les instances des concepts et des relations floues.
- Les instances des concepts et des relations précises sont des réalisations « précises » des éléments d'une ontologie floue. Elles ne sont pas caractérisées par des degrés d'appartenance à des ensembles flous : ce sont des instances ordinaires.
 - Les instances des concepts et des relations floues sont des réalisations « floues » des éléments d'une ontologie floue. Elles possèdent des degrés d'appartenance aux ensembles flous correspondants.

1.6 Méthode de développement des ontologies floues

FODM (Fuzzy Ontology Development Methodology) est un paradigme formel de développement d'ontologie floue est présenté sur la base des méthodes de développement d'ontologies existantes. Il met l'accent sur l'introduction de nouveaux changements apportés par des ontologies floues dans le processus de développement[18].

Le but du FODM est de fournir une abstraction formelle des activités qui doivent être réalisées tout au long du processus de développement. Elle vise à présenter la première approche méthodologique pour construire des ontologies floues à partir de zéro, plutôt que de convertir des ontologies existantes en ontologies floues. En général, toutes les activités ou tâches sont regroupées en onze phases pour former l'ensemble du cycle de vie de la construction d'une ontologie floue.

- **PHASE 1 :**Des questions de base doivent être soulevées et traitées explicitement afin de clarifier le but et la portée de l'ontologie.
 1. Quel est le domaine ou la portée des informations à modéliser ?
 2. L'ontologie est-elle la meilleure technique de modélisation par rapport aux autres solutions ?
 3. Quel est le type d'ontologie en fonction de la détermination du domaine ou de la portée ?
 4. Qui sera impliqué dans le développement de l'ontologie et quels rôles ils vont jouer ?
- **PHASE 2 :**La deuxième phase vise à identifier si le flou doit être introduit dans la conception de l'ontologie.

- **PHASE 3** : Suite à l'étape 2, une meilleure compréhension des informations doit être faite dans cette étape, qui vise à établir une distinction entre les informations précises et floues dans le domaine.
- **PHASE 4** : La vérification des ontologies standards existantes pertinentes pour le domaine ou la portée d'intérêt et la détermination de leur réutilisabilité sont les principales tâches définies dans cette phase.
- **PHASE 5** : Similaire à la phase précédente, cette phase vise à vérifier l'existence d'ontologies floues pour réutiliser un ou plusieurs éléments de ces dernières, cependant si aucune ontologie floue n'a été trouvée pour notre domaine, alors cette phase et la phase suivante seront ignorées.
- **PHASE 6** : Si dans la phase précédente, au moins une ontologie floue a été constatée comme utile grâce à ses concepts, on peut remodifier ces concepts pour les adapter à notre ontologie, ceci est le but de cette phase.
- **PHASE 7** : Le but de cette phase est de définir différents éléments d'ontologie floue pour fournir des approximations correctes de la nature des informations vagues et imprécises dans le domaine.
- **PHASE 8** : Cette phase se concentre sur le traitement de certaines connaissances dans le domaine. Outre les informations floues liées, le reste de la base de connaissances du domaine est défini comme des éléments d'ontologie précis en fonction de leurs attributs spécifiques.
- **PHASE 9** : Un certain langage doit être sélectionné pour formaliser l'ontologie conçue dans un format lisible par machine.
- **PHASE 10** : L'ontologie conçue doit passer par une vérification approfondie pour s'assurer qu'elle a représenté le modèle prévu du monde.
- **PHASE 11** : Documentation pour présenter les principes d'ingénierie de l'ontologie conçue, y compris les descriptions des différents éléments de l'ontologie, les détails de conception, la méthode d'utilisation et la maintenance.

1.7 Conclusion

Dans ce chapitre, nous avons défini l'ontologie classique avec ses limites, et la solution permettant de surmonter ces problèmes en utilisant des ontologies floues, en se basant sur la logique floue, elle marque un rôle important dans de nombreux domaines qui font face à une grande quantité de données imprécises et vagues.

Chapitre 2

RÉSEAU BAYÉSIEEN

2.1 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, se sont révélés des outils très pratiques pour la représentation des connaissances incertaines. Les réseaux bayésiens figurent parmi les modèles d'analyse probabiliste. L'expression de connaissances floues est l'un des principaux problèmes auxquels sont confrontés les utilisateurs de réseaux bayésiens. Pour cela Différents travaux de recherche ont récemment permis l'utilisation de la théorie des ensembles flous dans le cadre des réseaux bayésiens.

Dans ce chapitre nous allons définir les réseaux bayésiens, les réseaux bayésiens flous et illustrer les ontologies probabilistes avec ses différents langages. Ensuite, nous allons présenter notre étude sur les travaux existants.

2.2 Réseaux Bayésiens

Dans cette section, nous présentons les réseaux bayésiens, les différentes étapes de sa construction, les types d'inférence et le réseau bayésien flou avec un exemple.

2.2.1 Définition

Un réseau bayésien (RB) est un graphe acyclique dirigé dont les nœuds représentent des variables (attributs de l'utilisateur) et les liens représentent des interactions probabilistes d'impact. Ces variables peuvent être classées comme appartenant à un champ de connaissances, une base de connaissances ou un modèle cognitif. Chaque nœud représente un ensemble de croyances sur les valeurs possibles de la variable (niveaux, états). Par conséquent, à chaque nœud, la distribution de probabilité conditionnelle doit être fournie. Si les variables sont discrètes, un tableau peut être utilisé pour les représenter. Les tableaux de probabilité constituent les "paramètres" du modèle,

tandis que le graphique en constitue la "structure". Les experts peuvent donner la structure et les paramètres, ou ils peuvent être calculés à partir de données, bien que dans la plupart des cas, la structure est spécifiée par les experts et les paramètres sont calculés à partir de données expérimentales [19].

Le réseau bayésien est défini par $B = (G, C)$ [20] :

- $G = (X, E)$ est un graphe acyclique dirigé dont les nœuds sont associés à un ensemble de variables aléatoires $X = \{X_1, X_2, \dots, X_n\}$ et E est l'ensemble des arcs de G qui représentent les dépendances entre ces nœuds.
- $C = \{P(X_i | Pa(X_i))\}$ est un ensemble de probabilités selon lesquelles chaque nœud X_i est conditionnellement dépendant de l'état de son parent $Pa(X_i)$ dans G .

2.2.2 Construction d'un réseau bayésien

Les réseaux bayésiens utilisent des algorithmes d'apprentissage pour construire un modèle RB qui correspond mieux à la connaissance du problème à représenter et à traiter. L'apprentissage de la structure et des paramètres est, après tout, le problème d'apprentissage.

— Apprentissage de la structure

L'apprentissage de la structure d'un RB permet de déterminer automatiquement une structure optimale d'un RB à partir de l'information contenue dans les données observées. On distingue trois grandes familles d'approches d'apprentissage de la structure [21] :

1. Les méthodes basées sur les contraintes et recherche d'indépendances (constraint-based).
2. Les méthodes basées sur le score et la recherche (score and search based).
3. Les méthodes hybrides.

— Apprentissage des paramètres de réseaux bayésiens

Etant donnée une structure de RBG et un ensemble de données D , le principe de l'algorithme d'apprentissage des paramètres est d'estimer les distributions de probabilités (paramètres)

$\theta = \{\theta_i\}$ avec $\theta_i = P(X_i | Pa(X_i))$, table de probabilité conditionnelle constituée de $\theta_{ijk} = P(X_i = x_k | Pa(X_i = x_j))$.

Cette estimation peut être effectuée différemment selon que les données disponibles soient complètes ou incomplètes [21].

1. Apprentissage avec 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. Cette approche, appelée maximum de vraisemblance (MV), nous donnons la formule de calcul [22] :

$$P(X_i = x_k | P\alpha(X_i) = x_j) = \theta_{i,j,k}^{MV} = N_{i,j,k} / (\sum_k N_{i,j,k})$$

ou $N_{i,j,k}$ est le nombre d'événements dans la base de données pour lesquels la variable X_i est dans l'état x_k et ses parents sont dans la configuration x_j .

2. Apprentissage avec 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, ou d'autre. Après avoir constaté l'existence de différents types de données incomplètes, nous aborderons la méthode d'estimation la plus utilisée est celle qui se base sur l'algorithme EM (Expectation Maximisation) [22].

2.2.3 Inférence dans les réseaux bayésiens

L'inférence, ou la mise à jour des croyances, consiste à calculer la probabilité a posteriori au niveau de certaines variables en prenant en compte les informations sur d'autres variables appelées observations.

Les réseaux bayésiens permettent de mettre à jour les distributions de probabilités au niveau des variables : on commence par fixer des observations, c'est à dire affecter des valeurs aux variables d'observation, puis on utilise un algorithme d'inférence pour calculer les probabilités a posteriori des autres variables.

Pour un réseau bayésien dont l'ensemble des nœuds est $X = \{X_1, X_2, \dots, X_n\}$, les algorithmes d'inférence permettent de calculer les probabilités a posteriori $P(X_i|Y)$ d'une variable X_i sachant les observations Y ou $Y \subset X$ et $X_i \notin Y$. Aucun algorithme d'inférence n'applique directement le théorème de Bayes car cela conduit à des calculs très coûteux en temps. Il existe une multitude de travaux dans le domaine de l'inférence dans les réseaux bayésiens. Nous en présentons ici un échantillon représentatif. Deux grandes familles d'algorithmes d'inférence se distinguent : les algorithmes d'inférence exacte et les algorithmes d'inférence approchée [23].

1. Inférence Exacte

L'inférence exacte est un problème NP complet. En pratique, la complexité des algorithmes d'inférence dépend de plusieurs facteurs dont notamment : le nombre de nœuds du réseau bayésien, la structure du réseau bayésien (nombre de chemins multiples entre deux nœuds), la taille des domaines des variables et le nombre de parents d'un nœud. Ce constat peut amener à utiliser des algorithmes d'inférence approchées.

2. Inférence approchée

Les algorithmes d'inférence approchée sont plus souvent utilisés dans les réseaux bayésiens complexes, avec plusieurs cycles. Ces méthodes donnent une estimation des probabilités a posteriori et sont non déterministes. Ils sont principalement de deux types [24] :

- les algorithmes d'inférence approchée du premier type utilisent des méthodes exactes mais opèrent seulement sur une partie du graphe. L'idée est d'inhiber l'effet des arcs

jugés insignifiants ce qui permet d'éliminer des liens et de réduire la taille des tables de probabilités conditionnelles. Ceci accélère les calculs de façon significative avec des erreurs minimales sur les probabilités calculées [25]. Dans [26], les nœuds dont les probabilités ont les valeurs les plus importantes sont considérés comme des observations (probabilité égale 1). Cette méthode est inadaptée au cas des réseaux bayésiens dont certains nœuds ont une valeur extrêmement probable, car ils risqueraient d'être considérés à tort comme des observations.

- Les algorithmes d'inférence approchée du deuxième type utilisent des méthodes de simulation stochastique. Les plus connus sont les algorithmes de MonteCarlo ou probabilistic logic sampling [27], likelihood sampling [28] et backward sampling [29]. Ces algorithmes donnent une estimation des probabilités moyennant la génération d'échantillons d'instanciation du réseau. Le problème soulevé par ces méthodes est dû au nombre important d'instanciations nécessaires pour calculer une approximation des probabilités a posteriori. Pour des réseaux bayésiens de grande taille ou lorsque certaines variables ont des valeurs avec une probabilité proche de 0 ou 1, ces méthodes requièrent un très grand nombre d'échantillons.

2.3 Réseaux Bayésiens flous

Les réseaux bayésiens flous (RBF) combinent les capacités des réseaux bayésiens et de la logique floue afin de profiter des avantages des deux modèles simultanément. En effet, les RBF sont des modèles hybrides qui combinent la théorie des graphes, la théorie des probabilités et la logique floue pour gérer les connaissances floues dans les expériences statistiques et la fréquence des événements [30].

Les réseaux bayésiens sont des outils très puissants et sont utilisés dans de nombreuses situations et domaines différents. Ils constituent une méthode utile et compacte pour représenter les distributions de probabilité conjointes. De même, les ensembles flous permettent de représenter les données en termes linguistiques qui contribuent à améliorer leur compréhension.

En résumé, un RBF est une extension du réseau bayésien qui permet de traiter la probabilité d'un événement sous des observations ambiguës. Il comporte deux types de nœuds : les nœuds précis et les nœuds flous. Les nœuds précis ont des significations précises, tandis que les nœuds flous ont des significations floues.

2.4 Exemple d'un réseau bayésien flou

Nous présentons un réseau simple qui soit utilisé pour diagnostiquer un court-circuit de résistance avec un test mesurant le courant aux bornes de la résistance. Ce réseau est représenté à la figure 2.1. Ce réseau possède un nœud de test, Test de courant, et un nœud de diagnostic,

Court-circuit de résistance. Le nœud de test est traité comme un nœud de preuve, et le nœud de diagnostic est une variable d'interrogation [31]. Les tableaux 2.1 et 2.2 contiennent les tables de probabilité conditionnelle pour ce réseau bayésien, ainsi qu'une représentation des fonctions d'appartenance floues dans la figure 2.2.

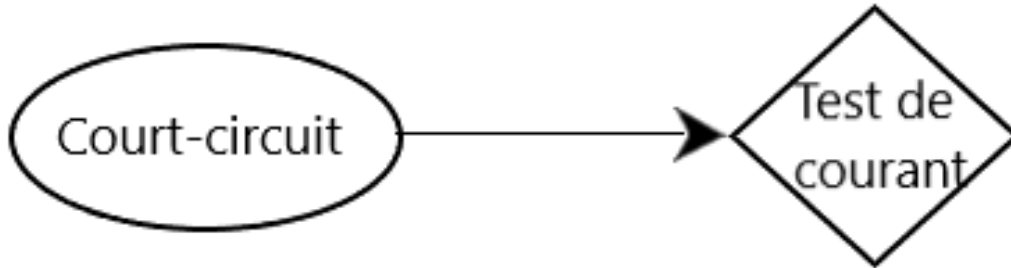


FIGURE 2.1 – Réseau bayésien simple.

$P(\text{Court} - \text{circuit} = \text{True})$	$P(\text{Court} - \text{circuit} = \text{False})$
0.15	0.85

TABLE 2.1 – Tableau de probabilité conditionnelle pour le nœud de court-circuit de la résistance.

$\text{Court} - \text{circuit} \rightarrow \text{Testdecourant}$	$P(\text{Testdecourant} = \text{High})$	$P(\text{Testdecourant} = \text{Normal})$
Court-circuit = True	0.99	0.01
Court-circuit = False	0.01	0.99

TABLE 2.2 – Tableau de probabilité conditionnelle pour le nœud de Test de Courant.

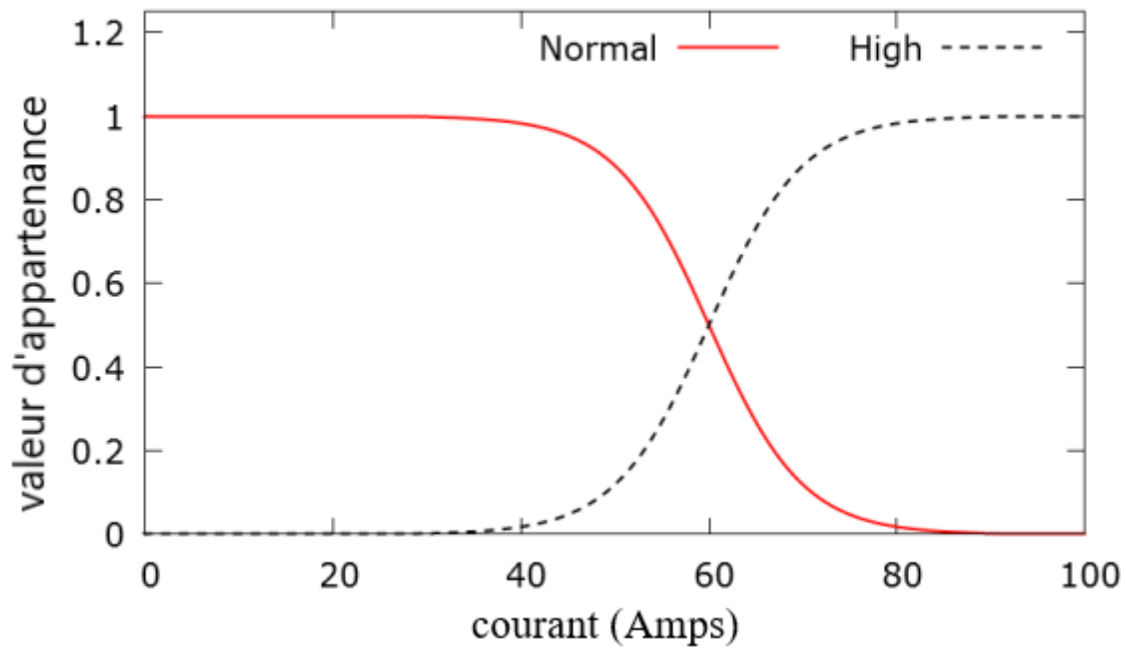


FIGURE 2.2 – Fonctions d'appartenance pour le réseau.

Supposons qu'un courant de 50 ampères soit lu aux bornes de la résistance au départ. A l'aide des fonctions d'appartenance μ_{High} et μ_{Normal} , La valeur d'appartenance floue pour High est $\mu_{High}(50Amps) = 0,119$, et la valeur d'appartenance pour Normal est $\mu_{Normal}(50Amps) = 0,881$.

Supposons qu'on veut calculer la requête suivante (Requête 1) :

$$P(Court - circuit\ de\ resistance = True | Test\ de\ courant)$$

Pour faciliter la notation, nous désignerons le court-circuit de résistance par R et le test de courant par T.

2.4.1 Méthode de pondération

L'idée de cette méthode est d'étendre les différentes règles utilisées dans les réseaux bayésiens en associant une valeur de degré d'appartenance à chaque loi, cette méthode consiste à étendre la loi Bayes afin d'incorporer les degrés d'appartenances dans les probabilités comme des poids lors de l'inférence.

En effet, les règles bayésiennes floues peuvent ensuite être définies pour prendre en charge l'inférence bayésienne floue dans un modèle RBF.

La Requête 1 sera traitée utilisant la méthode de pondération proposée dans [32] comme suit :

$$\begin{aligned}
P(R|T) &= \frac{\sum_{i \in I} \mu_T(T_i) P(T_i|R) P(R)}{P(T)} \\
&= \frac{\sum_{i \in I} \mu_T(T_i) P(T_i|R) P(R)}{\sum_{i \in I} \mu_T(T_i) P(T_i)} \\
&= \frac{\mu_T(T_{High}) P(T_{High} \vee R) P(R) + \mu_T(T_{Normal}) P(T_{Normal} \vee R) P(R)}{\mu_T(T_{High}) P(T_{High}) + \mu_T(T_{Normal}) P(T_{Normal})} \\
&= \frac{0.119 * 0.946 * 0.15 + 0.881 * 0.002 * 0.15}{0.119 * 0.157 + 0.881 * 0.843} \\
&= 0.023
\end{aligned}$$

Compte tenu des résultats du test de courant, il y a une probabilité de 0,023 que la résistance ait été court-circuitée par cette méthode.

2.4.2 Distribution probabiliste floue

Cette méthode consiste à construire des distributions probabilistes floues [[33],[34]]. Les degrés d'appartenances seront directement intégrés dans les distributions des probabilités pour construire des distributions probabilistes floues.

Ainsi, cette méthode est fondée sur deux étapes, la première étape sert à calculer les degrés d'appartenances et les représenter en notation particulière (par des vecteurs), lors de la deuxième étape, les degrés d'appartenances calculés en première étape seront incorporés directement dans les probabilités conditionnelles pour répondre à des requêtes.

Pour répondre à la Requête 1 en utilisant la méthode des distributions probabilistes floues, nous devons tout d'abord représenter les degrés d'appartenances relatifs au nœud le test de courant comme un vecteur.

$$T = \{high_{0.157}, normal_{0.843}\}$$

Par la suite, il s'agit d'incorporer les degrés d'appartenances dans la CPT (la table de probabilité conditionnelle) du nœud Court-circuit de résistance.

$$\begin{aligned}
R &= [\{P(R = true \vee T = high), P(R = false|T = high)\}_{\mu_{high}(50Amps)}, \\
&\{P(R = true \vee T = normal), P(R = false \vee T = normal)\}_{\mu_{normal}(50Amps)}] \\
&= [\{true_{0.945}, false_{0.054}\}_{0.119}, \{true_{0.002}, false_{0.998}\}_{0.881}] \\
&= [true_{0.945*0.119+0.002*0.881}, false_{0.054*0.119+0.998*0.881}] \\
&= [true_{0.114}, false_{0.886}]
\end{aligned}$$

Nous avons obtenu à la fin une distribution probabiliste floue pour la variable R en se basant sur les observations de la variable Test de courant. A partir de cette distribution, nous pouvons constater que $P(R = true|T) = 0.114$ et $P(R = flase|T) = 0.886$.

2.4.3 Méthode de l'évidence virtuelle

L'évidence virtuelle est une méthode permettant d'incorporer l'incertitude de l'évidence dans un réseau bayésien. Le processus d'utilisation de l'évidence virtuelle pour incorporer des valeurs floues dans un réseau bayésien est très simple. Une fois que le nœud de l'évidence virtuelle est ajouté, la preuve floue est incorporée directement en tant qu'évidence virtuelle. L'évidence virtuelle est représentée en manipulant la table de probabilité conditionnelle du nœud de l'évidence virtuelle. Nous illustrons ce processus avec l'exemple de réseau donné à la figure 2.3.

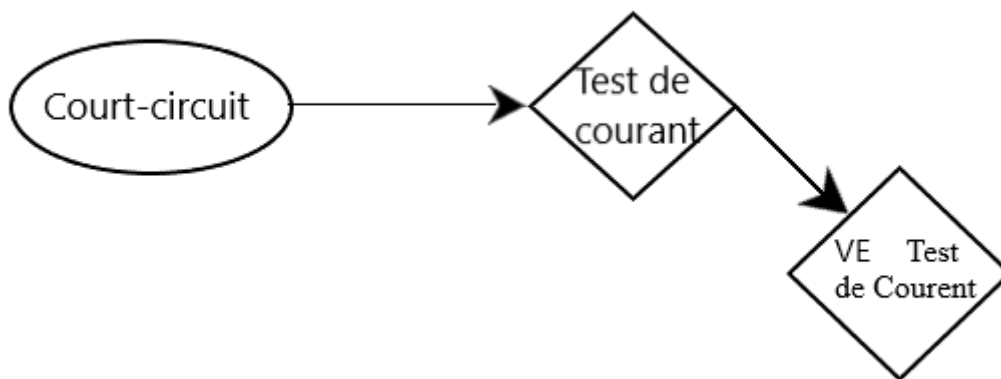


FIGURE 2.3 – Le Réseau Bayésien avec un nœud de l'évidence virtuelle.

Nous pouvons alors utiliser ces informations pour calculer $P(R_{true}|T_{high})$ en utilisant les valeurs floues comme évidence virtuelle. Cependant, au lieu d'utiliser T comme preuve, nous utilisons VE comme preuve, de sorte que ce que nous résolvons réellement est $P(R_{true}|VE_{high})$.

$$\begin{aligned}
 P(R_{true}|VE_{high}) &= \frac{P(VE_{high}\vee R_{true})P(R_{true})}{P(VE_{high})} \\
 &= \frac{P(VE_{high}\vee R_{true})P(R_{true})}{P(VE_{high}\vee T_{high}) + P(VE_{high}\vee T_{normal})} \\
 &= \frac{0.12662 * 0.15}{0.119 + 0.881} \\
 &= 0.0249
 \end{aligned}$$

Il convient de noter que les deux méthodes de l'évidence virtuelle et de pondération ont données le même résultat.

2.5 L'ontologie probabiliste

Dans cette partie, nous avons fait référence à l'ontologie probabiliste et à ses différentes langues.

2.5.1 Définition

Une ontologie probabiliste est une représentation formelle et explicite des connaissances qui exprime des connaissances sur un domaine d'application [3]. Cela comprend :

- Types d'entités qui existent dans le domaine.
- Les propriétés de ces entités.
- Les relations entre les entités.
- Les processus et les événements qui se produisent avec ces entités.
- Régularités statistiques qui caractérisent le domaine.
- Des connaissances non concluantes, ambiguës, incomplètes, non fiables et dissonantes liées aux entités du domaine.
- L'incertitude quant à toutes les formes de connaissances ci-dessus.

Où le terme entité désigne tout concept (réel ou fictif, concret ou abstrait) qui peut être décrit ou faire l'objet d'un raisonnement dans le domaine d'application [3]. Les ontologies probabilistes sont utilisées dans le but de décrire de manière exhaustive les connaissances relatives à un domaine et l'incertitude associée à ces connaissances d'une manière fondée sur des principes, structurée et partageable, idéalement dans un format qui peut être lu et traité par un ordinateur. Ils élargissent également les possibilités des ontologies standards en introduisant l'exigence d'une représentation appropriée des régularités statistiques et les preuves incertaines concernant les entités dans un domaine d'application [3].

2.5.2 Les langages des ontologies probabilistes

PR-OWL : PR-OWL est une extension d'OWL, ayant la capacité de faire des raisonnements probabilistes basé sur une logique bayésienne de premier ordre appelée réseaux bayésiens à entités multiples (MEBN : Multi Entity Bayesian Network), qui intègre la logique classique du premier ordre à la théorie des probabilités [3]. PR-OWL utilise une autre forme de représentation de classe. La figure 2.4 représente les classes sous forme d'ovales et les relations entre elles sont représentées par des flèches.

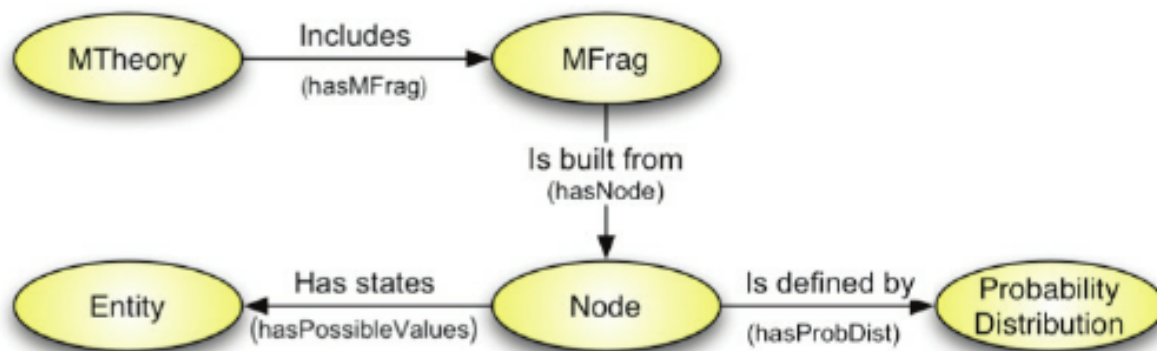


FIGURE 2.4 – Composition d’un MTheory de PR-OWL [3].

Dans ce diagramme, Une ontologie probabiliste doit avoir au moins un individu de la classe MTheory, qui est une étiquette reliant un groupe de MFrag qui forment collectivement une MTheory valide. Dans la syntaxe PR-OWL actuelle, ce lien est exprimé par la propriété d’objet hasMFrag (qui est l’inverse de la propriété d’objet isMFragIn). Un MFrag regroupe les variables aléatoires (Node) d’une classe sont représentées sous forme de nœuds résidents, d’entrée ou de contexte. Chaque nœud a des états qui lui sont propres appelés des entités (Entity). Enfin, les variables aléatoires ont des distributions de probabilité inconditionnelles ou conditionnelles, qui sont représentées par la classe Probability Distribution et liées à leurs nœuds respectifs par la propriété d’objet hasProbDist [3].

BayesOWL : est un framework probabiliste qui augmente et complète le langage ontologique du web sémantique OWL pour représenter et raisonner avec l’incertitude basée sur les réseaux bayésiens. BayesOWL fournit un ensemble de règles et de procédures pour la traduction directe d’une ontologie OWL en un graphe acyclique dirigé (DAG) de réseau bayésien. Il fournit également une méthode basée sur la procédure d’ajustement proportionnel itératif (IPFP) qui intègre les contraintes de probabilité disponibles lors de la construction des tables de probabilité conditionnelle (CPT) du RB [35].

La théorie Dempster-Shafer : elle est présentée comme une généralisation de la théorie des probabilités, elle permet de combiner des preuves provenant de différentes sources et de parvenir à un degré de croyance (représenté par une fonction de croyance) qui tient compte de toutes les preuves disponibles. En ce qui concerne l’ontologie, elle a encore été la plus appliquée dans des domaines tels que le traitement des incohérences dans les ontologies OWL et le mappage d’ontologies [7].

2.6 Travaux connexes

Dans cette section, nous décrivons les travaux liés à notre problème.

1. **A Fuzzy Ontology Generation Framework from Relational Schema :**

Dans [36] ils ont proposé un Framework général qui peut générer une ontologie floue à partir d'un schéma relationnel basé sur la théorie des ensembles flous. Le Framework basé sur les logiques de description n-aire et la description logiques f-DLRid n'est pas imposée la limitation à traiter uniquement la relation binaire et peut décrire la contrainte d'identification. En outre, la définition de l'ontologie floue présentée dans ce document est transposée dans le f-OWL correspondante grâce à la syntaxe et à la sémantique du f-DLRid flou.

2. **Fuzzy description logics under Gödel semantics :**

Les ontologies classiques ne sont pas adaptées à la représentation d'éléments d'information vagues, ce qui a conduit à la naissance de la logique de description floue comme formalisme approprié pour représenter ces éléments d'information. Différentes familles d'opérateurs flous mènent à des logiques de description floues avec des propriétés différentes. Dans cet article [37] les auteurs ont étudié les logiques de description floues sous une sémantique donnée par la famille de Gödel. L'objectif est d'étudier certaines propriétés logiques et montrons la décidabilité d'une extension floue de la logique SROIQ, base théorique du langage OWL, en fournissant une procédure de préservation du raisonnement pour en obtenir une représentation claire. En outre, ils ont montré comment représenter certains types de modificateurs de concepts et de rôle.

3. **OntoBayes : An Ontology-Driven Uncertainty Model :**

Dans [38], les auteurs présentent une intégration du langage d'ontologie web OWL avec les réseaux bayésiens, appelée OntoBayes. Cette approche utilise le langage OWL annoté par les probabilités et les dépendances pour représenter les informations incertaines dans les réseaux bayésiens.

4. **A Probabilistic Ontology for the Prediction of Author's Interests :**

Dans cet article [20], les auteurs ont proposé une méthode de construction d'ontologie probabiliste à partir d'une liste de publications (base dblp) d'un auteur particulier et de ses principaux co-auteurs. Cette méthode comprend les étapes suivantes : Construction de l'ontologie classique, Conversion des composants incertains en un graphe de réseau bayésien, Estimation de la distribution de probabilité dans le graphe obtenu basée sur l'apprentissage des paramètres du réseau bayésien et Conversion de l'ontologie classique (COdblp) en une ontologie probabiliste (POdblp).

5. **Probabilistic Ontology Definition Meta-Model :**

Dans [39], une extension probabiliste du méta-modèle OWL 2 est proposée, appelée

méta-modèle de définition d'ontologie probabiliste (PODM). Cela se fait en ajoutant de nouveaux composants la classe probabiliste, l'individu probabiliste, la propriété de données probabiliste, etc.

6. Knowledge oriented semantics modeling towards uncertainty reasoning :

Dernièrement, dans [40], une extension nommée ontologie probabiliste hybride basée sur les réseaux bayésiens hybrides est proposée. Le mérite de cette extension est qu'elle permet de traiter simultanément des distributions sur des quantités discrètes et continues dans l'ontologie.

2.7 Analyse et critique

Dans cette section, nous comparons les travaux existants de modélisation probabiliste et floue des connaissances en ontologie. Pour cela, nous utilisons les critères suivants :

- **Type d'ontologie** : Cette caractéristique représente le type d'ontologie utiliser en entré et en sortie, il s'agit d'une ontologie classique, d'une ontologie floue ou d'une ontologie probabiliste.
- **Type de connaissances** : Cette caractéristique indique si la solution traite les connaissances floues **CF** ou les connaissances probabilistes **CP**.
- **Réseau Bayésien** : Indique si le réseau bayésien a été abordé dans ces articles.

Le résultat de notre analyse est présenté dans le tableau.

Travaux	Type d'ontologie		Type de connaissances	Réseau bayésien
	Entré	Sortie		
Yanhui et al., [36]	Ontologie classique	Ontologie floue	CF	Non
Fernando et al., [37]	Ontologie classique	Ontologie floue	CF	Non
OntoByes [38]	Ontologie classique	Ontologie probabiliste	CP	Oui
Emna et al., [20]	Ontologie classique	Ontologie probabiliste	CP	Oui
PODM [39]	Ontologie classique	Ontologie probabiliste	CP	Oui
Mohammed et al., [40]	Ontologie classique	Ontologie probabiliste	CP	Oui
Notre Travail	Ontologie floue	Ontologie floue probabiliste	CP et CF	Oui

TABLE 2.3 – Tableau de comparaison entre les travaux existants.

D'après l'analyse menée sur les différents travaux de la littérature scientifique, nous avons remarqué que l'ontologie floue a imposé sa présence dans tous les domaines en raison des limites de l'ontologie classique qui ne peut pas traiter les informations vagues. A l'opposé, l'ontologies probabilistes représente des connaissances probabilistes en se basant sur les réseaux bayésiens. On relève, que la plupart des solutions proposées traitant les deux connaissances (probabiliste et floue) séparément. Cependant, ces deux types de connaissances apparaissent simultanément, très souvent liés et complémentaires dans le monde réel.

Notre travail présente un point fort par rapport aux travaux connexes existants : nous avons traité simultanément les connaissances floues et probabilistes afin d'adapter une ontologie floue probabiliste.

2.8 Conclusion

Ce chapitre est basé essentiellement sur les principaux aspects des réseaux bayésiens et des réseaux bayésiens flous. Ensuite, nous avons présenté les différents langages d'ontologie probabiliste. Enfin, quelques travaux connexes ont été proposés et analysés.

Chapitre 3

CONCEPTION DU SYSTÈME

3.1 Introduction

À la fin de la présentation de tout ce qui est relatif aux éléments essentiels de notre système (chapitres précédents), que constituent les ontologies floues et les réseaux bayésiens flous, nous ajoutons, l'étude de cas qui nous concerne, celle des maladies cardiovasculaires pour la construction de l'ontologie floue. Viendra ensuite la phase de la conception par laquelle nous pourrions résoudre notre problématique.

Au cours de ce chapitre, nous allons exposer la motivation de notre travail ainsi que la maladie de notre étude de cas. De plus, nous allons montrer la construction de l'ontologie floue. Nous présenterons ensuite le schéma global de notre système, qui sera suivi d'une explication de chaque étape de notre système.

3.2 Motivation

Les ontologies sont fréquemment utilisées dans de nombreuses disciplines différentes pour modéliser et exprimer les connaissances. Bien qu'une ontologie soit une spécification formelle d'une conceptualisation commune, en raison de leur incomplétude, de leur imprévisibilité, de leur imprécision, etc., les ontologies sont incapables de définir et d'exprimer formellement des informations incertaines.

Les ontologies sont devenues une question importante en raison d'une meilleure représentation et d'un meilleur raisonnement en matière d'incertitude. À cette fin, deux classes importantes de langages ont été distinguées : les ontologies probabilistes, qui sont typiquement basées sur les réseaux bayésiens pour traiter l'aléatoire, et les ontologies floues, qui sont typiquement basées sur l'imprécision et la théorie consensuelle floue, la majorité des domaines que nous essayons fréquemment de modéliser.

Les réseaux bayésiens peuvent représenter les connaissances incertaines dues à l'incomplétude, au hasard, etc. (connaissances probabilistes), mais elles sont incapables de traiter les

connaissances floues et/ou imprécises. Les réseaux bayésiens flous ont été proposés comme une solution hybride en intégrant l'imprécision dans les réseaux bayésiens standards afin de bénéficier des avantages de chaque formalisme. La logique floue, quant à elle, est considérée comme idéale pour représenter des connaissances floues et tirer des conclusions provisoires à partir de connaissances précises, mais elle ne peut pas représenter des connaissances probabilistes ni en tirer des conclusions.

De plus, les réseaux bayésiens flous sont capables de faire des raisonnements probabilistes basés sur des observations floues (c'est-à-dire que si un nœud est flou pendant l'inférence, il peut prendre plusieurs états en même temps, mais avec un certain degré d'appartenance). Comme le web sémantique englobe à la fois des connaissances probabilistes et floues, le système proposé dans cette recherche vise à prendre en compte une variété d'incertitudes et/ou de défauts liés à l'information. En se basant sur les réseaux bayésiens flous et les ontologies floues, nous y arriverons en convertissant une ontologie floue bien formée (Fuzzy OWL 2), qui contient des informations floues encodées à l'aide du langage fuzzy OWL 2, en une ontologie pouvant permettre la représentation de connaissances probabilistes dans un RBF.

Celle-ci est une solution générique, afin de bien présenter notre travail, nous avons réalisé une étude de cas qui s'intéresse aux maladies cardiovasculaires.

Les maladies cardiovasculaires (MCV) sont des troubles du cœur et des vaisseaux sanguins, notamment les maladies coronariennes (infarctus), les maladies cérébrovasculaires (AVC), l'insuffisance cardiaque (IC) et d'autres types de pathologies. Au total, les maladies cardiovasculaires causent chaque année la mort d'environ 17 millions de personnes dans le monde, le nombre de décès étant en hausse au Royaume-Uni pour la première fois depuis 50 ans. En particulier, l'insuffisance cardiaque survient lorsque le cœur est incapable de pomper suffisamment de sang vers l'organisme, et elle est généralement causée par le diabète, l'hypertension artérielle ou d'autres infections ou maladies cardiaques [41].

3.3 Dataset

Nous présentons ci-dessous le dataset utilisé. Il comporte les dossiers médicaux de 299 patients souffrant d'insuffisance cardiaque recueillis à l'Institut de cardiologie de Faisalabad et à l'hôpital Allied de Faisalabad (Punjab, Pakistan), entre avril et décembre 2015 [41].

Le dataset utilisé dans notre système contient 13 caractéristiques, qui rapportent des informations cliniques, corporelles et sur le mode de vie (Tableau 3.1). Certaines caractéristiques sont binaires : anémie, hypertension artérielle, diabète, sexe et tabagisme.

Le tableau 3.1 représente les significations, les unités de mesure et les intervalles de chaque caractéristique du dataset.

Caractéristique	Explication	La mesure	Intervalle
Âge	Âge du patient	Années	[40,...., 95]
Anémie	Diminution des globules rouges ou de l'hémoglobine	booléen	0, 1
Hypertension artérielle	Si un patient souffre d'hypertension	booléen	0, 1
Créatinine phosphokinase (CPK)	Niveau de l'enzyme CPK dans le sang	µg/L	[23,...., 7861]
Diabète	Si le patient est diabétique	booléen	0, 1
La fraction d'éjection (EF)	Pourcentage de sang sortant du cœur à chaque contraction	Pourcentage	[14,...., 80]
Sexe	Femme ou homme	Binaire	0, 1
Plaquettes	Plaquettes dans le sang	kiloplaquettes/mL	[25.01,...., 850.00]
Créatinine sérique (SC)	Niveau de créatinine dans le sang	mg/dL	[0.50,...., 9.40]
sodium sérique	Niveau de sodium dans le sang	mEq/L	[114,...., 148]
Fumeur	Si le patient fume	booléen	0, 1
Temps	Période de suivi	Journées	[4,....,285]
événement de décès (cible)	Si le patient est décédé pendant la période de suivi	booléen	0, 1

TABLE 3.1 – Significations, unités de mesure et intervalles de chaque caractéristique du dataset.

3.4 Construction de l'ontologie floue pour la maladie cardiovasculaire

Nous avons suivi la méthodologie FODM afin de construire l'ontologie floue de la maladie cardiovasculaire formalisée en Fuzzy-OWL 2.

Phase 1 : Objectif et portée de l'ontologie.

Afin de décrire le domaine cible, la motivation est de modéliser le diagnostic de la maladie cardio-vasculaire. Une liste de questions est présentée ci-dessous et les réponses à ces questions peuvent aider à déterminer le but et la portée de l'ontologie de façon précise.

Q1 : Quel type d'information doit être modélisé ?

La maladie cardiovasculaire qui va être inspectée et ses différents facteurs de risques sont l'intérêt de la modélisation.

Q2 : L'ontologie est-elle choisie comme technique de modélisation plutôt que d'autres solutions ?

Comme l'interopérabilité entre les médecins et les patients est censée être réalisée par l'utilisation d'un vocabulaire formalisé pour exprimer les facteurs risques de la maladie cardiovasculaire, l'ontologie est la technique de modélisation la plus prometteuse pour fournir cette spécification de conceptualisation.

Q3 : Quel sera le type de l'ontologie envisagée ?

Le type d'ontologie est une ontologie de domaine elle est utilisée pour représenter un domaine sous forme de base de connaissances.

Q4 : Qui participera au développement de l'ontologie et quels rôles joueront-ils ?

Les ingénieurs en ontologie pourraient être le principal participant, tandis que les experts (les médecins) pourraient apporter des connaissances approfondies à la modélisation.

Q5 : Comment les différentes personnes impliquées dans le développement peuvent-elles faciliter une collaboration étroite afin d'assurer le succès du développement de l'ontologie ?

Les ingénieurs en ontologie collecteront les connaissances auprès des experts. En conceptualisant les connaissances obtenues, un cadre général de l'ontologie pourra être construit.

Phase 2 : Identifier le besoin de flou.

Selon la description du domaine cible, le diagnostic médical de la maladie cardiovasculaire, et ses facteurs de risque sont les informations de modélisation prévues. Cependant il est obligatoire de prendre en compte l'implication de l'incertitude et surtout le flou donc il est à noter de concevoir des concepts flous sur chaque concept précis pour prendre en compte tous les cas de degrés d'appartenance d'une instance (patient) à ce concept par exemple pour l'âge, on s'y prendra ainsi en créant les concepts flous suivants :low age, high age, ainsi que les attributs flous spécifiques à ces derniers, ce qui fait que chaque patient aura des degrés d'appartenance à chacun de ces concepts.

Sur la base de cette analyse, le flou est nécessaire pour gérer l'imprécision inhérente à la classification de la maladie cardiovasculaire chez les patients. La conclusion tirée dans cette phase est qu'une ontologie floue, au lieu d'une ontologie classique, est déterminée pour modéliser le domaine de la maladie cardiovasculaire.

Phase 3 : Déterminer les informations floues et précises.

Dans cette étape, il convient de faire une distinction claire entre les informations floues liées au flou et certaines informations. Après avoir recueilli les connaissances et les suggestions des médecins, les ingénieurs ontologues établissent un diagnostic précis de la maladie entre les informations floues et les informations certaines. Les résultats sont présentés comme suit :

Informations précises	Informations floues
sexe	age
anémie	créatinine phosphokinase
hypertension artérielle	fraction d'éjection
tabagisme	plaquettes
	créatinine sérique
	sodium sérique
	temps (la période de suivi)

TABLE 3.2 – Les informations floues et précises.

Phase 4 : Envisager de réutiliser les ontologies existantes.

Aucune ontologie floue n'a été réutilisée dans la construction alors les phases 5 et 6 peuvent être ignorées.

Phase 7 : Définir les éléments d'ontologie floue.

Dans la phase 3, la base de connaissances de la maladie cardiovasculaire a été divisée en deux catégories : les informations précises et les informations floues. Afin de représenter les informations vagues et imprécises à l'aide de l'ontologie floue, différents éléments de l'ontologie floue sont définis dans cette phase (voir le tableau).

En principe, la définition du type de données floues vise à fournir une spécification correspondante pour le format de données de concept flou, tel que Low_CPK limite le niveau d'enzyme CPK dans le sang, et génère également une probabilité spécifique pour que l'enzyme soit classée comme LowCPK. Ainsi, une propriété de données floues, has_CPK, doit être définie afin de spécifier la relation entre les concepts flous (LowCPK, NormalCPK et HighCPK) et les types de données floues (Low_CPK, Normal_CPK et High_CPK).

On a obtenu les fonctions d'appartenance avec DATIL (DATatypes with Imprecision Learner) qui est un logiciel qui apprend automatiquement les types de données floues pour les ontologies floues à partir de différents types d'entrées. Il implémente plusieurs algorithmes d'apprentissage non-supervisé. L'algorithme d'apprentissage choisi est « Fuzzy C-means », qui fonctionne en attribuant un degré d'appartenance à chaque point de données correspondant à chaque centre de cluster sur la base de la distance entre le centre du cluster et le point de données. Plus les données sont proches du centre, plus leur appartenance au centre du cluster particulier est élevée.

concepts flous	types de données floues	fonction d'appartenance	axiome
age	Low_age	leftshoulder(40,95,51.59,71.51)	LowAge=age \cap \exists has_age.Low_age
	High_age	rightshoulder(40,95,51.59,71.51)	HighAge=age \cap \exists has_age.High_age
créatinine phosphokinase	Low_CPK	leftshoulder(23,7861, 301.12,2008.73)	LowCPK=CPK \cap \exists has_CPK.Low_CPK
	Normal_CPK	triangular(23,7861, 301.12,2008.73,6247.17)	NormalCPK=CPK \cap \exists has_CPK.Normal_CPK
	High_CPK	rightshoulder(23,7861, 2008.73,6247.17)	HighCPK=CPK \cap \exists has_CPK.High_CPK
fraction d'éjection	Low_EF	leftshoulder(14,80, 24.76,38.2)	LowEF=EF \cap \exists has_EF.Low_EF
	Normal_EF	triangular(14,80, 24.76,38.2,58.13)	NormalEF=EF \cap \exists has_EF.Normal_EF
	High_EF	rightshoulder(14,80, 38.2,58.13)	HighEF=EF \cap \exists has_EF.High_EF
plaquettes	Low_platelets	leftshoulder(25100,850000, 165171.78,268090.62)	LowPlatelets=platelets \cap \exists has_platelets.Low_platelets
	Normal_platelets	triangular(25100,850000, 165171.78, 268090.62,416777.08)	NormalPlatelets=platelets \cap \exists has_platelets.Normal_platelets
	High_platelets	rightshoulder(25100,850000, 268090.62,416777.08)	HighPlatelets=platelets \cap \exists has_platelets.High_platelets
créatinine sérique	Low_SC	leftshoulder(0.5,9.4, 1.04,2.34)	LowSC=SC \cap \exists has_SC.Low_SC
	Normal_SC	triangular(0.5,9.4, 1.04,2.34,7.12)	NormalSC=SC \cap \exists has_SC.Normal_SC
	High_SC	rightshoulder(0.5,9.4, 2.34,7.12)	HighSC=SC \cap \exists has_SC.High_SC
sodium sérique	Low_s_sodium	leftshoulder(113,148, 129.95,135.92)	LowSerumSodium= serum_sodium \cap \exists has_s_sodium.Low_s_sodium
	Normal_s_sodium	triangular(113,148, 129.95,135.92,140.81)	NormalSerumSodium= serum_sodium \cap \exists has_s_sodium.Normal_s_sodium
	High_s_sodium	rightshoulder(113,148, 135.92,140.81)	HighSerumSodium= serum_sodium \cap \exists has_s_sodium.High_s_sodium
temps	Low_time	leftshoulder(4,285, 68.17,210.9)	LowTime=time \cap \exists has_time.Low_time
	High_time	rightshoulder(4,285, 68.17,210.9)	HighTime=time \cap \exists has_time.High_time

TABLE 3.3 – Les concepts flous.

phase 8 : définir les éléments d'ontologie précis.

Dans cette phase, la partie restante de l'ontologie, à savoir les éléments ontologiques précis, doivent être défini pour modéliser des informations précises dans le domaine cible.

phase 9 : formalisation.

Dans ce cas d'utilisation, OWL 2 est choisi comme langage de formalisation pour représenter le modèle d'ontologie conçu. Pour effectuer facilement la transformation du modèle conceptuel en expressions formatées OWL 2, l'éditeur d'ontologie protégé et son extension Fuzzy OWL sont utilisés dans cette étape. Protégé permet de visualiser et de mettre en œuvre facilement l'ontologie floue de la maladie cardiovasculaire (voir la figure 3.1).

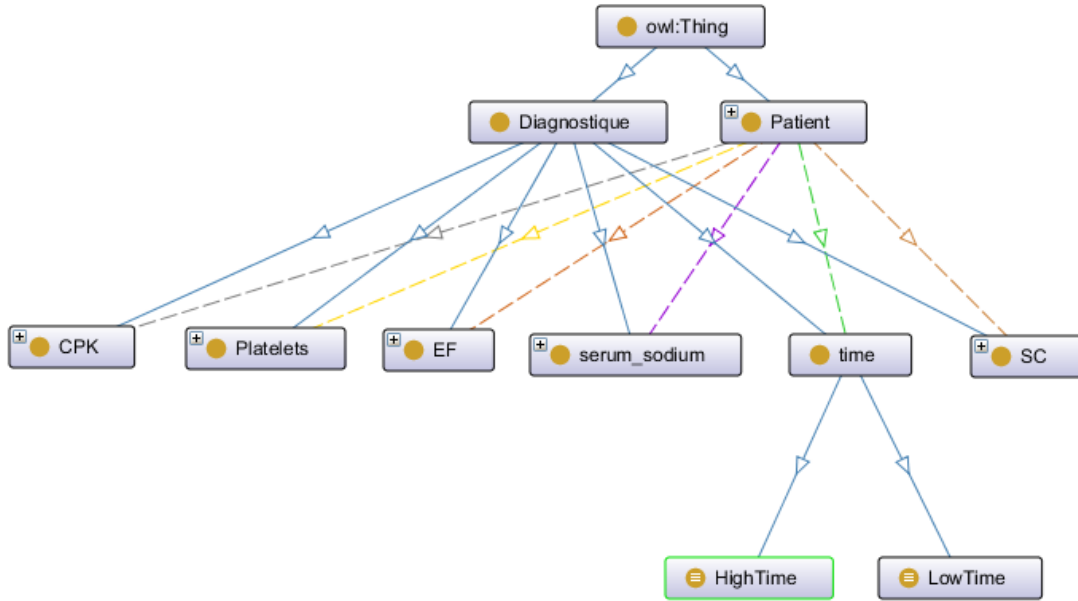


FIGURE 3.1 – L'ontologie floue de la maladie cardiovasculaire.

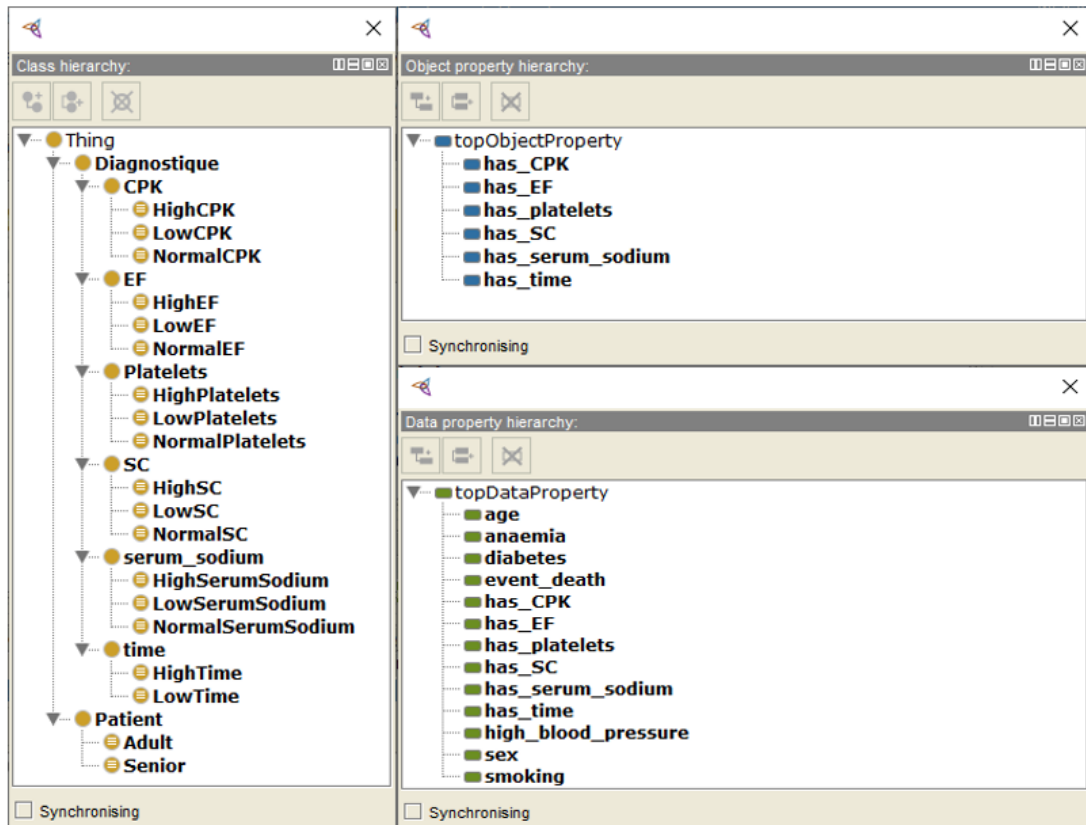


FIGURE 3.2 – Les classes, les propriétés d'objets, les propriétés de données de l'ontologie floue.

3.5 Aperçu de notre Système

La finalité de notre travail est de développer un système permettant l'intégration des connaissances probabilistes dans des ontologies floues, fondé sur une ontologie floue et un réseau bayésien flou.

Les différentes étapes de notre système sont illustrées dans la figure 3.3.

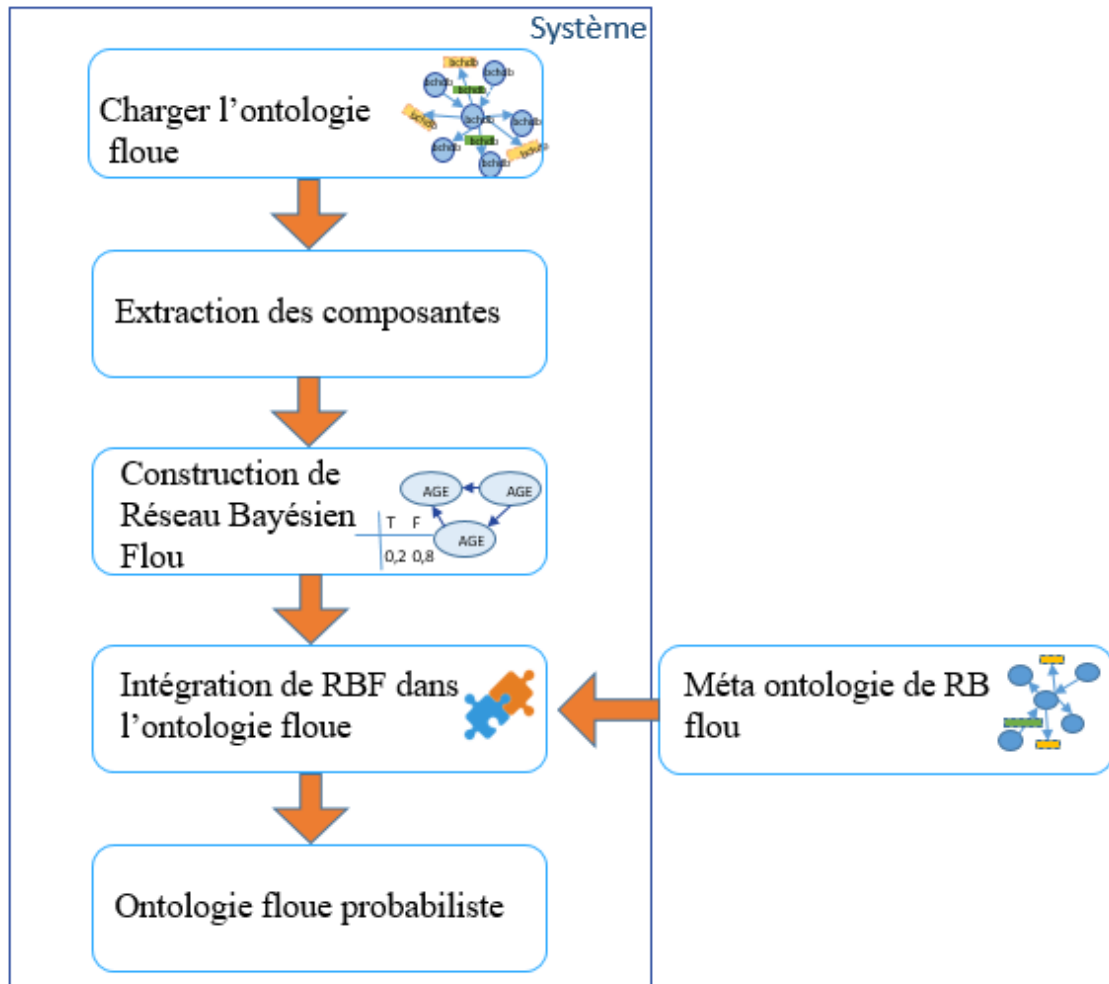


FIGURE 3.3 – Schéma global du système.

3.5.1 Charger l'ontologie floue

La lecture de l'ontologie floue permet à l'utilisateur de connaître ses composants (Class, Object property, Data property, ...) et effectuer des opérations dessus. L'ontologie floue a été construite dans la partie précédente, dans laquelle nous avons expliqué toutes ses étapes.

Par la suite, nous ferons la distinction entre les entrées et les sorties de cette étape.

Entrée : D'abord, nous avons besoin d'une ontologie floue qui est une ontologie classique avec une annotation "Fuzzy Label" 3.4 en plus du composant flou.

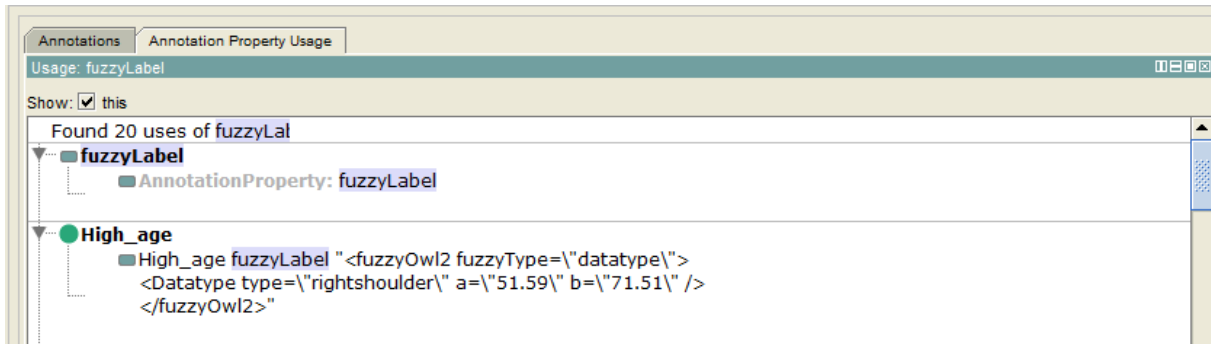


FIGURE 3.4 – Exemple d'une annotation "Fuzzy Label".

Sortie : La lecture de l'ontologie floue est bien effectuée et prête pour être utilisée.

3.5.2 Extraction des composantes

Le but de cette étape est d'extraire chaque composant de notre ontologie floue, et de fournir les éléments essentiels aux utilisateurs de notre système, afin de générer un réseau bayésien flou à partir de l'ontologie floue, et ainsi représenter les connaissances probabilistes qu'elle contient.

D'après les besoins de l'utilisateur, les propriétés des objets et les propriétés des données de l'ontologie floue servent de nœuds dans le réseau bayésien. Ce choix est basé sur le niveau de granularité de l'ontologie et les types de données qui représentent les états des nœuds dans le réseau bayésien, car il sert le même objectif dans les deux situations.

Les entrées et les sorties de cette étape :

Entrées : Comme entrée, nous utilise l'ontologie floue qui a été chargée dans l'étape précédente.

Sorties : La visualisation des différents composants de l'ontologie floue (Class, Data property, Object property et Data type) qui a été utilisée dans l'entrée. Comme le montre la figure 3.5.

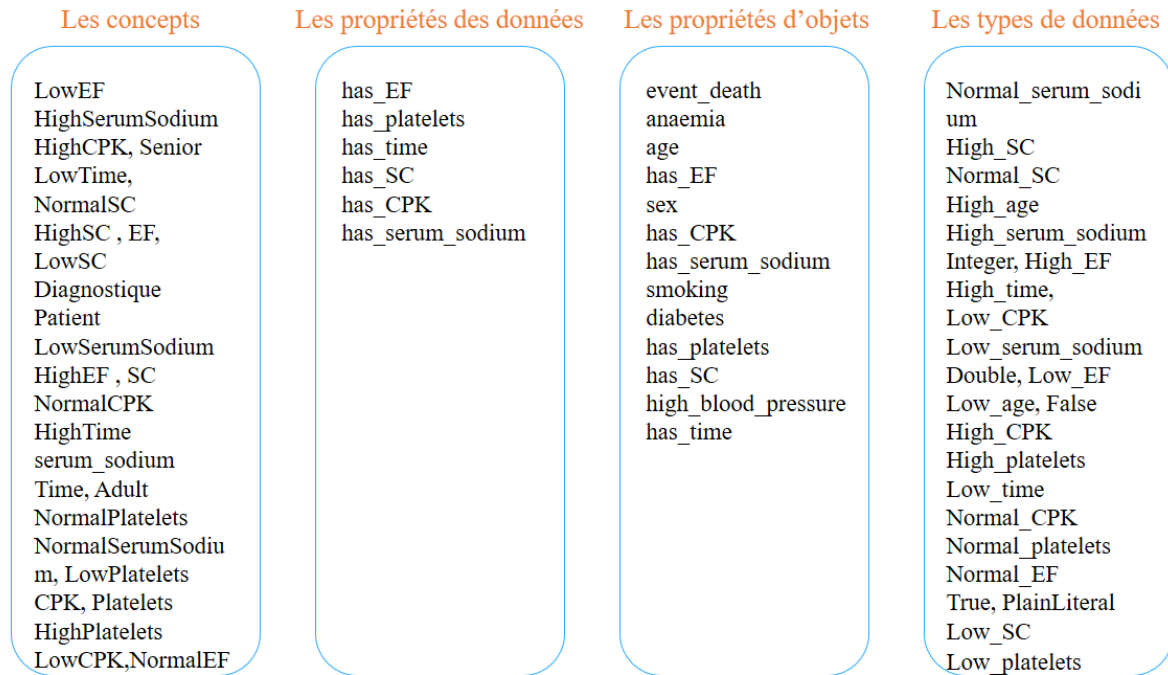


FIGURE 3.5 – Les composantes de l'ontologie floue.

3.5.3 Construction de Réseau Bayésien Flou

La construction du réseau bayésien flou est basée sur l'ontologie floue et ses composants flous.

L'apprentissage du réseau bayésien se divise en deux parties :

L'apprentissage de la structure

L'objectif de l'apprentissage des structures est de trouver, à partir des besoins de l'utilisateur et à l'aide de l'expert, une structure de graphe qui décrit le mieux un problème.

- Créer des nœuds : Parmi les nœuds, on distingue deux types, les nœuds flous et les nœuds précis : l'utilisateur choisit les nœuds (une propriété de données ou une propriété d'objet floue ou précis) qui répondent à son besoin.
- Attribuer les états au nœud choisi : Lorsqu'un nœud est choisi, l'utilisateur sélectionne les états souhaités pour chaque nœud dans une liste de types de données.
- Faire des relations entre les nœuds : Il suffit à l'utilisateur de sélectionner les nœuds père et fils pour établir une relation entre eux ; chaque nœud peut avoir un certain nombre de pères ou de fils.

Exemple :

Le nœud **Ejection_fraction** à un fils **DEATH_EVENT** et trois pères **Serum_creatinine**, **Smoking** et **Creatinine_phosphokin** .

La figure 3.6 représente la structure du réseau bayésien.

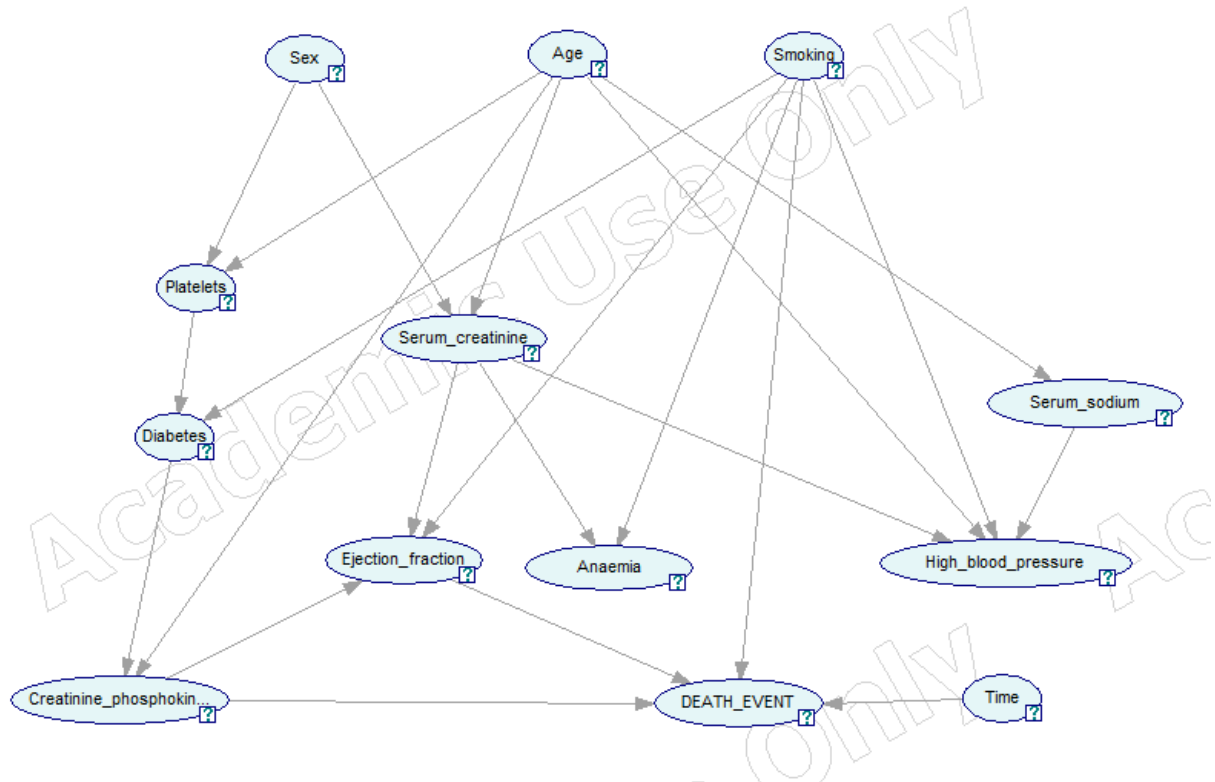


FIGURE 3.6 – La structure du réseau bayésien.

L'apprentissage des paramètres

Le processus d'apprentissage des paramètres consiste à estimer les probabilités de chaque nœud du réseau en utilisant l'approche **maximum de vraisemblance** (MV). La figure 3.7 indique l'apprentissage des paramètres.

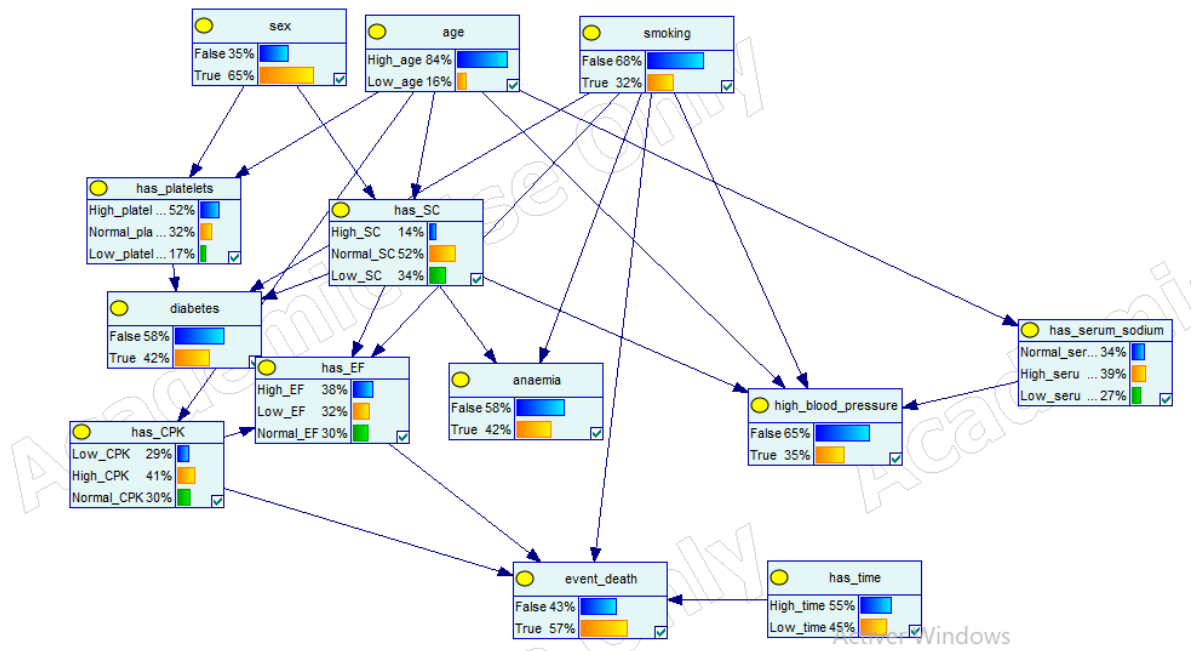


FIGURE 3.7 – Le réseau bayésien après l’apprentissage des paramètres.

3.5.4 Intégration des connaissances probabilistes dans l’ontologie floue

L’objectif de cette étape est de représenter le RBF construit à partir de l’ontologie floue dans une ontologie de haut niveau (méta-ontologie UOFBN), qui est utilisée pour représenter les connaissances probabilistes de manière formelle. Et intégrer cette dernière dans l’ontologie floue pour enrichir sa sémantique afin d’obtenir une ontologie floue probabiliste.

Méta-ontologie UOFBN

L’ontologie UOFBN est une ontologie de haut niveau des réseaux bayésiens flous qui permet de représenter sémantiquement les RBF, puis de les incorporer dans l’ontologie floue [42]. La méta-ontologie est représentée dans la figure 3.8.

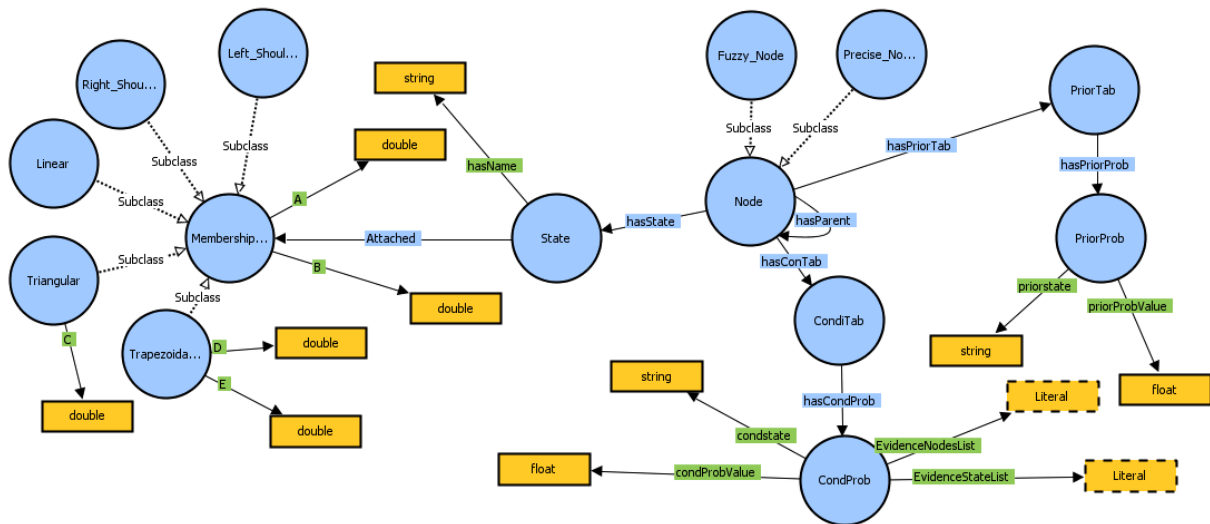


FIGURE 3.8 – Méta-ontologie UOFBN.

Les éléments de le méta-ontologie UOFBN sont les suivants :

- **Node** : une superclasse de "Precise Node" et "Fuzzy Node", qui comprend un nom, une table de probabilité et la possibilité d'avoir un ou plusieurs états comme instance de la classe State. Chaque nœud a la possibilité d'avoir un ou plusieurs parents ou enfants.
- **State** : Ensemble des états d'un nœud qui possède un nom et qui sont liés à un ou plusieurs fonctions d'appartenance.
- **Precise_Node** : Les nœuds précis (exact).
- **Fuzzy_Node** : Les nœuds flous connectés à des fonctions d'appartenance sont représentés par la classe Fuzzy_Node.
- **Membership_Function** : Représente la fonction d'appartenance de chaque nœud flou, peut prendre un ou plusieurs arguments comme instance et possède un nom.
- La table probabiliste de chaque nœud sera représentée en utilisant les classes "CondiTab" et "PriorTab".

Les composants de méta-ontologie UOFBN, sont représentés dans la figure 3.9 qui suit :

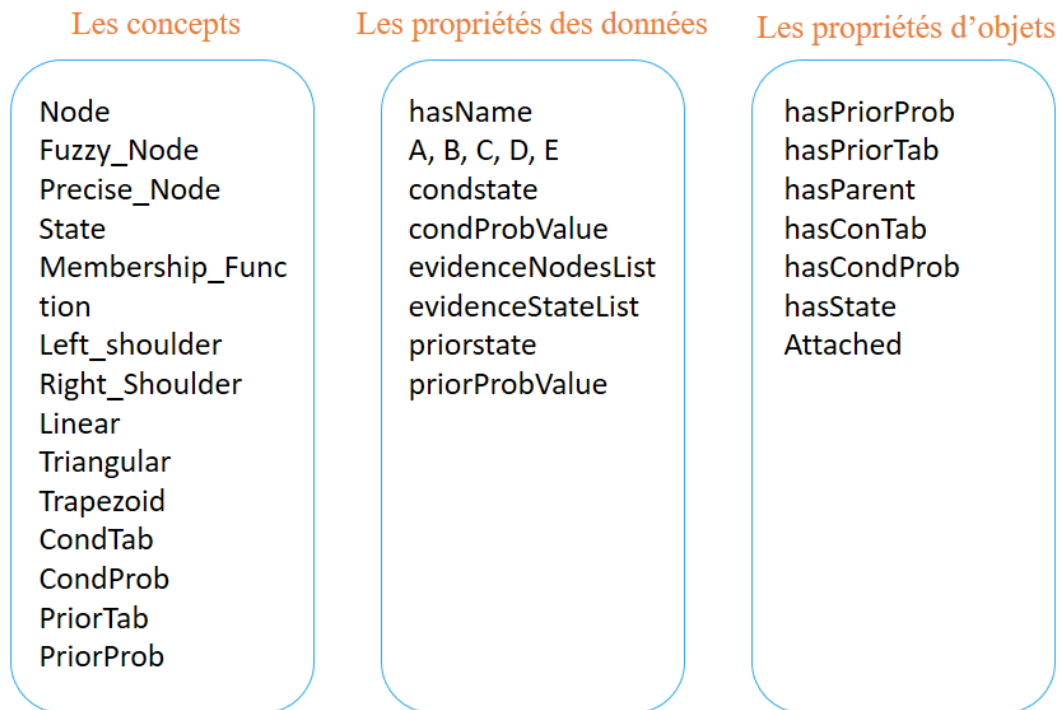


FIGURE 3.9 – Les composants de UOFBN.

En fait, les instances et les relations entre elles dans la méta-ontologie UOFBN seront utilisées pour représenter les nœuds, les états et les autres éléments du réseau bayésien construit à partir de l'ontologie floue.

Exemple d'instanciation :

Les figures 3.10, 3.11 représentent l'instanciation d'un nœud flou `has_time` dans le méta-ontologie.

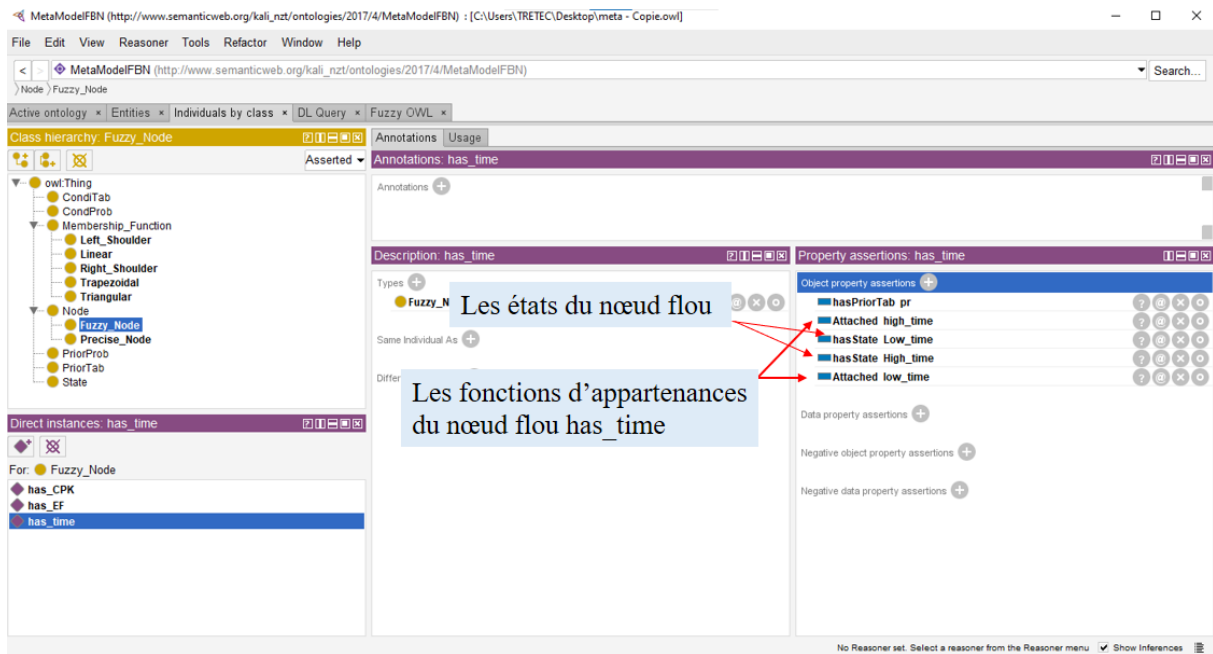


FIGURE 3.10 – Exemple des instances dans le méta-ontologie.

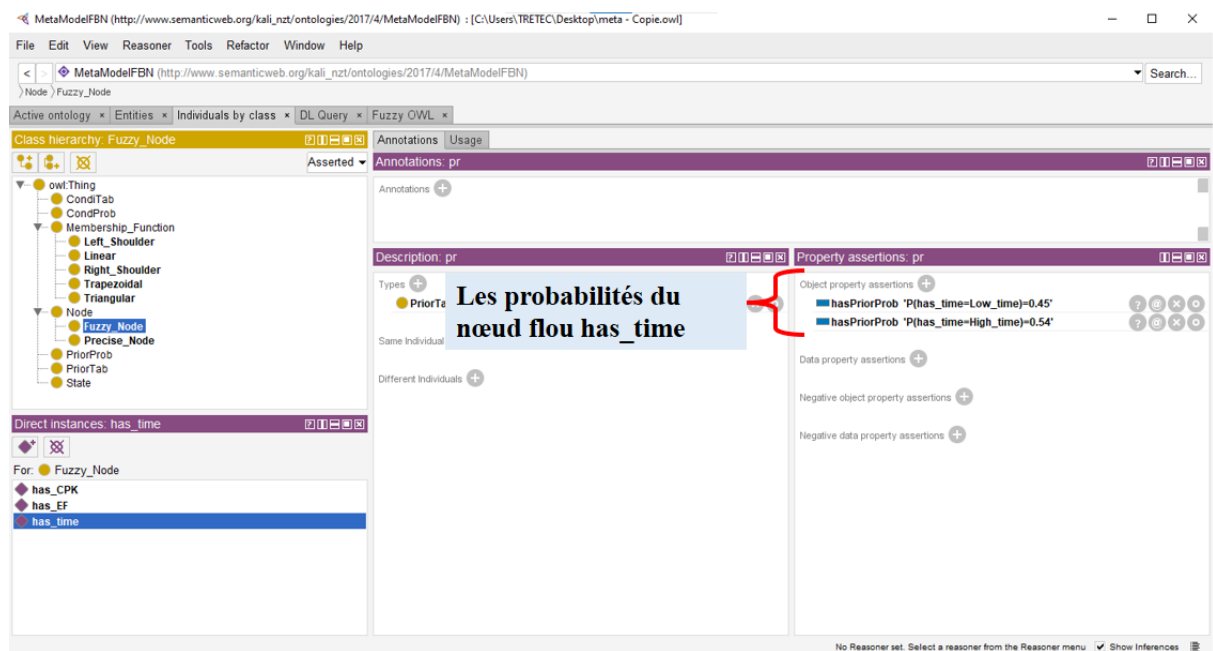


FIGURE 3.11 – Les instances de la classe PriorTab.

3.5.5 Ontologie floue et probabiliste

L'ontologie floue probabiliste est une extension de l'ontologie floue par l'intégration de méta-ontologie de RBF dedans. En réalité, il s'agit de fusionner les deux ontologies de telle sorte

que le nœud flou dans le méta-ontologie est identique à la classe floue dans l'ontologie floue. De ce fait, on obtient la nouvelle ontologie floue et probabiliste illustrée dans la figure 3.12 .

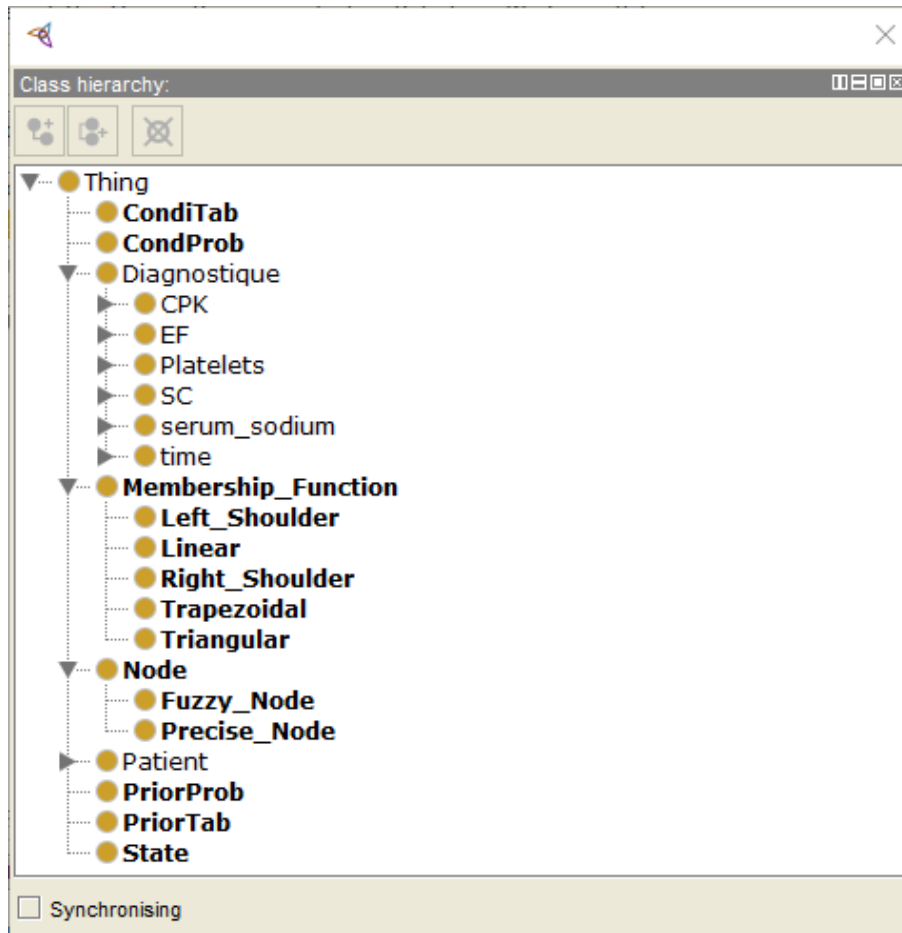


FIGURE 3.12 – La hiérarchie des classes de l'ontologie floue probabiliste.

3.6 Conclusion

Dans ce chapitre, nous avons illustré le processus de création de l'ontologie floue sur la maladie cardiovasculaire en utilisant la méthodologie FODM. Ensuite, nous avons expliqué chaque étape de notre système à partir d'une ontologie floue jusqu'à obtenir l'ontologie floue probabiliste. Ce chapitre est un point de départ pour la mise en œuvre de notre système.

Chapitre 4

IMPLEMENTATION ET TEST DU SYSTÈME

4.1 Introduction

Après l'étape conception du système, nous allons entamer l'étape finale qui consiste à l'implémentation de notre ontologie floue probabiliste.

Dans ce chapitre, nous allons décrire l'implémentation de notre système et montrer les différents outils utilisés, avec le test du système, afin de décrire visuellement l'implémentation de ce dernier, nous allons effectuer des captures d'écrans des différentes interfaces.

4.2 Implémentation de Notre Système

4.2.1 Outil de développement

Dans les lignes qui suivent, nous allons présenter les différents outils utilisés dans l'implémentation de notre système.

4.2.1.1 JAVA

JAVA a été développé par James Gosling chez Sun Microsystems Inc en 1995. Il s'agit d'un langage de programmation simple. Java facilite l'écriture, la compilation et le débogage des programmes. Il aide à créer du code réutilisable et des programmes modulaires. Java est un langage de programmation orienté objet, basé sur des classes, et conçu pour avoir le moins de dépendances possibles en matière d'implémentation. Il s'agit d'un langage de programmation polyvalent conçu pour que les développeurs puissent écrire une fois et exécuter n'importe quel code Java compilé sur toutes les plates-formes qui prennent en charge Java. Les applications Java sont compilées en code octet qui peut être exécuté sur toute machine virtuelle Java. La syntaxe de Java est similaire à celle de c/c++.

4.2.1.2 Protégé

Protégé est un système auteur pour la création d'ontologies. Il a été créé à l'université Stanford et est très populaire dans le domaine du Web sémantique et au niveau de la recherche en informatique. Protégé est développé en Java. Il est gratuit et son code source est publié sous une licence libre . Protégé peut lire et sauvegarder des ontologies dans la plupart des formats d'ontologies : RDF, RDFS, OWL, etc. La page de départ de protégé est illustrée dans la figure 4.1.

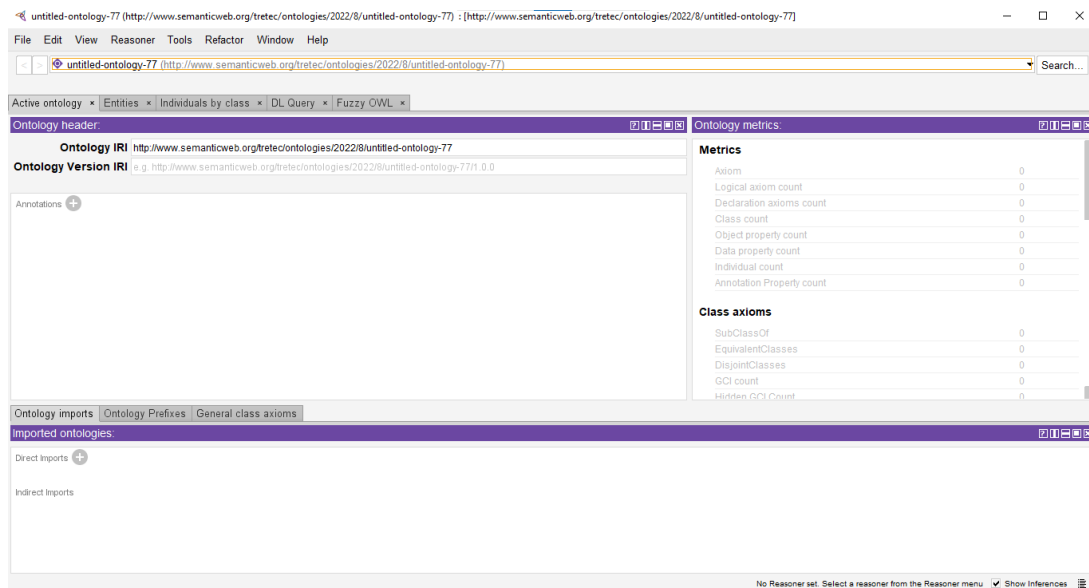


FIGURE 4.1 – La page de départ de protégé.

4.2.1.3 Netbeans

NetBeans IDE est un environnement de développement intégré (IDE) gratuit et open source qui vous permet de développer des applications de bureau, mobiles et Web. L'EDI prend en charge le développement d'applications dans divers langages, notamment Java, HTML5, PHP et C++. L'EDI fournit un support intégré pour le cycle de développement complet, de la création du projet au débogage, au profilage et au déploiement. L'EDI fonctionne sous Windows, Linux, Mac OS X et d'autres systèmes basés sur UNIX. La page d'accueil de Netbeans est décrite dans la figure 4.2.

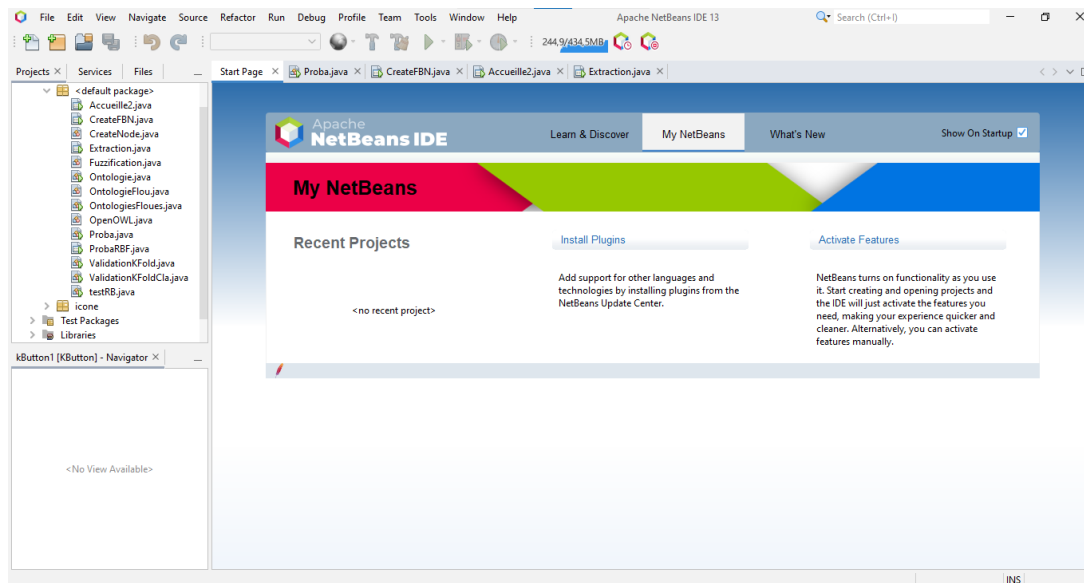


FIGURE 4.2 – La page d’accueil de Netbeans.

4.2.1.4 OWL API

L’API OWL, une interface de programmation d’application (API) de haut niveau qui prend en charge la manipulation de structures ontologiques, l’utilisation de moteurs de raisonnement, ainsi que l’analyse syntaxique et le rendu dans les syntaxes définies dans la spécification du W3C. L’API OWL est disponible depuis 2003 et a fait l’objet d’un certain nombre de révisions de conception, en particulier pour suivre l’évolution d’OWL lui-même. L’API OWL est implémentée en Java et est disponible en open source sous licence LGPL.

4.2.1.5 SMILE

SMILE (Structural Modeling, Inference, and Learning Engine) est une bibliothèque de fonctions entièrement indépendante de la plate-forme, qui met en œuvre des modèles graphiques probabilistes et décisionnels, tels que les réseaux bayésiens, les diagrammes d’influence et les modèles d’équations structurelles. Ses fonctions individuelles, définies dans l’interface API (Applications Programmer Interface) de SMILE, permettent de créer, d’éditer, de sauvegarder et de charger des modèles graphiques, et de les utiliser pour le raisonnement probabiliste et la prise de décision dans des conditions incertaines.

4.2.1.6 GeNie

Genie (Graphical network interface) est un outil logiciel développé à l’Université de Pittsburgh pour Microsoft Windows et disponible gratuitement à l’adresse Genie. Il est utile pour l’analyse des décisions et pour représenter graphiquement l’union de la probabilité et des occurrences en réseau. En particulier, Genie peut être utilisé pour l’analyse de réseaux bayésiens, ou de graphes acyliques dirigés (c’est-à-dire que les occurrences dans un réseau d’événements

sont conditionnellement indépendantes les unes des autres). La figure 4.3 représente la page principale de GeNie.

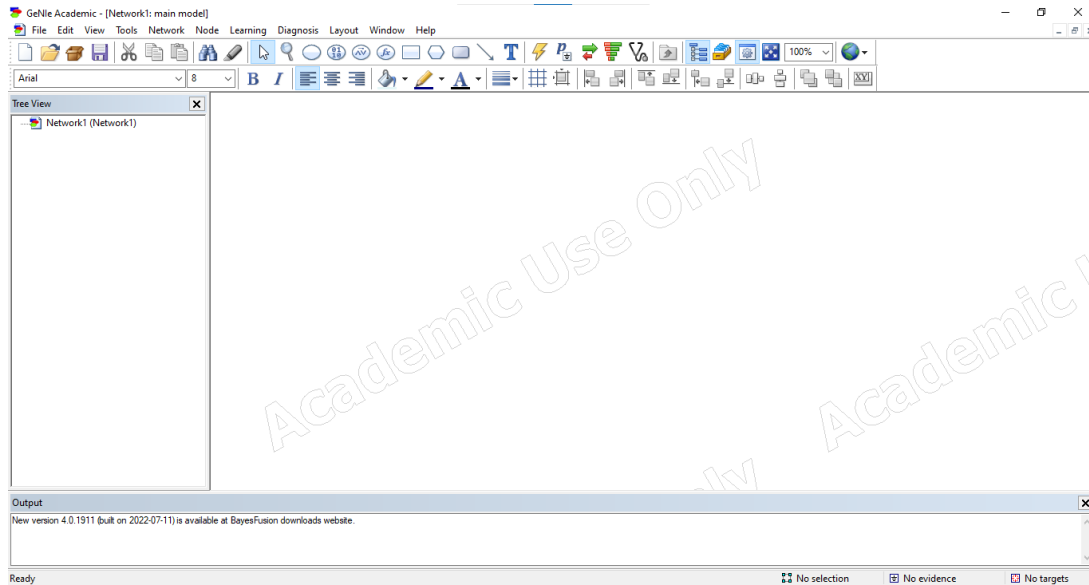


FIGURE 4.3 – la page principale de GeNie.

4.2.1.7 FUZZY OWL 2

FuzzyOWL2 (Fuzzy OWL 2) est un plugin pour Protege qui permet aux utilisateurs d'éditer, de sauvegarder des ontologies floues et de soumettre des requêtes au moteur d'inférence sous-jacent FuzzyDL. La fenêtre de Fuzzy OWL 2 est décrite dans la figure 4.4.

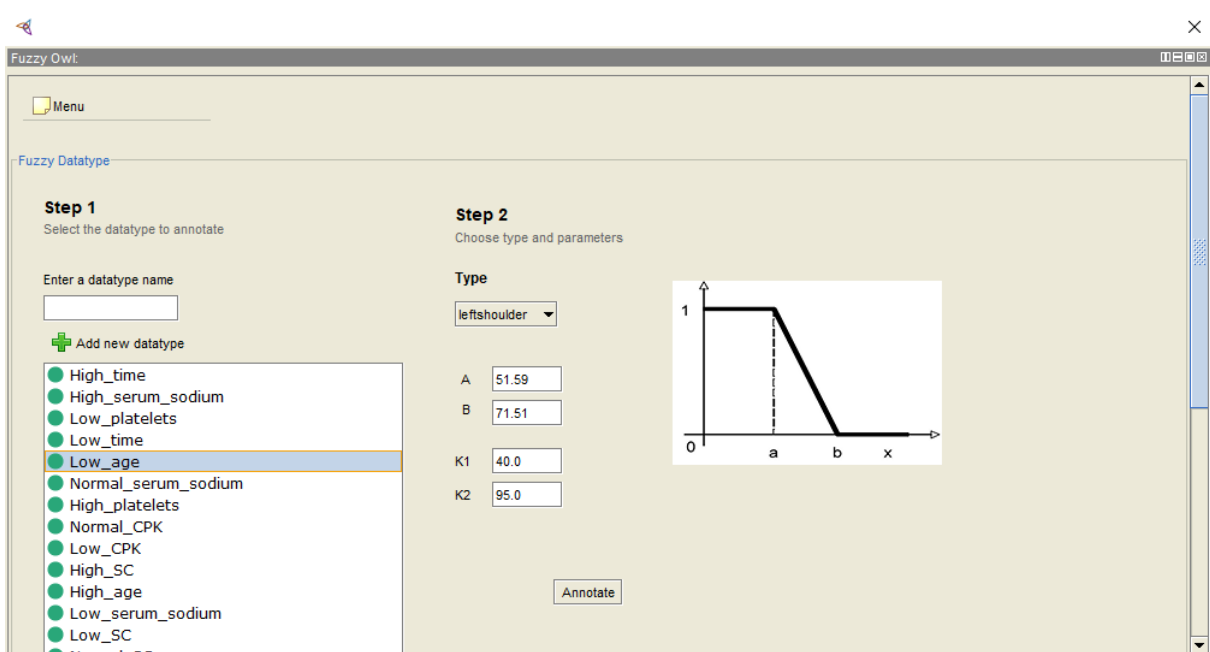


FIGURE 4.4 – La fenêtre de Fuzzy OWL 2.

4.2.2 Les interfaces graphiques

Nous allons présenter les différentes interfaces de notre application :

4.2.2.1 L'interface d'accueil

Il s'agit de l'interface principale de notre système. Elle contient le bouton « **Charger l'ontologie** » (voir la figure 4.5) qui permet d'importer l'ontologie floue de format OWL depuis un emplacement quelconque (voir la figure 4.6), 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.

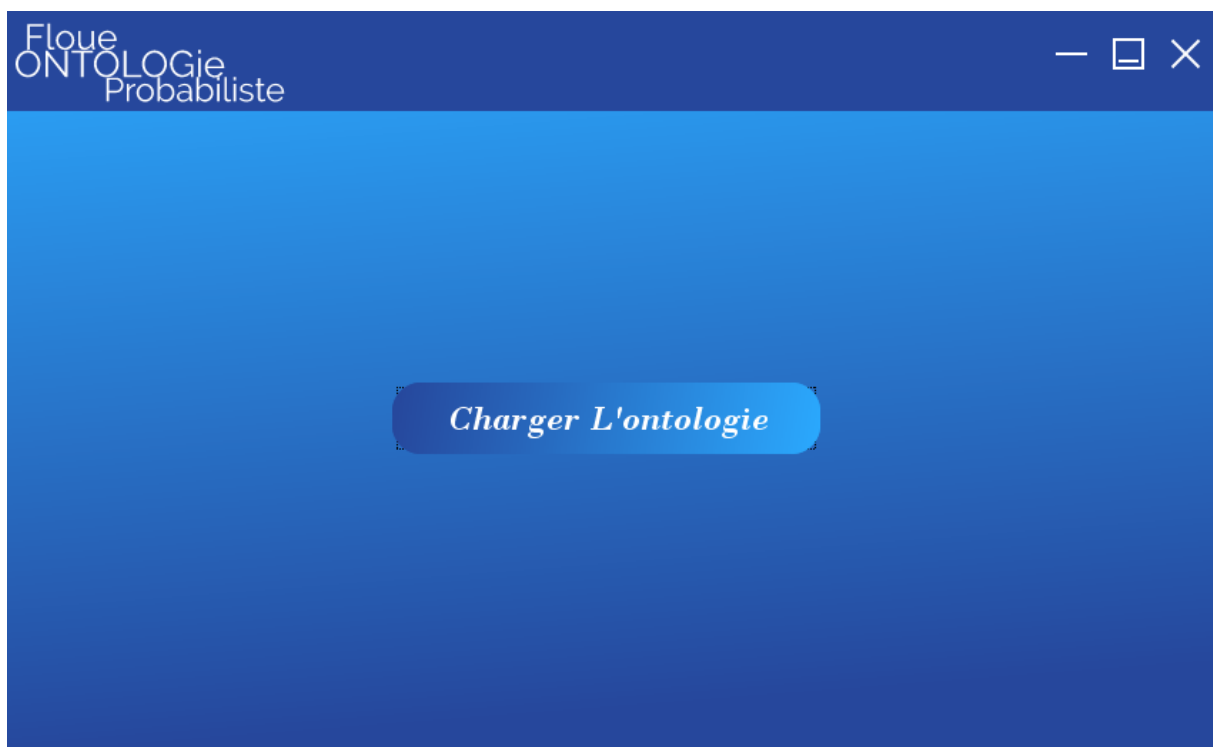


FIGURE 4.5 – L'interface d'accueil.

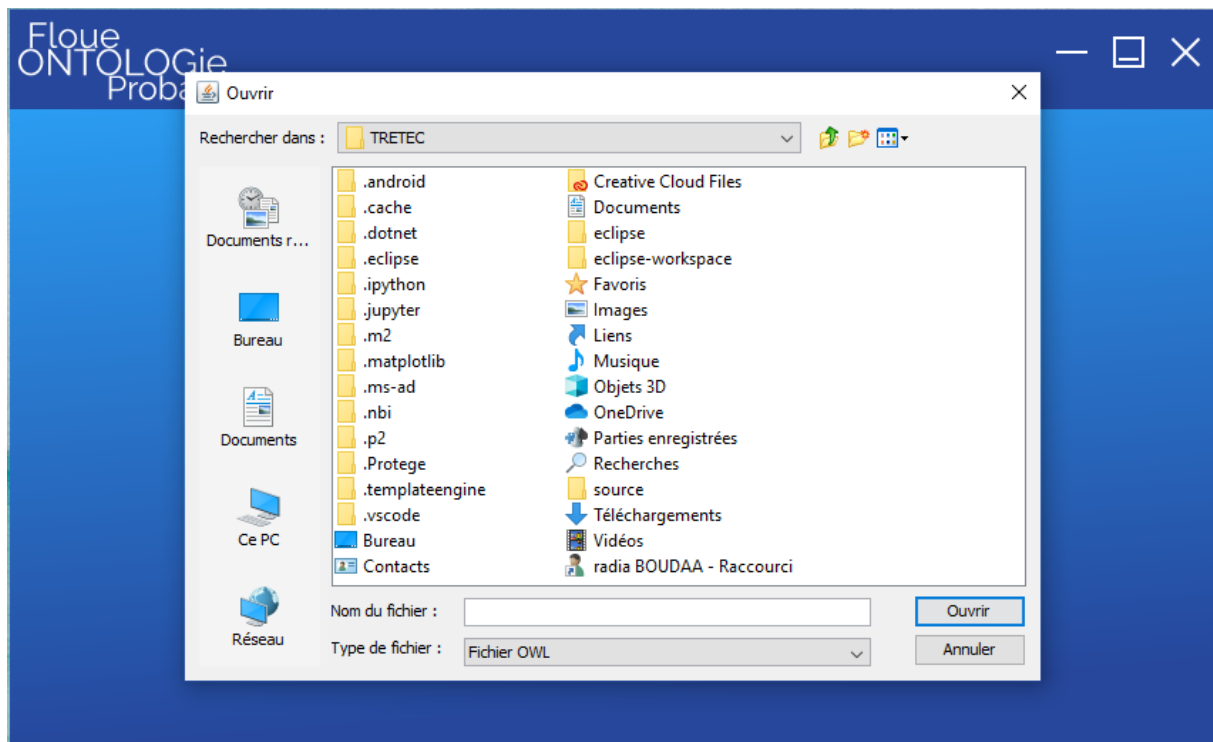


FIGURE 4.6 – Choisir le fichier OWL.

4.2.2.2 L'interface extraction des composants de l'ontologie floue

Cette interface s'affiche après avoir importé l'ontologie floue, elle indique les composants de l'ontologie floue (les classes, les propriétés d'objets, les propriétés des données, et les types de données). Nous trouvons le bouton « **créer réseau bayésien flou** » qui achemine l'utilisateur vers l'interface de la figure 4.8 pour lui permettre de construire un réseau bayésien flou à partir des composants de l'ontologie floue.

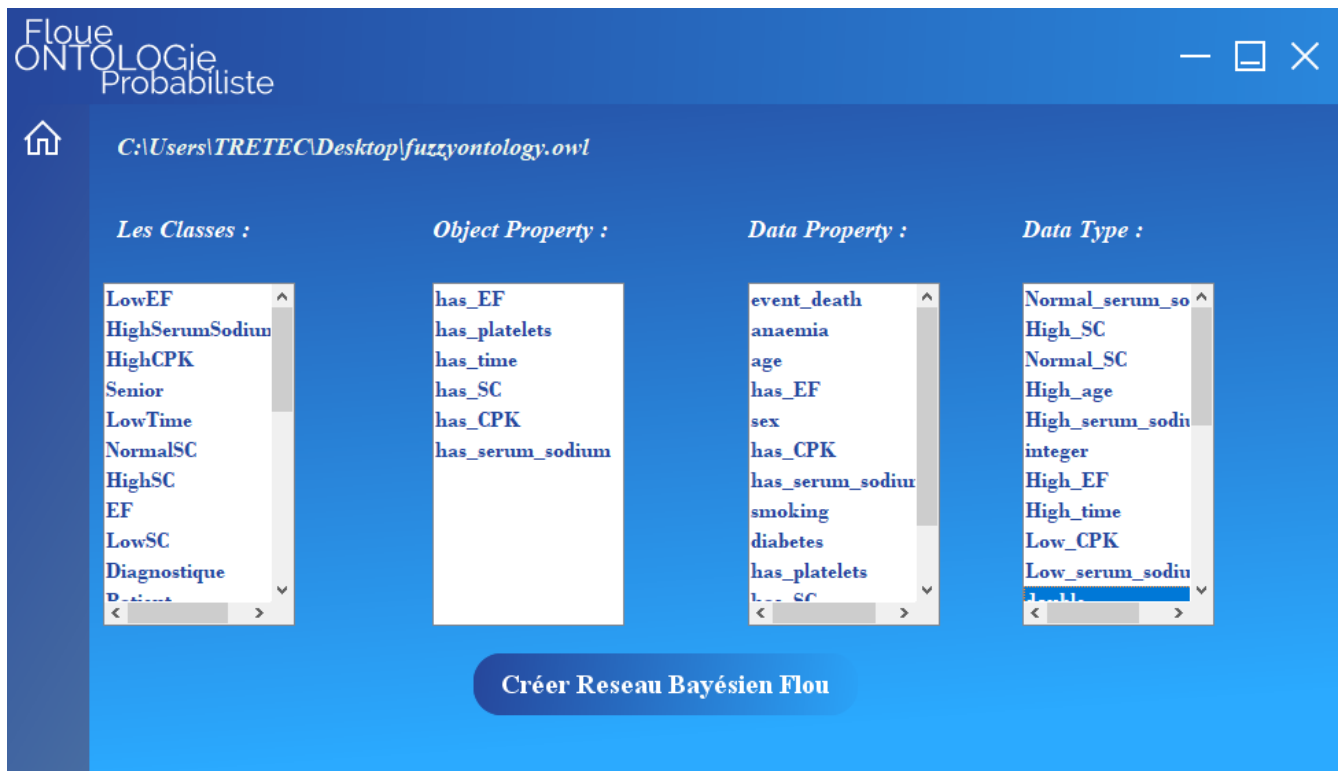


FIGURE 4.7 – L’interface d’extraction des composants de l’ontologie floue.

4.2.2.3 L’interface créer réseau bayésien flou

Cette interface permet la construction du réseau bayésien flou (apprentissage de la structure et de paramètre).

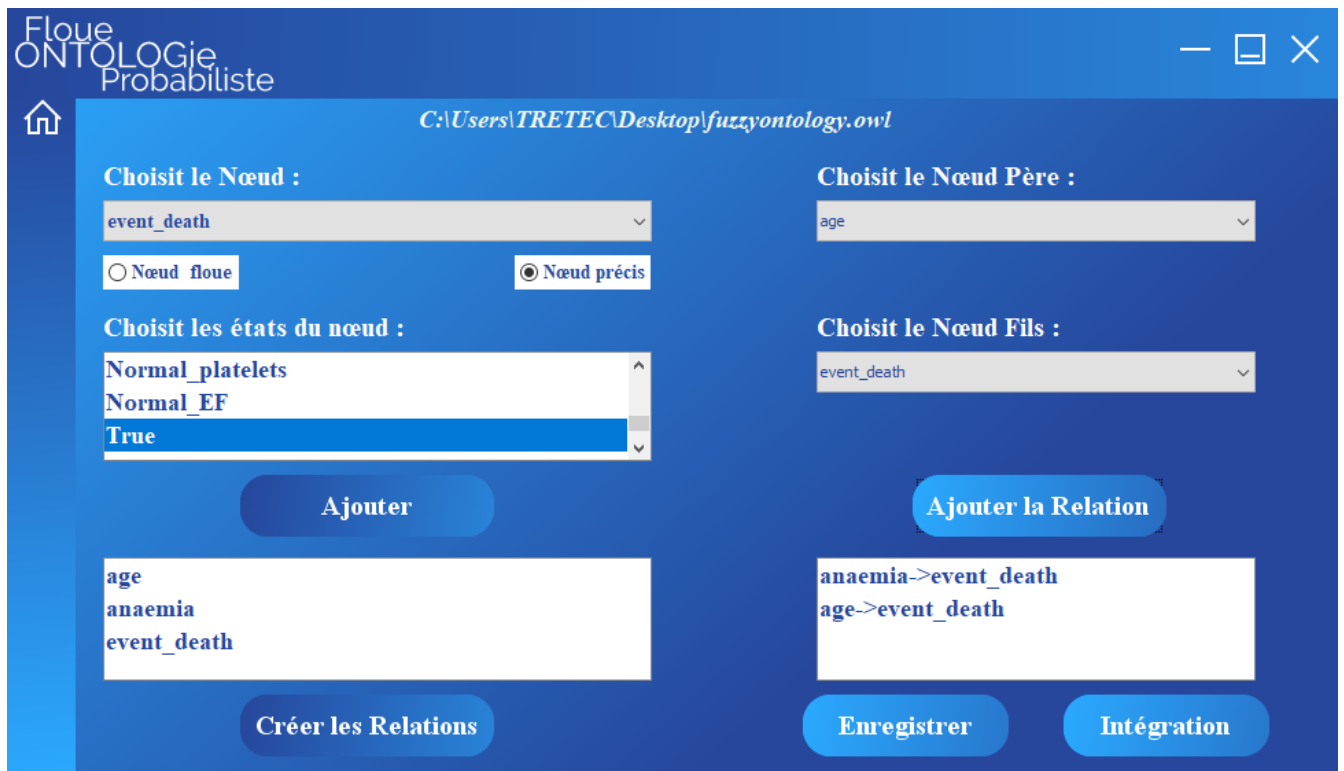


FIGURE 4.8 – L’interface de création de réseau bayésien flou.

Apprentissage de la structure :

-La création des nœuds avec ses différents états : Il faut d’abord choisir un nœud (précis ou flou), après quoi une liste d’états est affichée, et à l’utilisateur de choisir les états de ce nœud, et cliquer sur le bouton « **ajouter** » qui permet l’ajout d’un nœud avec ces états au RBF et affiche ce dernier dans une liste des nœuds créées pour passer à un autre nœud.

Une fois l’utilisateur créé tous les nœuds, le bouton « **créer les relations** » permet de passer à l’étape suivante.

-La création des relations entre les nœuds :

Choisir le nœud père : L’utilisateur doit choisir le nœud père à partir d’une liste qui contient tous les nœuds créés dans l’étape précédente.

Choisir le nœud fils : L’utilisateur doit choisir le nœud fils à partir d’une liste qui contient tous les nœuds créés dans l’étape précédente sauf le nœud choisi comme un père.

Le bouton « Ajouter la relation » : Permet d’ajouter la relation entre le nœud père et le nœud fils, et affiche cette relation dans une liste.

Apprentissage des paramètres :

une fois l’utilisateur termine l’étape de l’apprentissage de la structure il passe à l’apprentissage des paramètres. Cette étape est la dernière étape dans la construction du RBF, elle consiste à estimer les probabilités de chaque nœud du réseau en utilisant l’algorithme EM (Expectation

Maximisation) implémenté dans la classe EM, et enregistre le RBF créé dans un format "XDSL", en cliquant sur le bouton « **Enregistrer le RBF** ».

Le bouton « Intégration » : Permet de représenter le RBF dans une ontologie de haut niveau (méta-ontologie) pour se faire nous avons un méta-modèle pour un réseau bayésien flou illustré dans la figure 3.8. Les composants du réseau bayésien construit à partir de l'ontologie floue seront représentés sous forme d'instances et des relations entre ces derniers dans le méta-ontologie. Ensuite, il intègre ce dernier dans l'ontologie floue de départ pour qu'on obtient à la fin une ontologie floue probabiliste.

La figure 4.9 représente les composants de l'ontologie floue probabiliste.

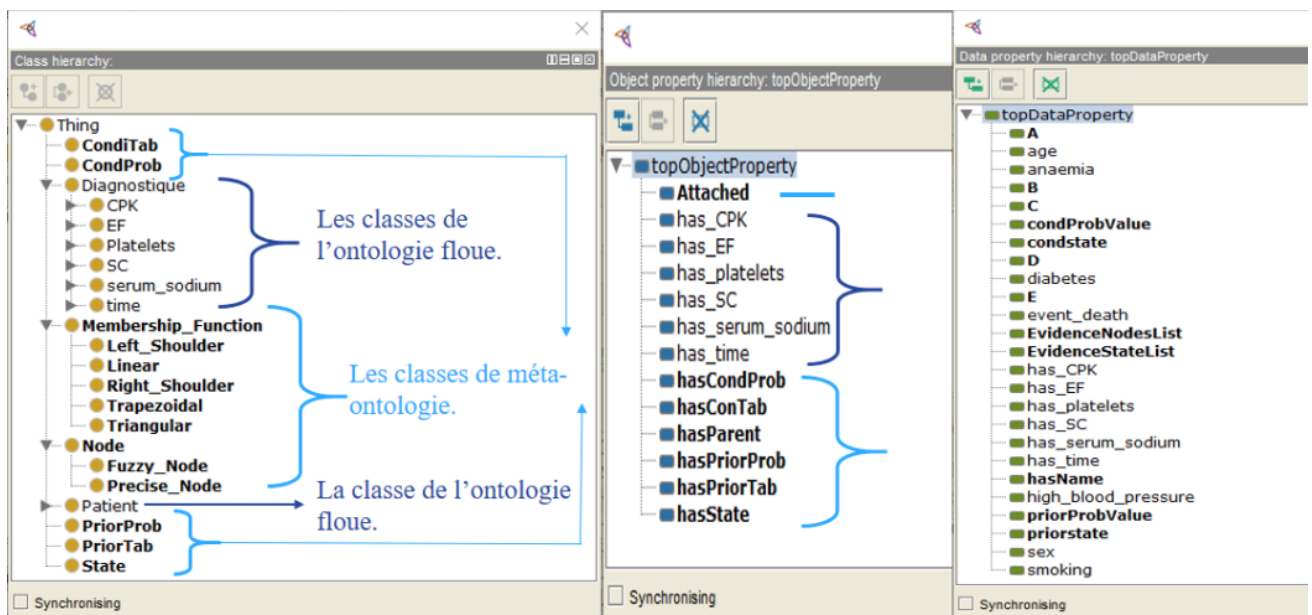


FIGURE 4.9 – Les classes, les propriétés d'objets, les propriétés de données de l'ontologie floue probabiliste.

4.3 Validation du système

Dans cette section on va faire la validation de notre système, afin de présenter les mesures de performance et les résultats obtenus.

4.3.1 Mesures d'évaluation

L'outil qui nous permet de mesurer les performances de notre système est la table de confusion ou tableau de contingence, elle sert à évaluer la qualité d'une classification. La matrice de confusion est un tableau à 4 valeurs représentant les différentes combinaisons de valeurs

réelles et valeurs prédites qui est l'inférence du noeud DEATH_EVENT de notre RBF dans ce cas, comme le montre la figure 4.10.

		Classe Prédite	
		Positive (Décès)	Négative (Non décès)
Classe Réelle	Positive	(TP) Vrais positifs	(FN) Faux négatifs
	Négative	(FP) Faux positifs	(TN) Vrais négatifs

FIGURE 4.10 – la table de confusion.

On classe les résultats en 4 catégories :

- **Vrais positifs (TP)** : La prédiction et la valeur réelle sont positives.
Exemple : Une personne morte et prévu morte.
- **Vrais négatifs (TN)** : La prédiction et la valeur réelle sont négatives.
Exemple : Une personne vivante et prévu vivante.
- **Faux positifs (FP)** : la prédiction est positive alors que la valeur réelle est négative.
Exemple : Une personne vivante et prévu morte.
- **Faux négatifs (FN)** : la prédiction est négative alors que la valeur réelle est négative.
Exemple : Une personne morte et prévu vivante.

4.3.2 Résultats obtenus

Nous avons choisi la méthode de validation croisée pour notre expérience, car elle nous permet d'utiliser l'ensemble des données pour l'apprentissage et la validation en même temps. En fait, nous divisons l'ensemble de données en K partis (folds). Nous choisissons une des K partis et l'utilisons comme partie de test pour chaque itération. La partie restante (l'union des K-1 restants) est utilisée pour l'apprentissage.

Pour chaque K sélectionné, nous avons exécuté une validation croisée K-fold pour notre RBF et un RB classique avec la même structure (au niveau des arcs). Nous avons ensuite effectué un ensemble de tests pour différentes valeurs de K. Ensuite, afin de comparer les résultats des tests de notre système avec ceux du RB classique, nous avons calculé les critères de performance. Les critères de performances sont :

- **L'accuracy** : Est une métrique qui décrit généralement les performances du modèle pour toutes les classes. Elle est utile lorsque toutes les classes sont d'importance égale. Elle est calculée comme le rapport entre le nombre de prédictions correctes et le nombre total de prédictions.

$$Accuracy = \frac{Vrai\ positif + Vrai\ négatif}{Total}$$

- **La précision** : Est calculée comme le rapport entre le nombre d'échantillons positifs correctement classés et le nombre total d'échantillons classés comme positifs (soit correctement, soit incorrectement). La précision mesure l'exactitude du modèle à classer un échantillon comme positif.

$$Precision = \frac{Vrai\ positif}{Vrai\ positif + Faux\ positifs}$$

- **Le rappel** : Est calculé comme le rapport entre le nombre d'échantillons positifs correctement classés comme positifs et le nombre total d'échantillons positifs. Le rappel mesure la capacité du modèle à détecter les échantillons positifs. Plus le rappel est élevé, plus le nombre d'échantillons positifs détectés est important.

$$Recall = \frac{Vrai\ positif}{Vrai\ positif + Faux\ négatifs}$$

Nous présentons ci-dessous les résultats obtenus après les tests : Le tableau 4.1 représente la comparaison des accuracy par les classifieurs pour K= 5, K=10 et la moyenne :

	Réseau bayésien flou	Réseau bayésien classique
K=5	85.96	80.61
K=10	87.68	83.17
moyenne	83.55	82.22

TABLE 4.1 – Tableau de comparaison des 'Accuracy' (%) obtenues par chaque classifieur.

La figure 4.11 illustre un histogramme pour les valeurs d'accuracy en fonction des K pour les 2 classifieurs :

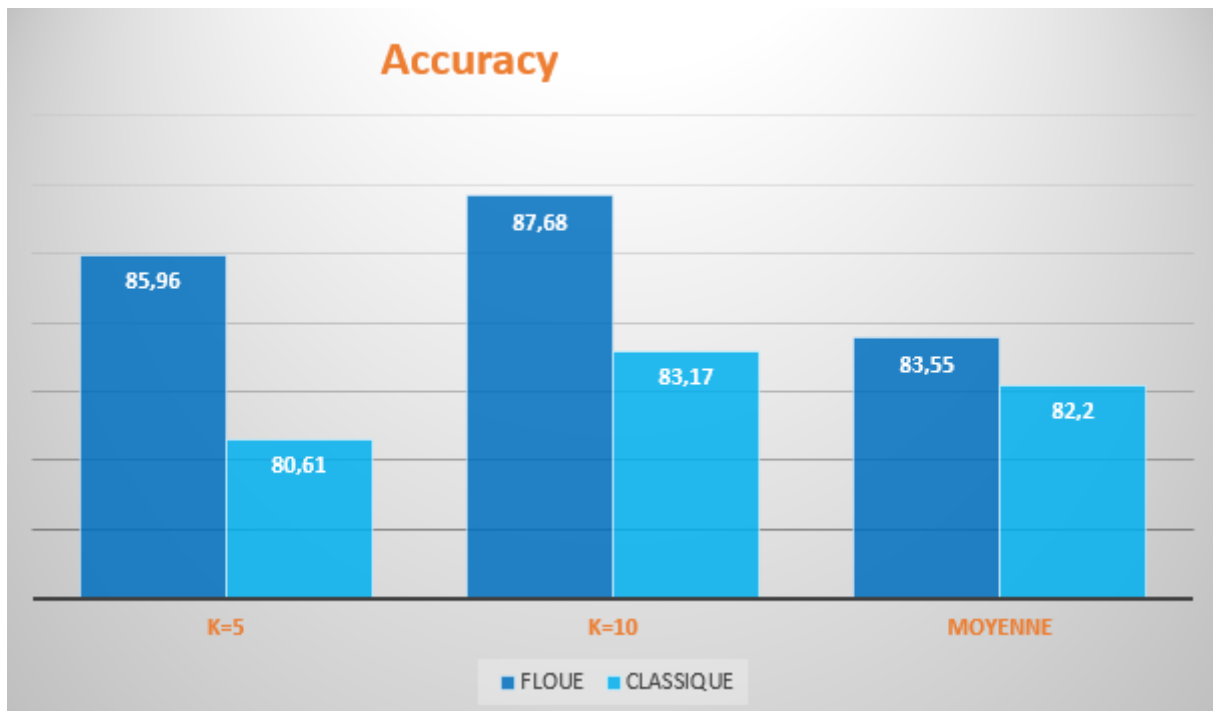


FIGURE 4.11 – Histogramme de l'accuracy.

Le tableau 4.3 représente la comparaison des rappels par les classifieurs pour $K = 5$ et $K = 10$:

	Réseau bayésien flou	Réseau bayésien classique
K=5	87.24	82.55
K=10	86.42	83.05
moyenne	85.38	81.05

TABLE 4.2 – Tableau de comparaison des 'Rappels' (%) obtenues par chaque classifieur.

La figure 4.12 illustre un histogramme pour les valeurs de rappel en fonction des K pour les 2 classifieurs :

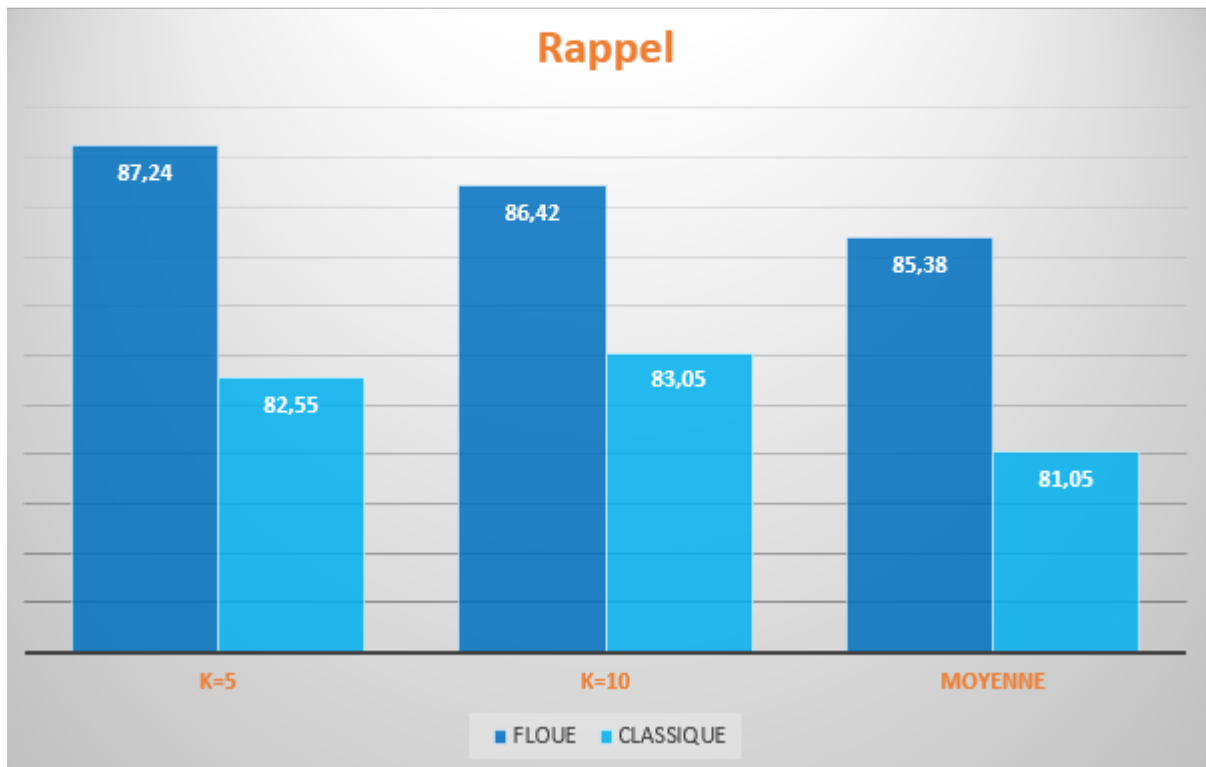


FIGURE 4.12 – Histogramme de rappel.

Le tableau 4.3 représente la comparaison des précisions par les classifieurs pour K =5 et K=10 :

	Réseau bayésien flou	Réseau bayésien classique
K=5	88.07	82.85
K=10	89.06	79.66
moyenne	85.11	80.76

TABLE 4.3 – Tableau de comparaison des 'Précisions' (%) obtenues par chaque classifieur.

La figure 4.13 illustre un histogramme pour les valeurs de précision en fonction des K pour les 2 classifieurs :

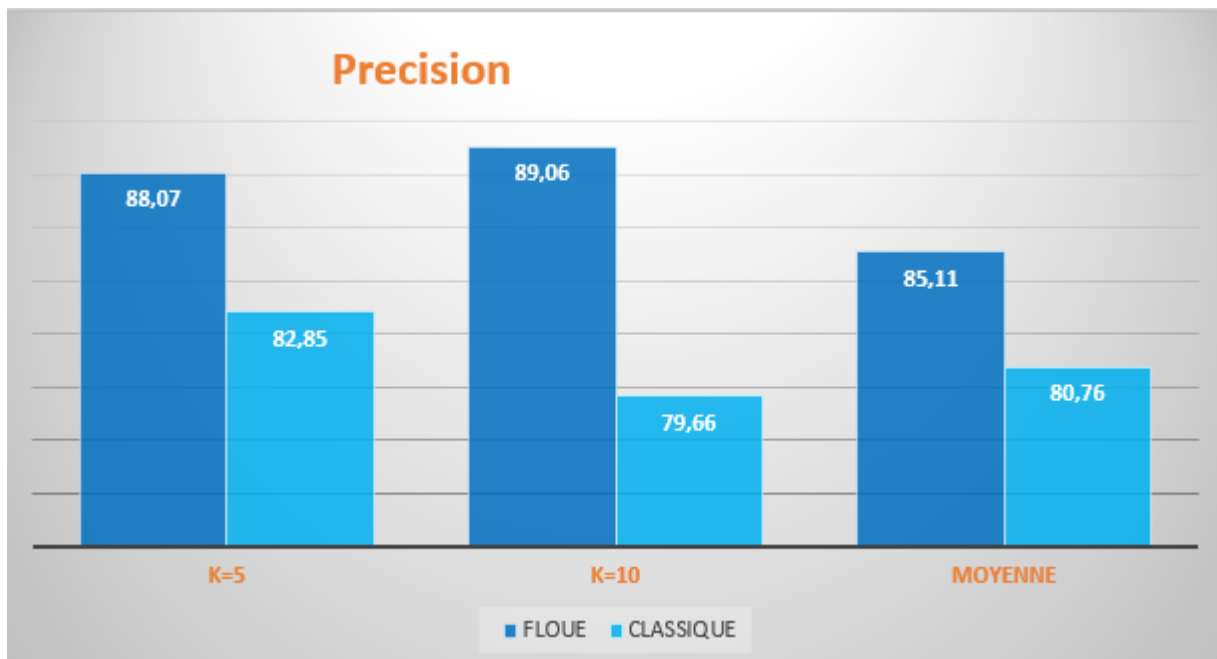


FIGURE 4.13 – Histogramme de précision.

Selon les résultats présentés par les histogrammes, on remarque que :

En effet, les meilleures performances sont observées dans la gamme des valeurs de K entre 5 et 10 (qui restent également supérieures aux autres classifieurs).

- Pour le K égal à 5, l’accuracy de notre système est de 85.96% face à 80.61% d’accuracy pour le RB classique, le rappel est de 87.24% face à 82.55% de rappel pour le RB classique, et la précision et de 88.07% face à 82.85% de précision pour le RB classique.
- Pour le K égal à 10, l’accuracy de notre système est de 87.68% face à 83.17% d’accuracy pour le RB classique, le rappel est de 86.42% face à 83,05% de rappel pour le RB classique, et la précision et de 89.06% face à 79,66% de précision pour le RB classique.

Pour conclure, un bon modèle doit avoir des critères de performance proches de 100%, l’accuracy, le rappel, et la précision de notre système sont proches de la valeur de perfection.

4.4 Conclusion

Au cours de ce dernier chapitre, nous avons exposé notre système et discuté des différents méthodes java, outils et API utilisés pour le construire. Nous avons également inclus des captures des interfaces du système. Et pour conclure le travail, nous avons évalué notre système afin de le valider.

Conclusion générale

La représentation et le traitement des connaissances incertaines dans les ontologies floues ont été améliorés grâce à une nouvelle approche que nous avons présentée dans notre travail. Une méthode générale qui prend une ontologie floue en entrée et produit une ontologie floue probabiliste. Notre concept a l'avantage de pouvoir représenter et mettre en œuvre des domaines à forte incertitude puisqu'il modélise simultanément des informations floues, incertaines et probabilistes.

Dans la première partie de ce mémoire nous avons présenté les ontologies classiques qui se sont avérées incapable de représenter les connaissances incertaines, les ontologies floue ensuite nous avons abordé les réseaux bayésiens et les réseaux bayésien floues.

La deuxième partie a été consacré à faire une étude de cas qui s'intéresse à la maladie cardiovasculaire en utilisant une dataset. Nous avons suivi la méthodologie "FODM" pour la construction de l'ontologie floue, ensuite la construction du réseau bayésien flou en passant par l'apprentissage de la structure se fait avec les composants de l'ontologie floue selon les besoins de l'utilisateur avec l'aide d'un expert du domaine. Après l'apprentissage des paramètres a été fait à partir d'une base d'apprentissage qui ne contient plus de données manquantes en utilisant l'approche maximum de vraisemblance (MV).

Pour l'intégration des connaissances probabilistes dans l'ontologie floue, nous avons représenté le réseau bayésien flou construit à partir de l'ontologie floue dans une ontologie de haut niveau (méta-ontologie UOFBN), qui est utilisée pour représenter les connaissances probabilistes de manière formelle afin de les intégrer dans l'ontologie floue pour arriver finalement à une ontologie floue probabiliste.

Afin de réaliser notre conception, nous avons implémenté notre système présenté par les différentes interfaces nécessaires à la construction de l'ontologie floue probabiliste.

Nos perspectives consistent à ajouter l'inférence à notre système pour répondre à des requêtes et tester le système sur des cas réels.

Bibliographie

- [1] Lazhari MOUAFFAK. *Développement d'une Ontologie Floue*. PhD thesis, Faculté des Mathématiques et de l'Informatique-UNIVERSITE MOHAMED BOUDIAF DE . . ., 2013.
- [2] Kenneth J Laskey and Kathryn B Laskey. Uncertainty reasoning for the world wide web : Report on the urw3-xg incubator group. *URSW*, 8 :108–116, 2008.
- [3] Paulo Cesar G Da Costa, Kathryn B Laskey, and Kenneth J Laskey. Pr-owl : A bayesian ontology language for the semantic web. In *Uncertainty Reasoning for the Semantic Web I*, pages 88–107. Springer, 2006.
- [4] Rudi Studer, V Richard Benjamins, and Dieter Fensel. Knowledge engineering : principles and methods. *Data & knowledge engineering*, 25(1-2) :161–197, 1998.
- [5] Valéry Psyché, Olavo Mendes, and Jacqueline Bourdeau. *Apport de l'ingénierie ontologique aux environnements de formation à distance*. PhD thesis, 2003.
- [6] Pierre Sidoine V Donfack Guefack. *Modélisation des signes dans les ontologies biomédicales pour l'aide au diagnostic*. PhD thesis, Université Rennes 1, 2013.
- [7] Amandine Bellenger and Sylvain Gatepaille. Uncertainty in ontologies : Dempster-shafer theory for data fusion applications. *arXiv preprint arXiv :1106.3876*, 2011.
- [8] Lotfi A Zadeh. Fuzzy logic. *Computer*, 21(4) :83–93, 1988.
- [9] Youness Chaabi. *Apport des Systèmes Multi-Agent et de la logique floue pour l'assistance au tuteur dans une communauté d'apprentissage en ligne*. PhD thesis, Université de Technologie de Belfort-Montbéliard, 2016.
- [10] Timothy J Ross. *Fuzzy logic with engineering applications*. John Wiley & Sons, 2005.
- [11] Robin Wikström. Fuzzy ontology for knowledge mobilisation and decision support. 2014.
- [12] Lotfi Asker Zadeh. The concept of a linguistic variable and its application to approximate reasoning—i. *Information sciences*, 8(3) :199–249, 1975.

- [13] Zahra Riahi Samani and Mehrnoush Shamsfard. The state of the art in developing fuzzy ontologies : A survey. *arXiv preprint arXiv :1805.02290*, 2018.
- [14] ZR Samani and M Shamsfard. A fuzzy ontology model for qualitative spatial reasoning. In *2011 6th International Conference on Computer Sciences and Convergence Information Technology (ICCIT)*, pages 1–6. IEEE, 2011.
- [15] Zahra Riahi Samani and Mehrnoosh Shamsfard. On the application of fuzzy ontology for qualitative spatial reasoning. *Int. J. Next Gener. Inf. Technol*, 3 :9–17, 2012.
- [16] Hanene Agrebi, Afef Bahri, and Rafik Bouaziz. A framework for fuzzy ontology models. 10 2008.
- [17] Umberto Straccia. Fuzzy alc with fuzzy concrete domains. In *Proceedings of the International Workshop on Description Logics (DL-05)*, volume 147, pages 96–103. Citeseer, 2005.
- [18] Xin Li, José-Fernán Martínez, and Gregorio Rubio. A new fuzzy ontology development methodology (fodm) proposal. *IEEE Access*, 4 :7111–7124, 2016.
- [19] Anouar Tadlaoui Mouenis, Khaldi Mohamed, and Aammou Souhaib. Towards probabilistic ontology based on bayesian networks. 2014.
- [20] Emna Hlel, Salma Jamoussi, and Abdelmajid Ben Hamadou. A probabilistic ontology for the prediction of author’s interests. In *Computational Collective Intelligence*, pages 492–501. Springer, 2015.
- [21] Hoai-Tuong Nguyen. *Réseaux bayésiens et apprentissage ensembliste pour l’étude différentielle de réseaux de régulation génétique*. PhD thesis, Université de Nantes, 2012.
- [22] Ph Leray. *Réseaux bayésiens : Apprentissage et diagnostic de systemes complexes*. PhD thesis, Université de Rouen, 2006.
- [23] Ali Ben Mrad. *Observations probabilistes dans les réseaux bayésiens*. PhD thesis, Université de Valenciennes et du Hainaut-Cambresis ; École nationale d . . . , 2015.
- [24] Patrick Naïm, Pierre-Henri Willemin, Philippe Leray, Olivier Pourret, and Anna Becker. Réseaux bayésiens. *Eyrolles, Paris*, 3 :120, 1999.
- [25] Uffe Kjærulff. Reduction of computational complexity in bayesian networks through removal of weak dependences. In *Uncertainty Proceedings 1994*, pages 374–382. Elsevier, 1994.
- [26] Bruce D’Ambrosio. Incremental probabilistic inference. In *Uncertainty in Artificial Intelligence*, pages 301–308. Elsevier, 1993.

- [27] Max Henrion. Propagating uncertainty in bayesian networks by probabilistic logic sampling. In *Machine intelligence and pattern recognition*, volume 5, pages 149–163. Elsevier, 1988.
- [28] Ross D Shachter and Mark Alan Peot. Simulation approaches to general probabilistic inference on belief networks. *arXiv preprint arXiv :1304.1526*, 2013.
- [29] Robert Fung and Brendan Del Favero. Backward simulation in bayesian networks. In *Uncertainty Proceedings 1994*, pages 227–234. Elsevier, 1994.
- [30] ISHAK RIALI. *Modélisation de la connaissance incertaine dans le web sémantique*. PhD thesis, Université Blida1-Saad Dahlab, 2020.
- [31] Nicholas Frank Ryhajlo et al. Fuzzy bayesian networks for prognostics and health management. 2013.
- [32] Hao Tang and Shi Liu. Basic theory of fuzzy bayesian networks and its application in machinery fault diagnosis. In *Fourth international conference on fuzzy systems and knowledge discovery (FSKD 2007)*, volume 4, pages 132–137. IEEE, 2007.
- [33] 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.
- [34] Nicholas Ryhajlo, Liessman Sturlaugson, and John W Sheppard. Diagnostic bayesian networks with fuzzy evidence. In *2013 IEEE AUTOTESTCON*, pages 1–8. IEEE, 2013.
- [35] Zhongli Ding, Yun Peng, and Rong Pan. Bayesowl : Uncertainty modeling in semantic web ontologies. In *Soft computing in ontologies and semantic web*, pages 3–29. Springer, 2006.
- [36] Yanhui Lv, Zong Min Ma, and Fu Zhang. A fuzzy ontology generation framework from relational schema. In *2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, volume 4, pages 435–439. IEEE, 2008.
- [37] Fernando Bobillo, Miguel Delgado, Juan Gómez-Romero, and Umberto Straccia. Fuzzy description logics under gödel semantics. *International Journal of Approximate Reasoning*, 50(3) :494–514, 2009.
- [38] Yi Yang and Jacques Calmet. Ontobayes : An ontology-driven uncertainty model. In *International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)*, volume 1, pages 457–463. IEEE, 2005.
- [39] Hlel Emna, Jamoussi Salma, Turki Mohamed, and Ben Hamadou Abdelmajid. Probabilistic ontology definition meta-model. In *International Conference on Intelligent Decision Technologies*, pages 243–254. Springer, 2016.

- [40] Abdul-Wahid Mohammed, Yang Xu, and Ming Liu. Knowledge-oriented semantics modeling towards uncertainty reasoning. *SpringerPlus*, 5(1) :1–27, 2016.
- [41] Davide Chicco and Giuseppe Jurman. Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC medical informatics and decision making*, 20(1) :1–16, 2020.
- [42] Ishak Riali, Messaouda Fareh, and Hafida Bouarfa. A semantic approach for handling probabilistic knowledge of fuzzy ontologies. In *ICEIS (1)*, pages 407–414, 2019.