NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCES
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

PROGRAM OF POSTGRADUATE STUDIES

PhD THESIS

Managing Uncertainty and Vagueness in Semantic Web

Loukia P. Karanikola

ATHENS

SEPTEMBER 2018

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

Υποστήριξη Αβεβαιότητας και Ασάφειας για το Σημασιολογικό Ιστό

Λουκία Π. Καρανικόλα

ΑΘΗΝΑ

ΣΕΠΤΕΜΒΡΙΟΣ 2018

PhD THESIS

Managing Uncertainty and Vagueness in Semantic Web

Loukia P. Karanikola

SUPERVISOR: Isambo Karali, Assistant Professor UOA

THREE-MEMBER ADVISORY COMMITTEE:

Isambo Karali, Assistant Professor UOA

Sally I. McClean, Professor Ulster University

Stathes Hadjiefthymiades, Associate Professor UOA

SEVEN-MEMBER EXAMINATION COMMITTEE

Isambo Karali,
Assistant Professor UOA

Sally I. McClean,
Professor Ulster University

Stathes Hadjiefthymiades,
Associate Professor UOA

George Vouros,
Professor University of Pireaus

Manolis Gergatsoulis,
Professor Ionian University

Panos Rondogiannis ,
Professor UOA

Andreas Stafylopatis ,
Professor NTUA

Examination Date: 27 September, 2018

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

Υποστήριξη Αβεβαιότητας και Ασάφειας για το Σημασιολογικό Ιστό

Λουκία Π. Καρανικόλα

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: Καράλη Ιζαμπώ, Επίκουρη Καθηγήτρια ΕΚΠΑ

ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ:

Καράλη Ιζαμπώ, Επίκουρη Καθηγήτρια ΕΚΠΑ

McClean Sally I., Καθηγήτρια Πανεπιστήμιο Ulster

Χατζηευθυμιάδης Ευστάθιος, Αναπληρωτής Καθηγητής ΕΚΠΑ

ΕΠΤΑΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

Καράλη Ιζαμπώ,
Επίκουρη Καθηγήτρια ΕΚΠΑ

McClean Sally I.,
Καθηγήτρια Πανεπιστήμιο Ulster

Χατζηευθυμιάδης Ευστάθιος,
Αναπληρωτής Καθηγητής ΕΚΠΑ

Βούρος Γεώργιος,
Καθηγητής Πανεπιστήμιο Πειραιά

Γεργατσούλης Εμμανουήλ,
Καθηγητής Ιόνιο Πανεπιστήμιο

Ροντογιάννης Παναγιώτης,
Καθηγητής ΕΚΠΑ

Σταφυλοπάτης Ανδρέας - Γεώργιος,
Καθηγητής ΕΜΠ

Ημερομηνία Εξέτασης: 27 Σεπτεμβρίου 2018

ABSTRACT

Semantic Web has been designed for processing tasks without human intervention. In this context, the term machine processable information has been introduced. In most Semantic Web tasks, we come across information incompleteness issues, aka uncertainty and vagueness. For this reason, a method that represents uncertainty and vagueness under a common framework has to be defined. Semantic Web technologies are defined through a Semantic Web Stack and are based on a clear formal foundation. Therefore, any representation scheme should be aligned with these technologies and be formally defined. As the concept of ontologies is significant in the Semantic Web for representing knowledge, any framework is desirable to be built upon it.

In our work, we have defined an approach for representing uncertainty and vagueness under a common framework. Uncertainty is represented through Dempster-Shafer model, whereas vagueness has been represented through Fuzzy Logic and Fuzzy Sets. For this reason, we have defined our theoretical framework, aimed at a combination of the classical crisp DL $\mathcal{ALC}$ with a Dempster-Shafer module. As a next step, we added fuzziness to this model. Throughout our work, we have implemented metaontologies in order to represent uncertain and vague concepts and, next, we have tested our methodology in real-world applications.

SUBJECT AREA: Uncertainty handling

KEYWORDS: Uncertainty, Vagueness, Dempster-Shafer Model, Description Logics, Semantic Web

ΠΕΡΙΛΗΨΗ

Ο Σημασιολογικός Ιστός στοχεύει στην διεκπεραίωση εργασιών σε υπολογιστικά συστήματα χωρίς την ανθρώπινη παρέμβαση. Προκειμένου να επιτευχθεί ο στόχος αυτός, εισάγεται η έννοια της πληροφορίας που είναι επεξεργάσιμη από μηχανές. Στα περισσότερα προβλήματα, η έννοια της πληροφορίας είναι συνυφασμένη με την έννοια της αβεβαιότητας και της ασάφειας. Και οι δύο έννοιες περιγράφονται με την κοινή ονομασία ατελής πληροφορία. Δεδομένου ότι ο Σημασιολογικός Ιστός απαρτίζεται από ένα σύνολο τεχνολογιών και των θεωριών που τις διέπουν, οποιαδήποτε μέθοδος αναπαράστασης θα πρέπει να βρίσκεται σε συμφωνία με άλλες υπάρχουσες. Συγκεκριμένα, το θεωρητικό πλαίσιο πρέπει να εντάσσεται ομαλά στη θεωρία που εφαρμόζεται στο Σημασιολογικό Ιστό. Η δε υλοποίησή του, ιδανικό είναι, να υποστηριχθεί με χρήση μεθόδων του Σημασιολογικού Ιστού, στις οποίες κυριαρχεί εκείνη των οντολογιών.

Στη διατριβή μας, ορίσαμε μία μέθοδο αναπαράστασης της αβεβαιότητας και της ασάφειας μέσω ενός ενιαίου πλαισίου. Το μοντέλο Dempster-Shafer χρησιμοποιήθηκε για την αναπαράσταση της αβεβαιότητας και το μοντέλο Ασαφούς Λογικής και Ασαφών Συνόλων για την αναπαράσταση της ασάφειας. Για το λόγο αυτό, ορίσαμε το θεωρητικό πλαίσιο, στοχεύοντας σε ένα συνδυασμό $\mathcal{ALC}$ Λογικών Περιγραφών (Description Logics) με το μοντέλο Dempster-Shafer. Κατά τη διάρκεια της έρευνάς μας υλοποιήσαμε μετα-οντολογίες για την αναπαράσταση της αβεβαιότητας και της ασάφειας και στη συνέχεια μελετήσαμε την συμπεριφορά τους σε πραγματικές εφαρμογές.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Χειρισμός αβεβαιότητας

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Αβεβαιότητα, Ασάφεια, Dempster-Shafer Model, Λογικές Περιγραφών, Σημασιολογικός Ιστός

ΣΥΝΟΠΤΙΚΗ ΠΑΡΟΥΣΙΑΣΗ ΤΗΣ ΔΙΔΑΚΤΟΡΙΚΗΣ ΔΙΑΤΡΙΒΗΣ

Το πλαίσιο του Σημασιολογικού Ιστού (Semantic Web) υποβάλλει τη διεκπεραίωση εργασιών σε υπολογιστικά συστήματα χωρίς την ανθρώπινη παρέμβαση. Όλες οι τεχνολογίες του Σημασιολογικού περιγράφονται μέσω της Στοίβας Σημασιολογικού Ιστού (Semantic Web Stack). Η έννοια της οντολογίας είναι μία απο τις βασικές τεχνολογίες που περιγράφονται στη Στοίβα Σημασιολογικού Ιστού. Ο ρόλος του *πράκτορα* (agent) έχει κεντρικό ρόλο στο περιβάλλον του Σημασιολογικού Ιστού. Ένας πράκτορας μπορεί να οριστεί σαν ένας μηχανισμός αναζήτησης πληροφορίας. Ένας τέτοιος μηχανισμός προϋποθέτει ένα πλαίσιο αποφασισιμότητας (decision making) και ανάκτησης πληροφορίας (information extraction). Η πληροφορία πολλές φορές χαρακτηρίζεται από αβεβαιότητα και ασάφεια, κάτι το οποίο πρέπει να ληφθεί υπόψη στο πλαίσιο αυτό.

Η *οντολογία* αποτελεί μία βασική έννοια του Σημασιολογικού Ιστού. Σε ένα περιβάλλον Σημασιολογικού Ιστού μία οντολογία ορίζεται μέσω της Web Ontology Language (OWL). Η πιο διαδεδομένη διάλεκτος της OWL βασίζεται στις Λογικές Περιγραφών (Description Logics). Οι Λογικές Περιγραφών ουσιαστικά επιτρέπουν την αναπαράσταση ενός πεδίου γνώσης. Η αναπαράσταση ουσιαστικά ανάγεται στην ανάθεση μίας τιμής αληθές / ψευδές, η οποία αποδίδεται μέσω μίας ερμηνείας (interpretation). Οι Λογικές Περιγραφών ορίζουν μία Βάση Γνώσης (Knowledge Base) μέσω:

- Ενός συνόλου αξιωμάτων εννοιών (TBox)
- Ενός συνόλου αξιωμάτων σχέσεων (RBox)
- Ενός συνόλου αξιωμάτων αναθέσεων τιμών (ABox)

Μία ερμηνεία ορίζεται ως $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, όπου το $\Delta^{\mathcal{I}}$ είναι ο τομέας ερμηνείας και το $\cdot^{\mathcal{I}}$ είναι η συνάρτηση ερμηνείας.

Τα περισσότερα προβλήματα στο Σημασιολογικό Ιστό χαρακτηρίζονται από την έννοια της *αβεβαιότητας* και της *ασάφειας*. Η αβεβαιότητα ουσιαστικά αναφέρεται σε περιπτώσεις ελλιπούς πληροφορίας, ενώ η ασάφεια περιγράφει πληροφορία που δεν είναι σαφώς ορισμένη. Ένα πλαίσιο το οποίο θα ενσωματώνει αβέβαιη και ασαφή πληροφορία σε ένα περιβάλλον Σημασιολογικού Ιστού θα συμβάλλει στην αποτελεσματικότερη αναπαράσταση και διαχείριση της πληροφορίας. Για μία τέτοια προσέγγιση απαιτείται η επέκταση μίας Λογικής Περιγραφών με έναν τέτοιο τρόπο ώστε να επιτρέπεται η ανάθεση μίας τιμής αλήθειας στο διάστημα [0,1]. Στην δική μας προσέγγιση ουσιαστικά επεκτείναμε τη Λογική Περιγραφών $\mathcal{ALC}$ με βαθμίδες αβεβαιότητας και ασάφειας.

Σαν παράδειγμα για την ανάγκη των παραπάνω, ας θεωρήσουμε ένα σύστημα συστάσεων (recommender system) το οποίο αφορά ξενοδοχεία. Έστω ότι έχουμε την πρόταση ”*το ξενοδοχείο h_1 έχει ένα κόστος ανάμεσα σε 50 και 100 EURO*”. Αυτό πρακτικά σημαίνει ότι δεν γνωρίζουμε το ακριβές κόστος, αλλά ένα εύρος κόστους. Επιπλέον, θεωρούμε το ερώτημα ”*Αναζητώ ένα ξενοδοχείο χαμηλού κόστους*”. Στο συγκεκριμένο πρόβλημα πρέπει να λάβουμε υπόψη:

- Αβεβαιότητα, λόγω ελλιπούς πληροφορίας όσον αφορά την ακριβή τιμή του κόστους
- Ασάφεια, λόγω του μη συγκεκριμένου ορισμού ”χαμηλό κόστος”

Στην προσέγγισή μας, για την αναπαράσταση της αβεβαιότητας εφαρμόσαμε το μοντέλο Dempster-Shafer, μία μέθοδο η οποία παραδοσιακά χρησιμοποιείται στην αναπαράσταση ελλιπούς πληροφορίας. Οι βασικές συναρτήσεις σε αυτό το μοντέλο είναι η *βασική ανάθεση πιθανότητας* (*basic probability assignment*), η *συνάρτηση Βεβαιότητας (Belief function)* και η *συνάρτηση Αληθοφάνειας (Plausibility function)*. Επιπλέον, θεωρούμε ένα *σύνολο διακριτών γεγονότων* (*frame of discernment*). Μέσω των συναρτήσεων μπορούμε να αναπαραστήσουμε ένα σύνολο από ”πεποιθήσεις” (beliefs) όσον αφορά στοιχεία του δυναμοσυνόλου του συνόλου διακριτών γεγονότων. Επιπρόσθετα, ο *κανόνας συνδυασμού του Dempster (Dempster’s rule of Combination)* επιτρέπει το συνδυασμό ανεξάρτητων ”πεποιθήσεων”.

Για την αναπαράσταση της ασάφειας χρησιμοποιείται η θεωρία των Ασαφών Συνόλων και της Ασαφούς Λογικής. Στον τομέα των Λογικών Περιγραφών υπάρχουν διάφορα πλαίσια που επεκτείνονται με Ασαφή Λογική. Αυτά τα πλαίσια ουσιαστικά αναθέτουν στα διάφορα αξιώματα ένα *βαθμό συμμετοχής (membership degree)* στο διάστημα [0,1].

Αρχικά, μελετήσαμε το πρόβλημα της αβεβαιότητας και της ασάφειας μεμονωμένα. Στην κατεύθυνση αυτή αρχικά η έρευνα εστίασε σε μία προσέγγιση για την ανάπτυξη μίας οντολογίας, η οποία βασίζεται στο μοντέλο Dempster-Shafer. Στην εν λόγω προσέγγιση, στοχεύσαμε στην αναπαράσταση της αβεβαιότητας της πληροφορίας, μέσω της μεθόδου Dempster-Shafer. Για το σκοπό αυτό, προσαρμόσαμε το μοντέλο Dempster-Shafer σε ένα περιβάλλον οντολογίας.

Στη συνέχεια, η έρευνά μας επικεντρώθηκε στην περιγραφή μίας οντολογίας αβεβαιότητας, καθώς επίσης και μίας μεθόδου συλλογιστικής για δεδομένα που χαρακτηρίζονται από αβεβαιότητα και ασάφεια. Η μέθοδος κάνει χρήση της έννοιας των ομότιμων (peers), οι οποίοι ορίζονται σαν μηχανές επεξεργασίας. Σε κάθε ομότιμο αντιστοιχεί ένα μέρος της πληροφορίας με τη μορφή κανόνων και γεγονότων, π.χ. «Όλα τα ξενοδοχεία 5 αστέρων έχουν πισίνα». Όσον αφορά το Σημασιολογικό Ιστό, η γλώσσα των οντολογιών είναι η OWL. Στο πλαίσιο αυτό, έγινε ο ορισμός μίας ασαφούς οντολογίας (fuzzy ontology) με επέκταση των Λογικών Περιγραφών (Fuzzy DL $\mathcal{SROIQ}(D)$) στις οποίες αυτή βασίζεται. Οι πιθανοθεωρητικές βάσεις γνώσης (probabilistic knowledge bases) χρησιμοποιούνται για να αναπαρασταθεί τόσο η αβέβαιη όσο και η ασαφής πληροφορία έχοντας σαν δεδομένο τη δυνατότητα αναπαράστασης της κάθε μιας με το φορμαλισμό της άλλης.

Δεδομένου του όγκου της πληροφορίας που ενδέχεται να διαχειρίζεται ο σημασιολογικός ιστός, η μελέτη μας συμπεριελαβε και περιβάλλοντα μεγάλων δεδομένων (big data). Η έννοια των μεγάλων δεδομένων αναπτύχθηκε τα τελευταία χρόνια σαν αποτέλεσμα της ραγδαίας αύξησης του όγκου των δεδομένων (data boom). Όσον αφορά τις οντολογίες, το πρόβλημα εντοπίζεται στον τρόπο με τον οποίο θα αποθηκεύσουμε την πληροφορία, καθώς και στον τρόπο με τον οποίο θα την επεξεργαστούμε. Από την άλλη, οι μέθοδοι αναπαράστασης αβέβαιης και ασαφούς πληροφορίας δεν μπορούν να εφαρμοστούν σε περιβάλλοντα μεγάλου όγκου δεδομένων. Για την διαχείριση τέτοιου όγκου δεδομένων, η διαίρεση της πληροφορίας σε ένα σύνολο από ομότιμους (peers) είναι εκμεταλλεύσιμη.

Η βασική συμβολή της διατριβής είναι στην θεωρητική προσέγγιση (Dempster-Shafer Fuzzy Description Logics). Όπως αναφέραμε προηγουμένως, στοχεύσαμε στην ενοποίηση της αβεβαιότητας και της ασάφειας σε ένα ενιαίο πλαίσιο. Για την ανάπτυξη ενός πλαισίου αναπαράστασης ατελούς πληροφορίας (imperfect information), έγινε η ενοποίηση τριών διαφορετικών μοντέλων – θεωριών:

1. Μοντέλο Dempster-Shafer
2. Μοντέλο Ασαφούς Λογικής
3. Μοντέλο Λογικής Περιγραφών

Ο στόχος της ενοποίησης αποτελεί την δυνατότητα ορισμού ενός συνόλου αξιωμάτων τα οποία θα έχουν βαθμούς αλήθειας (truthness degree) τόσο όσον αφορά την ασάφεια όσο και την αβεβαιότητα.

Το πλαίσιο αυτό, ουσιαστικά επεκτείνει ένα μοντέλο Fuzzy Description Logics με ένα μοντέλο Dempster-Shafer. Στη διατριβή μας έγινε ορισμός του συντακτικού (syntax) και της σημασιο-λογίας (semantics) του μοντέλου αυτού. Επίσης, διευρενήθηκαν θέματα αποφασισιμότητας και πολυπλοκότητας. Πιο συγκεκριμένα, σαν πρώτο βήμα ασχοληθήκαμε με την ανα-παράσταση της αβεβαιότητας, χωρίς να λάβουμε υπόψη θέματα ασάφειας. Προς αυτή την κατεύθυνση, θεωρήσαμε την Λογική Περιγραφών $\mathcal{ALC}$ την οποία επεκτείναμε με συνθήκες βεβαιότητας (belief degree conditions) και συνθήκες αληθοφάνειας (plausibility degree conditions). Για τον ορισμό της σημασιολογίας του πλαισίου, εισάγαμε την έννοια του πιθανού κόσμου (possible world) σαν ένα σύνολο αξιωμάτων που είναι αληθή στον εν λόγω πιθανό κόσμο. Το σύνολο των πιθανών κόσμων το διαχειριζόμαστε σαν ένα διακριτό σύνολο που αντιστοιχεί στο πλαίσιο αναφοράς. Επιπλέον, ορίζουμε μία ερμηνεία πιθανού κόσμου σαν μία βασική ανάθεση πιθανότητας με πεδίο ορισμού το δυναμοσύνολο του διακριτού συνόλου και πεδίο τιμών το διάστημα [0,1]. Επίσης, βασιζόμενοι στο κανόνα συνδυασμού του Dempster, ορίσαμε την *Συνδυασμένη Dempster-Shafer συνεπαγωγή* (*Combined Dempster-Shafer entailment*). Για να την εφαρμόσουμε, θεωρούμε δύο ανεξάρτητες βάσεις γνώσης και συνδυάζουμε γεγονότα που συνεπάγονται από τις δύο βάσεις.

Έπειτα, για την αναπαράσταση της ασάφειας, επεκτείναμε την ασαφή Λογική Περιγραφών $\mathcal{ALC}$ (Fuzzy $\mathcal{ALC}$), θεωρώντας τις ασαφείς ερμηνείες σαν πιθανούς κόσμους. Το σύνολο αυτό των πιθανών κόσμων εκφράζει το σύνολο διακριτών γεγονότων σε ένα πλαίσιο Dempster-Shafer. Ορίσαμε την Dempster-Shafer Fuzzy ερμηνεία (Dempster-Shafer Fuzzy Interpretation) η οποία βασίζεται πάνω στο δυναμοσύνολο του συνόλου των πιθανών κόσμων. Στη προσέγγισή μας κληθήκαμε να ορίσουμε βαθμούς πεποίθησης σε ασαφή σύνολα. Για να το πετύχουμε αυτό έπρεπε να λάβουμε υπόψη συναρτήσεις συμμετοχής (membership functions). Για παράδειγμα η πρόταση $< LowCost(hotel)\ 0.8\ \ :\ 0.9 >$ δηλώνει ένα ξενοδοχείο χαμηλού κόστους με ”πίστη” τουλάχιστον 0.9 και ”βαθμό συμμετοχής” τουλάχιστον 0.8.

Προκειμένου να εξεταστεί η χρήση του πλαισίου σε εφαρμογές, εστιάσαμε σε δύο διαφορετικά πεδία. Το πρώτο αναφερόταν στα περιβάλλοντα ταύτισης (matchmaking environments). Ένα πρόβλημα ταύτισης βασίζεται στα λεγόμενα ”κριτήρια” (constraints). Ένα παράδειγμα ενός τέτοιου πλαισίου είναι μία εφαρμογή ανεύρεσης εργασίας, όπου τόσο ο αναζητών εργασία όσο και ο προσφέρων εργασία θέτουν ένα σύνολο από κριτήρια. Ο στόχος είναι να ελεγθεί αν με βάση τα κριτήρια τα δύο μέρη μπορούν να ”ταιριάξουν”. Το πρόβλημα γίνεται ακόμη πιο περίπλοκο όταν αναφερόμεστε σε περιβάλλον Σημασιολογικού Ιστού, όπου ο όγκος της πληροφορίας είναι αρκετά μεγάλος. Τότε απαιτείται μία επιλογή των διαθεσίμων πηγών πληροφορίας, κάτι το οποίο οδηγεί σε αβεβαιότητα. Επίσης, όταν τα κριτήρια δεν είναι συγκεκριμένα, δηλαδή όταν χαρακτηρίζονται από ασάφεια, το πρόβλημα γίνεται ακόμη πιο έντονο. Σε ένα περιβάλλον Σημασιολογικού Ιστού η αναπαράσταση της πληροφορίας γίνεται μέσω μίας οντολογίας. Για το σκοπό αυτό δημιουργήθηκε μία οντολογία και ένα σύνολο κανόνων (rules) οι οποίοι έχουν σαν στόχο την εξαγωγή ενός παράγοντα ταύτισης(matchmaking degree). Για την εξαγωγή του συγκεκριμένου παράγοντα λάβαμε υπόψη τόσο έναν βαθμό συμμετοχής όσο και έναν βαθμό αβεβαιότητας (uncertainty degree). Η αναπαράσταση του βαθμού αβεβαιότητας έγινε μέσω ενός μοντέλου Dempster-Shafer. Στη μελέτη της περίπτωσής μας θεωρήσαμε τα παρακάτω στοιχεία:

- Ζητών εργασία (Job seeker)
- Προσφέρων εργασία (Job advertisement)
- Ασαφής Οντολογία (Fuzzy Ontology Repository)

Τα κριτήρια είναι ένα σύνολο από περιορισμούς που τίθενται από τον Ζητούντα και τον Προσφέροντα εργασία. Για κάθε κριτήριο υπάρχει μία ”ιδανική τιμή”. Όταν η τιμή αυτή μειώνεται ο βαθμός ικανοποίησης (satisfaction degree) επίσης μειώνεται. Μέσω αυτής της αναπαράστασης ορίζουμε τον ασαφή βαθμό ικανοποίησης (Fuzzy constraint degree) για τα δύο μέρη, το ”Ζητών εργασία” και το ”Προσφέρων εργασία”. Επιπλέον, ορίσαμε ένα σύνολο βαρών (weights) για όλα τα κριτήρια και συσχετίσαμε μία συνάρτηση ανάθεσης πιθανότητας με αυτά τα βάρη. Στο τέλος ορίσαμε το βαθμό ταύτισης για τα δύο μέρη ως εξής: $Matchmaking \equiv\ JobSeeker \sqcap\ JobAdvertisement$.

Το δεύτερο πεδίο εφαρμογών ήταν τα συστήματα συστάσεων, όπως για παράδειγμα ένα σύστημα όπου προτείνει ένα συγκεκριμένο ξενοδοχείο σε κάποιον χρήστη. Αρχικά, θεωρήσαμε μόνο περιπτώσεις αβεβαιότητας. Στην κατεύθυνση αυτή, ορίσαμε μία οντολογία αβεβαιότητας (uncertainty ontology) η οποία βασίστηκε στο μοντέλο Dempster-Shafer. Η οντολογία μας αποτελείται από ένα σύνολο από έννοιες όπως:

- Έννοια πιθανού κόσμου (Possible world concept)
- Έννοια δυναμοσυνόλου (Power set concept)

καθώς και ένα σύνολο σχέσεων όπως:

- ”έχει τιμή πεποίθησης” (hasBel)
- ”έχει τιμή βασικής ανάθεσης πιθανότητας” (hasBpa)

Σε επόμενο βήμα θεωρήσαμε την ένταξη ασαφών κριτηρίων. Ο χρήστης αναζητά ένα ξενοδοχείο, θέτοντας ασαφή κριτήρια, π.χ *ξενοδοχείο χαμηλού κόστους*, *κοντά στο μετρό* κλπ. Στην εφαρμογή αυτή θεωρήσαμε την πλήρη επέκταση της Γλώσσας Οντολογίας της διατριβής, ούτως ώστε να γίνεται αναπαράσταση της αβεβαιότητας και της ασάφειας. Για τον συνδυασμό των Dempster-Shafer Fuzzy OWL αξιωμάτων (OWL axioms) εφαρμόσαμε την τεχνική Μοντελοποίησης Ασαφών Συστημάτων (Fuzzy Systems Modelling Technique - FSM). Αυτή η τεχνική επιτρέπει τον ορισμό ενός συνόλου κανόνων, όπου τόσο το μέρος της συνθήκης όσο και το συμπέρασμα περιγράφονται μέσω ασαφών γλωσσικών όρων που συνοδεύονται από ένα βαθμό συμμετοχής. Η εφαρμογή αυτή που αποτελεί ένα πρόβλημα πραγματικής κλίμακας, αξιολογήθηκε από πλευράς ακρίβειας (precision) και ανάκλησης (recall) και αποδείχθηκε πολύ ικανοποιητική, αναδεικνύοντας την εφαρμοσιμότητα του πλαισίου.

To my daughter.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Problem statement

The Internet has paved the way for the evolution of alternative methods of communication. E-commerce, e-banking and online stores are some of them. Traditionally, computers were designed for performing numerical calculations. In addition, the content of Web information has been designed for human consumption, i.e. it is *human oriented*. The evolution of search engines gave a boost at the popularity of WWW, but at the same time made it necessary for the existence of a Web (or Web information) suitable for machines (or agents).

Towards this concept, Semantic Web was the vision of Tim Berners-Lee who stated: ”*Machines become capable of analyzing all the data on the Web - the content, links, and transactions between people and computers. ”A Semantic Web”, which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines, leaving humans to provide the inspiration and intuition. The ”intelligent agents” people have touted for ages will finally materialize*” [15].

The Semantic Web will contribute in the evolution of many web applications [5], such as:

- Knowledge Management
- Business-to-Computer
- Electronic Commerce
- Wikis

To sum up, the Semantic Web vision introduces the notion of machine-oriented information. This information comes as a result of data existing in various web sources. Information extraction from these sources can be very difficult in many cases. Reliability, ambiguity or incompleteness issues are usual problems considering Web information, resulting in deficient knowledge. Any method that represents machine-oriented information should provide a well-defined description of imprecise knowledge [90, 77].

Imprecise knowledge is usually divided into:

- *Uncertainty*
- *Vagueness*

Uncertainty refers to situations of information incompleteness whereas vagueness describes imprecise information, i.e concepts with not well-defined meaning. Generally, uncertainty and vagueness are considered two different notions and as such different theories have been defined for representing them. Probability theory, Dempster-Shafer theory and Possibility theory are some frameworks designed for uncertainty representation [42, 113]. On the other hand, Fuzzy Logic and Fuzzy Sets [142] is the theory that lies behind vagueness representation. In many cases, we come across situations where both uncertainty and vagueness coexist. Thus, we need a common framework in order to represent uncertainty and vagueness concepts. Both notions can be defined as *imperfect information*.

Regarding Semantic Web, *ontologies* is the core concept for knowledge representation. Ontologies are represented through the Web Ontology Language (OWL) with OWL2 being the current version [129]. Description Logics (DLs) [9] have been employed extensively in Semantic Web, as they are the logics behind the most widely used version of OWL, OWL-DL. DLs allow for the representation of a domain of knowledge, by providing *Concepts*, along with *Roles*. The necessity to capture uncertain and vague knowledge in Semantic Web has been employed in extensions of DLs, resulting in Probabilistic [83], Possibilistic [103] and Fuzzy extensions [118, 121]. These extensions capture the problem of uncertainty and vagueness separately and not as a common framework.

1.2 Objectives

1.2.1 Main Idea

The main goal of this dissertation is to define a framework for representing imperfect information, by extending crisp knowledge representation methods. By ”imperfect”, we refer either to uncertain or vague concepts. The general idea is to define a knowledge representation scheme, that allows for statements with uncertainty and vagueness degree conditions. This representation assigns a truth degree in the interval $[0, 1]$ rather than a true/false value. Our framework is aligned with semantic web knowledge representation frameworks and it is defined based on these theories. Thus, our approach can be defined as a ”*semantic web knowledge representation approach for representing uncertain and vague concepts*”.

1.2.2 Thesis steps - Achievements

More precisely, throughout our dissertation we have proceeded through the following steps. For each step the reached achievements are also presented:

- Propose a definition of an ”imperfect” Description Logic along with an ”imperfect” Ontology, that captures both uncertain and vague concepts. Towards this concept the following sub-goals have been achieved:
  - Define ontologies that capture uncertain and vague concepts: An uncertainty ontology and an entailment method which is based on Dempster-Shafer model

are described and implemented.

- Define an extension of a crisp DL with Belief - Plausibility Degrees:
  We propose a framework that employs Dempster-Shafer theory in a Description Logic Knowledge Base environment. More precisely, we have defined a Dempster-Shafer DL Knowledge Base, in order to represent uncertainty in a Description Logics framework. In addition, a combination method of independent Dempster-Shafer DL Knowledge Bases has been proposed, based on Dempster's rule of Combination.
- Define an extension of a fuzzy DL with Belief Degrees: Vague information has been emerged as a main issue in Semantic Web community. Vagueness is traditionally represented by Fuzzy Set theory. Besides vagueness, Semantic Web queries often have to deal with information incompleteness, aka uncertainty. This kind of information can be represented through Dempster-Shafer theory, that also enables distributed information fusion. Imperfect information, i.e uncertainty and vagueness, should be represented and manipulated under a common framework. We propose such a framework by defining a fuzzy Description Logic extended with Dempster-Shafer theory. Furthermore, we regard our method as a DL extension and we implemented it by a meta-ontology that captures Dempster-Shafer Fuzzy statements.

• Testing and evaluating our framework in real-world case studies: In order to test our methodology in real-world environments, we have tested two application areas, recommender systems and matchmaking environments. We have collected a set of data, detect uncertain and vague pieces of evidence and proceeded by employing suitable applications for manipulating them.

Consequently, for defining a unified framework for representing uncertainty and vagueness, we decided to combine the following theories:

- Fuzzy Logic
- Dempster-Shafer Theory
- Description Logics

2. BACKGROUND AND RELATED WORK

2.1 Semantic Web Concepts

At first, web data were designed taking into account human readers, with HTML being the most used language. The problem is that HTML does not provide for *metadata*, i.e. data about data. Metadata capture the semantic regarding Semantic Web data. Towards this concept, XML language have been employed.

In general, information processing within the Semantic Web is done by "*agents*". As it is referred in [5], a semantic web agent "*will receive some tasks and preferences from a person, seek information from web sources, communicate with other agents, compare information about user requirements and preferences, select certain choices, and give answers to the users*". It seems that the role of an agent actually demands a decision making mechanism, which in turn presupposes a method for handling uncertainty and vagueness tasks. They are generally characterized as pieces of software that operate autonomously and proactively. In Semantic Web, an agent usually employs the following technologies:

- Metadata
- Ontologies
- Logic

In the following of this section we overview the notions of Ontologies and Logic in the Semantic Web environment, starting by presenting the SW stack layer architecture.

2.1.1 Semantic Web Layers

Generally, the Semantic Web is regarded as a set of layers that form a stack, with each layer being built on top of another. In order to build this stack, two principles are followed [5]:

- *Downward compatibility*: An agent that knows a layer, necessarily knows lower layers
- *Upward partial understanding*: Awareness of a layer means that partial knows higher levels

At the bottom of the stack resides *XML*, which is a language that allows for structured web data with a user-defined vocabulary. Next, there is *RDF* and *RDF Schema*. For an overview of *XML*, *RDF* and *RDF Schema* see [34]. *RDF* is a data model that is employed for writing simple statements about Web objects (resources). In addition, *RDF Schema* provides for organizing Web objects into hierarchies. Though tools for writing ontologies are provided, there is a need for more advanced ontology languages. Thus, the next level is the *ontology languages*, that allow for representations of more complex relationships, through a variety of dialects. The *Logic* layer provides with the means for writing declarative knowledge. The *Proof* layer is the deductive process, along with the representation of proofs and proof validation. Finally, the *Trust* layer considers digital signatures and in general knowledge based on recommendations by trusted agents. Fig 1 summarizes the Semantic Web stack.

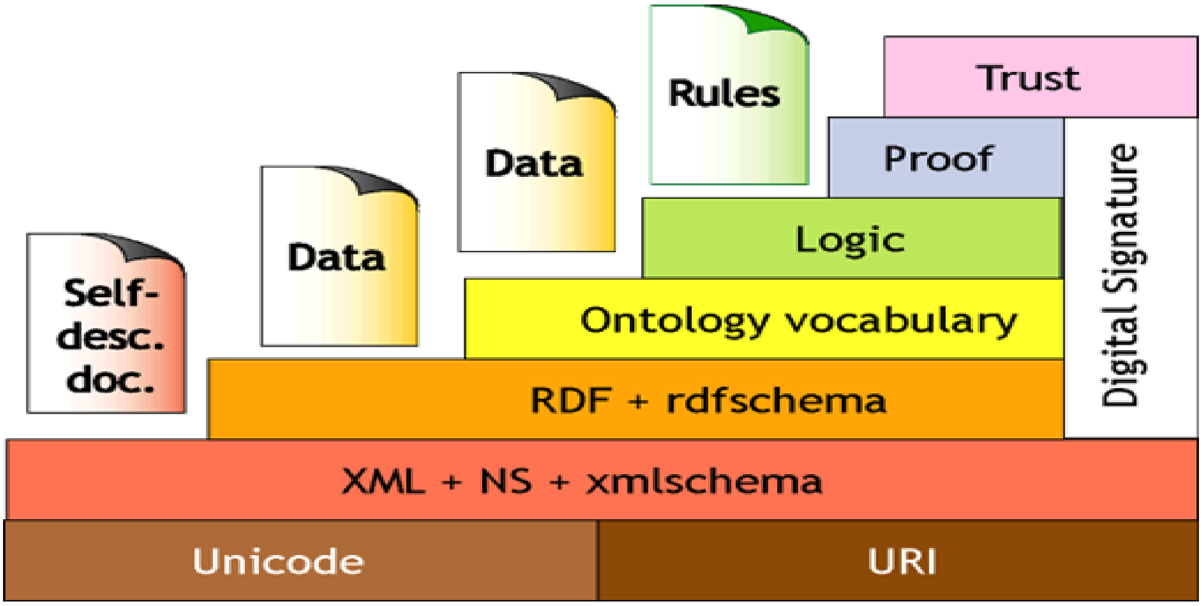

Figure 1: Semantic Web Stack

2.1.2 Ontologies

As previously mentioned, there is a need for web information to be represented in a way that is understandable by machines. To achieve this, the Semantic Web incorporates a lot of technologies, which are described in what we call a *semantic web stack*. In addition, in [60], the semantic web architecture is regarded as "two-towers" rather than a stack . *Ontologies* and *rules* are the most significant among these technologies. Generally, an ontology "*is an explicit and formal specification of a conceptualization*" [5]. That means that it is a *conceptualization of a domain* and provides a shared understanding of the domain. This term originates from philosophy and is the literal translation of the Greek word *οντολογία*. As it is referred in [62] definitions for objects as well as types of objects should be provided. We can consider that an ontology consists of:

1. Types of entities that describe a specific domain
2. Properties of those entities

These are expressed as:

- A finite list of terms: They denote important concepts (classes) of a domain
- Relationships between terms: They denote hierarchies of classes

In Semantic Web, ontologies are defined through the OWL (Web Ontology Language) family [129, 58], the ontology language recommended by W3C. There exist three dialects of OWL: OWL-Full, OWL-DL and OWL-Lite. OWL-DL and OWL-Lite are based on description logics, which is a logic-based knowledge representation formalism for modeling a domain in terms of concepts (classes), roles (properties / relations) and individuals [58], [128]. Ontologies represent the semantics of the domain (in the case of SW the semantics of the source).

The following are terms usually employed in Semantic Web ontologies [74]:

1. *Class*: It is a term employed to represent general qualities and properties of a group of objects
2. *Subclass*: It is a term employed to represent a part of the object group
3. *Individual*: It is called an object and represents a single item of our world
4. *Property*: It describes qualities common to all the individuals of the class and represents relationships in ontologies
5. *Property restriction*: They are employed in order to ”shape” properties (e.g cardinality restrictions)

2.1.3 Logic

As it is referred in [5] logic is the ”*foundation of knowledge representation*”. One of the main characteristics of logic is the proof systems that exist that provide a way to reason in order to inference new knowledge. In Semantic Web frameworks, logic aims at inferring new ontological knowledge and even better serves as a decision making mechanism. However, reasoning methods, usually, are suitable for *crisp logic*, i.e statements that are *true/false*. As we will see next, in cases where uncertainty and vagueness exist, these methods are not applicable.

Logic is described by the following properties:

- Formal languages aim at expressing knowledge
- Well-understood formal semantics
- Automated reasoners for inferring conclusions

When we refer to Logic, we usually mean predicate logic (or first-order logic). Predicate logic provides a set of rule systems. A subset of predicate logic with efficient proof systems is called *Horn logic*. In this case, a rule has the form:

$$C_1, \ldots, C_n \rightarrow \ D$$

where $C_i, i = 1, \ldots, n$ and D are atomic formulas. Moreover, rules are divided in two categories:

- Deductive rules: If $C_i, i = 1, \ldots, n$ is *true*, then D is true
- Reactive rules: If $C_i, i = 1, \ldots, n$ is *true*, then actions is D

Another characteristic of predicate logic is that it is monotonic. This means that if a conclusion is drawn, then the validity of this conclusion is preserved, even if new knowledge is added.

Another fragment of predicate logic is *Description Logics* [9]. There is a strong connection between Description Logics and the *Semantic Web*. More precisely, as we will see next, the most widely used dialect of the OWL language, OWL-DL, is based on Description Logics.

2.2 Description Logics

Description Logics is a family of *knowledge representation languages* and provide a way to ”represent knowledge in a structured and formally well-understood way” [9]. They belong to a more general category called *description languages*. These languages allow the description of worlds providing constructors for building them [97, 9]. Generally, DLs support expressions that are built from atomic concepts and atomic roles. Each DL offers a specific level of expressiveness. DLs are a fragment First Order Logic (FOL), achieving lower complexity in expense of limited expressivity.

At first, DLs were employed in order to define semantics for semantic networks and frames [96]. The history of DLs can be summarized in five phases, as follows [96, 10]:

- *Phase 0*: This phase is characterized by the definition of semantic networks and frames, aimed at representing structured knowledge [1965-1980].
- *Phase 1* - [1980-1990]: The first DL system has been defined in this phase [23].
- *Phase 2* - [1990-1995]: In this phase, tableau algorithms have been defined [109].
- *Phase 3* - [1995-2000]: A set of optimized reasoners has been defined in this phase [50]. In addition, the first approaches of fuzzy DLs have been emerged [118].
- *Phase 4* - [2000 - Today]: A set of commercial implementation of reasoners has been defined in this phase [57].

Generally, DLs provide a way to ”represent knowledge in a structured and formally well-understood way” [9].

The basic alphabets of a DL language (or *alphabet*) are the following:

- Concept names to name Atomic concepts - C, which can be regarded as unary predicates
- Role names to name Atomic roles - R, which can be regarded as binary predicates
- Individual names to name Individuals - I

DLs use a set of axioms, for describing a state of the world. These axioms are divided into the following categories:

- *ABox*, which represents knowledge about *Named Individuals*
- *TBox*, which represents knowledge between *Concepts*
- *RBox*, which represents knowledge between *Roles*

In general, a DL Knowledge Base is composed by the following items:

- *Extensive knowledge*, which is stored as *ABox*
- *Intensive knowledge*, which is stored as *TBox* and *RBox*

In addition, DLs come with a set of inference capabilities. Generally, the basic inference problems regarding DLs are the following [10]:

- Consistency:
  It aims at determining whether the knowledge base is non-contradictory.
- Subsumption:
  It aims at determining subconcept - superconcept relationships. More precisely, a concept C is *subsumed* by a concept D if all instances of C, are necessarily instances of D.
- Instantiation:
  It aims at determining instance relationships. More precisely, an individual i is an instance of a concept C.

A DL Knowledge Base $\mathcal{KB}$ is defined as a triple:

$$\mathcal{KB} = < \mathcal{A}, \mathcal{T}, \mathcal{R} >,$$

where $\mathcal{A}$ stands for $ABox$, $\mathcal{T}$ stands for $TBox$ and $\mathcal{R}$ stands for $RBox$. A DL knowledge base is often referred to as a *DL ontology*.

As already mentioned, DLs are expressed by a family of languages. The simplest of them is called $\mathcal{ALC}$, an acronym for $\mathcal{A}$ttribute $\mathcal{L}$anguage with $\mathcal{C}$omplement - DLs apply naming convention, in order to describe the characteristics of the language-. The syntax of the DL $\mathcal{ALC}$ considers a set of concept names, denoted as N_C and a set of role names, denoted as N_R. According to [10], the sets of $\mathcal{ALC}$ *concept descriptions* is the smallest sets such that:

- $\top, \bot$ and every concept name $A \in N_C$ is an $\mathcal{ALC}$ concept
- If C and D are $\mathcal{ALC}$ concepts and $r \in N_R$, then $C \sqcap D$, $C \sqcup D$, $\neg C$, $\forall r.C$ and $\exists r.C$ are $\mathcal{ALC}$ concepts.

The semantics of description languages (and hence description logics) is related to *interpretations* (and *interpretation functions*), which actually give semantics for concept and role descriptions.

Definition 1. A DL interpretation is defined as $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is an interpretation domain and $\cdot^{\mathcal{I}}$ is an interpretation function.

An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ maps every $\mathcal{ALC}$ concept to a subset of $\Delta^{\mathcal{I}}$ and every role name to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. More precisely:

$$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}, \ \bot^{\mathcal{I}} = 0$$
$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$$
$$(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$$
$$\neg C^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$$
$$(\exists r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \textit{There is some } y \in \Delta^{\mathcal{I}} \textit{ with } <x, y> \in r^{\mathcal{I}} \textit{ and } y \in C^{\mathcal{I}}\}$$
$$(\forall r.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \textit{For all } y \in \Delta^{\mathcal{I}} \textit{ if } <x, y> \in r^{\mathcal{I}}\textit{, then } y \in C^{\mathcal{I}}\}$$

The *TBox* of a DL KB contains a set of axioms, divided in two categories:

1. *Inclusions*: These are axioms of the form $C \sqsubseteq D$, where C, D are $\mathcal{ALC}$ concepts
2. *Equalities*: These are axioms of the form $C \equiv D$, where C, D are $\mathcal{ALC}$ concepts

The *RBox* of a DL KB contains a set of role inclusion axioms, of the form $R_1 \sqsubseteq R_2$, where R_1, R_2 are $\mathcal{ALC}$ roles.

Definition 2. An interpretation $\mathcal{I}$ is a model of an inclusion axiom $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. An interpretation $\mathcal{I}$ is a model of an equality axiom $C \equiv D$ if $C^{\mathcal{I}} = D^{\mathcal{I}}$.

Definition 3. An axiom α is a consequence of a set of axioms $\mathcal{O} = \{\tau_1, \ldots, \tau_n\}$ or $\mathcal{O} = \{\tau_1, \ldots, \tau_n\}$ entails α, written as $\mathcal{O} \models \alpha$ if α holds in every model of $\mathcal{O}$.

The *ABox* of a DL KB contains a set of axioms, divided in two categories:

- Individual assertion: It states that an individual is an instance of a given concept. It has the form $x : C$, where x is an individual name and C is a concept name.
- Pair of individuals assertion: It states that a pair of individuals in an instance of given role. It has the form $(x, y) : r$, where x, y are individual names and r is a role name.

Definition 4. An interpretation $\mathcal{I}$ is a model an assertional axiom $x : C$ if $x^{\mathcal{I}} \in C^{\mathcal{I}}$. In addition, an interpretation $\mathcal{I}$ is a model an assertional axiom $(x, y) : r$ if $(x^{\mathcal{I}}, y^{\mathcal{I}}) \in r^{\mathcal{I}}$.

Definition 5. It is stated that an interpretation $\mathcal{I}$ is a model of a *TBox* (or an *ABox*) if it satisfies every *TBox* (or *ABox*) axiom.

As is described in [9], Description Logics offer a structured and formally well-understood way to describe specific knowledge in a certain application domain. Using the atomic concepts (or concept names) and atomic roles (or role names), as well as the Boolean constructors conjunction $\sqcap$, negation $\neg$, existential restriction $\exists R.C$, universal restriction $\forall R.C$ etc., complex concepts or *complex descriptions* can be built. The set of constructors provided each time defines the description logics' language that is used. A description can be defined as an equality axiom, whose left-hand side is an atomic concept.

As an example the concept "*Human and not Female*" in a DL syntax is represented as:

$$Human \sqcap \neg Female$$

As we see in this example, in order to define the description, the Boolean constructors *conjunction* ($\sqcap$) and ($\neg$) are employed. Other constructors are disjunction ($\sqcup$), existential restriction ($\exists\ r.C$) and value restriction ($\forall\ r.C$). The descriptions are employed in order to build *ABox*, *TBox* and *RBox* axioms of the KB. For example, the description

$$Working \sqcap Female \sqcap \exists\ parent.Naughty$$

denotes the concept of an exhausted woman.

ABox axioms are employed in order to describe knowledge about individuals. For example, a concept assertion has the form:

$$loukia : Mother$$

This assertion actually states that "*Loukia is an instance of the concept Mother*".

2.2.1 The most well-known DLs

In this subsection we will overview some of the most well-known DLs. As we have stated, each DL follows a naming convention as a string of capital letters. $\mathcal{AL}$(*Attributive* $\mathcal{L}$*anguage*) is the most simplified DL. $\mathcal{AL}$ provides the following concept constructors:

Table 1: DL Constructors

Label	Constructor
$\mathcal{U}$	Concept Union
$\mathcal{E}$	Unrestricted Existential Restriction
$\mathcal{C}$	Complement, Concept Negation
$\mathcal{R}_+$	Transitive Roles
$\mathcal{H}$	Role Hierarchies
$\mathcal{O}$	Nominals
$\mathcal{I}$	Inverse Roles
$\mathcal{F}$	Functional Roles
$\mathcal{N}$	Number Restrictions
(D)	Datatypes
$\mathcal{R}$	Additional Role Constructors
(o)	Role Composition
$\mathcal{Q}$	Qualified Number Restrictions

1. Top and bottom concepts
2. Atomic concepts
3. Negation of atomic concepts
4. Concept intersection
5. Universal quantifications
6. Existential restrictions, restricted to the top concept

In addition, $\mathcal{AL}$ can be extended with the constructors summarized in Table 1, resulting in various DLs. Some well known DLs are $\mathcal{ALC}$ $\mathcal{FL}^-$ [22] $\mathcal{EL}$ [75] $\mathcal{EL}++$ [8] DL-Lite [6] $\mathcal{S}$, which is an equivalent to $\mathcal{ALCIF}$ [6] $\mathcal{SHIF}(\mathbf{D})$ [65] $\mathcal{SHOIN}(\mathbf{D})$ [63] $\mathcal{SROIQ}(\mathbf{D})$ [59]

2.2.2 DLs and OWL

DLs have gained their popularity due to their applicability in ontology languages. As it is referred in paragraph 2.1.2 OWL is the language provided for defining Semantic Web ontologies. The building blocks of OWL resemble DLs ones. More precisely, OWL employs *classes* (instead of concepts) and *properties* (instead of roles). OWL classes can either be simple classes or defined based on other classes through a set of constructors (Table 2).

There exist a set of OWL axioms summarized in Table 3. In addition, in this table the equivalence between OWL axioms and DL syntax is outlined.

OWL comes in three sublanguages:

Table 2: OWL Constructors

Constructor	DL syntax
$intersectionOf$	$C_1 \sqcap \cdots \sqcap C_n$
$unionOf$	$C_1 \sqcup \cdots \sqcup C_n$
$complementOf$	$\neg C$
$oneOf$	$\{x_1, \ldots, x_n\}$
$allValuesFrom$	$\forall p.C$
$someValuesFrom$	$\exists r.C$
$hasValue$	$\exists r.\{x\}$
$minCardinality$	$(\geq n \ r)$
$maxCardinality$	$(\leq n \ r)$
$inverseOf$	(r^-)

Table 3: OWL Axioms

OWL Axiom	DL syntax
$subClassOf$	$C_1 \sqsubseteq C_2$
$equivalentClass$	$C_1 \equiv C_2$
$subPropertyOf$	$P_1 \sqsubseteq P_2$
$equivalentProperty$	$P_1 \equiv P_2$
$disjointWith$	$C_1 \sqsubseteq \neg C_2$
$sameAs$	$\{x_1\} \equiv \{x_2\}$
$differentFrom$	$\{x_1\} \sqsubseteq \neg \{x_2\}$
$TransitiveProperty$	P transitive role
$FunctionalProperty$	$\top \sqsubseteq (\leq 1P)$
$InverseFunctionalProperty$	$\top \sqsubseteq (\leq 1P^-)$
$SymmetricProperty$	$P \equiv P^-$

- OWL Full: It employs all the OWL language primitives and it equivalent to first-order logic.
- OWL DL: It is defined as a sublanguage of OWL Full, providing a restricted set of constructors. It is named due to its correspondence with DLs and allows for efficient reasoning support.
- OWL Lite: It is defined as a subset of OWL DL, providing a limited set of constructors. Although it is easy to be implemented, the disadvantage is its restricted expressivity.

As we have stated previously, DLs are a fragment of First-Order-Logic (FOL). As such, DLs preserve the following properties:

- Open-world assumption: It states that we cannot conclude a statement to be false because we cannot show that it is true. The opposite is called closed-world assumption.
- Non-unique-name assumption: It states that when two individuals are known by different names, then they are not necessarily different ones.

When dealing with ontologies, we may come across situations when it is required to employ closed-world assumption, along with unique-name assumption. Towards this demand, OWL DLP [48] has been defined, where DLP is an acronym for Description Logic Programming. This fragment of OWL is defined as the largest fragment where the choice between open/closed-world assumption and unique-name assumption makes no difference.

2.3 Uncertainty and Vagueness

Imperfect information includes uncertainty and vagueness concepts, which are described as follows:

- *Uncertainty*: It refers to situations when information incompleteness exist in order to decide about the truthness of a fact.
- *Vagueness*: It describes imprecise concepts, or concepts lacking clarity of definition

A good example of uncertainty and vagueness is given in [84], where the word “*degree*” is used to describe both uncertainty and vagueness measurements, but with different meaning. For example,

1. ”*To some degree birds fly*” (uncertainty)
2. ”*To some degree Jim is blond and young*” (vagueness)

Table 4: The Family of Fuzzy Logics

Family	t-norm $\alpha \bigotimes \beta$	t-conorm $\alpha \bigoplus \beta$	complement $\ominus\alpha$	implication $\alpha \Rightarrow \beta$
Zadeh	$min\{\alpha, \beta\}$	$max\{\alpha, \beta\}$	$1 - \alpha$	$max\{1 - \alpha, \beta\}$
Gödel	$min\{\alpha, \beta\}$	$max\{\alpha, \beta\}$	$\left\{ \begin{array}{ll} 1, & \alpha = 0 \\ 0, & \alpha > 0 \end{array} \right\}$	$\left\{ \begin{array}{ll} 1, & \alpha \leq \beta \\ \beta, & \alpha > \beta \end{array} \right\}$
Lukasiewicz	$max\{\alpha + \beta - 1, 0\}$	$min\{\alpha + \beta, 1\}$	$1 - \alpha$	$min\{1 - \alpha + \beta, 1\}$
Product	$\alpha \times \beta$	$\alpha + \beta - \alpha \times \beta$	$\left\{ \begin{array}{ll} 1, & \alpha = 0 \\ 0, & \alpha > 0 \end{array} \right\}$	$\left\{ \begin{array}{ll} 1, & \alpha \leq \beta \\ \beta/\alpha, & \alpha > \beta \end{array} \right\}$

3. ”*Tomorrow, it will be a nice day*” (uncertainty and vagueness)

The need for representing imperfect information in Semantic Web environments resulted in the definition of ontologies with uncertain and vague concepts [84].

2.4 Fuzzy Logic and Fuzzy Sets

In this section, we overview the basics of Fuzzy Logic and Fuzzy Sets [150, 145, 11]. Fuzzy logic [142] is the logic of imprecision and approximate reasoning [150]. It is the framework for describing *vagueness*, by assigning truth values to linguistic variables [144] and aims at representing the human way of thinking. The general idea is that Fuzzy Sets' elements can belong to some degree to the set. More precisely, vagueness actually considers statements that are true to a certain degree, taken in the truth space [0,1]. In other words, statements are *graded*. Vagueness is associated with a set of vague concepts, e.g *low cost*. What is more is that vagueness is the result of ambiguity that describes information. For example a \$100 ticket can be considered expensive for some people and low cost for others. The intuition behind the degree of membership is that the higher it is the more related is the object to the vague concept.

A characteristic of vague statements is that they are truth functional. There are four categories of fuzzy logic [19], and actually we talk about ”*families or fuzzy logics*” (Table 4).

In [142], a fuzzy set is defined as ”*a class of objects with a continuum of degrees of membership*”. In order to define a fuzzy set, a space of points, X, is determined first. The space of points constitutes the vague concept. For example, if we are looking for low cost tickets, then X represents all the tickets that are *considered* low cost. The degree of membership for each object that belongs to X is assigned through a membership function $f_A(x)$ where A is a fuzzy subset of X. This degree of membership is a number between 0 and 1 that is assigned to each class object.

Membership functions are mathematical tools for indicating flexible membership to a set, modeling and quantifying the meaning of symbols. In order to represent a fuzzy set, one way is to employ a set of membership functions. Some of them are:

1. Trapezoidal
2. Triangular
3. Left-shoulder
4. Right-shoulder

Another characteristic of fuzzy sets are the *fuzzy modifiers* [16]. When applying to a fuzzy set, this modifier actually changes the membership function. Formally, it is represented as a function:

$$f_m : [0, 1] \rightarrow [0, 1]$$

For example, the modifier *very* is usually defined as $f_{very}(x) = x^2$.

To sum up, Fuzzy Logic can be defined as a logic of imprecise reasoning. It is a multiple-valued logic with a continuum of truth values. As we will see in following sections, Fuzzy Logic is employed in Description Logics environment allowing for knowledge representation and reasoning.

2.5 Uncertainty Modelling

Uncertainty can be modelled through various methods [156, 107, 35, 130]. Generally, uncertainty can be divided into [111, 53]:

1. *Aleatory Uncertainty*: It results from the fact that a system can behave in random ways. In that case, uncertainty is represented by relative frequencies.
2. *Epistemic Uncertainty*: It results from the lack of knowledge about a system. In that case, uncertainty reflects subjective assessments of likelihood.

The main idea of the aforementioned notions is that, generally, people do not use the probability measure to describe ignorance. This is the most usual cause of uncertainty on the Semantic Web [51].

Generally, uncertainty is modelled through the notion of possible worlds often called *states* or *elementary outcomes*. More precisely, an uncertainty framework, or an agent that operates on an uncertainty environment, is defined over the following concepts [51]:

- $\mathcal{W}$: Set of possible worlds
- $\mathcal{U} \subseteq \mathcal{W}$: A subset of $\mathcal{W}$
- $\mathcal{A} \subseteq \mathcal{W}$: Subset that the agent considers possible, i.e. a qualitative measure of uncertainty

- $\mathcal{U} \bigcap \mathcal{A} \neq \emptyset$: $\mathcal{U}$ is possible
- $\mathcal{A} \subseteq \mathcal{U}$: The agent knows $\mathcal{U}$

A possible world $\mathcal{I}$ is defined as:

$$\mathcal{I} : \mathcal{W} \rightarrow \{0, 1\}$$

If we consider an elementary outcome $\phi \in \mathcal{W}$ and a world $\mathcal{I}$ then, $\mathcal{I}(\phi) = 1$ means that ϕ is true in $\mathcal{I}$, denoted as $\mathcal{I} \models \phi$. When dealing with uncertainty, we do not know which possible world prevails. Towards this, many approaches exist. One may consider a probability distribution on $\mathcal{W}$. As we will see later, in cases of epistemic uncertainty Dempster-Shafer theory is the most usual representation framework. As such, in our approach, we consider this theory for modelling uncertainty as it is considered more suitable for the SW environment.

2.6 Dempster-Shafer Model and Dempster's rule of Combination

In the Semantic Web environment, usually, uncertainty comes as a result of ignorance, which in turn, is due to *incomplete information*. In other words, we talk about epistemic uncertainty. In those cases, the classical notion of probability cannot be considered suitable for the following reasons [51]:

1. Probability is not as good at representing ignorance.
2. An agent cannot always define probabilities for all sets of possible worlds.
3. In some cases, the computational effort demanded for probability definition, might be prohibitive.

Dempster-Shafer theory [113, 114, 79] is considered a mathematical theory of evidence, that quantifies uncertainty in cases of ignorance and comes as a generalization of the Bayesian theory of subjective probability judgement. This theory is also known as *Theory of Belief Functions* or *Evidence Theory*. Bayesian theory quantifies judgements by assigning probabilities to the set of possible answers. Dempster-Shafer theory allows for deriving degrees of belief for a specific question based on probabilities for another related question.

Dempster-Shafer theory is defined over two main ideas:

- Obtaining degrees of belief for one question from subjective probabilities for a related question
- Dempster's rule of Combination, for combining such degrees of belief in cases of independent pieces of evidence

The basic notion of Dempster-Shafer theory is the *belief function*, or support function. This theory attaches likelihood to events. It can be regarded as a generalization of probability theory in the sense that probabilities are assigned to sets rather than singletons. The strength of this model resides on the fact that it allows for *levels of precision*. In literature there exist some approaches that employ Dempster-Shafer theory in real-world applications, e.g. [55, 1, 14, 91].

Dempster-Shafer theory employs a set of functions, namely:

- Basic probability assignment
- Belief function
- Plausibility function

The belief function can be described as *a measure of evidence that supports an event*. Dempster-Shafer theory considers a *frame of discernment*, which is defined as the set of *different and mutually exclusive events*. Another characteristic of the frame of discernment is that the propositions contained are *exhaustive* [131].

Dempster-Shafer theory considers the following [148]:

- Combination of evidence
- Data fusion

Dempster-Shafer model provides us with the ability to ”assess *belief* on some space Y on which the existence of probability measure is acknowledged, but not precisely known in that the probability is known for some of its subsets, not for all of them”[111]. This theory has been evolved as a method for representing incomplete information, by employing the concept of the *basic probability assignment*. More precisely, Dempster-Shafer theory considers two spaces, X, Y. On these spaces, a compatibility relation C is defined, as true/false statement, in the following way:

$x \in X$ compatible to $y \in Y$, denoted as xCy, if it is possible x is an answer to X and y to Y in the same time

The compatibility relation is used in order to define the granule of an element $x \in X$:

$$G(x) = \{y \mid y \in Y, xCy\}$$

Given a probability distribution on space X, the basic probability assignment of a subset A of Y, where Y is called the Frame of Discernment, in Dempster-Shafer theory is defined as following:

$$m(A) = \frac{\sum_{G(x_i)=A} p(x_i)}{1 - \sum_{G(x_i)=\emptyset} p(x_i)}$$

In the definition above A is called a *focal element*. Basic probability assignment is used in order to define *belief* and *plausibility* functions, which constitute lower and upper probability measures, given a probability distribution on Y, respectively:

$$Bel(B) = \sum_{A \subseteq B} m(A)$$

$$Pl(B) = \sum_{A \bigcap B \neq \emptyset} m(A)$$

From the definition of Belief and Plausibility formulas, it follows that:

$$Pl(B) = 1 - Bel(\overline{B})$$

where $\overline{B}$ is the complement of B.

Taking into account an $A \subseteq Y$ and a mapping $Bel : 2^Y \to [0,1]$, then Bel is a Belief function if and only if the following hold:

- Axiom 1: $Bel(\emptyset) = 0$
- Axiom 2: $Bel(Y) = 1$
- Axiom 3:
  For $A_1, A_2, \ldots, A_n \subset Y$, $Bel(\cup_{i=1}^{n} A_i) \geq \sum_{I \subset \{A_1, A_2, \ldots, A_n\}, I \neq \emptyset} (-1)^{|I|+1} Bel(\cap_{i \in I} A_i)$

In cases of $n = 2$, axiom 3 can be written as: $Bel(A_1 \cup A_2) \geq Bel(A_1) + Bel(A_2)$. As we see, a belief function in non-additive, as opposed to a probability function.

In addition, for any proposition, the plausibility degree cannot be less that its belief degree, i.e:

$$Bel(A) \leq Pl(A), A \in Y$$

Given a belief function for $A \subseteq Y$, the basic probability assignment is constructed as follows:

$$m(A) = \sum \{(-1)^{|A-B| Bel(B) | B \subseteq A}\}$$

where $|A - B|$ is the cardinality of $A - B$.

Dempster's rule of Combination is defined on two basic probability assignments m_1, m_2, derived from independent sources:

$$m_1 \bigoplus m_2(B) = \frac{\sum_{A_i \bigcap A_j = B} m_1(A_i) m_2(A_j)}{1 - \sum_{A_i \bigcap A_j = \emptyset} m_1(A_i) m_2(A_j)}$$

Dempster's rule can be employed in cases of statistically independent evidence. The factor $1 - \sum_{A_i \bigcap A_j = \emptyset} m_1(A_i) m_2(A_j)$ which describes conflicting evidence is called a normalization factor. A discussion on the validity of normalization can be found in [141]. Dempster's rule of combination has gained more and more popularity in cases of information combination. Such an application is described in [2], where two different classifiers are combined through Dempster-Shafer model. In addition, in [134], another approach that aims on sensor fusion is described.

2.6.1 Computing combined beliefs - Monte Carlo Algorithms

When dealing with Dempster-Shafer theory, an important issue we have to tackle is the complexity of Dempster's rule of Combination, since the straightforward application of the rule is exponential. For calculating Dempster-Shafer combined belief, Monte Carlo algorithms are employed [88, 131, 132].

These algorithms aim at providing an approximate value of the combined belief. An algorithm considers a frame of discernment Θ and a source triple over $\Theta = (\Omega, P, \Gamma)$, where Ω is a finite set, P is a probability function on Ω and Γ is a function from Ω to 2^{Θ}, such that $\forall \omega \in \Omega,\ \Gamma(\omega) \neq \emptyset$ and $P(\omega) \neq \emptyset$. In addition, a mass function (and a belief function) is associated with a source triple, defined as $m(X) = \sum_{\omega:\Gamma(\omega)=X} P(\omega)$ and $Bel(X) = \sum_{\omega:\Gamma(\omega)\subseteq\, X} P(\omega)$, where $X \subseteq \Theta$. Each belief function represents an independent piece of evidence. For calculating combined evidence through Dempster's rule, a mapping of a set of source triples $(\Omega i, P_i, \Gamma_i), i = 1, \ldots, m$ to a single triple (Ω, P_{DS}, Γ) is defined as follows: Let $\overline{\Omega} = (\Omega_1 \times \Omega_2 \times \Omega_m)$. Following, if $\omega \in \overline{\Omega}$, then ω_i is its ith component and ω is written as $\omega = (\omega_1, \ldots, \omega_m)$. In addition, $\Gamma' : \overline{\Omega} \rightarrow 2^{\Theta}$ is defined by $\Gamma'(\omega) = \cap_{i=1}^{m} \Gamma_i(\omega_i)$ and probability function P' on $\overline{\Omega}$ by $P'(\omega) = \Pi_{i=1}^{m} P_i(\omega_i), \omega \in \overline{\Omega}$. Let Ω a set defined as $\{\omega \in \overline{\Omega} : \Gamma'(\omega) \neq \emptyset\}$, let Γ be Γ' restricted to Ω and let probability function P_{DS} on Ω be P' conditioned on Ω, i.e $P_{DS}(\omega) = P'(\omega)/P'(\Omega), \omega \in \Omega$. Then, the combined belief measure over Θ is defined as $Bel(X) = P_{DS}(\{\omega \in \Omega : \Gamma(\omega) \subseteq X\}), X \in \Theta$. As Θ is getting larger, and taking into account that there are exponentially subsets of Θ, it is not feasible to calculate belief measure in all of these subsets.

A simple Monte Carlo algorithm considers a large set of trials. In each trial, a random $\omega \in \Omega$ with chance $P_{DS}(\omega)$ is selected. Then, if $\Gamma(\omega) \subseteq\ X, X \in \Theta$, the value of the trial equals 1, otherwise equals 0. Following, $Bel(X)$ is estimated through the average value of the trials. The time demanded is roughly proportional to $|\Theta| m/P'(\Omega)$. However, in cases of strongly conflicting evidence, the algorithm in inefficient. An alternative to this simple algorithm is the Markov-Chain Monte-Carlo Algorithm [88]. In this case, trials are not independent, but form a Markov Chain. This allows for each trial result to depend on the result of the previous trial. Another method for efficiently calculating belief degrees is outlined in [132]. This algorithm is almost linear in the size of the frame of discernment. In [115], an algorithm for computing combined belief in cases of hierarchical evidence is described. In [152], a combination method based on rule-based systems is outlined.

2.6.2 Evidential operations

In [55, 1, 126, 140], approaches based on evidential operations are outlined. These approaches represent relationships between a set of network nodes for propagating belief distributions in order to infer activities. For this reason, a *discounted mass function* is defined as:

$$m^r(A) = \left\{ \begin{array}{ll} (1-r)m(A), & A \subset \mathcal{W} \\ r + (1-r)m(\mathcal{W}), & A = \mathcal{W} \end{array} \right\}$$

where $\mathcal{W}$ is a frame of discernment, A is an element of the power set, $m(A)$ is the mass function of A and r is a discounting rate, taking values in $[0, 1]$. We distinguish among the following values of r:

- $r = 0$: The source is absolutely reliable
- $r = 1$: The source is completely unreliable
- $0 < r < 1$: The source is reliable with a rate reduction r

In addition, a multivalued mapping, Γ serves as a way to represent the relationship between two sources (frames of discernment), Θ_A, Θ_B. These sources represent evidence for the same problem. Γ is defined as a function mapping, as follows:

$$\Gamma : \Theta_A \leftarrow \ 2^{\Theta_B}$$

The mapping function assigns to each element $e_i, i = 1, \ldots, n$ of Θ_A a subset $\Gamma(e_i)$ of Θ_B.

When the relationship between an element $e_i \in \Theta_A$ and a subset $\Gamma(e_i) \subseteq \Theta_B$ is uncertain, then an evidential mapping Γ^* is employed. The evidential mapping assigns probabilities to an element $e_i \in \Theta_A$. Then, belief distributions of Θ_A are propagated to Θ_B, using evidential mapping. More precisely, the evidential mapping is defined as:

$$\Gamma^*(e_i) = \{(H_{ij}, f(e_i \rightarrow \ H_{ij})), \ldots, (H_{im}, f(e_i \rightarrow \ H_{im}))\}$$

where $e_i \in \Theta_A$, $H_{ij} \subseteq \Theta_B$, $i = 1, \ldots, n$ and $j = 1, \ldots, m$, satisfying:

$$\begin{aligned}
&H_{ij} \neq \emptyset, j = 1, \ldots, m \\
&f(e_i \rightarrow \ H_{ij}) > 0, j = 1, \ldots, m \\
&\sum_{j=1}^{m} f(e_i \rightarrow \ H_{ij}) = 1 \\
&\Gamma^*(\Theta_A) = \{(\Theta_B, 1)\}
\end{aligned}$$

Finally, the propagation is defined as:

$$m'(H_j) = \sum_i m(e_i) f(e_i \rightarrow \ H_j)$$

where $H_j \in \{H_{i1}, \ldots, H_{im}\}$ and $\Gamma^*(e_i) = \{(H_{ij}, f(e_i \rightarrow \ H_{ij})), \ldots, (H_{im}, f(e_i \rightarrow \ H_{im}))\}$.

2.6.3 Generalizing Dempster-Shafer theory for Fuzzy Sets

In [139], a generalization of Dempster-Shafer framework for fuzzy sets, along with Dempster's rule of combination, is outlined. More precisely, the compatibility relation is not defined as a yes/no answer, but it is defined as a joint possibility distribution, between two spaces, X and Y, i.e. $C(x,y) = \Pi_{\Psi,\Omega}(x,y), x \in X, y \in Y$, where Ψ,Ω are variables that take values from spaces X and Y respectively. Yen assumes a frame of discernment $\mathcal{W} = \{x_1, \ldots, x_k\}$ and n subsets of $\mathcal{W}$, denoted as $A_1, \ldots, A_n$. An A_j is called a *focal element* of a function: $m : \mathcal{W} \rightarrow [0,1]$, iff $m(A_j) > 0, j = 1, \ldots, n$.

Then, Belief and Plausibility of a subset $B \subseteq \mathcal{W}$ are considered as lower and upper probabilities of this subset. The Belief function is regarded as an optimization problem [137, 89] which is to find $min \sum_{x_i \in B} \sum_{j=1}^{n} m(x_i : A_j)$. This optimization problem has to be aligned to the following constraints:

$$m(x_i : A_j) \geq 0, i = 1, \ldots, k$$
$$m(x_i : A_j) = 0, \forall x_i \notin A_j$$
$$\sum_i m(x_i : A_j) = m(A_j), j = 1, \ldots, n$$

where $m(x_i : A_j)$ is a variable that denotes the probability mass assigned to x_i due to the basic probability assignment of focal A_j. The formula $\sum_{x_i \in B} \sum_{j=1}^{n} m(x_i : A_j)$ corresponds to the probability of subset B. The key-point in the optimization procedure considers that focals' masses do not interact with one another. This means that the optimization can be proceeded independently. The optimal value is defined as the sum of a set of subproblems, each one defined as $min \sum_{x_i \in B} m(x_i : A_j)$.

Considering Plausibility degree, min is replaced by max in the above formulae, thus giving the following formula: $max \sum_{x_i \in B} m(x_i : A_j)$. The optimal solutions to the above subproblems are denoted as $m_*(B : A_j)$ and $m^*(B : A_j)$ for Belief and Plausibility, respectively.

The core idea of this method resides on the fact that Belief and Plausibility measures are assigned to fuzzy subsets. A similar framework is defined in [154], where fuzzy sets are generalized in a way that a value in $[0,1]$ is assigned to a set of elements, rather than a single element.

In [73], a formalism for representing probabilistic knowledge and fuzzy knowledge through Dempster-Shafer theory is outlined. Since both types of knowledge are representing through Dempster-Shafer model, then Dempster's rule of Combination is employed for combining these two types of knowledge.

2.7 Probabilistic Knowledge Bases - Probabilistic Description Logics - Probabilistic Ontologies

A Knowledge Base is a part of a Knowledge Based System, which constitutes a mechanism for solving complex problems. Ontologies have been emerged as a way to tackle information sharing among different knowledge bases. In this context, uncertainty issues are represented through a *probabilistic knowledge base* [84, 82] and probabilistic logic [45, 92]. More precisely, a probabilistic knowledge base consists of the following:

- A finite non-empty set of basic events $\Phi = \{p_1, p_2, \ldots, p_n\}$

- An event ϕ is a boolean combination of basic events
- A set of constraints
  - Logical Constraints, denoted as $\nu \Leftarrow \mu$, interpreted as μ *implies* ν, where ν, μ are events
  - Conditional Constraints, denoted as $(\nu \mid \mu)[k, m]$, interpreted as the conditional probability of ν given μ is in the interval $[k, m], k, m \in [0, 1]$. Conditional constraints (also called interval restrictions for conditional probabilities) are divided into *strict*, i.e. statements that always hold, and *defeasible*, i.e. constraints that represent weaker conditions.
- The strict and defeasible constraints are defined as ($\mu|\nu)[k, m]$ and $(\mu \parallel \nu)[k, m]$, $[k, m] \in [0, 1]$ respectively.
- A probabilistic default theory is defined as $T = (P, D)$, where P is a finite set of strict conditional constraints and D is a finite set of defeasible conditional constraints.
- Set of strict probabilistic formulas: It is defined as the closure of the set of all strict conditional statements under the Boolean operators $\wedge$ and $\neg$.
- Set of probabilistic formulas: It is defined as the closure of the set of all conditional statements under the Boolean operators $\wedge$ and $\neg$.
- A possible world is a truth assignment $I : \Phi \rightarrow \{true, false\}$
- I_Φ denotes the set of all possible worlds
- Probabilistic interpretation - Pr: It is a probability function that assigns to each possible world I a number [0,1]. The probabilistic interpretation suggests a probability ordering among possible worlds.
- Satisfaction of probabilistic formula: A probabilistic interpretation Pr satisfies (or Pr is a model of) a probabilistic formula (strict or defeasible) iff $Pr(\nu \mid \mu)\epsilon[k, m]$.
- Verification of a default: A Pr verifies a default $Pr(\nu \parallel \mu)\epsilon[k, m]$ if $Pr(\mu) = 1$ and Pr satisfies $Pr(\nu \parallel \mu)\epsilon[k, m]$. In addition, a set of defaults D *tolerates* a default d *under* a set of strict conditional constraints P iff $P \bigcup D$ has a model, i.e. a Pr that satisfies it, that also satisfies d. If such model does not exist, it is stated that D is *in conflict* with d.
- Default ranking σ: It is defined on a set of defaults D as: $\sigma : d \rightarrow \{0...1\}$, for each $d \in D$. It is admissible with $T = (P, D)$ iff each $D' \subseteq D$ that is under P in conflict with a default d, has a $d' : \sigma(d') < \sigma(d)$. If such an admissible default ranking exists, then T is σ *-consistent*.

- Probability ranking κ : It is a mapping that assigns a ranking number to each probability interpretation, i.e $\kappa : \quad P_r \rightarrow \quad \{0...1\} \bigcup \infty$, where $\kappa(P_r) = 0$ for at least one P_r. Also for any satisfiable formula F, $\kappa(F) = min(\kappa(P_r) \mid P_r \models F)$. If F is not satisfiable, then $\kappa(F) = \infty$. Also, if $\kappa(\neg F) = \infty$ then κ is *admissible* with F. Considering a default formula D, then κ is *admissible with* D iff $\kappa((\mu \mid \top)[1,1])$ and $\kappa((\mu \mid \top) \wedge [1,1](\nu \mid \mu)[k,m]) < \kappa((\nu \mid \top)[1,1] \wedge \neg(\mu \mid \nu)[k,m])$

A set of logical constraints $\mathcal{L}$ and a set of conditional constraints $\mathcal{C}$ define a probabilistic Knowledge Base as $\mathcal{KB} = (\mathcal{L}, \mathcal{C})$.

The semantics of a probabilistic knowledge base considers a set of all possible worlds $\mathcal{I}$, where each possible world $I \in \mathcal{I}$ assigns a true/false value in each $p_i \in \Phi, i = 1, \ldots, n$. If an event ϕ is true in I, it is denoted as $I \models \phi$. A probabilistic knowledge base assigns a probability interpretation $\mathcal{P}$ on $\mathcal{I}$. The probability of an event ϕ is defined as:

$$\mathcal{P}(\phi) = \sum_{I \models \phi} \mathcal{P}(I)$$

The conditional probability $\mathcal{P}(\nu \mid \mu)$ is defined as:

$$\mathcal{P}(\nu \mid \mu) = \frac{\mathcal{P}(\nu \wedge \mu)}{\mathcal{P}(\mu)}$$

Based on the conditional probability definition, the assignment of a truth/false value of a logical $\nu \Leftarrow \mu$ and conditional $(\nu \mid \mu)[k,m]$ constraint, where $\mathcal{P}(\mu) > 0$ is defined as:

$$\mathcal{P} \models \nu \Leftarrow \mu, \text{ iff } \mathcal{P}(\nu \wedge \mu) = \mathcal{P}(\mu)$$
$$\mathcal{P} \models \nu \mid \mu[k,m], \text{ iff } \mathcal{P}(\nu \mid \mu) \in [k,m]$$

A probabilistic interpretation $\mathcal{P}$ is a *model* of a probabilistic knowledge base $\mathcal{KB} = (\mathcal{L}, \mathcal{C})$ iff:

$$\mathcal{P} \models \mathcal{K}, \forall \mathcal{K} \in \mathcal{L} \cup \mathcal{C}$$

A conditional constraint $(\nu \mid \mu)[k,m]$ is a logical consequence of a probabilistic $\mathcal{KB}$, denoted as $\mathcal{KB} \mid\models (\nu \mid \mu)[k,m]$, iff every model of $\mathcal{KB}$ is also a model of $(\nu \mid \mu)[k,m]$.

Extensions of DLs with probabilistic framework are described in [52, 47, 49]. In [86], a probabilistic DL that applies a probability distribution on a set of possible worlds is defined. More precisely, a probabilistic interpretation $\mathcal{I}$ is defined as follows:

$$\mathcal{I} = (\Delta^{\mathcal{I}}, \mathcal{W}, (\mathcal{I}_w)_{w \in \mathcal{W}}, \mu)$$

In the definition above, $\Delta^{\mathcal{I}}$ is a DL interpretation domain, $\mathcal{W}$ is the set of possible worlds and μ is a probability distribution on $\mathcal{W}$. In addition $\mathcal{I}_w$ is a classical DL interpretation with domain $\Delta^{\mathcal{I}}$.

Probabilistic ontologies [27, 84] are defined through a probabilistic knowledge base. In [84], a probabilistic ontology is defined as a twofold notion:

- Considering terminological knowledge, i.e concepts and roles, this is extended into terminological probabilistic knowledge
- Considering assertional knowledge, i.e. instances of concepts and roles, this is extended into assertional probabilistic knowledge

These probabilistic ontologies are called into existence in order to fulfil the following tasks:

- Representation of terminological and assertional probabilistic knowledge
- Information retrieval
- Ontology matching
- Probabilistic data integration

In [27], another approach that aims at probabilistic ontological mapping is outlined. For this reason, a suitable logic-based representation formalism is employed, allowing for reasoning, uncertainty representation and inconsistency handling.

In [32], a probabilistic ontology is defined as an explicit, formal knowledge representation that expresses knowledge about a domain of application. The extension of OWL is defined as PR-OWL that allows for the representation of complex Bayesian probabilistic models. In addition, the semantics of OWL is extended in order to be aligned with the semantics of first-order Bayesian logic. In [36], Bayesian networks, a graphic model for knowledge representation under uncertainty is employed in OWL environment. For this reason, at first, OWL is extended with probabilistic markups. Practically, this means that probability measures annotate individual OWL properties and concepts. In addition, a set of rules is defined in order to map these annotation in a Bayesian network. Other approaches for representing uncertainty in Semantic Web are described in [27, 26].

2.7.1 Entailment approaches

In [82], a set of techniques for probabilistic reasoning based on statistical knowledge and degrees of belief is outlined. More precisely, three entailment methods are described:

z-entailment This method considers a $\sigma - consistent$ probabilistic default theory $T = (P, D)$. Following, the notions of the ordered partition D, the default ranking z and probability ranking κ^z are considered. The z-partition of D is defined as $(D_0, ..., D_n)$, where each $D_i, i = 1, \ldots, n$ is the set of defaults in $D - \cup\{D_j \mid 0 \leq j < 1\}$ that are tolerated under P by $D - \cup\{D_j \mid 0 \leq j < 1\}$. Following, the the default ranking z is defined as: For $j = 0, \ldots, k$ each $d \in D_j$ is assigned the value j under z. In addition, the probability ranking κ^z is defined as:

$$\kappa^z(P_r) = \left\{ \begin{array}{lr} \infty, & \text{if } P_r \not\models P \\ 0, & \text{if } P_r \models P \cup D \\ 1 + max_{d \in D : P_r \not\models d} z(d), & \text{otherwise} \end{array} \right\}$$

In [82], it is showed that z is a default ranking admissible with T and κ^z is a probability ranking admissible with T. κ^z suggests an ordering on P_r, and it is stated that P_r is z-preferable to $P_{r'}$ iff $z(P_r) < z(P_{r'})$. In addition, a model P_r of a set of probabilistic formulas $\mathcal{F}$ is a $z - minimal$ model of $\mathcal{F}$ iff no model of $\mathcal{F}$ is z-preferable to P_r. A strict probabilistic formula $\mathcal{F}$ is a $z - consequence$ of a knowledge base KB iff each $z - minimal$ model of $P \cup \{KB\}$ satisfies $\mathcal{F}$. In addition, a strict conditional constraint $(\psi \mid \phi)[l, u]$ where $l, u \in [0, 1]$ is a *tight* $z - consequence$ of KB iff l (resp. u) is the infimum (resp. supremum) of $P_r(\psi \mid \phi)$ subject to all $z - minimal$ models P_r of $P \cup \{KB\}$ with $P_r(\phi) > 0$.

Lexicographic entailment Let $T = (P, D)$ a σ-consistent probabilistic default theory. As in the previous case, a z-partition D is defined, denoted as $(D_0, ..., D_n)$. Each $D_i, i = 1, \ldots, n$ contains a set of defaults that are tolerated under P. A probabilistic interpretation P_r is *lexicographically preferable* to P'_r iff there exists $i \in \{0, \ldots, n\}$ such that $| \{d \in D_i | P_r \models d\} | > | \{d \in D_i | P'_r \models d\} |$ and $| \{d \in D_j | P_r \models d\} | = | \{d \in D_j | P'_r \models d\} |$, $\forall\ i < j \leq n$. A model P_r of a set of probabilistic formulas $\mathcal{F}$ is a *lexicographically minimal model* of $\mathcal{F}$ iff no model of $\mathcal{F}$ is lexicographically preferable to P_r. In addition, a strict probabilistic formula $\mathcal{F}$ is a *lexicographic consequence* of KB iff each lexicographically minimal model of $P \cup \{KB\}$ satisfies $\mathcal{F}$. A strict conditional constraint $(\psi \mid \phi)[l, u]$ is a tight lexicographic consequence of KB iff l (resp. u is the infimum (resp. supremum) of $P_r(\psi \mid \phi)$ subject to all lexicographically minimal models P_r of $P \cup \{KB\}$ with $P_r(\phi) > 0$.

Conditional entailment Let $T = (P, D)$ a probabilistic default theory. A *priority ordering* on D is defined, denoted as $\prec$, as an irreflexive and binary relation on D. Following, it is stated that $\prec$ is admissible with T iff each set of defaults $D' \subseteq D$ that is under T in conflict with some default $d \in D$ contains a default d' such that $d' \prec d$. In addition, T is $\prec$-consistent iff a priority ordering on D exists that is admissible with T. Let P_r, P'_r two probabilistic interpretation and $\prec$ a priority ordering on D. Then, it is stated that P_r is $\prec$-preferable to P'_r iff $\{d \in D | P_r \not\models d\} \neq \{d \in D | P'_r \not\models d\}$ and for each $d \in D$ such that $P_r \neq d$ and $P'_r \models d$ a default $d' \in D$ exists such that $d \prec d'$, $P_r \models d'$ and $P'_r \neq d'$. Following, a model P_r of a set of probabilistic formulas $\mathcal{F}$ is a $prec$-*minimal model* of $\mathcal{F}$ iff no model of $\mathcal{F}$ is $\prec$-preferable to P_r. In addition, a model P_r is a conditional minimal model of $\mathcal{F}$ iff P_r is a $\prec$-*minimal model* for some priority ordering $\prec$ admissible with T. Finally, it is stated that a strict probabilistic formula is a conditional consequence of KB iff each conditional minimal model $P \cup \{KB\}$ satisfies this formula. A strict conditional constraint $(\psi | \phi)[l, u]$ is a tight conditional constraint of KB iff l (resp. u) is the infimum (resp. supremum) of $P_r(\psi | \phi)$ subject to all conditionally minimal models P_r of $P \cup \{KB\}$ with $P_r(\phi) > 0$.

For performing entailment, these methods incorporate generic knowledge, i.e. statistical knowledge, which comes under the term ”objective knowledge”, and evidence, which comes under the term ”subjective knowledge”. The concept of these methods is the following:

- A *Probabilistic default theory* - T as defined above is considered.
- Some evidence is also given, which constitutes a “knowledge base - KB”.
- The set of strict conditional constraints, as well as the evidence, should always be satisfied, i.e. there should be a probabilistic interpretation Pr model of them, while performing any entailment method.
- The above probabilistic interpretation should also be a model of a subset of the set of defaults in a way that any member of this set is not in conflict with any other member under the set of strict conditional constraints and the evidence.
- Also, they ignore irrelevant information, show property inheritance to globally nonexceptional subclasses, and respect the principle of specificity.

Considering inconsistency handling, those methods do not always entail ”intuitively” expected conclusions. There are situations where some methods have better performance than the others. Moreover, z-entailment and lexicographic entailment consider the principle of specificity and conditional entailment entails ignorance. Another characteristic of these approaches is that in order to perform reasoning, they entail conclusions from subjective knowledge using objective knowledge.

2.8 Possibilistic Knowledge Bases Possibilistic Description Logics and Possibilistic Ontologies

Another framework for representing uncertainty resides in Possibility theory [44]. This theory employs a pair if dual set functions, denoted as possibility and necessity measures.

Towards this, a possibilistic knowledge base, [84], based on Possibility theory [44] and possibilistic logic [39, 43, 38], annotates an event ϕ with *possibility* and *necessity* measures. Like in the probabilistic approach a set of possible worlds $\mathcal{I}$ is defined, along with a possibilistic interpretation on it:

$$\pi : \mathcal{I} \rightarrow [0, 1]$$

The possibility and necessity measures of an event ϕ, under a possibilistic interpretation π are defined as:

$$Poss(\phi) = sup\{\pi(I), I \in \mathcal{I}, I \models \phi\}$$
$$Necc(\phi) = 1 - Poss(\neg\phi)$$

where $I \models \phi$ means that ϕ is true in I.

The truthness of an event ϕ is twofold:

$$\pi \models P(\phi), \phi \geq\ m,\ \textit{iff}\ Poss(\phi) \geq\ m$$
$$\pi \models N(\phi), \phi \geq\ m,\ \textit{iff}\ Necc(\phi) \geq\ m$$

The reasoning process in a possibilistic knowledge base, can be performed by a reduction to classical entailment, in the following way [54].

Let $\mathcal{KB}$ be a possibilistic knowledge base

Let: $\mathcal{KB}_\varepsilon = \{\phi | N\phi \geq \iota \in \mathcal{KB}, \iota \geq \varepsilon\}$

$\mathcal{KB}^\varepsilon = \{\phi | N\phi \geq \iota \in \mathcal{KB}, \iota > \varepsilon\}$

Then, the following entailments exist:

$\mathcal{KB} \models\ N\phi \geq \varepsilon$ iff $\mathcal{KB}_\varepsilon \models \phi$

$\mathcal{KB} \models\ P\phi \geq \varepsilon$ iff $\mathcal{KB}^0 \models \phi$ or

$\exists\ P\psi \geq \iota \in \mathcal{KB},$ with $\iota \geq \varepsilon$ and KB$^{1-\iota} \bigcup \{\psi\} \models \phi$

A possibilistic DL is based on a possibilistic interpretation. Such an extension is described in [103, 102, 127, 40].

In [103], the syntax of a possibilistic DL is defined based on the syntax of a classical DL. A possibilistic axiom is defined as a pair (ϕ, α), where ϕ is a DL axiom and $\alpha \in (0, 1]$. Then, a possibilistic $TBox$ (resp. $ABox$ and $RBox$) is defined as a finite set of possibilistic axioms (ϕ, α), where ϕ is a $TBox$ axiom. In addition, a possibilistic DL Knowledge Base $\mathcal{B} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$ consists of a possibilistic $RBox$ $\mathcal{R}$, $TBox$ $\mathcal{T}$ and $ABox$ $\mathcal{A}$. The semantics of a possibilistic DL is defined by a possibility distribution π over the set $\mathbb{I}$ of all classical DL interpretations, i.e: $\pi : \mathbb{I} \rightarrow [0, 1]$. Then, $\pi(I), I \in \mathbb{I}$ represents the degree of compatibility of I with the information available.

2.9 Fuzzy Description Logics - Fuzzy Ontologies

The need for representing vague statements, resulted in the extension of classical DLs, with Fuzzy Set Theory and Fuzzy Logic. In [122], an overview regarding managing vagueness in Semantic Web is presented. In addition, fuzzy ontologies have been defined [25, 116, 61, 153, 21, 94]. In Semantic Web, there exist a lot of cases where vague statements are engaged, such as:

- Matchmaking
- Distributed information retrieval

- Ontology alignment

Fuzzy Description Logics [118], [84], [96], [121], [119],[120] have been defined towards this direction. Practically, this means that the DL axioms hold with a degree $\alpha \in [0, 1]$. For example, $CheapHotel \geq\ 0.7$ states that the *hotel is cheap with a degree at least 0.7*. Rough Description Logics [20] are also considered an alternative suitable only for vague knowledge representation, whereas uncertainty issues are not considered.

In [118], a fuzzy DL has been introduced by considering formulas of the form:

$$< a \quad n >, n \in [0, 1]$$

where a is classical crisp DL formula. This statement means that a *is true with degree at least* n.

In order to define semantics for a fuzzy DL, the family of fuzzy logics is employed (Table 4). This means that semantics can be defined based on a certain fuzzy logic. For example, in [118], Zadeh's fuzzy logic is used for this reason.

For the fuzzy DL semantics, fuzzy interpretations [121], [118] are introduced as an extension of a crisp DL. A fuzzy interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ assumes a domain interpretation $\Delta^{\mathcal{I}}$ and an in interpretation function $\cdot^{\mathcal{I}}$. In the crisp case, each individual (or pair of individuals) can belong on a concept $\mathcal{C}$ (or role $\mathcal{R}$), or not. The fuzzy extension actually assigns a membership degree in $[0, 1]$ for individuals (or pair of individuals). More precisely, a fuzzy axiom $\tau \geq \mu$ or $\tau \leq \nu$ holds in a fuzzy interpretation $\mathcal{I}$ iff $\mathcal{I}(\tau) \geq \mu$ or $\mathcal{I}(\tau) \leq \nu$.

As we will see later, in our framework, we apply the fuzzy $\mathcal{ALC}$ defined in [118].

Another issue that we have to note here, is the methodology for constructing fuzzy sets. As stated in [123], fuzzy sets are, usually, constructed by:

- Employing some well-known predefined membership functions, e.g a trapezoidal or triangular function.
- Employing fuzzy clustering. Fuzzy C-Means is a well known fuzzy clustering algorithm. Such a method applied in a semantic web ontology can be found in [78]. In addition, Genetic Algorithms (GAs) [112] are a framework for fuzzy clustering procedures. In [112], a set of fuzzy rules are constructed using Genetic Algorithms (GA) to learn both the antecedent and the consequent part of a rule. A two-stage approach is defined, firstly, by considering fuzzy clustering for deriving an initial rule-based model and after by optimizing this model by a GA subject to a set of constraints.

As we have mentioned in 2.1, ontologies and logic are the most significant technologies (layers) of the semantic web stack. In order to combine ontologies and rules, Description Logic Programs can be employed. In [85], such a method is presented. This approach tackles uncertainty and vagueness in ”rules layer”. In [117], OWL is extended with Fuzzy Set theory, resulted in fuzzy-OWL (f-OWL). The extension is based on OWL facts, resulting in *fuzzy facts*. The semantics is provided by a *fuzzy interpretation*. In [124, 125], the state of the art of fuzzy knowledge representation and reasoning in Semantic Web languages (RDF/RDFS and OWL) is summarized. In [81], a fuzzy ontology is defined suitable for representing imprecise knowledge. The method combines different frameworks, such as ontology-based intelligent fuzzy agents and Fuzzy Markup Language. Finally, the method is applied in the field of medical data, by providing suitable semantic decision making systems.

2.10 Probability Theory - Possibility Theory - Dempster Shafer Theory - Fuzzy Logic

Probability Theory, Possibility Theory, Dempster-Shafer Theory and Fuzzy Logic are some well known frameworks for representing imperfect information [41, 113, 106, 46, 149]. Though they are designed for representing different kinds of imperfect information, a relation among them can sometimes be detected. In addition, Zadeh's z-numbers [151] can be considered as a framework for representing imperfect knowledge. A z-number is defined as $Z = (A, B)$, where A a restriction on the values of a real-valued uncertain variable and B is a measure of reliability of the first component.

In [80, 146, 104], relations between Probability - Possibility theory and Fuzzy Logic is outlined. More precisely, it is stated that the *membership function of a fuzzy set may be interpreted as a conditional probability*. In the simple case, suppose there exist a fuzzy set A in a universe of discourse U. Then, from a Fuzzy Logic point of view, $\mu_A(u)$ is the degree of membership of u in A, where $u \in U$. In the world of Probabilities, let X be a random variable taking two values A and A', where A' stands for *not* A. Then, $\mu_A(u)$ is the conditional probability $P_r(A \mid u)$, where P_r is a probability function. As a more formal definition, if $V = \{V_1, V_2, ..., V_n\}$ is a collection of *voters* and each $V_i, i = 1, \ldots, n$ constitutes a vote for classifying u as A or A', then $P_r(A \mid u)$ can be seen as the probability that a random voter would classify u as A. Through this equivalence of the membership function of fuzzy logic and the conditional probability, we can consider a common term, or in other words, an *imperfection factor*, that covers both uncertainty and vagueness.

In [12], the method introduced in [80] is employed for defined a probabilistic alternative to Fuzzy Logic Controllers. Another approach is introduced in [143]. More precisely, let Ω denote the set of all possible outcomes. Let $\mathcal{F}$ denote a set whose members are subsets of Ω, i.e. a family of sets. In addition, a subset A of Ω is called an event. The uncertainty of the event A is described by a number $\mathcal{P}(A)$, taking values in $[0, 1]$. Then, a probability measure space is defined as a triple $(\Omega, \mathcal{F}, \mathcal{P})$. Based on this triple, Zadeh considered fuzzy subsets $\overline{A}$, denoted as *fuzzy events* and defined the probability measure of these subsets.

In [148], the correlation between Dempster-Shafer model and the theory of Possibility is outlined. More precisely, the Belief and Plausibility measures, which constitute lower and upper probability measures, respectively, are interpreted as certainty and possibility measures, respectively. In [14], an ontology for representing uncertainty is outlined.

Finally, in [64], the relation between possibility theory and fuzzy sets is described. More precisely, it is stated that a possibility distribution can be interpreted as a membership function.

3. THE INFRASTRUCTURE: DEMPSTER-SHAFER ONTOLOGICAL REPRESENTATION

As we have stated in the introduction, the Semantic Web vision introduces the concept of machine-processable information. In cases of imperfect information, i.e uncertainty and vagueness, the classical concept of ontology should be extended for capturing imperfect knowledge. Towards this concept, in this chapter we aim at representing imperfect knowledge in an ontological environment.

In [71], an ontology for manipulating uncertainty, based on Dempster-Shafer theory, is described. The basic concepts of Dempster-Shafer model are represented through a Semantic Web ontology. Following, a set of entailment methods as described in 2.7.1, is combined through a method based on Dempster's rule of Combination.

In [72], an approach for representing uncertainty and vagueness is outlined. This approach considers vague knowledge represented through a fuzzy DL. In addition, an ontology is employed for representing information in a rule/event form, in order to perform reasoning.

In [68], an approach suitable for imperfect knowledge in a matchmaking case study is outlined. Matchmaking problems [67] can be considered as a case study of Semantic Web applications. In general, a matchmaking application considers a set of criteria, set by two parts. Towards this, we propose a matchmaking method of web data based on fuzzy criteria. Our method employs Dempster-Shafer theory and Dempster's rule of Combination in order to derive a combined constraint degree that represents the degree of matchmaking between the two parts (the seeker and the offer).

In the following of this section, we describe the aforementioned approaches.

3.1 Uncertain knowledge representation and Ontologies

In [71], a method that performs on probabilistic knowledge bases and employs Dempster's rule of Combination is outlined. As we have stated in 2.7, a probabilistic knowledge base is defined as $\mathcal{KB} = (\mathcal{L}, \mathcal{C})$, where $\mathcal{L}$ is a set of logical constraints and $\mathcal{C}$ is a set of conditional constraints. These constraints are defined as:

1. **Strict conditional constraint**: They have the form $(\phi|\psi)[l, u]$ and represent generic knowledge that always hold, where ϕ, ψ are events and $l, u \in [0, 1]$.

2. **Defeasible (default) conditional constraint**: They have the form $(\phi \parallel \psi)[l, u]$ and represent weaker generic knowledge that can be defeated, where ϕ, ψ are events and $l, u \in [0, 1]$.

In addition, a probabilistic default theory, T, is defined as $T = (P, D)$, where P is a finite set of strict conditional constraints and D is a finite set of defeasible conditional constraints.

Following, in 2.7.1, a set of entailment methods for reasoning upon probabilistic knowledge bases is outlined. These are z-entailment, lexicographic entailment and conditional entailment. In our approach, these methods are combined in order to define a new (combined) entailment method. For this reason, Dempster's rule of Combination is employed.

Our method performs by considering the following:

1. *Ignores irrelevant information*
2. *Applies property inheritance to globally nonexceptional subclasses*
3. *Applies the principle of specificity*

The entailment methods defined in 2.7 derive a conclusion as a strict probabilistic formula $(\phi|\top)[l, u]$, where ϕ is an event and $l, u \in [0, 1]$. In order to combine these conclusions through Dempster's rule of Combination, a frame of discernment, $\mathcal{W}$ of mutually exclusive events should be defined. In our framework, we are interested in deciding on the truthness of the event ϕ. Thus, we consider our frame of discernment as:

$$\mathcal{W} = \{\phi, \neg\phi\}$$

In addition, subsets of $\mathcal{W}$ are associated with a basic probability assignment, bpa. In our case, there are four subsets of $\mathcal{W}$, i.e:

$$2^{\mathcal{W}} = \{\{\emptyset\}, \{\phi\}, \{\neg\phi\}, \{\phi, \neg\phi\}\}$$

The third element of the power set has a zero bpa value, since we have no information about the truthness of $\neg\phi$. In addition, the $\{\emptyset\}$ element has, by definition, a zero bpa value. Taking this into account, we regard two focal elements, $\{\phi\}$ and $\{\phi, \neg\phi\}$.

The next step is about assigning values for these focal elements. Based on the information given, we decide on three ways for representing the bpa:

1. Pessimistic approach

$$bpa(\{\phi\}) = l$$
$$bpa(\{\phi, \neg\phi\}) = 1 - l$$

2. Optimistic approach

$$bpa(\{\phi\}) = u$$
$$bpa(\{\phi, \neg\phi\}) = 1 - u$$

3. Middle approach

$$bpa(\{\phi\}) = \frac{(l + u)}{2}$$
$$bpa((\{\phi, \neg\phi\}) = 1 - \frac{(l + u)}{2}$$

Based on these bpa values, the belief values of the subsets of $\mathcal{W}$ are defined as:

1. Pessimistic approach

$$\begin{aligned}
&Bel(\{\emptyset\} = 0 \\
&Bel(\{\phi\}) = l \\
&Bel(\{\neg\phi\} = 0 \\
&Bel(\{\phi, \neg\phi\}) = 1
\end{aligned}$$

2. Optimistic approach

$$\begin{aligned}
&Bel(\{\emptyset\} = 0 \\
&Bel(\{\phi\}) = u \\
&Bel(\{\neg\phi\} = 0 \\
&Bel(\{\phi, \neg\phi\}) = 1
\end{aligned}$$

3. Middle approach

$$\begin{aligned}
&Bel(\{\emptyset\} = 0 \\
&Bel(\{\phi\}) = \frac{(l+u)}{2} \\
&Bel(\{\neg\phi\} = 0 \\
&Bel(\{\phi, \neg\phi\}) = 1
\end{aligned}$$

Having defined the representation scheme, in the following we describe the entailment method of our approach.

3.1.1 Entailment method for interval

Taking into account the strict probabilistic formulas, $(\phi|\top)[l, u]$, derived from the entailment methods, we propose an entailment method, based on Dempster's rule of Combination, More precisely, as we will show next, our method combines, through Dempster's rule of Combination, two of the entailment methods described in 2.7.1 in order to derive a combined entailment.

The steps for deriving a combined entailment result are the following:

1. Select two of the entailment methods for probabilistic knowledge bases, i.e. z, lexicographic and conditional entailment

2. Perform entailment for each method and derive a separate result as a strict probabilistic formula $(\phi|\top)[l, u]$

3. Select an approach for assigning bpa values, i.e pessimistic, optimistic and middle approach

4. Combine the results based on Dempster's rule of Combination, by consider the common frame of discernment, $\mathcal{W} = \{\phi, \neg\phi\}$, along with the bpa assignments, denoted as bpa_1 (for the first entailment method) and bpa_2 (for the second entailment method):

$$bpa_{combined}(\phi) = \frac{\sum_{x,y \in \mathcal{W}: x \cap y = \phi} bpa_1(x) \times bpa_2(y)}{1 - \sum_{x,y \in \mathcal{W}: x \cap y = \emptyset} bpa_1(x) \times bpa_2(y)}$$

5. Repeat, if necessary, steps 1-3 for adding another entailment result, by considering the new basic probability assignment, bpa_3 (for the third entailment method) and the combined $bpa_{combined}$

6. Assign $Bel_{combined}(\{\phi\}) = bpa_{combined}(\{\phi\})$

Since the entailment methods derive conclusions separately, we regard them as ”independent pieces of evidence”. The independence is a necessary precondition for employing Dempster's rule of Combination. In addition, since we are interested in subsets of the frame of discernment with a single item, i.e. $\{\phi\}$, the belief measure equals the bpa measure. Next, we overview our method through an example.

Example We consider the following probabilistic default theory:
$P = \{(bird|penguin)[1, 1]\}$
$D = \{(fly \parallel bird)[0.95, 1], (fly \parallel penguin)[0, 0.05],$
$(easy\_to\_see \parallel yellow)[0.95, 1.0]\}$
We also have the following evidence:
$KB = \{(penguin \wedge yellow|\top)[1, 1]\}$. The entailment methods described in 2.7.1 derive the following conclusions:

- z-entailment: $easy\_to\_see \mid \top[0, 1]$
- lexicographic entailment: $easy\_to\_see \mid \top[0.95, 1]$
- conditional entailment: $easy\_to\_see \mid \top[0.95, 1]$

Our goal is to derive a combined conclusion for two (or more) of the entailment methods. In our example, we choose the first and the last conclusion, i.e. z and $conditional$ entailment. We consider the following frame of discernment: $X = \{easy\_to\_see, \neg easy\_to\_see\}$. Then: $2^X = \{\emptyset, \{easy\_to\_see\}, \{\neg easy\_to\_see\}, \{easy\_to\_see, \neg easy\_to\_see\}\}$. For employing the Rule of Combination, the basic probability assignments are defined as follows:

- $bpa_z(\emptyset) = 0$, $bpa_z(\{easy\_to\_see\}) = \frac{0+1}{2} = 0.5$
  $bpa_z(\{easy\_to\_see, \neg easy\_to\_see\}) = 1 - 0.5 = 0.5$

- $bpa_{cond}(\emptyset) = 0$
  $bpa_{cond}(\{easy\_to\_see\}) = \frac{0.95+1}{2} = 0.975$
  $bpa_{cond}(\{easy\_to\_see, \neg easy\_to\_see\}) = 1 - 0.975 = 0.025$

In addition, Dempster's Rule of Combination produces the following result: $bpa_{comb}(\{easy\_to\_see\}) = 0.9875$. Hence, $Bel_{comb}(\{easy\_to\_see\}) = 0.9875$.
As we observe, our belief about *easy to see* is greater than the initial beliefs, which is intuitively correct, since Bel_z increases initial Bel_{cond} (and vice versa).

3.1.2 Entailment method for a set of values

The approaches described above can be applied in situations where we have an interval estimation, i.e. lower and higher value. Besides that, there are case studies where we have more values than a higher and lower ones. Let us consider, for example, a hotel recommendation web site. A list of ratings describes each hotel entry. In that case, we have to take into account all the information given, i.e. all the list values. For this reason, the pessimistic and optimistic approaches are not suitable as they take into account only lower/upper bounds. A generalization of the middle approach considers the *mean value* of the set of values $l_1, l_2, \ldots, l_n, n > 0$, as follows:

$$bpa(\{\phi\}) = \frac{(l_1 + l_2 + \ldots l_n)}{n}$$
$$bpa((\{\phi, \neg\phi\}) = 1 - \frac{(l_1 + l_2 + \ldots l_n)}{n}$$

So, the mean value can constitute a bpa in cases where there exist a set of values rather than an interval. Next, we overview a case study that operates on a set of values.

3.1.3 Dempster-Shafer ontology and Uncertainty - A Hotel Metaclassifier case study

Following, we consider the representation of Dempster-Shafer modules through an ontological framework. Our uncertainty ontology is represented through a list of classes (concepts) (Fig 2):

1. *Possible World*: It represents the set of possible states and is regarded as a frame of discernment
2. *PowerSet*: It is the power set of the set of possible states
3. *Agent*: It represents a semantic web agent
4. *Result*: It represents the combined belief degree, based on Dempster's rule of Combination

5. *HasFor*: It serves as a ”referee” concept for joining an Agent concept with a PowerSet concept

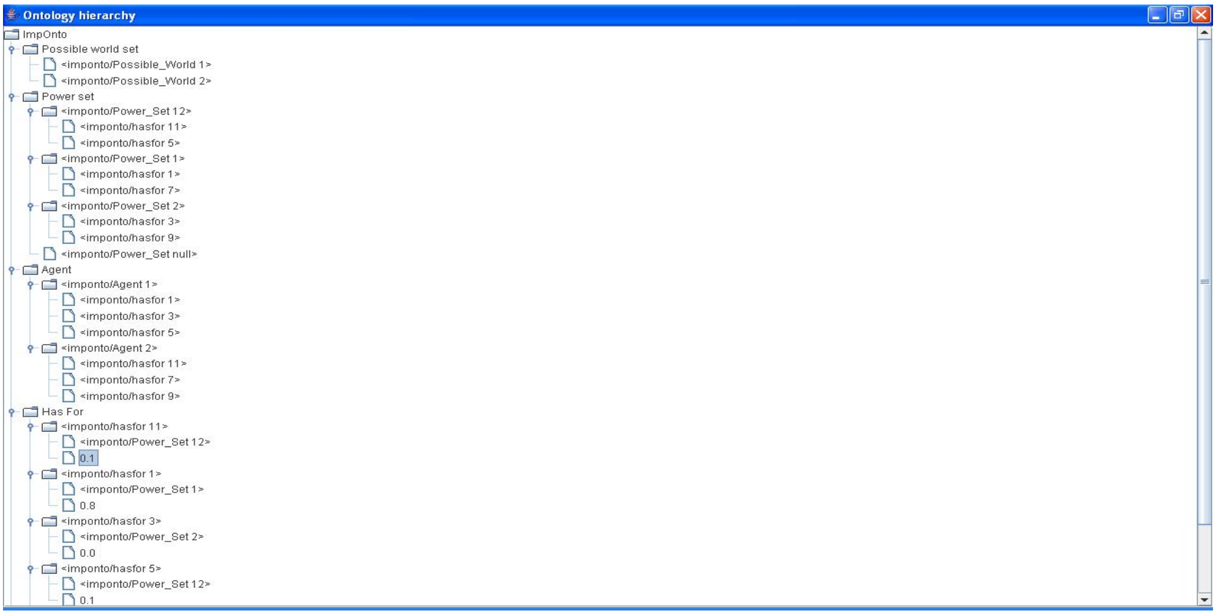

Figure 2: Ontology Hierarchy

In addition, our ontology contains relations among concepts:

1. *hasBel*: It represents the belief function defined over the elements of the power set

2. *hasBpa*: It represents the basic probability assignment that is defined over the elements of the power set

3. *hasPl*: It represents the plausibility function defined over the elements of the power set

4. hasfor: It connects an Agent individual with a PowerSet individual along with a bpa value, through a HasFor individual

Classification of web data is a representative area where uncertainty prevails. Our method can be employed in cases of combination of two (or more) web classifiers, i.e. it can be considered as a metaclassifier approach.

Let us consider two well-known hotel classifiers, for example *www.booking.com* and *www.tripadvisor.co.uk*. Each classifier assigns each hotel a rating value. Our application takes as input the two classifiers considering hotel estimations. There exist a number of estimations for each hotel entry. Our method takes as input these estimations of the two classifiers and returns a belief number about how much proposed the hotel is.

The following steps are implemented:

- The ratings are mapped into a number $[0, 1]$
- A frame of discernment, $\mathcal{W}$, is defined as $\mathcal{W} = \{p, \neg p\}$, where p stands for *proposed hotel*
- Assign bpa values, based on the general middle approach described in 3.1.2
- Perform the combination, based on Dempster's rule

Our approach has been implemented in Protégé [56], for defining and manipulating ontologies. In Chapter 6, we overview in detail applications in hotel recommendation systems.

3.2 Imperfect knowledge representation and Ontologies

The approach described in 3.1, takes into account cases described by uncertainty. Besides uncertainty, vagueness is another factor that describes information. In the previous section, we presented an ontology that classifies a hotel as being (or not) proposed, based on a set of hotel ratings. When dealing with recommendation sites, users usually employ criteria such as:

I'm looking for a low-cost hotel

These criteria are described by vagueness, since they do not have clear-cut meaning. In addition, there are cases where sources of information are described by uncertainty. Let us consider the following statements:

80% of 3-star hotels provide a swimming pool
Hotel A is a 3-star hotel

Based on the two statements, we can say that:

It is 80% likely that hotel A provides a swimming pool

In this section, we present a methodology to reason upon imprecise and uncertain information. Towards this, we built an ontology suitable for representing and processing imperfect information, that it is distributed among a number of peers, a concept employed in Big Data environments. As we address a distributed environment, such as the Web and Semantic Web ones, we consider information distributed in a number of processing able sites (peers). In addition, we present an application of our method using as an example a hotel recommendation task as it is stated above. Since our method also considers Big Data issues, next, we overview some of them.

3.2.1 Semantic Web and Big Data

The *data boom* of the last years, i.e. the exponential growth of the amount of data available, resulted in the evolution of *Big Data* concept. The term *Big Data* is associated with methods for handling large amounts of data. As a concept it was created in the database world but it applies in Web as well, in order to index and query its content. In WWW world, Big Data methods can be used in situations when vast amount of data exists, for example:

- Historical content in web pages
- Social networks
- Sensor information

When dealing with Big Data, we have to master the *3 V's* [76]:

- Volume: this refers to the amount of Web Data
- Velocity: this refers to the high speed that data appears
- Variety: this refers to data heterogeneity

In a Semantic web environment, Big-Data issues affect the semantic web stack. Considering ontologies, the problem is how to store the ontological content and how to reason with it [133]. So, the problem can be divided in the following sub-problems:

1. Big semantic data ontology representation
2. Reasoning methods suitable for large scale data

For this reason, suitable methods for reasoning over big-data environments exist, e.g. [4]. In this situation, a partitioning of information into a set of machines is necessary. Then, the processing can be achieved by each machine independently, using classical logic concepts. The problem here is that data that reside in each machine may not always derive a conclusion. The method used in these situations is the following:

1. The initial set of data is sent randomly to machines
2. The set of data that reside in each machine is combined in order to produce new data (if possible).

Suppose, for example, that we have the following data:

1. parent(Argy,Lucy), which means that Argy is Lucy's parent
2. female(Argy), which means that Argy is female

3. male(Pit), which means that Pit is male

In addition, let us consider a set of machines (or agents) that have a reasoning engine that contains the following rule:
$parent(X,Y) \wedge female(X) \rightarrow mother(X,Y)$.
If the statements 1 and 2 coexist in the same machine, then the conclusion *Argy is the mother of Lucy can be derived*. On the other hand if statements 1 and 3 or statements 2 and 3 coexist, then new knowledge cannot be derived.

The current methods for handling imperfect information, i.e. uncertainty and vagueness as well the reasoning techniques used, cannot be employed in situations where large volume of data is considered. In addition, as it is referred in [29], large scale information is usually characterized by *inconsistency*, *incoherence* and *heterogeneity*.

3.2.2 Fuzzy DLs and Uncertainty

Following, we propose a method for representing imperfect information, taking into account Big Data environments [72]. For representing vagueness, we employ fuzzy DL $\mathcal{SROIQ}(\mathbf{D})$ [17]. Following, we overview syntax and semantics of this fuzzy DL.

Syntax The fuzzy DL $\mathcal{SROIQ}(\mathbf{D})$ employs three alphabets of symbols:

- Concepts
- Roles
- Individuals

The following notations are used:

- Fuzzy Atomic concept - A: It describes concepts not constructed of other concepts, e.g. *Human*.
- Fuzzy Concept - C, D: It is constructed of atomic concepts, using logical constructors, e.g *Father≡Male⊓Parent*, or it can be a fuzzy atomic concept.
- Abstract individuals a,$b \in \Delta^{\mathcal{I}}$
- Concrete individual $v \in \Delta_{\mathbf{D}}$
- Atomic fuzzy role - R_A
- Abstract fuzzy role - R: As it is referred in [116], a role is abstract if *it connects two abstract individuals*, e.g *mother(Argie,Lucy)*. It can be either atomic or complex.

- Simple fuzzy role - S: As it is referred in [116], a role is simple if it is *neither transitive nor it has any transitive sub-role*.
- Concrete role - T: As it is referred in [116], a role is concrete if it connects *an abstract individual with a concrete individual*, e.g. *hasAge(Lucy,"29")*.
- Fuzzy concrete predicate **d**: A fuzzy concrete predicate is defined over $[k_1, k_2] \subseteq Q^+ \cup \{0\}$ through the following membership functions:
  1. Triangular membership function
  2. Trapezoidal membership function
  3. Left shoulder function
  4. Right shoulder function

  For example, a low cost value can be described as a function $lowCost : \mathbb{N} \rightarrow [0, 1]$, as a right membership function, as follows: $lowCost(x) = R(50, 100)$.
- Natural numbers n, m, $n \geq 0$, $m > 0$
- Fuzzy modifier mod: A fuzzy modifier is a function $f_m : [0, 1] \rightarrow [0, 1]$ and is applied to a fuzzy set to change its membership function. For example, $very(x) = x^2$.

A Fuzzy Knowledge Base is defined as a finite set of axioms. These axioms, called *fuzzy axioms*, are DL axioms annotated with a degree of truth in $[0, 1]$.

Semantics The semantics of this fuzzy DL, given as a *fuzzy interpretation* $\mathcal{I}$ with respect to a fuzzy concrete domain $\Delta_{\mathbf{D}}$, is a pair $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the *interpretation domain* that is disjoint with $\Delta_{\mathbf{D}}$ and $\cdot^{\mathcal{I}}$ is the interpretation function. In addition, the semantics, as given in [18], considers the following mappings:

1. *fuzzy abstract individual* α onto an element $\alpha^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
2. *fuzzy concrete individual* d onto an element $v_{\mathbf{d}} \subseteq \Delta_{\mathbf{D}}$
3. *fuzzy concept* C onto a function $C^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow [0, 1]$
4. *fuzzy abstract role* R onto a function $R^{\mathcal{I}}$:$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0, 1]$
5. *fuzzy concrete role* T onto a function $T^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta_{\mathbf{D}} \rightarrow [0, 1]$
6. *n-ary fuzzy concrete domain* **d** onto a function $\mathbf{d}^{\mathcal{I}} : \Delta^n_{\mathbf{D}} \rightarrow [0, 1]$
7. *fuzzy modifier* mod onto a function $f_{mod} : [0, 1] \rightarrow [0, 1]$

The syntax and semantics of fuzzy concepts, roles and axioms are summarized in Tables 5,6 and 7.

Table 5: DL Syntax and Semantics: Concepts

Syntax - Concept(C)	**Semantics -** $C^{\mathcal{I}}(x)$
$\top$	1
$\bot$	0
A	$A^{\mathcal{I}}(x)$
$C \sqcap D$	$C^{\mathcal{I}}(x) \otimes D^{\mathcal{I}}(x)$
$C \sqcup D$	$C^{\mathcal{I}}(x) \oplus D^{\mathcal{I}}(x)$
$\neg C$	$\ominus C^{\mathcal{I}}(x)$
$\forall R.C$	$inf_{y \in \Delta^{\mathcal{I}}}\{R^{I}(x,y) \Rightarrow C^{\mathcal{I}}(y)\}$
$\exists R.C$	$sup_{y \in \Delta^{\mathcal{I}}}\{R^{I}(x,y) \otimes C^{\mathcal{I}}(y)\}$
$\forall T.\mathbf{d}$	$inf_{v \in \Delta_{\mathbf{D}}}\{T^{\mathcal{I}}(x,v) \Rightarrow \mathbf{d}_{\mathbf{D}}(v)\}$
$\exists T.\mathbf{d}$	$sup_{v \in \Delta_{\mathbf{D}}}\{T^{\mathcal{I}}(x,v) \otimes \mathbf{d}_{\mathbf{D}}(v)\}$
$\{\alpha_1/o_1, ..., \alpha_n/o_n\}$	$sup\{i \mid x = o_i^{\mathcal{I}}\}\alpha_i$
$\geq mS.C$	$sup_{y_1,...,y_m \in \Delta^{\mathcal{I}}}(min_{i=1}^{m}\{S^{\mathcal{I}}(x,y_i)$ $\otimes C^{\mathcal{I}} y_i)\}) \otimes ((\otimes)_{1 \leq j<k \leq m}\{y_j \neq y_k\})$
$\leq nS.C$	$inf_{y_1,...,y_{n+1} \in \Delta^{\mathcal{I}}}(min_{i=1}^{m}\{S^{\mathcal{I}}(x,y_i)$ $\otimes C^{\mathcal{I}}(y_i)\}) \Rightarrow ((\oplus)_{1 \leq j<k \leq n+q}\{y_j = y_k\})$
$\geq mT.\mathbf{d}$	$sup_{v_1,...,v_m \in \Delta_{\mathbf{D}}}(min_{i=1}^{m}\{T^{I}(x,v_i)$ $\otimes \mathbf{d}^{\mathcal{I}}(v_i)\}) \otimes ((\otimes)_{j<k}\{v_j \neq vd_k\})$
$\leq nT.\mathbf{d}$	$inf_{v_1,...,v_{n+1} \in \Delta_{\mathbf{D}}}(min_{i=1}^{m}\{T^{I}(x,v_i)$ $\otimes \mathbf{d}^{\mathcal{I}}(v_i)\}) \Rightarrow ((\oplus)_{j<k}\{v_j = v_k\})$
$\exists S.Self$	$S^{\mathcal{I}}(x,x)$
$mod(C)$	$f_{mod}(C^{\mathcal{I}}(x))$
$[C \geq \alpha]$	1 if $C^{\mathcal{I}}(x) \geq \alpha$, 0 otherwise
$[C \leq \beta]$	1 if $C^{\mathcal{I}}(x) \leq \beta$, 0 otherwise
$\alpha_1 C_1 + ... + \alpha_k C_k$	$\alpha_1 C_1^{\mathcal{I}}(x) + ... + \alpha_k C_k^{I}(x)$
$C \rightarrow D$	$C^{\mathcal{I}}(x) \Rightarrow D^{I}(x)$

Table 6: DL Syntax and Semantics: Roles

Syntax - Role (R)	**Semantics -** $R^{I}(x,y)$
R_A	$R_{A^{\mathcal{I}}(x,y)}$
U	1
R^{-}	$R^{\mathcal{I}}(y,x)$
$mod(R)$	$f_{mod}(R^{\mathcal{I}}(x,y)$
$[R \geq \alpha]$	1 if $R^{\mathcal{I}}(x,y) \geq \alpha$, 0 otherwise
T	$T^{\mathcal{I}}(x,v)$

Table 7: DL Syntax and Semantics: Assertions

Syntax - axiom τ	**Semantics -** τ^I
$a : C$	$C^{\mathcal{I}}(a^{\mathcal{I}})$
$(a,b) : R$	$R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}})$
$(a,b) : \neg\, R$	$\ominus\, R^{I}(a^{\mathcal{I}}, b^{\mathcal{I}})$
$(a,v) : T$	$T^{\mathcal{I}}(a^{\mathcal{I}}, v^{\mathbf{D}})$
$(a,v) : \neg\, T$	$\ominus\, T^{\mathcal{I}}(a^{I}, v_{\mathbf{D}})$
$C \sqsubseteq\ D$	$inf_{x\in\Delta^{\mathcal{I}}} C^{\mathcal{I}}(x) \Rightarrow\ D^{\mathcal{I}}(x)$
$R_1 ... R_m \sqsubseteq\ R$	$inf_{x_1,x_{n+1}} sup_{x_2,x_n}(R_1^{\mathcal{I}}(x1,x2)\otimes$ $... \otimes\ R_n^{\mathcal{I}}(x_n, x_{n+1})) \Rightarrow\ R^{\mathcal{I}}(x_1, x_{n+1})$ where $x_1, x_{n+1} \in\ \Delta^{\mathcal{I}}$
$T_1 \sqsubseteq\ T_2$	$inf_{x\in\Delta^{\mathcal{I}}, v\in\Delta_{\mathbf{D}}} T_1^{\mathcal{I}}(x,v) \Rightarrow\ T_2^{\mathcal{I}}(x,v)$

Fuzzy inverse concrete predicate In our approach, we introduce the notion of *fuzzy inverse concrete predicate*, denoted as $invp$. It operates in a way similar to a fuzzy modifier, i.e. it changes the membership function of a fuzzy set. More precisely, it is a function $f_{invp} : [0,1] \to [0,1]$ and denotes the inverse notion of a fuzzy concrete predicate. It is defined as:

$$invp(x) = 1 - f(x), x \in \Delta_{\mathbf{D}}$$

where $f(x)$ can be one of triangular, left, right, triangular and trapezoidal membership functions. Based on the definition of $invp$, we have the following equivalences:

$$invp(x) = R(k1,k2), \text{ \textit{if} } f(x) = L(k1,k2)$$
$$invp(x) = L(k1,k2), \text{ \textit{if} } f(x) = R(k1,k2)$$

In addition, fuzzy modifiers can also be applied to fuzzy inverse predicates as well.

Example Let us consider the concept $Hotel \sqcap \exists hasCost.L(50,100)$ that denotes the set of low cost hotels. The cost is described by the left membership function $L(50,100)$. Then, $Hotel \sqcap invp(\exists hasCost.L(50,100))$ denotes the set of high cost hotels.

For representing fuzzy statements in an ontology environment, we map the fuzzy DL axioms into fuzzy facts. As fuzzy extensions of OWL bare a close connection to the fuzzy DL $\mathcal{SROIQ}(\mathbf{D})$, we employ the DL notation instead of the OWL one, a method suggested in [19].

Apart from vagueness, uncertainty is another issue we should take into account. Fuzzy DL axioms allow for representation of statements α *is true with a factor* $f_f, f_f \in [0,1]$. In cases of uncertainty, we allow for statements of the form α *is true with a factor* $f_u, f_u \in [0,1]$. In both cases, f_u, f_f represent degrees of factor. More precisely, f_u represents an uncertainty factor whereas f_f represents a fuzzy factor. In our approach, we consider these factors under the common term *imperfection factor*. In order to do this, we consider the equivalence between fuzzy logic and probabilities, as described in 2.10. This equivalence also serves as a way to use probabilistic knowledge bases in order to represent both uncertain and vague concepts.

For representing statements annotated with an imperfection factor, we consider a probabilistic knowledge base as described in 2.7. Then, we consider the following:

- Concept *Generic Rule*: It represents generic knowledge that always hold. It has the form $(b|a)[l,l]$, where a, b are events and $l \in [0,1]$.
- Concept *Event*: It represents evidence. It has the form $(e|\top)[l,l]$, where e is an event and $l \in [0,1]$.

In general, probabilistic knowledge bases annotate constraints with an interval $[l,u]$. In our method, we consider a single value, denoted as l. The reason for this is that fyzzy DL axioms are described with a single fuzzy factor. Since we consider a single value rather than an interval, generic rules and events have the following form:

- **Rule:(**$b \mid \quad a[f_r]$**)**, meaning that if a holds then b generally holds with imperfection factor f_r, where $f_r \in [0,1]$.
- **Event:(**$e \mid \top[f_e]$**)**, meaning that the event is true with imperfection factor f_e, where $f_e \in [0,1]$.
  The *imperfection factor* can be either an uncertainty or fuzzy factor.

The scope of defining rules and events is to detect new knowledge. More precisely, let us consider the following:

$$a \mid \top[f_e]$$

Then, we aim at defining an imperfection factor for:

$$b \mid \top[f_c], \text{ where } f_c \in [0,1]$$

In order to do this, we employ the propagation formula, described in 2.6.2. More precisely, we consider a frame of discernment $\Theta_A = \{a, a'\}$, where a' stands for $\neg\, a$. In addition, we consider a frame of discernment $\Theta_B = \{b, b'\}$, where b' stands for $\neg\, b$. Then, the propagation formula for b is defined as:

$$m'(b) = f_e f_r$$

This propagation formula actually constitutes the imperfection factor for b, i.e f_c.

3.2.3 Information Distribution

The Big-Data era introduces the distributed reasoning approach. In an environment described by uncertainty, the problem is how to extend the probabilistic knowledge base into a *distributed probabilistic knowledge base*. In order to address this complexity, the set of rules as well as evidence could be distributed in a set of peers. In addition, the selection of the initial set of data automatically results in uncertainty as a result of incomplete information. For example, in order to perform a *recommendation* about a *three star hotel*, we can use all the services provided (pool, restaurant, etc.) or, instead, we can use only the *partial information* ”three star” along with a list of services that three star hotels *usually* provide and draw a conclusion with some uncertainty. This step may increase the imperfection factor in information, as the step is characterized by uncertainty and vagueness. So, our problem can be summarized as follows:

1. How the choice of the subset of data is made?
2. How the distribution of data is made?

The main problem with step 2 is that the coexistence of some events with some rules will never come up to a conclusion, as they may be *unrelated*. For example, if we have generic knowledge about five star hotels, e.g. *All five star hotels have a restaurant*, then we are interested only in events that have references to 5 star hotels. Thus, the distribution of rules and events among peers should be done in a way that statements that are *correlated* should (if possible) coexist in the same peer.

If an entity is associated with another entity it is stated that there exist a *correlation*. As we have previously stated, correlated entities should be processed by the same peer. For this reason, we annotate generic rules and events with a *key value* concept, as follows:

1. $b \mid\ a[f_1], k$, meaning that if a holds then b generally holds with imperfection factor f_1, where $f_1 \in [0, 1]$ and key value $k, k \in \mathbb{N}$.
2. $a \mid \top[f_2]$, where $f_2 \in [0, 1]$, meaning a is true with imperfection factor f_2 and key value $k, k \in \mathbb{N}$.

The key value k serves as a peer descriptor, i.e it is unique number that serves as the peer identity.

Having defined the key value for each entity, the correlation for two entities is defined as follows:

If $e1$ and $e2$ are entities, then they are correlated iff they have the same key value

So, the first step in our method is to distribute the events and rules among peers according to their *key value*. For example, if we have rules about hotels that describe different hotel categories (e.g ”all five star hotel have a swimming pool”), then the key here is the hotel rating, and hence hotels can be distributed into peers according to their category. The algorithm is the following:

Define number of peers equal to the number of key values
for *i=1 to number of entities* **do**
| *Associate entity with peer having the same number as entity's key value;*
end
for *i=1 to number of peers* **do**
| *Find a conclusion considering events and rules in those peer;*
end

Algorithm 1: Peer distribution

After peer distribution, each peer contains a number of events and rules. The next step is the reasoning process. This process is executed *independently* in each peer, without taking into consideration the other peers. This permits peer processing to be done in parallel. The problem here is that *conflicts* may arise in situations when two rules result in the same conclusion with different *imperfection factors*. For example:

1. *Three star hotels have a swimming pool with factor 0.5*
2. *A resort hotel always has a swimming pool*

If we have a resort hotel in our data, then the second rule should be applied instead of the first. In general, those conflicts may arise in situations where there exist *exceptional* classes (as resort hotel) and *exceptional rules*, and hence the *exceptional class* is preferred as opposed to the *general class*. The reasoning algorithm, i.e the last step in the first algorithm, in each peer is the following:

Assign the priority 2 *to each exceptional rule;*
Assign the priority 1 *to rules without exceptions;*
Run the rule engine from higher to lower priority;
Retract the facts when the rule "fires";

Algorithm 2: Peer reasoning algorithm

The fact retraction serves as a way to stop rule firing in lower priorities, so if the rule that fires has priority *2*, then by retracting the fact that caused firing, the rule with priority *1* won't fire.

For representing the concepts above, we consider an "imperfect" ontology, that employs the following concepts:

- *Knowledge*: It represents the entities information, i.e. rules and events
  - *Rule*: It represents all the rules, i.e. the generic knowledge individuals

  - *Event*: It represents all the events individuals

- *Peer*: It represents all the peers individuals
- *Condition*: It is the condition part of the rule
- *Conclusion*: It is the conclusion part of the rule

In addition, our ontology employs the following object properties:

- *if* : It is an object property that connects a Rule individual to a Condition individual
- *then*: It is an object property that connects a Rule individual to a Conclusion individual

Finally, our ontology employs the following data properties:

- *hasFactor*: It represents the imperfection (regarding information) for each entity, i.e. rules and events
- *hasPeer*: It represents the peer number associated with each entity, or in other words the key value

In order to perform reasoning our ontology needs to be provided with the key value, that determines the number of peers, as well as the set of generic rules and events. Let us consider a hotel recommendation site, along with the query *I'm looking for a hotel around, not expensive, having a gym*. Our goal is to propose a set of hotels, based on this query. In addition, let us consider that the information available is hotel's rating, i.e 3-star, 4-star, etc. We have ignorance regarding cost or services,i.e. in our problem uncertainty is a result of incomplete information. In addition, the user provides us with vague information about cost, since *expensive* does not have a clear meaning. For example, in Greece *an expensive* hotel is around 100 Euros. In our example, the fuzzy concrete predicate *expensive* is defined as $expensive(x) = L(70, 130)$, where L is the $Left - function$. Following, if we want to represent *not expensive*, we define the $invp(expensive) = R(70, 130)$ where R stands for $Right - function$. We also have a set of generic knowledge about hotels, for example, *4-star hotels have spa services with factor 0.40*. Here, we have to notice that exceptions may arise in some situations, for example the hotel "Habtoor Grand Beach Resort & Spa" definitely has a spa centre since the word Spa exists in its name. In this case, the rule *4-star hotels have spa services with factor 0.40* cannot and shouldn't be applied.

After data insertion, i.e. the set of hotels as well as the set of statistics considering the star category, our ontology has a set of event individuals and rule individuals. The next step is the distribution among peers. The key value here is the rating of the hotels, i.e. their stars. So, we have 6 peer individuals, one for each rating. All hotels that have the same rating belong to the same peer. The total number of events and rules that exist in all peers, are only those that we are interested in. For example, if we look only for 2 and 3-star hotels that have some characteristics, then rules having a condition considering one, four or five star are omitted. This leads into the reduction of the number of peers as well. In our example, we look for a hotel that costs around 50 to 100 euro and has spa and gym services. The only information we have is some statistics considering the hotel's star category, e.g. *80 per cent of 3-star hotels cost around 50 euros and provide restaurant services*. After peer distribution, we have a set of events and rules, as well as their peer number. The final step in our method is the reasoning process. Each peer proceeds into reasoning independently, without taking into consideration rules and events that exist in other peers.

In this approach, we have studied situations where uncertainty and vagueness coexist, by considering an ”imperfect” ontology and employing our method in a hotel recommendation system. Our approach has been implemented in a Protégé plugin for Java language. In Chapter 6, we will present a more detailed case-study of hotel recommendation systems along with evaluation results.

3.3 Fuzzy knowledge representation and Ontologies - A Matchmaking case study

Matchmaking problems [67] can be considered as a case study of Semantic Web, as uncertainty and vagueness often describe them. A matchmaking method always takes into account a set of constraints. A typical matchmaking problem consists of two groups, denoted as ”sellers” and ”buyers”. The point is to create pairs of sellers and buyers with an optimal way, for example by their maximum similarity. A common matchmaking application can be found in e-Commerce environments. The critical issue in a matchmaking task is knowledge representation. Another issue considers sellers' and buyers' constraints. In general, constraints can be divided into hard and soft constraints, based on the flexibility regarding the fulfilment of the constraint.

Let's take the example of a job recruitment problem. In order to perform a recruitment, some criteria should be provided from both the recruiter and the seeker. Let us consider the following example:

- A seeker that searches for a job with salary around $3,000$
- A recruiter that can pay around $2,000$ and also requires applicants to have enough experience

As we see, in our problem, the constraints are not explicitly defined, in other words, a notion of vagueness describe them. If we consider a semantic web agent that automates the whole process, then the steps would be the following:

- The agent should select a set of sources relevant to the query
- The agent should execute the query and classify the results, by ordering them from most to less proposed

The selection of sources results in information incompleteness, which is described by uncertainty theory. In addition, the theory for representing vagueness is Fuzzy Logic. Moreover, the Semantic Web employs ontologies for representing knowledge. So, an extension of OWL with fuzziness and uncertainty, or an ontology that functions with "imperfect" information, suitable for matchmaking environments, is necessary in order to represent meaningful and machine processable web information.

Coming back to our introductory example, in a DL formalism, the constraints should have the following formalism:

$$Seeker \equiv \exists\ hasSalary.Around\ 3.000$$
$$Recruiter \equiv \exists\ hasSalary.Around\ 2.000 \sqcap\ hasExperience.EnoughExperience$$

In the rest of this section, we describe a matchmaking case study, outlined in [68], that operates with fuzzy constraints. In addition, Dempster's rule of Combination is employed for combining different matchmaking degrees.

3.3.1 Fuzzy Dempster-Shafer Ontology

For representing a fuzzy matchmaking case study, first we consider the fuzzy DL $\mathcal{SHOIN}(\mathbf{D})$ as described in [121]. The crisp version, $\mathcal{SHOIN}(\mathbf{D})$, allows for reasoning with concrete datatypes, called concrete domains. A concrete domain, D, is defined as a pair $< \Delta_D, \Phi_D >$, where Δ_D is an interpretation domain and Φ_D is the set of concrete domain predicates d with arity $n, n \in \mathbb{N}$ and interpretation $d^D \subseteq \Delta_D^n$. As an example, we may consider $Hotel \sqcap \exists\, cost. \geq\ 100$, which denotes hotels with a cost greater than 100. The semantics of the crisp DL $\mathcal{SHOIN}(\mathbf{D})$ is defined through an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$.

The fuzzy DL $\mathcal{SHOIN}(\mathbf{D})$ allows for reasoning with concrete datatypes and concrete domains based on fuzzy sets. A core concept in this approach is the concrete fuzzy domain, which is defined as a pair <Δ_D, Φ_D>, where Δ_D is an interpretation domain and Φ_D is a set of concrete fuzzy predicates d with arity $n, n \in \mathbb{N}$ and interpretation $d^D : \Delta_D^n \rightarrow [0, 1]$. These predicates are used in order to represent concepts like *Young Person*, *Fast Car* etc. For example, $Hotel \sqcap \exists\, cost.Low$, denotes *low cost hotels*.

Regarding semantics, the core concept is that the fuzzy extension allows for the interpretation of concepts and roles as fuzzy subsets of the interpretation domain. This allows for the annotation of axioms with a degree of truth in $[0, 1]$. More precisely, a fuzzy interpretation is defined as a pair $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is the interpretation domain and $\cdot^{\mathcal{I}}$ is an interpretation function. This interpretation performs the following mappings, considering a fuzzy Concept, C, a fuzzy Role, R and a concrete fuzzy predicate, d:

$$C \subseteq \Delta^{\mathcal{I}}, \text{with} \mu_C(x) : \Delta^{\mathcal{I}} \rightarrow [0, 1], x \in \Delta^{\mathcal{I}}$$

$$R \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}, \text{with} \mu_R(x, y) : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0, 1], x, y \in \Delta^{\mathcal{I}}$$

$$d^D : \Delta_D^n \rightarrow [0, 1]$$

In our approach, we introduce the notion of the fuzzy focal domain, denoted as

$$W \; : \; < \; \Delta_W, \Phi_W >$$

,where Δ_W is a frame of discernment and Φ_W is a set of fuzzy focal elements $A_i, i = 1, 2, \ldots, n$, with basic probability assignment $m(A_i) : 2^{\Delta_W} \rightarrow [0, 1]$. The set Δ_W represents mutually exclusive events, whereas the elements of Φ_W are fuzzy subsets over $\mathcal{W}$. For example, if we want to represent the fuzzy datatype $lowCost$, then $\Delta_W = \mathbb{N}$ and Φ_W are all the fuzzy subsets of Δ_W.

For defining Belief and Plausibility degrees on fuzzy subsets of $B \subseteq \Delta_W$, with membership function μ_B we employ the formulas described in [139]. More precisely:

$$Bel(B) = \sum_A m(A) \sum_{a_i} [a_i - a_{i-1}] \times \; inf_{x \in \; A_{a_i}} \mu_B(x)$$

$$Pl(B) = \sum_A m(A) \sum_{a_i} [a_i - a_{i-1}] \times \; sup_{x \in \; A_{a_i}} \mu_B(x)$$

where $A \in \Phi_W$, $a_i, i = 1, \ldots, n$ are α-level sets of A and $x \in \Delta_W$.

3.3.2 Fuzzy Dempster-Shafer Ontology and Matchmaking

As we have stated in the introduction of this section, the matchmaking job recruitment process requires two entities, job seeker and job advertisement. Thus, a selection of job advertisements is necessary, in order to perform the matchmaking process. More precisely, for the job recruitment problem, the method considers the following components:

- *Job Seeker*, who searches for a specific position, based on a set of criteria, like salary, working hours etc.
- *Job Advertisement*, who publishes a certain job, stating the necessary and optional qualifications
- *Fuzzy Ontology Repository*, which represents all the knowledge, about the problem's domain
- *Fuzzy Matchmaking Engine*, which proceeds in the matchmaking process based on a set of vague criteria, related to the problem

The domain of knowledge is represented through a *job search ontology*. The following concepts are defined:

$JobSeeker \equiv \; Person \sqcap_{\geq 1} hasCV.CV \sqcap \; SeekerRequirements \sqcap \; SeekerPreferences$
$JobAdvertisement \equiv \; Person \sqcap \exists_{=1} hasJobOffer.JobOffer \sqcap \; SkillRequirements$
$\sqcap \; SkillPreferences$

In addition, the concept $JobOffer$ has the following form:

$$JobOffer \equiv \exists_{=1} hasSalary.Integer \sqcap \exists_{\geq 1} hasWorkingHours.WorkingHours$$

A CV concept has two subclasses, $Degree$ and $Experience$, defined as follows:

$$Degree \equiv CV \sqcap \exists_{=1} hasDegreeRate.Double$$
$$Experience \equiv CV \sqcap \exists_{=1} hasCompany.Company$$
$$\sqcap \exists_{=1} hasJobType.JobType \sqcap \exists_{=1} yearsOfExperience.Integer$$

The $JobType$ and $WorkingHours$ are enumerated classes of the following form:

$$JobType \equiv ComputerProgrammer, DatabaseAdministrator, DataAnalyst$$
$$WorkingHours \equiv FullTime, PartTime$$

Seeker and Offer requirements are defined based on a set of membership functions. In our example, for the Salary constraint, these are stated as follows:

$$\mu_{Seeker_{Salary}}(x) = \left\{ \begin{array}{lr} 0, & \text{for } 0 \leq x \leq 2000 \\ \frac{x-2000}{1000}, & \text{for } 2000 \leq x \leq 3000 \\ 1, & \text{for } 3000 \leq x \end{array} \right\}$$

$$\mu_{Offer_{Salary}}(x) = \left\{ \begin{array}{lr} 1, & \text{for } 0 \leq x \leq 2000 \\ \frac{3000-x}{1000}, & \text{for } 2000 \leq x \leq 3000 \\ 0, & \text{for } 3000 \leq x \end{array} \right\}$$

where $x \in \mathbb{N}$.

In addition, for the Experience constraint, we have:

$$\mu_{Offer_{Experience}}(x) = \left\{ \begin{array}{lr} 0, & \text{for } 0 \leq x \leq 5 \\ \frac{x-5}{5}, & \text{for } 5 \leq x \leq 10 \\ 1, & \text{for } 10 \leq x \end{array} \right\}$$

where $x \in \mathbb{N}$.

Each Seeker and Offer preference is assigned a membership degree, based on the predefined membership function. For defining the total membership degree of Seeker's (Offer's) preferences, we employ any fuzzy logic defined in Table (4). This membership degree represents the *degree of satisfaction*, based on the input data values, i.e salary and experience. These are denoted as m_{Seeker} and m_{Offer} for Seeker and Offer, respectively.

Example Let us consider the membership functions:

$$\mu_{Seeker_{Salary}}, \mu_{Offer_{Salary}} \text{ and } \mu_{Offer_{Experience}},$$

as previously defined. In addition, we consider the input values:

$$x_{salary} = 2500 \textit{ and } x_{experience} = 8$$

According to the membership functions, we get the following assignments:

$$\mu_{Seeker_{Salary}}(x_{salary}) = 0.5$$
$$\mu_{Offer_{Salary}}(x_{salary}) = 0.5$$
$$\mu_{Offer_{Experience}}(x_{experience}) = 0.6$$

In addition, according to Zadeh's Fuzzy Logic (Table 4), we have the following degrees of satisfaction for $Seeker$ and $Offer$:

$$m_{Seeker} = \mu_{Seeker_{Salary}}(x_{salary}) = 0.5$$
$$m_{Offer} = \mu_{Offer_{Salary}}(x_{salary}) \otimes \mu_{Offer_{Experience}}(x_{experience}) = min\{0.5, 0.6\} = 0.5$$

Having defined the representation scheme of Seeker and Offer concepts, in the following, we present the matching process, i.e. the reasoning method.

3.3.3 Matching Process

The Matching Process is the reasoning method of our system. The method attempts to match Seeker's Preferences along with Offer's Preferences. As it is not always possible to match exactly Seeker's and Offer's preferences, we try to find the best agreement between them.

Formally, this is defined as follows:

$$BestAgreement \equiv \ SeekerPreferences \sqcap \ OfferPreferences$$

The interpretation of the fuzzy intersection $SeekerPreferences \sqcap \ OfferPreferences$ can be resolved using Dempster's rule of Combination. The reason for employing Dempster-Shafer framework and Dempster's rule of Combination is because this theory is ideal for preference fusion situations. Seeker Preferences as well as Offer Preferences are associated with a degree of satisfaction, as previously defined.

In order to to compute the combined degree of $BestAgreement$, we consider Seeker and Offer as independent sources of information. In addition, a frame of discernment, $\mathcal{W}$, is defined as $\mathcal{W} = \{\alpha, \neg\alpha\}$, where α stands for "has degree of satisfaction". Following this convention, the basic probability assignment is defined as follows:

$$
\begin{aligned}
&bpa_{Seeker}(\emptyset) = 0\\
&bpa_{Seeker}(\{\alpha\}) = m_{Seeker}\\
&bpa_{Seeker}(\{\neg\alpha\}) = 0\\
&bpa_{Seeker}(\{\alpha, \neg\alpha\}) = 1 - m_{Seeker}\\
&bpa_{Offer}(\emptyset) = 0\\
&bpa_{Offer}(\{\alpha\}) = m_{Offer}\\
&bpa_{Offer}(\{\neg\alpha\}) = 0\\
&bpa_{Offer}(\{\alpha, \neg\alpha\}) = 1 - m_{Offer}
\end{aligned}
$$

Finally, we define the combined constraint degree through Dempster's rule of Combination:

$$
\begin{aligned}
&combined_{Seeker,Offer} =\\
&(\{\alpha\}) = m_{Seeker} \times Offer + m_{Seeker} \times (1 - m_{Offer}) + m_{Offer} \times (1 - m_{Seeker})
\end{aligned}
$$

Our approach has been implemented in a Protégé tool along with the rules plugin provided. In Chapter 6, we will present a real-world case study of a matchmaking approach.

4. TOWARDS THE DEFINITION OF THE FRAMEWORK

In this Chapter, we propose a framework that employs Dempster-Shafer theory in a Description Logic Knowledge Base environment. We name our model a *Dempster-Shafer DL Knowledge Base*.

As we have stated in the introduction, while developing Semantic Web applications, we often come across information incompleteness issues. As an example, let us consider a data source that contains information about *hotels*. We assume each hotel h to be assigned an interval cost per night rather than a crisp value, e.g:

$$h : [50 - 150]$$

In this case, if we want to make a reservation, we do not know exactly what the cost is but we know a lower-upper bound of the cost value. Moreover, consider the following query:

I'm looking for a hotel with cost no greater than 100

In a crisp logic framework, where each hotel has a unique value cost, the query could be answered with a yes/no statement. In our case, where we have to deal with interval value form, a yes/no statement cannot fully answer this query. The introduction of a *degree* notion seems to be more suitable to describe this kind of information.

More precisely, in a Description Logics environment, if we consider a concept *DesiredHotel*, defined as:

$$DesiredHotel \equiv \; Hotel \sqcap \exists cost. \leq_{100}$$

then, the answer to our query is to decide whether a hotel individual is a member of the Class *DesiredHotel*.

Information incompleteness can be classified as an uncertainty problem. Other uncertainty problems consider information randomness and data inconsistency [37]. Dempster-Shafer theory, along with Dempster's rule of Combination [111], is a framework for dealing with information incompleteness, allowing integration of information from different independent sources. In our approach, we propose an adaptation of Dempster-Shafer theory in a logic context.

In this chapter, we define an extension of crisp Knowledge Bases with Dempster-Shafer modules. The concept of *Dempster-Shafer DL Knowledge Base* is introduced and it is served as a way to tackle information incompleteness. Dempster-Shafer Theory is more well-suited in modelling beliefs regarding the truthness of an event. Our method is an extension of the crisp DL $\mathcal{ALC}$. In our framework, we consider crisp DL axioms annotated with Dempster-Shafer belief and plausibility degree conditions.

4.1 Dempster-Shafer Theory and Logical Extensions

A Logical representation of Dempster-Shafer theory has been examined extensively in literature [101, 100, 108, 98, 155]. The key-point in all representations is the annotation of statements or formulas with belief and plausibility measures.

In [108], the Dempster-Shafer logical model is defined as a set of *bf-formulas*, denoted as:

$$F : [a, b], \;\; a, b \in [0, 1]$$

where F is a classical first-order sentence and $[a, b]$ constitutes a Belief-Plausibility interval, in a Dempster-Shafer framework. The logic defined, denoted as *BFL*, constitutes a method for representing uncertain knowledge. More precisely, a belief measure is assigned to classical first-order logic formulas. In the formula above, a, b constitute belief and plausibility degrees respectively, i.e a *BFL* formula is classical FOL formula, annotated with an interval constraint. In addition, the concept of *hyper-interpretation* is introduced, as a set of interpretations. Then, the entailment relation is extended for hyper-interpretations. If an $\mathcal{I}$ is a hyper-interpretation, then a *bf-interpretation*, $\mathcal{M}$, is defined as:

$$\mathcal{M} : 2^{\mathcal{I}} \rightarrow [0, 1]$$

Such an interpretation should be aligned with the following:

$$\begin{aligned}&(1) \;\; Bel_{\mathcal{M}}(2^{\mathcal{I}}) = 1\\&(2) \;\; if \quad X \cup Y \neq 2^{\mathcal{I}}, then \quad Bel_{\mathcal{M}}(X \cap Y) \geq Bel_{\mathcal{M}}(X) \times Bel_{\mathcal{M}}(Y)\end{aligned}$$

The notions of validity, satisfaction and entailment are defined over BFL formulas.

Another approach is defined in [101, 100]. The notion of provability is employed, by defining a set of focal propositions, Θ, along with its corresponding literals x. A set of clauses X denotes provability relations considering $2^{\times}$.

4.2 Dempster-Shafer Description Logics

An approach which introduces the concept of the *Dempster-Shafer DL Knowledge Base*, based on the logic defined in [108], is defined in [69]. This approach is denoted as $\mathcal{DS} - \mathcal{ALC}$, as an extension to crisp $\mathcal{ALC}$. In this approach, we extend classical DL axioms with Belief degree conditions and Plausibility degree conditions. Then, we interpret these axioms to hold with a Belief degree lower bound or Plausibility degree lower bound.

Following, we define our $\mathcal{DS} - \mathcal{ALC}$ syntax and semantics.

4.3 The Description Logic $\mathcal{ALC}$

In this Section, we overview the DL $\mathcal{ALC}$. $\mathcal{ALC}$ is considered the basic DL language. Its syntax uses the following sets: N_C (the set of concept names), N_R (the set of role names) and N_I (the set of individuals). In order to build complex rules, we apply a set of syntax rules. More precisely, $\mathcal{ALC}$ *concepts* are the following:

- $\top, \bot, A$, where A is a primitive concept
- If C, D are $\mathcal{ALC}$ *concepts*, then $C \sqcap D$, $C \sqcup D$, $\neg C$, $\forall r.C$ and $\exists r.C$, where r is a *DL Role*, are $\mathcal{ALC}$ *concepts*.

An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ performs the following mapping:

$$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}, \bot^{\mathcal{I}} = \emptyset, C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}, r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$$
$$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}, (C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$$
$$(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \backslash C^{\mathcal{I}}$$
$$(\forall r.C)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} : \forall d', (d, d') \in r^{\mathcal{I}} \textit{ implies } d' \in C^{\mathcal{I}}\}$$
$$(\exists r.C)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} : \exists d', (d, d') \in r^{\mathcal{I}} \textit{ and } d' \in C^{\mathcal{I}}\}$$

In addition, $\mathcal{ALC}$ considers two kinds of assertions for an individual α: $C(\alpha)$, meaning that α is an instance of C (*concept assertion*) and $r(\alpha, \beta)$, meaning that there is a relation r between α, β (*role assertion*). A set of concept assertions $\{C(\alpha_1), \ldots, C(\alpha_n)\}$ is *satisfied* in an interpretation $\mathcal{I}$, iff $\alpha_i^{\mathcal{I}} \in C^{\mathcal{I}}, i = 1, \ldots, n$. A set of role assertions $\{r(\alpha_1, \beta_1), \ldots, r(\alpha_n, \beta_n)\}$ is *satisfied* in an interpretation $\mathcal{I}$, iff $(\alpha_i^{\mathcal{I}}, \beta_i^{\mathcal{I}}) \in r^{\mathcal{I}}, i = 1, \ldots, n$.

4.3.1 Syntax of $\mathcal{DS} - \mathcal{ALC}$

A *Dempster-Shafer DL knowledge base* is described by the following:

- A set $\Phi = \{p_1, p_2, \ldots, p_n\}$, where $p_i, i = 1, \ldots, n$ is a basic crisp DL $\mathcal{ALC}$ assertional axiom.
- Any assertion ϕ is an atomic assertion, or a boolean combination of assertions.
- A set of constraints:
  - *Belief Constraints*: They have the form $\phi \quad B \geq \alpha$, and interpreted as ϕ *is true with Belief degree at least* α.
  - *Plausibility Constraints*: They have the form $\phi \quad P \geq \alpha$, and interpreted as ϕ *is true with Plausibility degree at least* α.

Definition 1. A Dempster-Shafer DL Knowledge Base is defined as a set of Belief Constraints $\mathcal{B}$ and a set of Plausibility Constraints $\mathcal{P}$, as:

$$\mathcal{KB} = (\mathcal{B}, \mathcal{P})$$

4.3.2 Semantics of $\mathcal{DS} - \mathcal{ALC}$

Before defining the semantics of our framework, we introduce the concept of a *possible world* I to be a subset of the set of basic crisp DL assertions Φ. In that sense, a possible world I specifies the set of assertions that are *true* in that world. We denote as $\mathcal{W}$ the set of possible worlds I, i.e $\mathcal{W} = 2^{\Phi}$. Since Φ is finite, $\mathcal{W}$ is also finite. Given a crisp DL Knowledge Base, $\mathcal{KB}_{crisp}$, and a possible world I, the satisfaction of $\mathcal{KB}_{crisp}$ is defined as:

Definition 2. A possible world I *satisfies* (or it is a *model* of) $\mathcal{KB}_{crisp}$ iff:

$$\{p \mid p \in I\} \cup \mathcal{KB}_{crisp}$$

is satisfiable.

Next, we will prove that the satisfaction (entailment) of a $\mathcal{KB}_{crisp}$ is a necessary and sufficient condition for the existence of a model $\mathcal{I}$ of this Knowledge Base. Our proof is based on the one defined in [83], adapted in our DL.

Proposition 1. Let Φ be a finite set of DL assertions and let $\mathcal{KB}_{crisp}$ be a crisp $\mathcal{ALC}$ Knowledge Base out of Φ. Then $\mathcal{KB}_{crisp}$ has a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ iff there exists a possible world I that satisfies $\mathcal{KB}_{crisp}$.

Proof $(\Rightarrow)$Suppose that $\mathcal{KB}_{crisp}$ has a model. This means that an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ exists. Then, the set of DL assertions that are satisfiable under $\mathcal{I}$ constitutes a subset of Φ, i.e a possible world I. This means that $\mathcal{KB}_{crisp}$ has also a model I.
$(\Leftarrow)$Now, suppose that there exists an I model of $\mathcal{KB}_{crisp}$. This means that $\mathcal{KB}_{crisp}$ is satisfiable, so a model $\mathcal{I}$ exists. $\Box$

The set of possible worlds $\mathcal{W}$ can be considered as a Dempster-Shafer frame of discernment, since the elements of $\mathcal{W}$ are mutually exclusive. We define a *Dempster-Shafer interpretation* m, as a basic probability assignment function on subsets of the set $\mathcal{W}$. Based on this assignment, we define belief and plausibility degrees, induced from bpa's on sets $T \subseteq \mathcal{W}$. In addition, the power set of $\mathcal{W}$, denoted as $\mathcal{PW}$ is defined over the following function:

$$\mathcal{PW} = 2^{\mathcal{W}}$$

In this context, we consider that a crisp $\mathcal{ALC}$ axiom can be true in a subset of $\mathcal{W}$. We define this subset as a set-interpretation, i.e:

Definition 3. Let us consider the set of all possible worlds (or interpretations) $\mathcal{W}$, with power-set $2^{\mathcal{W}}$. Any $K \in 2^{\mathcal{W}}$ is called a *set-interpretation*. Our set-interpretation is defined in an analogous way to a hyper-interpretation [108].

The entailment of an axiom ϕ under a set-interpretation is defined as:

Definition 4. An entailment of an axiom ϕ from a set-interpretation K, where $K \in 2^{\mathcal{I}}$ is defined as:

$$K \models \phi \text{ iff }, \forall\, I \in\ K,\ I \models_{DL} \phi$$
$$K \not\models \phi \text{ iff }, \exists\, I \in\ K,\ I \not\models_{DL} \phi$$
$$K \models \neg\phi \text{ iff }, \forall\, I \in\ K,\ I \not\models_{DL} \phi$$

In the definition above, $\models_{DL}$ denotes classical crisp DL entailment.

Definition 5. A *Dempster-Shafer* interpretation m is defined as a basic probability assignment, as follows:

$$m : 2^{\mathcal{W}} \rightarrow [0, 1]$$

As we operate on a Dempster-Shafer framework, a constraint that we have to preserve is the following:

$$\sum_{T \in\ 2^{\mathcal{W}}} m(T) = 1$$

Our Dempster-Shafer DL knowledge base assumes a set of possible worlds $\mathcal{W}$ and assigns a *Dempster-Shafer interpretation* to subsets of this set. Any $A \subseteq \mathcal{W}$ such that $m(A) > 0$ constitutes a *focal set-interpretation*. Following, we define Belief and Plausibility Degrees of assertions ϕ from these focal interpretations, based on *entailment* notion related to $\mathcal{PW}$.

Definition 6. The Belief Degree of an axiom ϕ under a Dempster-Shafer interpretation m is defined as:

$$Bel_m(\phi) = \sum_{PW \models \phi} m(PW), PW \in \mathcal{PW}$$

Definition 7. The Belief Degree of an axiom $\neg\phi$ under a Dempster-Shafer interpretation m is defined as:

$$Bel_m(\neg\phi) = \sum_{PW \not\models \phi} m(PW), PW \in \mathcal{PW}$$

In a Dempster-Shafer framework, the following relation holds for an axiom ϕ:

$$Pl_m(\phi) = 1 - Bel_m(\neg\phi)$$

Proposition 2. A Plausibility Degree for an axiom ϕ is equal to:

$$Pl_m(\phi) = 1 - \sum_{PW \not\models \phi} m(PW), PW \in \mathcal{PW}$$

Proof Based on the relation $Pl_m(\phi) = 1 - Bel_m(\neg\phi)$, we have the following:

$$\begin{aligned} Pl_m(\phi) &= 1 - Bel_m(\neg\phi) \Rightarrow \\ Pl_m(\phi) &= 1 - \sum_{PW \not\models \phi} m(PW), PW \in \mathcal{PW} \end{aligned}$$

□

Definition 8. The truthness of a Dempster-Shafer axiom ϕ_{DS}, where $\phi_{DS} \equiv \phi\ \mathcal{B} \geq \alpha$ or $\phi_{DS} \equiv \phi\ \mathcal{P} \geq \alpha$, under a Dempster-Shafer interpretation m is defined as:

$$m \models \phi\ \mathcal{B} \geq \alpha \text{ iff } Bel(\phi) \geq \alpha$$
$$m \models \phi\ \mathcal{P} \geq \alpha \text{ iff } Pl(\phi) \geq \alpha$$

Definition 9. A Dempster-Shafer interpretation m is a *model* of a Dempster-Shafer DL Knowledge Base $\mathcal{KB} = (\mathcal{B}, \mathcal{P})$ iff $m \models \mathcal{U}, \quad \forall \mathcal{U} \in \mathcal{B} \cup \mathcal{P}$.

Definition 10. A Dempster-Shafer axiom ϕ_{DS} is a *logical consequence* of a Dempster-Shafer DL Knowledge Base $\mathcal{KB}$, denoted as $\mathcal{KB} \models \phi_{DS}$, iff every model of $\mathcal{KB}$ is also a model of ϕ_{DS}.

In addition, a Dempster-Shafer DL Knowledge Base is *consistent* if a model exists for $\mathcal{KB}$.

An important issue considers consistency checking. Knowledge Base consistency partially stems from the relation between Belief and Plausibility degrees, i.e:

$$Bel(\phi) = 1 - Pl(\phi)$$

Consistency checking refers to the Belief - Plausibility Degrees of a formula ϕ and its negation $\neg\phi$.

Syntactic Consistency: Let us consider a Dempster-Shafer DL Knowledge Base $\mathcal{KB}$ with the following Belief and Plausibility constraints:

$$\begin{array}{c} \phi \quad \mathcal{B} \geq \alpha, \quad \phi \quad \mathcal{P} \geq \beta \\ \neg\phi \quad \mathcal{B} \geq \gamma, \quad \neg\phi \quad \mathcal{P} \geq \delta \end{array}$$

where $\alpha, \beta, \gamma, \delta \in [0, 1]$.

Then, based on the Belief-Plausibility relation, we conclude that a consistent Knowledge Base should be aligned with the following:

$$\beta \leq (1 - \gamma) \qquad \delta \leq (1 - \alpha)$$

Semantic Consistency: From a semantics point of view, if m a model of $\mathcal{KB}$, i.e:

$$\begin{array}{c} m \models \phi \quad \mathcal{B} \geq \alpha, \quad m \models \phi \quad \mathcal{P} \geq \beta \\ m \models \neg\phi \quad \mathcal{B} \geq \gamma, \quad m \models \neg\phi \quad \mathcal{P} \geq \delta \end{array}$$

Then, following our semantics definition, we conclude that:

$$Bel(\phi) \geq \alpha, \quad Bel(\phi) \geq \beta, \quad Pl(\neg\phi) \geq \gamma, \quad Pl(\neg\phi) \geq \delta$$

For $\mathcal{KB}$ being a consistent Knowledge Base, we have to preserve the following:

$$Bel(\phi) + Pl(\neg\phi) = 1$$

Example In order to illustrate our method, let us consider the following Dempster-Shafer DL Knowledge Base:

$$\begin{array}{c} < h_1 : Hotel \sqcap \exists cost. \leq_{100} > \quad \mathcal{B} \geq 0.5 \\ < h_1 : Hotel \sqcap \exists cost. \leq_{100} > \quad \mathcal{P} \geq 0.7 \end{array}$$

Our knowledge base is consistent, based on our consistency checking formulas. Now, let us suppose that we add the following axiom:

$$\neg < h_1 : Hotel \sqcap \exists cost. \leq_{100} > \quad \mathcal{B} \geq 0.9$$

Based on the consistency checking, we must have $0.7 \leq 0.1$, which makes our knowledge base inconsistent.

Semantics and Soundness

As a final issue we consider soundness of the semantics. In the following, we will prove that our $\mathcal{DS} - \mathcal{ALC}$ is sound with respect to crisp $\mathcal{ALC}$ semantics. We will follow the notation described in [119]. More precisely, we consider the transformation $\sharp(\cdot)$ which takes as input a Belief or a Plausibility constraint and outputs the crisp part of this constraint. More precisely,

$$\sharp(\phi \quad B \geq \alpha) \mapsto \phi$$
$$\sharp(\phi \quad P \geq \alpha) \mapsto \phi$$

In addition, if $\mathcal{KB} = (\mathcal{B}, \mathcal{P})$ a Dempster-Shafer DL Knowledge Base then:

$$\sharp\mathcal{KB} = \{\sharp\phi_{DS} : \phi_{DS} \in \mathcal{B} \cup \mathcal{P}\}$$

Then, the following proposition holds:

Proposition 3. Let $\mathcal{KB} = (\mathcal{B}, \mathcal{P})$ a Dempster-Shafer DL Knowledge Base and $\phi \quad B \geq \alpha$, $\phi \quad P \geq \beta$, $\alpha, \beta \in [0,1]$ DS axioms. If $\mathcal{KB} \models \phi \quad B \geq \alpha$ ($\phi \quad P \geq \beta$), then $\sharp(\mathcal{KB}) \models_{DL} \sharp(\phi \quad B \geq \alpha)$ ($\sharp(\mathcal{KB}) \models_{DL} \sharp(\phi \quad P \geq \beta)$).

Proof If $\mathcal{KB} \models \phi \quad B \geq \alpha$, it follows that any model of $\mathcal{KB}$ is also a model of $\phi \quad B \geq \alpha$. In other words, $\phi \quad B \geq \alpha$ can be considered as a part of $\mathcal{KB}$. Hence, $\sharp(\phi \quad B \geq \alpha)$ is also a part of $\sharp(\mathcal{KB})$. Let $\mathcal{I}$ a model of $\sharp(\mathcal{KB}$, then if $\phi \in \sharp\mathcal{KB}$, then $\mathcal{I} \models \phi$. □

The opposite of this proposition does not, generally, hold.

4.3.3 Combined Dempster-Shafer entailment

In this Section, a new notion of entailment, named *Combined Dempster-Shafer entailment* and denoted as $\models_{DS_{combined}}$ is defined. The Combined Dempster-Shafer entailment is applied on two different independent Knowledge Bases, named $\mathcal{KB}_1 = (\mathcal{B}_1, \mathcal{P}_1)$ and $\mathcal{KB}_2 = (\mathcal{B}_2, \mathcal{P}_2)$ and combine assertions that are entailed (with a Belief-Plausibility degree) by both Knowledge Bases.

Let us suppose that the following hold:

$$\mathcal{KB}_1 \models \phi : \mathcal{B} \geq \gamma$$
$$\mathcal{KB}_2 \models \phi : \mathcal{B} \geq \delta$$

where $\gamma, \delta \in [0,1]$.

This means that, if m_1 is a model of $\mathcal{KB}_1$ and m_2 is a model of $\mathcal{KB}_2$, then we have the following:

$$Bel_1(\phi) \geq \gamma$$
$$Bel_2(\phi) \geq \delta$$

In addition, we consider $T_i, i = 1, \ldots, n$ the focal set-interpretations of $\mathcal{KB}_1$ and $T_j, j = 1, \ldots, m$, the focal set-interpretations of $\mathcal{KB}_2$.

In our framework, Dempster's rule of Combination is applied in order to define a *Combined Belief Degree*. This rule provides for combination of a set of bpa's $m_1, \ldots, m_n$. In our approach, the Belief Degree of ϕ equals its bpa value, as subsets of ϕ are not feasible to be defined, i.e. $Bel(\phi) = m(\phi)$. Belief and Plausibility degrees can be defined as a combination of m_1, m_2. For defining a Combined Belief Degree we take into account the definition of the Belief Degree of a formula, defined previously and consider intersections of the form $T_i \cap T_j$, as in the case of Dempster's rule of Combination. Then, we introduce the following definition:

Definition 11. The Combined Belief Degree $Bel_{1,2}$ over models m_1 and m_2, is defined as:

$$Bel_{1,2}(\phi) = \frac{\sum_{T_i \cap T_j \models \phi} m_1(T_i) \times m_2(T_j)}{1 - \sum_{T_i \cap T_j = \emptyset} m_1(T_i) \times m_2(T_j)}$$

The Combined Plausibility Degree $Pl_{1,2}$ over models m_1 and m_2, is derived from the following formula:

$$Pl_{1,2}(\phi) = 1 - Bel_{1,2}(\neg\phi)$$

Let us consider $\mathcal{KB}_1$ with model m_1 and $\mathcal{KB}_2$ with model m_2. Then, the Dempster-Shafer Combined entailment is defined as follows:

Definition 12. An axiom $\phi \quad \mathcal{B} \geq \varepsilon$, $\varepsilon \in [0, 1]$ is Dempster-Shafer Combined entailed, under $\mathcal{KB}_1$ and $\mathcal{KB}_2$, denoted as $\mathcal{KB}_1 \oplus \mathcal{KB}_2 \models_{DS_{combined}} \phi \quad \mathcal{B} \geq \varepsilon$, iff $\varepsilon \geq Bel_{1,2}(\phi)$.

Definition 13. An axiom $\phi \quad \mathcal{P} \geq \varepsilon$, $\varepsilon \in [0, 1]$ is Dempster-Shafer Combined entailed, under $\mathcal{KB}_1$ and $\mathcal{KB}_2$, denoted as $\mathcal{KB}_1 \oplus \mathcal{KB}_2 \models_{DS_{combined}} \phi \quad \mathcal{P} \geq \varepsilon$, iff $\varepsilon \geq Pl_{1,2}(\phi)$.

Example Continuing the previous example, let us suppose, that we have two Dempster-Shafer DL Knowledge Bases, $\mathcal{KB}_1$ and $\mathcal{KB}_2$ consisting of the following axioms:

$$\mathcal{KB}_1 : < h_1 : Hotel \sqcap \exists cost. \leq_{100} > \quad \mathcal{B} \geq 0.5$$
$$\mathcal{KB}_2 : < h_1 : Hotel \sqcap \exists cost. \leq_{100} > \quad \mathcal{B} \geq 0.7$$

We consider $\mathcal{W} = \{I_1, I_2\}$, two possible worlds, where $< h_1 : Hotel \sqcap \exists cost. \leq_{100} >$ is false in I_1 and true in I_2, i.e, there exist two DL interpretations, $\mathcal{I}_1$ and $\mathcal{I}_2$ such that:

$$\mathcal{I}_1 \not\models < h_1 : Hotel \sqcap \exists cost. \leq_{100} >$$
$$\mathcal{I}_2 \models < h_1 : Hotel \sqcap \exists cost. \leq_{100} >$$

Table 8: Dempster-Shafer Interpretation

$\mathcal{PW}$	$\models \phi$	m_1	m_2
$\{I_1\}$	0	0.3	0.2
$\{I_2\}$	1	0.5	0.7
$\{I_1, I_2\}$	0	0.2	0.1

We consider two Dempster-Shafer interpretations, m_1, m_2 as described in Table 8. In addition, we consider m_1 a model of $\mathcal{KB}_1$ and m_2 a model of $\mathcal{KB}_2$. For our convenience we name $< h_1 : Hotel \sqcap \exists cost. \leq_{100} >$ as ϕ.

By applying the Combination, based on our formula defined in the previous section, we derive a result of $Bel_{combined}(\phi)$ of 0.78.

Based on the Dempster-Shafer Combined entailment, the following holds:

$$\mathcal{KB}_1 \oplus \mathcal{KB}_2 \models_{DS_{combined}}$$
$$< h_1 : Hotel \sqcap \exists cost. \leq_{100} > \quad \mathcal{B} \geq 0.78$$

4.4 Decidability and Reasoning in Dempster-Shafer Description Logics

In this Section, we provide a method for reasoning over a Dempster-Shafer DL Knowledge Base, $\mathcal{KB}_{DS}$, which actually contains ABox. Reasoning in DLs is usually accomplished through tableaux procedures [24]. The decidability problem in our framework can be reduced in finding a method for deciding whether $\mathcal{KB}_{DS} \models \tau$, where τ is a Dempster-Shafer assertion axiom. Deciding satisfiability in a Dempster-Shafer DL Knowledge Base should take into account a basic probability assignment on subsets of interpretations (or possible worlds). It has to be noted that our axioms are described by Belief and Plausibility conditions in a similar way to axioms defined in [118] where axioms are annotated with membership degree conditions. Having taken this into consideration, we adapt the method described in [118] and extend it in order to capture Belief and Plausibility conditions.

More precisely, we consider $\mathcal{O}$ as an alphabet of symbols (DL individuals) in the same way as it is referred in [24]. Moreover, we consider an alphabet of variable symbols $\mathcal{V}$ along with an ordering $\prec$ on $\mathcal{V}$. In addition, the common term *object* is employed for describing either a DL individual or a variable, in other words an object is an element of $\mathcal{O} \cup \mathcal{V}$. The symbols s, t are used to denote an object element. A *constraint* σ is defined as one of the following:

$$s : C$$
$$sPt$$

where C is a DL concept and P is a DL role. Following, a *constraint system* is defined as a finite nonempty set of constraints. Also, by $\neg\sigma$, we denote $s : \neg C$ or $s \neg Pt$.

Based on these concepts, we define a *Belief constraint* as follows:

Definition 14. A Belief constraint is defined as:

$$\sigma \;\; \mathcal{B} \bowtie n$$

where $\bowtie$ is one of $<, >, \leq, \geq$.

In addition, we defined a *Plausibility constraint* as follows:

Definition 15. A Plausibility constraint is defined as:

$$\sigma \;\; \mathcal{P} \bowtie n$$

where $\bowtie$ is one of $<, >, \leq, \geq$.

Definition 16. A *Dempster-Shafer constraint system* is defined as a set of Belief and Plausibility constraints.

Definition 17. An interpretation m *satisfies* a Belief Constraint

$$s : C \;\; \mathcal{B} \bowtie n \qquad (sPt \;\; \mathcal{B} \bowtie n)$$

iff $Bel_m(C(s)) \bowtie n$ (Resp. $Bel_m(P(s,t)) \bowtie n$).

Definition 18. An interpretation m *satisfies* a Plausibility Constraint

$$s : C \;\; \mathcal{P} \bowtie n \qquad (sPt \;\; \mathcal{P} \bowtie n)$$

iff $Pl_m(C(s)) \bowtie n$ (Resp. $Pl_m(P(s,t)) \bowtie n$).

In addition, m satisfies a constraint system $\mathcal{S}_{\mathcal{KB}}$ iff m satisfies every Dempster-Shafer constraint in it.

A Dempster-Shafer DL Knowledge Base $\mathcal{KB}_{DS}$ can be mapped into a Dempster-Shafer constraint system $\mathcal{S}_{\mathcal{KB}}$, defined as:

$$\begin{aligned} \mathcal{S}_{\mathcal{KB}} = \{a : C \;\; \mathcal{B} \geq n \mid C(a) \quad \mathcal{B} \geq n\} \quad \cup \\ \{a : C \;\; \mathcal{P} \geq n \mid C(a) \quad \mathcal{P} \geq n\} \quad \cup \\ \{aPb \;\; \mathcal{B} \geq n \mid P(a,b) \quad \mathcal{B} \geq n\} \quad \cup \\ \{aPb \;\; \mathcal{P} \geq n \mid P(a,b) \quad \mathcal{P} \geq n\} \end{aligned}$$

Then we have the following:

$\mathcal{KB}_{DS} \models C(a) \quad \mathcal{B} \geq n$ *iff* $\mathcal{S}_{\mathcal{KB}} \cup a : C \;\; \mathcal{B} < n$ *is not satisfiable*
$\mathcal{KB}_{DS} \models C(a) \quad \mathcal{P} \geq n$ *iff* $\mathcal{S}_{\mathcal{KB}} \cup a : C \;\; \mathcal{P} < n$ *is not satisfiable*
$\mathcal{KB}_{DS} \models P(a,b) \quad \mathcal{B} \geq n$ *iff* $\mathcal{S}_{\mathcal{KB}} \cup (aPb) \;\; \mathcal{B} < n$ *is not satisfiable*
$\mathcal{KB}_{DS} \models P(a,b) \quad \mathcal{P} \geq n$ *iff* $\mathcal{S}_{\mathcal{KB}} \cup (aPb) \;\; \mathcal{P} < n$ *is not satisfiable*

In order to examine constraint satisfiability of $\mathcal{S}_{\mathcal{KB}}$, we consider a set of constraint propagation rules. These rules actually add constraints to $\mathcal{S}_{\mathcal{KB}}$ until a *contradiction (or clash)* happens or the current constraint system is *complete* (i.e an m that satisfies a constraint to be added can be obtained from the current constraint system). Any complete set $\mathcal{S}_{\mathcal{KB}_{comp}}$ derived from an initial set $\mathcal{S}_{\mathcal{KB}_{init}}$, by applying a set of propagation rules, is called a *completion* of $\mathcal{S}_{\mathcal{KB}_{init}}$.

A set of Dempster-Shafer constraints $\mathcal{S}$ contains a contradiction iff it contains one of the following:

1. $\top, \bot$ contradictions:

$$\begin{array}{l} s : \bot\mathcal{B} \geq n, \quad s : \bot\mathcal{P} \geq n, \quad s : \bot\mathcal{B} > n, n > 0 \\ s : \bot\mathcal{P} > n, \quad s : \bot\mathcal{B} < 0, \quad s : \bot\mathcal{P} < 0, n > 0 \\ s : \top\mathcal{B} \leq n, n < 1, \quad s : \top\mathcal{P} \leq n, n < 1, \quad s : \top\mathcal{B} < n \\ s : \top\mathcal{P} < n, \quad s : \top\mathcal{B} > 1, \quad s : \top\mathcal{P} > 1 \end{array}$$

2. $<, >, \leq, \geq$ relationships contradictions:

$$\begin{array}{l} \sigma\ \mathcal{B} \geq n \ \text{ and } \ \sigma\ \mathcal{B} < m \ \text{ and } \ n \geq m \\ \sigma\ \mathcal{B} \geq n \ \text{ and } \ \sigma\ \mathcal{B} \leq m \ \text{ and } \ n > m \\ \sigma\ \mathcal{B} > n \ \text{ and } \ \sigma\ \mathcal{B} < m \ \text{ and } \ n \geq m \\ \sigma\ \mathcal{B} > n \ \text{ and } \ \sigma\ \mathcal{B} \leq m \ \text{ and } \ n \geq m \\ \sigma\ \mathcal{P} \geq n \ \text{ and } \ \sigma\ \mathcal{P} < m \ \text{ and } \ n \geq m \\ \sigma\ \mathcal{P} \geq n \ \text{ and } \ \sigma\ \mathcal{P} \leq m \ \text{ and } \ n > m \\ \sigma\ \mathcal{P} > n \ \text{ and } \ \sigma\ \mathcal{P} < m \ \text{ and } \ n \geq m \\ \sigma\ \mathcal{P} > n \ \text{ and } \ \sigma\ \mathcal{P} \leq m \ \text{ and } \ n \geq m \end{array}$$

In [119], the propagation rules have the following form:

$$\Phi \rightarrow \Psi \ \textit{if} \ \Gamma$$

where Φ, Ψ are sequences of Dempster-Shafer constraints and Γ is a condition. A rule fires if the following hold:

1. The condition Γ holds
2. The current set of Dempster-Shafer constraints contains a set of constrains that match Φ

After firing the constraints of Ψ are added to S.

In our rules, though, the condition Γ is part of the rule left hand side. The rules are the following:

1.$(\neg \geq) < \neg\sigma \ \mathcal{B} \geq k > \rightarrow < \sigma \ \mathcal{P} \leq (1-k) >$

2.$(\neg \geq) < \neg\sigma \ \mathcal{P} \geq k > \rightarrow < \sigma \ \mathcal{B} \leq (1-k) >$

3.$(\neg \leq) < \neg\sigma \ \mathcal{B} \leq k > \rightarrow < \sigma \ \mathcal{P} \geq (1-k) >$

4.$(\neg \leq) < \neg\sigma \ \mathcal{P} \leq k > \rightarrow < \sigma \ \mathcal{B} \geq (1-k) >$

5.$(\sqcap \geq) < \sigma_1 \sqcap \ \sigma_2 \ \mathcal{B} \geq k > \rightarrow < \sigma_1 \ \mathcal{B} \geq \ k \ _{and} \, \sigma_2 \ \mathcal{B} \geq \ k >$

6.$(\sqcap \geq) < \sigma_1 \sqcap \ \sigma_2 \ \mathcal{P} \geq k > \rightarrow < \sigma_1 \ \mathcal{P} \geq \ k \ _{and} \, \sigma_2 \ \mathcal{P} \geq \ k >$

7.$(\sqcap \leq) < \sigma_1 \sqcap \ \sigma_2 \ \mathcal{B} \leq k > \rightarrow < \sigma_1 \ \mathcal{B} \leq \ k \ _{or} \, \sigma_2 \ \mathcal{B} \leq \ k >$

8.$(\sqcap \leq) < \sigma_1 \sqcap \ \sigma_2 \ \mathcal{P} \leq k > \rightarrow < \sigma_1 \ \mathcal{P} \leq \ k \ _{or} \, \sigma_2 \ \mathcal{P} \leq \ k >$

9.$(\sqcup \geq) < \sigma_1 \sqcup \ \sigma_2 \ \mathcal{B} \geq k > \rightarrow < \sigma_1 \ \mathcal{B} \geq \ k \ _{or} \, \sigma_2 \ \mathcal{B} \geq \ k >$

10.$(\sqcup \geq) < \sigma_1 \sqcup \ \sigma_2 \ \mathcal{P} \geq k > \rightarrow < \sigma_1 \ \mathcal{P} \geq \ k \ _{or} \, \sigma_2 \ \mathcal{P} \geq \ k >$

11.$(\sqcup \leq) < \sigma_1 \sqcup \ \sigma_2 \ \mathcal{B} \leq k > \rightarrow < \sigma_1 \ \mathcal{B} \leq \ k \ _{and} \, \sigma_2 \ \mathcal{B} \leq \ k >$

12.$(\sqcup \leq) < \sigma_1 \sqcup \ \sigma_2 \ \mathcal{P} \leq k > \rightarrow < \sigma_1 \ \mathcal{P} \leq \ k \ _{and} \, \sigma_2 \ \mathcal{P} \leq \ k >$

13.$(\neg \ >) < \neg\sigma \ \mathcal{B} > k > \rightarrow < \sigma \ \mathcal{P} < (1-k) >$

14.$(\neg >) < \neg\sigma \ \mathcal{P} > k > \rightarrow < \sigma \ \mathcal{B} < (1-k) >$

15.$(\neg <) < \neg\sigma \ \mathcal{B} < k > \rightarrow < \sigma \ \mathcal{P} > (1-k) >$

16.$(\neg <) < \neg\sigma \ \mathcal{P} < k > \rightarrow < \sigma \ \mathcal{B} > (1-k) >$

17.$(\sqcap >) < \sigma_1 \sqcap \ \sigma_2 \ \mathcal{B} > k > \rightarrow < \sigma_1 \ \mathcal{B} > \ k \ _{and} \, \sigma_2 \ \mathcal{B} > \ k >$

18.$(\sqcap >) < \sigma_1 \sqcap \ \sigma_2 \ \mathcal{P} > k > \rightarrow < \sigma_1 \ \mathcal{P} > \ k \ _{and} \, \sigma_2 \ \mathcal{P} > \ k >$

19.$(\sqcap <) < \sigma_1 \sqcap \ \sigma_2 \ \mathcal{B} < k > \rightarrow < \sigma_1 \ \mathcal{B} < \ k \ _{or} \, \sigma_2 \ \mathcal{B} < \ k >$

20.$(\sqcap <) < \sigma_1 \sqcap \ \sigma_2 \ \mathcal{P} < k > \rightarrow < \sigma_1 \ \mathcal{P} < \ k \ _{or} \, \sigma_2 \ \mathcal{P} < \ k >$

21.$(\sqcup >) < \sigma_1 \sqcup \ \sigma_2 \ \mathcal{B} > k > \rightarrow < \sigma_1 \ \mathcal{B} > \ k \ _{or} \, \sigma_2 \ \mathcal{B} > \ k >$

22.$(\sqcup >) < \sigma_1 \sqcup \ \sigma_2 \ \mathcal{P} > k > \rightarrow < \sigma_1 \ \mathcal{P} > \ k \ _{or} \, \sigma_2 \ \mathcal{P} > \ k >$

23.$(\sqcup <) < \sigma_1 \sqcup \ \sigma_2 \ \mathcal{B} < k > \rightarrow < \sigma_1 \ \mathcal{B} < \ k \ _{and} \, \sigma_2 \ \mathcal{B} < \ k >$

24.$(\sqcup <) < \sigma_1 \sqcup \ \sigma_2 \ \mathcal{P} < k > \rightarrow < \sigma_1 \ \mathcal{P} < \ k \ _{and} \, \sigma_2 \ \mathcal{P} < \ k >$

25.$(\forall \geq) < s : \forall \, R.C \ \mathcal{B} \geq \ k >, \ < sRt \ \mathcal{B} > (1-k) > \rightarrow \ < t : C \ \mathcal{B} \geq \ k >$

26.$(\forall >) < s : \forall \, R.C \ \mathcal{B} > \ k >, \ < sRt \ \mathcal{B} > (1-k) > \rightarrow \ < t : C \ \mathcal{B} > \ k >$

27.$(\forall \leq) < s : \forall \, R.C \ \mathcal{B} \leq \ k > \rightarrow \ < sRt \ \mathcal{B} \geq (1-k) >, < t : C \ \mathcal{B} \leq k >$

28.$(\forall <) < s : \forall \, R.C \ \mathcal{B} < \ k > \rightarrow \ < sRt \ \mathcal{B} > (1-k) >, < t : C \ \mathcal{B} < k >$

29.$(\exists \leq) < s : \exists \, R.C \ \mathcal{B} \leq \ k >, \ < sRt \ \mathcal{B} > k > \rightarrow \ < t : C \ \mathcal{B} \leq \ k >$

30.$(\exists <) < s : \exists \, R.C \ \mathcal{B} < \ k >, \ < sRt \ \mathcal{B} > k > \rightarrow \ < t : C \ \mathcal{B} < \ k >$

31.$(\exists \geq) < s : \exists \, R.C \ \mathcal{B} \geq \ k > \rightarrow \ < sRt \ \mathcal{B} \geq k >< t : C \ \mathcal{B} \geq k >$

32.$(\exists >) < s : \exists \, R.C \ \mathcal{B} > \ k > \rightarrow \ < sRt \ \mathcal{B} > k >< t : C \ \mathcal{B} > k >$

33.$(\forall \geq) < s : \forall \, R.C \ \mathcal{P} \geq \ k >, \ < sRt \ \mathcal{P} > (1-k) > \rightarrow \ < t : C \ \mathcal{P} \geq \ k >$

34.$(\forall >) < s : \forall \, R.C \ \mathcal{P} > \ k >, \ < sRt \ \mathcal{P} > (1-k) > \rightarrow \ < t : C \ \mathcal{P} > \ k >$

35.$(\forall \leq) < s : \forall \, R.C \ \mathcal{P} \leq \ k > \rightarrow \ < sRt \ \mathcal{P} \geq (1-k) >, < t : C \ \mathcal{P} \leq k >$

36.$(\forall <) < s : \forall \, R.C \ \mathcal{P} < \ k > \rightarrow \ < sRt \ \mathcal{P} > (1-k) >, < t : C \ \mathcal{P} < k >$

37.$(\exists \leq) < s : \exists\, R.C \;\; \mathcal{P} \leq \; k >, \;\; < sRt \;\; \mathcal{P} > k > \rightarrow \; < t : C \;\; \mathcal{P} \leq \; k >$
38.$(\exists <) < s : \exists\, R.C \;\; \mathcal{P} < \; k >, \;\; < sRt \;\; \mathcal{P} > k > \rightarrow \; < t : C \;\; \mathcal{P} < \; k >$
39.$(\exists \geq) < s : \exists\, R.C \;\; \mathcal{P} \geq \; k > \rightarrow \; < sRt \;\; \mathcal{P} \geq k >< t : C \;\; \mathcal{P} \geq k >$
40.$(\exists >) < s : \exists\, R.C \;\; \mathcal{P} > \; k > \rightarrow \; < sRt \;\; \mathcal{P} > k >< t : C \;\; \mathcal{P} > k >$

Example Let us consider the following Knowledge Base:

$$\mathcal{KB} = \{a : C \quad \mathcal{B} \geq 0.7, \qquad \neg a : D \quad \mathcal{P} \geq 0.9\}$$

In addition we consider the following assertions:

$$\gamma_1 : a : C \quad \mathcal{B} \geq 0.5$$
$$\gamma_2 : \neg a : D \quad \mathcal{P} \geq 0.8$$

We will show that $\mathcal{KB} \models \gamma_1$ and $\mathcal{KB} \models \gamma_2$.

In the first case, we have to derive a *clash* for the $\mathcal{S}_{\mathcal{KB}} \cup \{a : C\mathcal{B} < 0.5\}$. Based on the relationships contradictions and assigning $n = 0.7$ and $m = 0.5$, we derive a clash. Hence, $\mathcal{KB} \models \gamma_1$.

In the second case, we have to derive a *clash* for the $\mathcal{S}_{\mathcal{KB}} \cup \{\neg a : D \;\; \mathcal{P} < 0.8\}$. We apply the following substitutions, based on the $\neg \geq$ rule defined previously:

$$\neg a : D \;\; \mathcal{P} \geq 0.9 \rightarrow a : D \;\; \mathcal{B} \leq 0.1 \tag{4.1}$$
$$\neg a : D \;\; \mathcal{P} < 0.8 \rightarrow a : D \;\; \mathcal{B} > 0.2 \tag{4.2}$$

Hence, we have clash, i.e $\mathcal{KB} \models \gamma_2$.

Proposition 4. A finite set of Belief and Plausibility constraints $\mathcal{S}_{\mathcal{KB}}$ is satisfiable iff there exists a contradiction-free completion of $\mathcal{S}_{\mathcal{KB}}$.

Proof $(\Rightarrow)$ Based on [119], given the termination property, which actually states that any completion of a finite set of constraints $\mathcal{S}_{\mathcal{KB}}$ can be obtained after a finite number of rule applications, we have that the propagation rules are *sound*, i.e. if we consider a satisfiable set $\mathcal{S}_{init}$, then there is a satisfiable completion $\mathcal{S}_{comp}$ of $\mathcal{S}_{init}$.
$(\Leftarrow)$ Suppose we have a contradiction-free completion $\mathcal{S}_{comp}$ of $\mathcal{S}_{init}$. Then, an interpretation $\mathcal{I}_{DS}$ can be defined that satisfies $\mathcal{S}_{comp}$. As $\mathcal{S}_{init} \subseteq \mathcal{S}_{comp}$ it follows that $\mathcal{I}_{DS}$ also satisfies $\mathcal{S}_{init}$.$\square$

In the following chapter, we adapt a Dempster-Shafer framework in a fuzzy DL environment.

5. DEMPSTER-SHAFER FUZZY DESCRIPTION LOGIC

As it is referred in the introduction, there is a need for representing uncertainty and vagueness through a common framework, especially in web-application areas. In the previous chapter, we have defined a DL extended with Dempster-Shafer concepts, in order to represent ignorance in Semantic Web environments. As a final step, we consider a theory for representing a fuzzy DL extended with a Dempster-Shafer framework. This framework is described in [70]. In this chapter, we describe our Dempster-Shafer Fuzzy Description Logic. Our framework constitutes a generalization scheme of a crisp DL with fuzzy conditions along with a Dempster-Shafer module.

Taking into account the fuzzy DL interpretations introduced in [118], our framework considers any such interpretation as a *possible world*. The set of possible worlds is regarded as a frame of discernment. Thus, a basic probability assignment function is assigned on subsets of this set. This measure constitutes the uncertainty framework of our method.

As we have stated in Chapter 2, a classical DL, assumes a universe $\mathcal{X}$ and subsets $\mathcal{A} \subseteq \mathcal{X}$, that constitute a DL *Concept*. Any element $x \in \mathcal{X}$ belongs to $\mathcal{A}$ or not, which is interpreted as a true/false value. The fuzzy extension assumes truthness interval on $[0, 1]$, where $\mathcal{A}$ is a *Fuzzy subset* and it is associated with a membership function $\mu_{\mathcal{A}}(x) : \mathcal{X} \to [0, 1]$. Any DL axiom, either crisp or fuzzy, has a truth value in a fuzzy interpretation $\mathcal{I}$. Our innovation, called *Dempster-Shafer Fuzzy Description Logic*, assigns probability masses into sets of fuzzy interpretations. As already stated, a fuzzy interpretation can be considered as a possible world.

Let $\mathcal{W}$ a set of fuzzy DL interpretations. Let's denote a basic probability assignment function, m_{DS} on $2^{\mathcal{W}}$ as $m_{DS} : 2^{\mathcal{W}} \to [0, 1]$. Then, the extension of our method employs sets of fuzzy DL interpretations $\mathcal{I} \in \mathcal{W}$ in order to define Belief Degrees of fuzzy subsets of an interpretation domain $\Delta^{\mathcal{I}}$ (or $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$). This means that we assume a Fuzzy Description Logic and define Belief Degrees Conditions for axioms of this logic. In our case, we have considered the DL $\mathcal{ALC}$ and based on a fuzzy extension of it, we define our Dempster-Shafer Fuzzy DL. Since we extend fuzzy $\mathcal{ALC}$ based on Zadeh fuzzy logic, we also employ this logic in our framework.

5.1 Basics Adapted from Fuzzy $\mathcal{ALC}$

The syntax of our Dempster-Shafer fuzzy DL is an extension of the fuzzy $\mathcal{ALC}$ Description Logic [118]. In this approach a fuzzy DL statement is defined as $\tau_{fuzzy} \equiv < a \quad n >, n \in [0, 1]$, where a is a crisp DL assertion. This statement means that a *is true with a membership degree at least* n. In our approach, this statement will be called a *fuzzy axiom*.

Based on this syntax, a *fuzzy DL interpretation*, $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, considers *Concepts* as fuzzy subsets of a domain $\Delta^{\mathcal{I}}$ and *Roles* as fuzzy subsets of a domain $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

More precisely, this Fuzzy DL defines a set of symbols, named *Primitive concepts*, denoted as A, *Primitive roles*, denoted as R and *Individuals*, denoted as a, b.

Any crisp $\mathcal{ALC}$ Concept C,D is defined as:

$$\begin{aligned} C, D \rightarrow & \\ & \top \mid \textit{(top concept)} \\ & \bot \mid \textit{(bottom concept)} \\ & A \mid \textit{(primitive concept)} \\ & C \sqcap D \mid \textit{(concept conjunction)} \\ & C \sqcup D \mid \textit{(concept disjunction)} \\ & \neg C \mid \textit{(concept negation)} \\ & \forall R.C \mid \textit{(universal quantification)} \\ & \exists R.C \mid \textit{(existential quantification)} \end{aligned}$$

Then, a fuzzy DL interpretation assigns:

- To each Concept C a function $C^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow [0, 1]$
- To each Role R a function $R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0, 1]$

The assignments defined above, actually constitute membership functions. Each DL individual a is interpreted as an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. Following, $C^{\mathcal{I}}(a^{\mathcal{I}})$ is the membership degree of $a^{\mathcal{I}}$ being in $C^{\mathcal{I}}$. The same applies for $R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}})$.

Zadeh's fuzzy logic is employed in order to define semantics for fuzzy $\mathcal{ALC}$. More precisely, a fuzzy interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ assigns the following values to an individual d:

$$\begin{aligned} \top^{\mathcal{I}}(d) = 1,\ \ \bot^{\mathcal{I}}(d) = 0 \\ (C \sqcap D)^{\mathcal{I}}(d) = min\{C^{\mathcal{I}}(d), D^{\mathcal{I}}(d)\} \\ (C \sqcup D)^{\mathcal{I}}(d) = max\{C^{\mathcal{I}}(d), D^{\mathcal{I}}(d)\} \\ (\neg C)^{\mathcal{I}}(d) = 1 - C^{\mathcal{I}}(d) \\ (\forall R.C)^{\mathcal{I}}(d) = min_{d' \in \Delta^{\mathcal{I}}} max\{\{1 - R^{\mathcal{I}}(d, d'), C^{\mathcal{I}}(d')\}\} \\ (\exists R.C)^{\mathcal{I}}(d) = max_{d' \in \Delta^{\mathcal{I}}} min\{\{R^{\mathcal{I}}(d, d'), C^{\mathcal{I}}(d')\}\} \end{aligned}$$

5.2 Syntax of Dempster-Shafer Fuzzy DL

Extending the logic above, our Dempster-Shafer Fuzzy DL considers fuzzy $\mathcal{ALC}$ axioms, annotated with belief degree conditions.

Definition 1. A Dempster-Shafer fuzzy assertion, τ, is defined as $\tau \equiv < \tau_{fuzzy} \quad : k >$, where τ_{fuzzy} is a fuzzy $\mathcal{ALC}$ axiom and $k \in [0, 1]$.

Definition 2. A *Dempster-Shafer Fuzzy ABox* is defined as a set of assertion axioms

$$< C(i) \quad n \quad : k >, < R\,(\,i_1\,,\,i_2) \quad n \quad : \,k >$$

where C is a DL Concept, R is a DL Role, i, i_1, i_2 are DL individual names and $n, k \in [0, 1]$

As in the crisp DL case, the definition above, actually, sets apart two kinds of assertions, Concepts and Roles. For example, $< CheapHotel(a) \quad 0.8 \quad : 0.9 >$, denotes that a is a $CheapHotel$ with membership degree at least 0.8 (*Fuzzy Degree Condition*) and Belief degree at least 0.9 (*Belief Degree Condition*). Also, $< CloseTo(a, b) \quad 0.8 \quad : \; 0.9 >$, denotes that a is related to b through $CloseTo$ with Fuzzy Degree Condition ”at least 0.8” and Belief Degree Condition ”at least 0.9”.

Definition 3. A Dempster-Shafer Fuzzy KB $\mathcal{K}$ is defined as a set of Dempster-Shafer fuzzy assertions.

5.3 Semantics

The semantics of any Description Logic is defined through *interpretations*, *satisfiability* and *logical consequence*.

5.3.1 Interpretation

As in the classical DL, we consider an interpretation, $\mathcal{I}_{DS}$, which defines a domain, named $\Delta^{\mathcal{I}_{\mathcal{DS}}}$. On this domain, we consider fuzzy subsets regarded as Fuzzy Concepts. Also, on $\Delta^{\mathcal{I}_{\mathcal{DS}}} \times \Delta^{\mathcal{I}_{\mathcal{DS}}}$ we define fuzzy subsets regarded as Fuzzy Roles.

Any DL framework is based on an interpretation that is defined for the particular DL. For example, a probabilistic DL is based on a probabilistic interpretation. In any case, the notion of *possible world* is employed, in order to define an *uncertainty environment*. Since our method applies a Dempster-Shafer model in a Fuzzy Logic model, we call it a *Dempster-Shafer Fuzzy interpretation*. We define a Dempster-Shafer Fuzzy interpretation over the power set of the set of possible worlds, $\mathcal{W}$, where $\mathcal{W}$ is the (infinite) set of *fuzzy DL interpretations* $\mathcal{I}$. The key-point here, is that we have to define Belief Degrees to account for fuzzy sets. This means that, for computing Belief Degrees, we have to take into consideration the membership function μ that describes any fuzzy set. For example, the statement $< a : CheapHotel >$ cannot be considered true/false in any interpretation, since this concept is described by fuzziness, and as such a membership degree is always associated with an individual a.

For applying a Dempster-Shafer framework, a frame of discernment, as well as its power set, needs to be defined. We consider the frame of discernment, $\mathcal{W}$, as the set of all possible worlds or interpretations i.e $\mathcal{W} = \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \dots\}$, where $\mathcal{I}_i, i = 1, 2, 3, \dots$ is a Fuzzy DL interpretation. This means, that the power set is defined as $2^{\mathcal{W}}$.

As in the fuzzy $\mathcal{ALC}$ [118], we introduce the following definitions:

Definition 4. A fuzzy assertion $< C(a) \quad n >$, where C is a Concept and a is an individual name, *holds* in a fuzzy interpretation $\mathcal{I}_j \in \mathcal{W}$, denoted as $\mathcal{I}_j \models < C(a) \quad n >$, iff $C^{\mathcal{I}_j}(a^{\mathcal{I}_j}) \geq n$, where $C^{\mathcal{I}_j}(a^{\mathcal{I}_j})$ is the *membership degree* of $a^{\mathcal{I}_j}$ being in $C^{\mathcal{I}_j}$.

Definition 5. A fuzzy assertion $< R(a,b) \quad n >$, where R is a Role and a, b are individual names, *holds* in a fuzzy interpretation $\mathcal{I}_j \in \mathcal{W}$, denoted as $\mathcal{I}_j \models < R(a,b) \quad n >$, iff, $R^{\mathcal{I}_j}(a^{\mathcal{I}_j}, b^{\mathcal{I}_j}) \geq n$, where $R^{\mathcal{I}_j}(a^{\mathcal{I}_j}, b^{\mathcal{I}_j})$ is the *membership degree* of $(a^{\mathcal{I}_j}, b^{\mathcal{I}_j})$ being in $R^{\mathcal{I}_j}$.

Any fuzzy axiom, τ_{fuzzy}, is associated with a membership degree, $\mu_{\tau_{fuzzy}}$. The definition of $\mu_{\tau_{fuzzy}}$ is based on a Fuzzy DL, as described in [121]. For example, $(a : \ CheapHotel)$ has a membership degree $\mu_{CheapHotel}(a)$, where $\mu_{CheapHotel}$ is the membership function of the Fuzzy Concept $CheapHotel$.

Definition 6. A set of possible worlds $\mathcal{T} = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_m\}$ *satisfies* a fuzzy axiom τ_{fuzzy}, denoted as $\mathcal{T} \ \models \ \tau_{fuzzy}$, *iff* $\quad \forall \, \mathcal{I}_i \in \ \mathcal{T}, \mathcal{I}_i \models \tau_{fuzzy}$.

Definition 7. A set of possible worlds $\mathcal{T} = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_m\}$ *does not satisfy* a fuzzy axiom τ_{fuzzy}, denoted as $\mathcal{T} \ \not\models \ \tau_{fuzzy}$, *iff* $\quad \exists \, \mathcal{I}_i \in \ \mathcal{T}, \mathcal{I}_i \not\models \tau_{fuzzy}$.

Definition 8. A *Dempster-Shafer fuzzy interpretation*, $\mathcal{I}_{\mathcal{DS}}$, is defined as

$$\mathcal{I}_{\mathcal{DS}} = (\Delta^{\mathcal{I}_{\mathcal{DS}}}, \cdot^{\mathcal{I}_{\mathcal{DS}}}, \mathcal{W}, m_{DS}),$$

where $\Delta^{\mathcal{I}_{\mathcal{DS}}}$ is the interpretation domain, $\cdot^{\mathcal{I}_{\mathcal{DS}}}$ is a fuzzy DL interpretation function, $\mathcal{W}$ is the set of possible worlds and m_{DS} is a basic probability assignment on subsets of $\mathcal{W}$. Any $\mathcal{I}_j \in \mathcal{W}$, such that $m_{DS}(\mathcal{I}_j) > 0$ is called a *focal possible world*.

Our Dempster-Shafer Fuzzy interpretation considers an m_{DS} assignment on subsets of $\mathcal{W}$. In this context, any $\mathcal{T} \subseteq \mathcal{W}$ with $m_{DS}(\mathcal{T}) > 0$ can play the role of a non-fuzzy focal element, to enable us to define a Belief function.

In that sense, we define the *Belief degree* of a fuzzy assertion τ_{fuzzy}, under $\mathcal{W}$, as follows:

Definition 9. The *Belief degree* of a fuzzy concept assertion τ_{fuzzy}, under $\mathcal{W}$ is defined as:

$$Bel_{m_{DS}}(\tau_{fuzzy}) = \sum_{\mathcal{T} \models \tau_{fuzzy}} m_{DS}(\mathcal{T}) \times inf_{\mathcal{I}_j \in \mathcal{T}} \mu_{\mathcal{I}_j}(a)$$

where $\mu_{\mathcal{I}_j}(a)$ is the *membership degree* of $a^{\mathcal{I}_j}$ in $C^{\mathcal{I}_j}$ in possible world $\mathcal{I}_j$.
The *Belief degree* of a fuzzy role assertion τ_{fuzzy}, under $\mathcal{W}$ is defined as:

$$Bel_{m_{DS}}(\tau_{fuzzy}) = \sum_{\mathcal{T} \models \tau_{fuzzy}} m_{DS}(\mathcal{T}) \times inf_{\mathcal{I}_j \in \mathcal{T}} \mu_{\mathcal{I}_j}(a,b)$$

where $\mu_{\mathcal{I}_j}(a,b)$ is the *membership degree* of $(a^{\mathcal{I}_j}, b^{\mathcal{I}_j}) \in R^{\mathcal{I}_j}$ in possible world $\mathcal{I}_j$.

Definition 10. The *Plausibility degree* of a fuzzy concept assertion τ_{fuzzy}, under $\mathcal{W}$ is defined as:

$$Pl_{m_{DS}}(\tau_{fuzzy}) = \sum_{\mathcal{T} \models \tau_{fuzzy}} m_{DS}(\mathcal{T}) \times sup_{\mathcal{I}_j \in \mathcal{T}} \mu_{\mathcal{I}_j}(a)$$

where $\mu_{\mathcal{I}_j}(a)$ is the *membership degree* of $a^{\mathcal{I}_j}$ in $C^{\mathcal{I}_j}$ in possible world $\mathcal{I}_j$.
The *Plausibility degree* of a fuzzy role assertion τ_{fuzzy}, under $\mathcal{W}$ is defined as:

$$Pl_{m_{DS}}(\tau_{fuzzy}) = \sum_{\mathcal{T} \models \tau_{fuzzy}} m_{DS}(\mathcal{T}) \times sup_{\mathcal{I}_j \in \mathcal{T}} \mu_{\mathcal{I}_j}(a,b)$$

where $\mu_{\mathcal{I}_j}(a,b)$ is the *membership degree* of $(a^{\mathcal{I}_j}, b^{\mathcal{I}_j}) \in R^{\mathcal{I}_j}$ in possible world $\mathcal{I}_j$.

The definitions of Belief and Plausibility measures are based on the fact that they are lower and upper probabilities of τ_{fuzzy} in a way similar to [139]. In order to define these measures, we consider *Belief and Plausibility degrees of Fuzzy sets derived from non-fuzzy focal elements*. More precisely, we consider subsets of the set of interpretations, as non-fuzzy subsets. Therefore, a Dempster-Shafer fuzzy interpretation assigns a mass degree on each subset. Any mass degree value greater than zero, results in a *non-fuzzy focal element*, or a *focal-set-interpretation*. This can be derived by considering a minimization linear programming problem as it has been defined in [139], and is described in the following. Viewing the Belief degree as a minimization problem is justified by the fact that the Belief degree of an $\mathcal{A} \subseteq \mathcal{W}$ can be considered as the minimal amount of belief assigned to $\mathcal{A}$ subject to the constraints imposed by $m_{\mathcal{DS}}$ [138]. In the same way, the Plausibility degree can be viewed as a maximization problem. Before proceeding into the minimization (maximization) process, we introduce the sets $\mathcal{W}_\tau$ and $\models_\tau$ of a Dempster-Shafer Fuzzy axiom τ.

Definition 11. The $\mathcal{W}_\tau$ set of a Dempster-Shafer Fuzzy axiom is defined as a fuzzy set

$$\{\mu_{\mathcal{I}_1}/\mathcal{I}_1, \mu_{\mathcal{I}_2}/\mathcal{I}_2, \dots\}$$

where each $\mu_{\mathcal{I}_i}, i = 1, 2, \dots$ is the membership degree of τ_{fuzzy} under fuzzy DL interpretation $\mathcal{I}_i$

Definition 12. The $\models_\tau$ set of a Dempster-Shafer Fuzzy axiom is defined as a fuzzy set

$$\{\mu_{\mathcal{I}_{a_1}}/\mathcal{I}_{a_1}, \mu_{\mathcal{I}_{a_2}}/\mathcal{I}_{a_2}, \dots\}$$

where each $\mathcal{I}_{a_i}, i = 1, 2, \dots$ entails τ_{fuzzy}, i.e $\mathcal{I}_i \models \tau_{fuzzy}$.

In addition, we denote the $\models_{\tau-}$ set of $\models_\tau$ as $\models_{\tau-} \equiv \{\mathcal{I}_{a_1}, \mathcal{I}_{a_2}, \dots\}$.

In order to apply this minimization (maximization) process in our Dempster-Shafer Fuzzy DL, we consider the following associations:

- The fuzzy set $\models_\tau$ is regarded as a Fuzzy subset B.
- A focal-set possible world $\mathcal{T}_j$ is regarded as a non-fuzzy focal element A_i.
- $\mathcal{T}_j \models \tau_{fuzzy}$ iff $\mathcal{T}_j \subseteq\models_{\tau-}$, a straightforward conclusion of the definition of $\models_{\tau-}$.

By using the correspondence above, the Belief degree of $\models_{\tau-}$ (in cases of crisp sets) is defined as a minimization problem as follows:

$$\sum_{\mathcal{I}_i \in \models_{\tau-}} \sum_j m_{DS}(\mathcal{I}_i : \mathcal{T}_j)$$

where $m_{DS}(\mathcal{I}_i : \mathcal{T}_j)$ is the probability mass allocated to $\mathcal{I}_i$ from a basic probability assignment of $\mathcal{T}_j$, with $\mathcal{T}_j$ being a focal-set possible world.

In an analogous way, the Plausibility degree of $\models_{\tau-}$ is defined as a maximization problem.

In addition, $m_{DS}(\mathcal{I}_i : \mathcal{T}_j)$ is constrained by the following:

$$\begin{aligned} &m_{DS}(\mathcal{I}_i : \mathcal{T}_j) \geq \ 0, j = 1, \ldots, l \\ &m_{DS}(\mathcal{I}_i : \mathcal{T}_j) = 0, \forall \mathcal{I}_i \notin \mathcal{T}_j \\ &m_{DS}(\mathcal{I}_i : \mathcal{T}_j) = m_{DS}(\mathcal{T}_j), \forall j = 1, 2, \ldots, l \end{aligned}$$

In order to compute the optimal solutions, we divide the initial problem into subproblems, each one considers the allocation of the mass of a single $\mathcal{T}_j, j = 1, \ldots, l$. The optimal solutions of these aforementioned problems are denoted as, $m_*(\models_\tau : \mathcal{T}_j)$ and $m^*(\models_\tau : \mathcal{T}_j)$. Hence, the Belief and Plausibility measures are computed by adding the optimal solutions for all $\mathcal{T}_j$, $j = 1, \ldots, l$, i.e:

$$Bel(\models_\tau) = \sum_{T_j \subseteq \mathcal{W}} m_*(\models_\tau : \mathcal{T}_j)$$

$$Pl(\models_\tau) = \sum_{T_j \subseteq \mathcal{W}} m^*(\models_\tau : \mathcal{T}_j)$$

The optimal solutions of the subproblems are the following [139]:

$$m_*(\models_\tau : \mathcal{T}_j) = \left\{ \begin{array}{ll} m(\mathcal{T}_j), & \text{if } \mathcal{T}_j \subset\models_\tau \\ 0, & otherwise \end{array} \right\}$$

$$m^*(\models_\tau : \mathcal{T}_j) = \left\{ \begin{array}{ll} m(\mathcal{T}_j), & \text{if } \mathcal{T}_j \cap \models_\tau \neq \emptyset \\ 0, & otherwise \end{array} \right\}$$

In order to account for fuzzy sets the minimization (maximization) process is defined as:

$$\sum_{\mathcal{I}_i \in \models_{\tau-}} \sum_j m_{DS}(\mathcal{I}_i : \mathcal{T}_j) \times \ \mu_{\models_{\tau-}}(\mathcal{I}_i)$$

The optimal solutions for subproblems are computed by assigning all the mass of $\mathcal{T}_j$ to the element of $\mathcal{T}_j$ that has the lowest (or highest) membership degree in $\models_\tau$:

$$m_*(\models_\tau\colon \mathcal{T}_j) = m_{DS}(\mathcal{T}_j) \times inf_{x \in \mathcal{T}_j} \mu_{\mathcal{I}}$$
$$m^*(\models_\tau\colon \mathcal{T}_j) = m_{DS}(\mathcal{T}_j) \times sup_{x \in \mathcal{T}_j} \mu_{\mathcal{I}}$$

As in the crisp case, by adding these optimal solutions, we get Belief and Plausibility degree of $\models_\tau$:

$$Bel(\models_\tau) = \sum_{T_j \subseteq \mathcal{W}} m_{DS}(\mathcal{T}_j) \times inf_{x \in \mathcal{T}_j} \mu_x$$
$$Pl(\models_\tau) = \sum_{T_j \subseteq \mathcal{W}} m_{DS}(\mathcal{T}_j) \times sup_{x \in \mathcal{T}_j} \mu_x$$

Finally, if $\mathcal{T} \subseteq \mathcal{W}$ is a focal element, we make the following assumptions:

- $Bel(\models_\tau) \equiv Bel(\tau_{fuzzy}),\ Pl(\models_\tau) \equiv Pl(\tau_{fuzzy})$
- $T_j \subseteq \mathcal{W} \equiv \mathcal{T} \models \tau_{fuzzy}$, since $m_{DS}(\mathcal{I}_i : \mathcal{T}_j) = 0, \forall \mathcal{I}_i \notin \mathcal{T}_j$

By considering the assumptions, we get the formulas of Belief and Plausibility as defined above.

Definition 13. A Dempster-Shafer Fuzzy interpretation $\mathcal{I}_{\mathcal{DS}}$ is a model (or satisfies) of a Dempster-Shafer Fuzzy Assertion Axiom $< \ \tau_{fuzzy} \quad : \ k >$ iff $Bel(\tau_{fuzzy}) \geq \ k$.

Definition 14. A Dempster-Shafer Fuzzy interpretation $\mathcal{I}_{\mathcal{DS}}$ is a model of a set of Dempster-Shafer Fuzzy axioms Ψ iff it satisfies each $\tau \in \Psi$.

Definition 15. A Dempster-Shafer Fuzzy axiom τ is a logical consequence of a Dempster-Shafer Fuzzy Knowledge Base $\mathcal{K}$, denoted as $\mathcal{K} \models \tau$ iff every model of $\mathcal{K}$ satisfies τ.

5.3.2 Combination of Dempster-Shafer Fuzzy Assertions

As a final step of our method, we introduce the concept of Dempster-Shafer Fuzzy Combined entailment, denoted as $\models_{DSF_{comb}}$. In order to perform this kind of entailment, we consider two Dempster-Shafer Fuzzy assertions:

$$\tau_1 :< \ a \ \ n \quad : k_1 >$$
$$\tau_2 :< a \ \ n \quad : k_2 >$$

In addition, we consider:

$$\mathcal{I}_{\mathcal{DS}_1} = (\Delta^{\mathcal{I}_{\mathcal{DS}}}, \cdot^{\mathcal{I}_{DS_1}}, \mathcal{W}, m_{DS_1}), \textit{ a model of } \tau_1$$
$$\mathcal{I}_{\mathcal{DS}_2} = (\Delta^{\mathcal{I}_{\mathcal{DS}}}, \cdot^{\mathcal{I}_{DS_2}}, \mathcal{W}, m_{DS_2}) \textit{ a model of } \tau_2$$

with sets $\models_{\tau_1}$ and $\models_{\tau_2}$, respectively. This means that the two sets contain the focal possible worlds of $m_{\mathcal{DS}_1}$ and $m_{\mathcal{DS}_2}$ that entail τ_1 and τ_2, respectively.

We consider $\mathcal{T}_{11}, \ldots \mathcal{T}_{1n}$, the focal points of $\models_{\tau_1 -}$ and $\mathcal{T}_{21}, \ldots \mathcal{T}_{2m}$, the focal points of $\models_{\tau_2 -}$. Then, we define the following:

Definition 16. A Dempster Shafer Fuzzy assertion $\tau \equiv < a \quad n \quad : k >$ is Dempster-Shafer Fuzzy Combined entailed under $\mathcal{I}_{\mathcal{DS}_1}$ and $\mathcal{I}_{\mathcal{DS}_2}$, denoted as $\mathcal{I}_{\mathcal{DS}_1} \oplus \mathcal{I}_{\mathcal{DS}_2} \models_{DSF_{comb}} \tau$, iff $k \geq \ Bel_{1,2}(\tau_{fuzzy})$,where $Bel_{1,2}(\tau_{fuzzy})$ is the Combined Belief Degree, defined as:

$$Bel_{1,2}(\tau_{fuzzy}) = \frac{\sum_{\mathcal{T}_{1i} \cap \mathcal{T}_{2j} \models \ \tau_{fuzzy}} m_{\mathcal{DS}_1}(\mathcal{T}_{1i}) \times \ m_{\mathcal{DS}_2}(\mathcal{T}_{2j})}{1 - \sum_{\mathcal{T}_{1i} \cap \mathcal{T}_{2j} = \emptyset} m_{\mathcal{DS}_1}(\mathcal{T}_{1i}) \times \ m_{\mathcal{DS}_2}(\mathcal{T}_{2j})}$$

5.4 Decidability and Reasoning in Dempster-Shafer Fuzzy Description Logics

As a final issue of our method, we discuss reasoning procedure. Decidability is a synonym to the deductive procedure that combines TBox and ABox in order to retrieve new information. DL decidability is resolved through tableaux procedures [24]. More precisely, we aim at deciding on the satisfiability of the formula $\mathcal{K} \models \tau$, where $\mathcal{K}$ is a Dempster-Shafer Fuzzy DL Knowledge Base and τ is a Dempster-Shafer Fuzzy axiom. In order to decide satisfiability, we have to take into account the basic probability assignment on subsets of fuzzy DL interpretations along with the membership degree conditions.

In our approach, we adapt and extend the decidability procedure described in [118, 119] defined over fuzzy $\mathcal{ALC}$, in order to account for Dempster-Shafer Degree Conditions. This approach was first introduced in [28] in a propositional logic framework.

Our method is based on a constraint system, first introduced in [93]. To begin with, we define as $\mathcal{O}$ an alphabet of symbols regarded as DL individuals in the same way as the one defined in [24]. We employ the term *object* for referring into an element in $\mathcal{O}$. We employ the symbols s, t for describing an object element. Following, a constraint (crisp) σ is defined as $\sigma \equiv s : C \quad or \quad \sigma \equiv sPt$, where C is a concept name and P is a role name. In addition, a Fuzzy Constraint is defined as $\sigma_f : \sigma \quad \bowtie n$, where $\bowtie$ is one of $<, >, \leq, \geq$ and σ is a DL constraint.

A *Belief Fuzzy Constraint* and a *Plausibility Fuzzy Constraint* are defined as $\sigma_f \ \ \mathcal{B} \bowtie k$ and $\sigma_f \ \ \mathcal{P} \bowtie k$, respectively, where $\bowtie$ is one of $<, >, \leq, \geq$ and σ_f is a fuzzy DL constraint.

A Dempster-Shafer Fuzzy Constraint system is defined as a finite non-empty set of Dempster-Shafer Fuzzy Belief and Plausibility Constraints.

We state that an interpretation $\mathcal{I}_{\mathcal{DS}} = (\Delta^{\mathcal{I}_{\mathcal{DS}}}, \cdot^{\mathcal{I}_{DS}}, \mathcal{W}, m_{DS})$ ***satisfies*** a Belief Fuzzy Constraint $\sigma_f \ \ \mathcal{B} \bowtie k$ iff $Bel_{m_{DS}}(\sigma_f) \bowtie k$. In addition, we state that an interpretation $\mathcal{I}_{\mathcal{DS}} = (\Delta^{\mathcal{I}_{\mathcal{DS}}}, \cdot^{\mathcal{I}_{DS}}, \mathcal{W}, m_{DS})$ ***satisfies*** a Plausibility Fuzzy Constraint $\sigma_f \ \ \mathcal{P} \bowtie k$ iff $Pl_{m_{DS}}(\sigma_f) \bowtie k$.

We state that an interpretation $\mathcal{I}_{DS}$ satisfies a constraint system $\mathcal{S}$ iff m_{DS} satisfies every Dempster-Shafer Fuzzy constraint in it.

Following, we map a Dempster-Shafer Fuzzy DL Knowledge Base, $\mathcal{K}$, into a constraint system as follows:

$$\begin{aligned} \mathcal{S}_{\mathcal{K}} = \{\sigma_f \ \ \mathcal{B} \geq k \mid \ < \tau_f \quad : k > \in \mathcal{K}\} \quad \cup \\ \{\neg\sigma_f \ \ \mathcal{P} \leq 1 - k \mid \ < \tau_f \quad : k > \in \mathcal{K}\} \end{aligned}$$

We have that $\mathcal{K} \models \tau_f \quad : k$ iff $\mathcal{S}_{\mathcal{K}} \cup \sigma_f \ \ \mathcal{B} < k$ is not satisfiable. In addition, σ_f is the equivalent fuzzy constraint of τ_f.

Satisfiability issues are resolved by employing a set of constraint propagation rules. Constraints are added progressively to $\mathcal{S}_{\mathcal{K}}$ until a *contradiction (or clash)* happens or the current constraint system is *complete*. A constraint system is considered complete if an interpretation $\mathcal{I}_{DS}$ that satisfies $\mathcal{S}_{\mathcal{K}}$ plus a constraint to be added can be obtained from the current constraint system.

A set of Dempster-Shafer constraints contains a contradiction iff it contains one of the following:

$\top, \bot$ Belief - Plausibility contradictions:

$$\begin{aligned} &\sigma_{f_\bot} \ \ \mathcal{B} \geq k, \quad \sigma_{f_\bot} \ \ \mathcal{P} \geq k, \ \ k > 0 \\ &\sigma_{f_\bot} \ \ \mathcal{B} > k, \quad \sigma_{f_\bot} \ \ \mathcal{P} > k, \ \ k > 0 \\ &\sigma_{f_\top} \ \ \mathcal{B} \leq k, \quad \sigma_{f_\top} \ \ \mathcal{P} \leq k, \ \ k < 1, \\ &\sigma_{f_\top} \ \ \mathcal{B} < k, \quad \sigma_{f_\top} \ \ \mathcal{P} < k, \ \ k < 1 \\ &\sigma_{f_\top} \ \ \mathcal{B} > 1, \quad \sigma_{f_\top} \ \ \mathcal{P} > 1, \quad \sigma_{f_\bot} \ \ \mathcal{B} < 0, \quad \sigma_{f_\bot} \ \ \mathcal{P} < 0 \end{aligned}$$

$<, >, \leq, \geq$ Belief relationships contradictions:

$$\begin{aligned} &\sigma_f \ \ \mathcal{B} \geq k \ \ \textit{and} \ \ \sigma_f \ \ \mathcal{B} < m \ \ \textit{and} \ \ k \geq m \\ &\sigma_f \ \ \mathcal{B} \geq k \ \ and \ \ \sigma_f \ \ \mathcal{B} \leq m \ \ and \ \ k > m \\ &\sigma_f \ \ \mathcal{B} > k \ \ and \ \ \sigma_f \ \ \mathcal{B} < m \ \ \textit{and} \ \ k \geq m \\ &\sigma_f \ \ \mathcal{B} > k \ \ \textit{and} \ \ \sigma_f \ \ \mathcal{B} \leq m \ \ \textit{and} \ \ k \geq m \end{aligned}$$

$<, >, \leq, \geq$ Plausibility relationships contradictions:

$$\begin{aligned} &\sigma_f \ \ \mathcal{P} \geq k \ \ \textit{and} \ \ \sigma_f \ \ \mathcal{P} < m \ \ \textit{and} \ \ k \geq m \\ &\sigma_f \ \ \mathcal{P} \geq k \ \ \textit{and} \ \ \sigma_f \ \ \mathcal{P} \leq m \ \ \textit{and} \ \ k > m \\ &\sigma_f \ \ \mathcal{P} > k \ \ \textit{and} \ \ \sigma_f \ \ \mathcal{P} < m \ \ \textit{and} \ \ k \geq m \\ &\sigma_f \ \ \mathcal{P} > k \ \ \textit{and} \ \ \sigma_f \ \ \mathcal{P} \leq \ m \ \ \textit{and} \ \ k \geq m \end{aligned}$$

As in [119], the propagation rules have the form $\Phi \rightarrow \Psi$ *if* Γ, where Φ, Ψ are sequences of Dempster-Shafer Fuzzy constraints and Γ is a precondition. A rule fires if the precondition Γ holds and the current set of Dempster-Shafer Fuzzy constraints contains a set of constrains that match Φ. After firing the constraints of Ψ are added to the current set of constraints. Since the constraints can be one of $>, <, \geq, \leq$, connectives are one of $\sqcap, \sqcup, \neg, \forall, \exists$ and we consider two types of constraints (Belief and Plausibility), then we have the following rules, taking into account that the condition Γ is part of the rule's left hand side.

1.$(\neg \geq) < \neg\sigma_f \ \mathcal{B} \geq k > \rightarrow < \sigma_f \ \mathcal{P} \leq (1-k) >$
2.$(\neg \geq) < \neg\sigma_f \ \mathcal{P} \geq k > \rightarrow < \sigma_f \ \mathcal{B} \leq (1-k) >$
3.$(\neg \leq) < \neg\sigma_f \ \mathcal{B} \leq k > \rightarrow < \sigma_f \ \mathcal{P} \geq (1-k) >$
4.$(\neg \leq) < \neg\sigma_f \ \mathcal{P} \leq k > \rightarrow < \sigma_f \ \mathcal{B} \geq (1-k) >$
5.$(\sqcap \geq) < \sigma_{1_f} \sqcap \ \sigma_{2_f} \ \mathcal{B} \geq k > \rightarrow < \sigma_{1_f} \ \mathcal{B} \geq \ k \ _{and} \ \sigma_{2_f} \ \mathcal{B} \geq \ k >$
6.$(\sqcap \geq) < \sigma_{1_f} \sqcap \ \sigma_{2_f} \ \mathcal{P} \geq k > \rightarrow < \sigma_{1_f} \ \mathcal{P} \geq \ k \ _{and} \ \sigma_{2_f} \ \mathcal{P} \geq \ k >$
7.$(\sqcap \leq) < \sigma_{1_f} \sqcap \ \sigma_{2_f} \ \mathcal{B} \leq k > \rightarrow < \sigma_{1_f} \ \mathcal{B} \leq \ k \ _{or} \ \sigma_{2_f} \ \mathcal{B} \leq \ k >$
8.$(\sqcap \leq) < \sigma_{1_f} \sqcap \ \sigma_{2_f} \ \mathcal{P} \leq k > \rightarrow < \sigma_{1_f} \ \mathcal{P} \leq \ k \ _{or} \ \sigma_{2_f} \ \mathcal{P} \leq \ k >$
9.$(\sqcup \geq) < \sigma_{1_f} \sqcup \ \sigma_{2_f} \ \mathcal{B} \geq k > \rightarrow < \sigma_{1_f} \ \mathcal{B} \geq \ k \ _{or} \ \sigma_{2_f} \ \mathcal{B} \geq \ k >$
10.$(\sqcup \geq) < \sigma_{1_f} \sqcup \ \sigma_{2_f} \ \mathcal{P} \geq k > \rightarrow < \sigma_{1_f} \ \mathcal{P} \geq \ k \ _{or} \ \sigma_{2_f} \ \mathcal{P} \geq \ k >$
11.$(\sqcup \leq) < \sigma_{1_f} \sqcup \ \sigma_{2_f} \ \mathcal{B} \leq k > \rightarrow < \sigma_{1_f} \ \mathcal{B} \leq \ k \ _{and} \ \sigma_{2_f} \ \mathcal{B} \leq \ k >$
12.$(\sqcup \leq) < \sigma_{1_f} \sqcup \ \sigma_{2_f} \ \mathcal{P} \leq k > \rightarrow < \sigma_{1_f} \ \mathcal{P} \leq \ k \ _{and} \ \sigma_{2_f} \ \mathcal{P} \leq \ k >$
13.$(\neg \ >) < \neg\sigma_f \ \mathcal{B} > k > \rightarrow < \sigma_f \ \mathcal{P} < (1-k) >$
14.$(\neg >) < \neg\sigma_f \ \mathcal{P} > k > \rightarrow < \sigma_f \ \mathcal{B} < (1-k) >$
15.$(\neg <) < \neg\sigma_f \ \mathcal{B} < k > \rightarrow < \sigma_f \ \mathcal{P} > (1-k) >$
16.$(\neg <) < \neg\sigma_f \ \mathcal{P} < k > \rightarrow < \sigma_f \ \mathcal{B} > (1-k) >$
17.$(\sqcap >) < \sigma_{1_f} \sqcap \ \sigma_{2_f} \ \mathcal{B} > k > \rightarrow < \sigma_{1_f} \ \mathcal{B} > \ k \ _{and} \ \sigma_{2_f} \ \mathcal{B} > \ k >$
18.$(\sqcap >) < \sigma_{1_f} \sqcap \ \sigma_{2_f} \ \mathcal{P} > k > \rightarrow < \sigma_{1_f} \ \mathcal{P} > \ k \ _{and} \ \sigma_{2_f} \ \mathcal{P} > \ k >$
19.$(\sqcap <) < \sigma_{1_f} \sqcap \ \sigma_{2_f} \ \mathcal{B} < k > \rightarrow < \sigma_{1_f} \ \mathcal{B} < \ k \ _{or} \ \sigma_{2_f} \ \mathcal{B} < \ k >$

20.$(\sqcap <) < \sigma_{1_f} \sqcap \ \sigma_{2_f} \ \mathcal{P} < k > \rightarrow < \sigma_{1_f} \ \mathcal{P} < \ k \ _{or} \, \sigma_{2_f} \ \mathcal{P} < \ k >$
21.$(\sqcup >) < \sigma_{1_f} \sqcup \ \sigma_{2_f} \ \mathcal{B} > k > \rightarrow < \sigma_{1_f} \ \mathcal{B} > \ k \ _{or} \, \sigma_{2_f} \ \mathcal{B} > \ k >$
22.$(\sqcup >) < \sigma_{1_f} \sqcup \ \sigma_{2_f} \ \mathcal{P} > k > \rightarrow < \sigma_{1_f} \ \mathcal{P} > \ k \ _{or} \, \sigma_{2_f} \ \mathcal{P} > \ k >$
23.$(\sqcup <) < \sigma_{1_f} \sqcup \ \sigma_{2_f} \ \mathcal{B} < k > \rightarrow < \sigma_{1_f} \ \mathcal{B} < \ k \ _{and} \, \sigma_{2_f} \ \mathcal{B} < \ k >$
24.$(\sqcup <) < \sigma_{1_f} \sqcup \ \sigma_{2_f} \ \mathcal{P} < k > \rightarrow < \sigma_{1_f} \ \mathcal{P} < \ k \ _{and} \, \sigma_{2_f} \ \mathcal{P} < \ k >$
25.$(\forall \geq) < s : \forall \, R.C \quad n \ \mathcal{B} \geq \ k >, \ < sRt \quad n \ \mathcal{B} > (1-k) > \rightarrow \ < t : C \quad n \ \mathcal{B} \geq \ k >$
26.$(\forall >) < s : \forall \, R.C \quad n \ \mathcal{B} > \ k >, \ < sRt \quad n \ \mathcal{B} > (1-k) > \rightarrow \ < t : C \quad n \ \mathcal{B} > \ k >$
27.$(\forall \leq) < s : \forall \, R.C \quad n \ \mathcal{B} \leq \ k > \rightarrow \ < sRt \quad n \ \mathcal{B} \geq (1-k) >, < t : C \quad n \ \mathcal{B} \leq k >$
28.$(\forall <) < s : \forall \, R.C \quad n \ \mathcal{B} < \ k > \rightarrow \ < sRt \quad n \ \mathcal{B} > (1-k) >, < t : C \quad n \ \mathcal{B} < k >$
29.$(\exists \leq) < s : \exists \, R.C \quad n \ \mathcal{B} \leq \ k >, \ < sRt \quad n \ \mathcal{B} > k > \rightarrow \ < t : C \quad n \ \mathcal{B} \leq \ k >$
30.$(\exists <) < s : \exists \, R.C \quad n \ \mathcal{B} < \ k >, \ < sRt \quad n \ \mathcal{B} > k > \rightarrow \ < t : C \quad n \ \mathcal{B} < \ k >$
31.$(\exists \geq) < s : \exists \, R.C \quad n \ \mathcal{B} \geq \ k > \rightarrow \ < sRt \quad n \ \mathcal{B} \geq k >< t : C \quad n \ \mathcal{B} \geq k >$
32.$(\exists >) < s : \exists \, R.C \quad n \ \mathcal{B} > \ k > \rightarrow \ < sRt \quad n \ \mathcal{B} > k >< t : C \quad n \ \mathcal{B} > k >$
33.$(\forall \geq) < s : \forall \, R.C \quad n \ \mathcal{P} \geq \ k >, \ < sRt \quad n \ \mathcal{P} > (1-k) > \rightarrow \ < t : C \quad n \ \mathcal{P} \geq \ k >$
34.$(\forall >) < s : \forall \, R.C \quad n \ \mathcal{P} > \ k >, \ < sRt \quad n \ \mathcal{P} > (1-k) > \rightarrow \ < t : C \quad n \ \mathcal{P} > \ k >$
35.$(\forall \leq) < s : \forall \, R.C \quad n \ \mathcal{P} \leq \ k > \rightarrow \ < sRt \quad n \ \mathcal{P} \geq (1-k) >, < t : C \quad n \ \mathcal{P} \leq k >$
36.$(\forall <) < s : \forall \, R.C \quad n \ \mathcal{P} < \ k > \rightarrow \ < sRt \quad n \ \mathcal{P} > (1-k) >, < t : C \quad n \ \mathcal{P} < k >$
37.$(\exists \leq) < s : \exists \, R.C \quad n \ \mathcal{P} \leq \ k >, \ < sRt \quad n \ \mathcal{P} > k > \rightarrow \ < t : C \quad n \ \mathcal{P} \leq \ k >$
38.$(\exists <) < s : \exists \, R.C \quad n \ \mathcal{P} < \ k >, \ < sRt \quad n \ \mathcal{P} > k > \rightarrow \ < t : C \quad n \ \mathcal{P} < \ k >$
39.$(\exists \geq) < s : \exists \, R.C \quad n \ \mathcal{P} \geq \ k > \rightarrow \ < sRt \quad n \ \mathcal{P} \geq k >< t : C \quad n \ \mathcal{P} \geq k >$
40.$(\exists >) < s : \exists \, R.C \quad n \ \mathcal{P} > \ k > \rightarrow \ < sRt \quad n \ \mathcal{P} > k >< t : C \quad n \ \mathcal{P} > k >$

In addition, since a plausibility value, is always greater than the belief value, we have the following rule:

$$(_{B \leq \ P}) < \sigma_f \ \mathcal{B} \geq k > \rightarrow < \sigma_f \ \mathcal{P} \geq k >$$

As an example, let us consider the following:

Example Let us consider the following Knowledge Base:

$$\mathcal{K} = \{R(a,b) \quad n \quad : 1.0, \qquad (\forall \, R.C)(a) \quad n \quad : \ 0.7\}$$

where n denotes a membership degree condition. In addition, we consider the assertion $\gamma_1 : C(b) \ \ n \quad : 0.7$.

We will show that $\mathcal{K} \models \gamma_1$.

First, we map our DL statements into a set of constraints, based on our constraint system definition, i.e:

$$aRb \ \ n \ \ \mathcal{B} \geq 1.0, \ \ a : \forall \, R.C \ \ n \ \ \mathcal{B} \geq 0.7, \ \ b : C \ \ n \ \ \mathcal{B} \geq 0.7$$

In order to prove that $\mathcal{K} \models \gamma_1$, we have to detect a *clash* for:

$$\{aRb \ \ n \ \ \mathcal{B} \geq \ 1.0, \ \ a : \forall \ R.C \ \ n \ \ \mathcal{B} \geq \ 0.7\} \cup$$
$$\{b : C \ \ n \ \ \mathcal{B} < \ 0.7\}$$

Then, we employ our rule contradictions as follows:

$$a : \forall \ R.C \ \ n \ \ \mathcal{B} \geq \ 0.7, \quad aRb \ \ n \ \ \mathcal{B} > \ 1.0 (> 0.3) \rightarrow$$
$$b : C \ \ n \ \ \mathcal{B} \geq \ 0.7$$

Then, as we have a *clash*, it follows that $\mathcal{K} \models C(b) \ \ n \quad : \ 0.7$.

The decidability process operates on a set of Belief and Plausibility (Fuzzy) constraints in a way analogous to Section 4.4. In addition, the concepts of *clash* and *completion* are defined similarly. As in Section 4.4 the following proposition holds:

Proposition 1. A finite set of Belief and Plausibility Fuzzy constraints $\mathcal{S}_{\mathcal{K}}$ is satisfiable iff there exists a contradiction-free completion of $\mathcal{S}_{\mathcal{K}}$.

Proof $(\Rightarrow)$ Given the termination property, which actually states that any completion of a finite set of constraints $\mathcal{S}_{\mathcal{K}}$ can be obtained after a finite number of rule applications, we have that the propagation rules are *sound*, i.e. if we consider a satisfiable set $\mathcal{S}_{init}$ (where in our case $\mathcal{S}_{init} = \mathcal{S}_{\mathcal{K}}$), then there is a satisfiable completion $\mathcal{S}_{comp}$ of $\mathcal{S}_{init}$.
$(\Leftarrow)$ Suppose we have a contradiction-free completion $\mathcal{S}_{comp}$ of $\mathcal{S}_{init}$ (where in our case $\mathcal{S}_{init} = \mathcal{S}_{\mathcal{K}}$). Then, a Dempster-Shafer fuzzy interpretation, $\mathcal{I}_{DS}$, can be defined that satisfies $\mathcal{S}_{comp}$. As $\mathcal{S}_{init} \subseteq \mathcal{S}_{comp}$ it follows that $\mathcal{I}_{DS}$ also satisfies $\mathcal{S}_{init}$.$\square$

5.4.1 Complexity issues

As a final point, we shall make some remarks on complexity issues of our framework. In [118], the relationship between Fuzzy DL satisfiability and crisp DL satisfiability is discussed. In our framework, the proofness of this relation is based on the following:

Proposition 2. Let $\mathcal{K} = \{\tau_1, \tau_2, \ldots, \tau_o\}$ a Dempster-Shafer Fuzzy DL KB. Then, the crisp counterpart of $\mathcal{K}$ is defined as a crisp KB, $\overline{\mathcal{K}} = \{\tau_{1_{crisp}}, \tau_{2_{crisp}}, \ldots, \tau_{o_{crisp}}\}$. Since any crisp assertion can be considered as a Dempster-Shafer Fuzzy assertion (by assigning on Belief, Plausibility and membership conditions the value 1), then we have that if $\mathcal{K} \models \tau_i, i = 1, \ldots, o$, then $\overline{\mathcal{K}} \models_{crisp} \tau_{i_{crisp}}, i = 1, \ldots, o$. In addition, if $\overline{\mathcal{K}} = \{\tau_{1_{crisp}}, \tau_{2_{crisp}}, \ldots, \tau_{o_{crisp}}\}$ a crisp KB, then the fuzzy counterpart of it is denoted as $\mathcal{K}_{fuzzy} = \{\tau'_1, \tau'_2, \ldots, \tau'_o\}$, where $\tau'_i, i = 1, \ldots, o$ is defined as $\tau'_i \equiv \tau_i \quad 1.0 \quad : 1.0$.

This proposition, actually states that any Dempster-Shafer Fuzzy DL entailment, presupposes crisp entailment. By employing this proposition, the Dempster-Shafer Fuzzy $\mathcal{ALC}$ satisfiability is restricted into $\mathcal{ALC}$ satisfiability. In addition, as the entailment problem in $\mathcal{ALC}$ is PSPACE-complete [110], then, we have that Dempster-Shafer Fuzzy entailment decidability is PSPACE-complete. This is because we have a PSPACE-hard problem, based on our proposition, and additionally our propagation rules can be seen as a set of trace rules [119] [110]. Thus, our rules ensure polyonomial space and therefore, we regard our framework as a PSPACE-complete problem.

Another complexity issue considers Dempster's rule of Combination. Based on [95], the combination procedure can be regarded as a $\#P$-complete problem. However, in [131] a set of algorithms is proposed in order to manage better complexity. In addition, approximation methods for Dempster-Shafer theory are also examined in [13]. Monte Carlo algorithms [131] are employed as an alternative solution to complexity issues. In those algorithms, the combined Belief Degree is estimated through a large number trials of a random algorithm. These algorithms are efficient in cases of non-conflicting evidence. Non-conflicting evidence can be defined as ABox inconsistencies in a DL environment. In [91] a method for resolving these inconsistencies is outlined. The algorithm defined is based on a Belief network construction. Thus, in order to achieve better complexity in our DL ontologies, one way is to define consistent DL ontologies, based on the steps defined above, and then employ the Monte-Carlo algorithm. However, this approach, is presented as an example of possible optimizations and has not been employed in our framework.

5.5 A Dempster-Shafer Fuzzy Meta-Ontology

In this Section, we define a meta-ontology for representing Dempster-Shafer Fuzzy axioms, along with a set of rules for reasoning upon them, in order to extend DL with our imprecision support mechanisms. Our ontology is defined by employing classical crisp ontology concepts, i.e. Classes, Relationships, Attributes, Concrete domains and Individuals. Then, we are able to annotate OWL axioms with Dempster-Shafer Fuzzy Degree Conditions in a way similar to [117]. As ontologies are defined through the OWL language, it is straightforward to consider Dempster-Shafer Fuzzy OWL axioms to implement our Dempster-Shafer Fuzzy DL.

More precisely, the DS Fuzzy axioms, as defined in Paragraph 5.2, are mapped into the following Dempster-Shafer Fuzzy OWL axioms:

$$< C(o) \quad n \quad : k > \mapsto Individual(o)\ \ type(C)\ \ : n \quad \mathcal{B} \geq\ k$$

$$< R(o_1, o_2) \quad n \quad : k > \mapsto Individual(o_1)\ \ value(R, o_2)\ \ : n \quad \mathcal{B} \geq\ k$$

We define a Dempster-Shafer Fuzzy ontology $\mathcal{O}$, as a set of Dempster-Shafer Fuzzy OWL axioms. This meta-ontology is based on the meta-ontology defined in [136].

In our metaontology, we combine the axioms with fuzzy rules, in order to perform reasoning. These rules serve as a means to classify our data. In order to do this, we incorporate the Fuzzy Systems [3] and Fuzzy Systems Modelling technique (FSM) [99]. A rule has the following form:

WHEN U_1 IS X_1 AND $\ldots$ AND U_n IS X_n THEN Y IS Z

Here $X_i, i = 1, \ldots, n$ denote linguistic terms, e.g *Cheap Hotel*, $U_i, i = 1, \ldots, n$ is a set of antecedent variables, Y is the output variable and Z is a linguistic term that represents the output as vague concept, e.g *Proposed Hotel*. An example of an FSM rule is the following:

WHEN U_1 IS $GoodReviewsHotel$ AND U_2 IS $CheapHotel$
THEN Y IS $ProposedHotel$

In our example, we consider $U_1 = U_2 = Y$. The firing level of each rule is defined as: $f_{DS} = Min_i[\mu_{X_i}(x) \times Bel_{X_i}(x))]$, where x represents an individual, μ_{X_i} is a membership degree and Bel_{X_i} is a Belief degree regarding x. If we want to classify an individual x as being Z, then the firing level of the rule serves as a classification score value. In the f_{DS} definition, we actually consider a reduction of the membership degree by a factor equal to the belief degree according to the ideas followed in Section 5.3 for computing the Belief Degree. Next, we combine those reduced membership degrees based on Zadeh's Fuzzy Logic *t-norm* operator. In addition, in [41], the equivalence between belief functions and probability theory, as well as the equivalence between probability theory and membership functions is outlined. According to that, by interpreting the membership function as conditional probability, we can consider the firing levels as Belief Degree values.

Let us consider a rule r with two different firing levels, f_{DS1} and f_{DS2}, respectively. Then, we can combine the two firing levels, by considering them as Belief Degree values and employing Dempster's Rule of Combination. The combination of them is performed through a Combined Belief Degree formula, by considering a frame of discernment $W = \{Z, \neg Z\}$. Then, we proceed by employing Dempster's Rule of Combination. This technique will allow us to combine classification score values from different sources.

In the following Chapter, we overview some applications related to the frameworks described until now.

6. APPLICABILITY AND EVALUATION

In this Chapter, we present some applications suitable for uncertainty and vagueness in Semantic Web environments. As we will see next, we operate on two different case studies:

- Recommender systems
- Matchmaking environments

In both approaches, we consider suitable ontologies that capture vagueness and uncertainty degrees. In addition, the set of statements is represented through a syntax aligned with the framework defined in Chapter 5.

In cases of recommender systems, we aim at hotel recommendation sites. Towards this we employ the meta-ontology concept defined in the previous chapter for classifying a set of hotels from various recommendation sites. Information is represented through a set of statements of the following form:

$$< LowCost(h)\ 0.4\ : 1.0 >$$

The statement above denotes that *hotel h is a low cost hotel with fuzzy degree condition 0.4 and belief degree condition 1.0*. As we see in this example, the representation scheme follows our Dempster-Shafer Description Logic framework defined in Chapter 5. Following, based on this representation scheme, a reasoning technique suitable for Fuzzy Systems is performed in order to classify the hotels with a proposed degree value. In addition, we operate on a Big-Data case study in which case, we consider a set of rules where conditions are either fuzzy or uncertain statements. The conclusion of the rule is a proposed degree for each hotel element. An imprecision ontology is employed in order to serve as a repository for rules and constraints.

In matchmaking case studies, we model the criteria through a set of fuzzy degrees, suitable for a job advertisement - seeker case study. The set of criteria is annotated with a set of weights, represented through a Dempster-Shafer framework. Each job individual is annotated with a fuzzy degree (based on set of membership functions). In addition, according to the weights assigned to both matchmaking parts, a belief degree is also assigned. Thus, our representation scheme is aligned with the framework described in Chapter 5.

6.1 Dempster-Shafer Fuzzy Metaontology - An application

Recommender systems are proved an ideal framework for applying our Dempster-Shafer Fuzzy DL approach. Generally, a recommender system can be regarded as a method for developing strategies in order to perform affordable, personal and high-quality recommendations [66]. A hotel recommender system can be considered as such a system.

Table 9: Hotel Attributes

Source 1	Source 2	Source 3
cost per night	cost per night	cost per night
review score	review score	review score
reviews number	reviews number	reviews number
distance	distance	distance
location review		demand rating
booking rate		cancellation policy

In order to test our methodology in a real-world environment, we considered a dataset of hotels in London derived from various hotel recommendation sites, namely, ”www.booking.com” (source 1), ”www.airtickets.gr” (source 2) and ”www.priceline.com” (source 3). Our system has been implemented in Java language. Each recommendation site provides us with different information that describe a hotel, like *cost per night*, *review score*, etc (see Table 9). These are referred to as *hotel attributes*.

In addition, we defined a set of queries that employ imprecise properties, e.g *I'm looking for a low cost hotel that has good review score*. Our goal is to classify each hotel based on the query's criteria as a *proposed* or *not proposed* one.

The evaluation data consists of a set of approximately 2,854 hotels gathered from these recommendation sites. The evaluation method used 254 queries. These queries are actually formed by the combination of all attributes (and values) that describe a hotel element.

Initially, we perform the hotel classification for each recommendation site, separately. In order to do this, we detect the fuzzy and uncertainty properties. For example, the *cost per night* and *review score* determine the fuzzy properties *cheap* and *good*, whereas the *number of reviews* is a belief degree and describes *how certain* we are regarding the review score. Next, we define a *recommendation degree* which comes as a combination of the review score and the number of reviews and it is a Dempster-Shafer fuzzy property. The recommendation degree is defined based on our Belief degree definition and is computed by multiplying the review score by a *belief degree value*. The belief degree value is defined as:

$$BeliefDegree_{recommendationDegree} = \\ number\_of\_reviews/max\_number\_of\_reviews$$

Then, the recommendation degree formula is defined as:

$$recommendationDegree = \\ review\_Score \times \; BeliefDegree_{recommendationDegree}$$

For example, a hotel entry from *www.booking.com* has the following form:

name: ”Melbourne House Hotel”, review score: ”8.2”, cost: ”139”, number of reviews: ”676”

Then, based on a set of pre-defined membership functions along with the *recommendation degree* formula, our method translates the raw data above into a set of Dempster-Shafer Fuzzy assertions. These fuzzy assertions are based on the syntax defined in Paragraph 5.2. More precisely, if we map the aforementioned hotel as domain individual h, then the above statement is translated as:

$$< GoodReview(h)\ 0.8\ : 0.69 >, < LowCost(h)\ 0.4\ : 1.0 >$$

These assertions are inserted into our FSM and a new assertion of the form ”Melbourne House Hotel is *proposedHotel* with Belief Degree of 0.83”, is generated.

In the implementation, our target is to classify hotels from various sites based on a set of criteria by assigning each hotel individual a *proposed score value*. A hotel may exist in more than one recommendation sites, so the proposed score value comes as a combination result. Thus, we proceed by combining the results of the sources. In case of missing attributes in one source, in this source we assume ignorance (i.e. $Bel \geq 0$). Let's consider the combination of sources 1 and 2, as they have a lot of attributes in common, and, thus, we avoid many occurences of ignorance. Since the two data sources are independent, a combination of them, based on Dempster's Rule, can be performed. More precisely, the combination considers statements such as, $\tau_1 \equiv\ hotelA\ \ proposedHotel\ \ k_1$ and $\tau_2 \equiv\ hotelA\ \ proposedHotel\ \ k_2$, derived from sources 1 and 2, respectively, were k_1, k_2 denote Belief Degree conditions. Then, we are able to compute a combined degree and derive the statement: $hotelA\ \ proposedHotel\ \ k_{1,2}$. The combination is computed by considering a frame of discernment $\mathcal{W} = \{p, \neg\, p\}$, where p means ”proposed hotel”. Based on Dempster's rule of combination, we get the following (simplified) formula:

$$Combined_{1,2} = k_1 \times\ k_2 + (1 - k_1) \times\ k_2 + k_1 \times (1 - k_2)$$

In the example above, the first source derives the statement ”Melbourne House Hotel is proposedHotel with Belief Degree of 0.83”, whereas the second derives the statement ”Melbourne House Hotel is proposedHotel with Belief Degree of 0.81”. This means that we have $k_1 = 0.83$ and $k_2 = 0.81$. The combination of these two statements derives the new one ”Melbourne House Hotel is proposedHotel with Belief Degree of 0.95”. Although the Dempster's rule of combination produces counterintuitive results, in the general case, this does not occur in our case as we do not expect strongly conflicting evidence among a set of recommendation sites.

We evaluated our implementation in terms of precision and recall measurements. When dealing with fuzzy sets, the classical measurements for recall and precision cannot be employed, since they are designed for crisp sets [87]. In that case, the scalar cardinality defined in [147] is proposed. *Sigma cardinality*, denoted as sigma-count, of a fuzzy set $F : X \rightarrow [0, 1]$ is defined as $sc(F) = \sum_{x\in\ X} F(x)$.

In our case, we measure precision and recall as follows:

$$precision_f = \frac{sc_{rel} \cap\ sc_{retr}}{sc_{retr}}, \qquad recall_f = \frac{sc_{rel} \cap\ sc_{retr}}{sc_{rel}}$$

In the formulae above, sc_{rel} and sc_{retr} denote the sigma-count of relevant and retrieved fuzzy sets of elements, respectively. In addition, the intersection is interpreted as the minimum. The classification of fuzzy queries' output that classifies a hotel as *proposed* is

Table 10: Recall and Precision

Source	Size (#)	Queries (#)	Recall (%)	Precision (%)	Time) (min)
Sources 1	1100	107	85	97	3.53
Source 2	884	97	86	98	2.12
Source 3	475	107	87	97	3.23
Source 1,2	1706	36	85	96	13.10

another aspect that has to be treated differently. In [7], the concept of *linguistic variable* is employed. In our application, this can be defined by considering the set of possible Belief Degree condition values i, where $i \in [0, 1]$, which constitutes the base variable for the linguistic value $proposed$ (Fuzzy restriction). The fuzzy restriction $proposed$ is defined as follows:

$$\mathcal{F}_{proposed}(i) = \left\{ \begin{array}{ll} 0, & \text{for } 0 \leq i \leq a \\ \frac{i-a}{b-a}, & \text{for } a \leq i \leq b \\ 1, & \text{for } b \leq i \leq 1 \end{array} \right\}$$

In our case study, after experimentation, we assigned $a = 0.7$, $b = 0.8$. This restriction serves as a way to detect relevant elements, aka compute *precision* and *recall*.

The results of our evaluation are summarized in Table 10. In this table, execution time measurements are also included. The recall, precision and time measurements are the average of the recall, precision and time of all the executed queries. Regarding precision, the classification method returns accurate results, where ”accurate” depends on the way we fuzzify attributes. Regarding recall, we get a lower percentage. This happens because we came across situations of ignorance (of ”null” attributes). In that case, we are not able to fuzzify these attributes. More precisely, our recall measures are estimated on around 85%. Note that the source combination resulted in a much longer execution time. This is due to the complexity of Dempster’s rule of combination. As stated later, as a future work we may consider performance efficiency issues of our framework. Taking into account the performance of the implemented system and the quality of the results, the feasibility of our method has been proven. The main contribution of our method is in terms of expressiveness as, to our knowledge, it is the only one that allows for representation of imperfect information, either vague or uncertain, in a unified semantic web framework.

6.2 Big Data Case Study

In real-world applications, the large scale data demand specific processing methods. Regarding reasoning process, interleaving reasoning and selection can be applied in order to deal with Big Data. Generally, interleaving reasoning is based on the selection of a meaningful, limited initial dataset. The selection of this consistent dataset becomes more difficult is cases of web data where information is usually inconsistent. The selection procedure excludes the unnecessary items from a dataset for achieving better processing.

In our approach, we consider a reasoning process suitable for Fuzzy Sets and Fuzzy Logic, denoted as Fuzzy Systems Modelling (FSM). FSM actually allows for the definition of a set of if then rules, by considering a semantic understanding of fuzzy concepts. More precisely, an FSM [105] serves as a way to develop semantically rich rule based representations in order to model *complex, nonlinear multiple input-output systems*. In addition, FSMs provide methods for reasoning with fuzziness. As it is referred in [135], the following characteristics describe an FSM system:

- FSM provide a formal reasoning and manipulation
- FSM provide semantics for human conceptualization

Moreover, one of the basic characteristics of an FSM is its potential to formulate new conclusions out of the consequents of a rule. We apply this modelling in uncertain and vague frameworks. We aim at defining a common degree for representing both uncertainty and vagueness. Also, we consider a method for combining a set of FSMs rules, by employing Dempster-Shafer theory along with Dempster's rule of Combination.

An FSM considers a set of n rules. Each rule $i = 1, \ldots, n$ has the following form:

When U_1 *is* A_{i1} *and* U_2 *is* A_{i2} *and* $\ldots$ U_r *is* A_{ir} *then* V *is* D_i

Each $U_j, j = 1, \ldots r$ is a variable, whereas each A_{ij} denotes a linguistic variable, represented as a fuzzy subset over a domain X_j of the variable U_j. In addition, D_i is also represented as a fuzzy subset of a domain Y of V. The antecedent describes a condition and defines a fuzzy region of the space $X_1 \times X_2 \times \cdots \times X_r$. Hence, if the input is in this region, the consequent holds.

Large scale data reasoning has been taken into account. In our method, we consider a step for data elimination, in order to produce a meaningful set of data. Following, as a case study, a metaclassifier process is described and tested. As in the previous case, we aim at classifying a hotel as being (or not) proposed.

At first, we assume a set of n sources, each providing some information about a specific domain. In our case study, we consider a set of hotel recommendation sites, where each site represents such a source.

Our goal is to classify a hotel as being (or not) proposed, i.e:

Hotel $\mathcal{A}$ *is proposed with a degree of* 0.8

Throughout the classification procedure, situations of missing information may arise. Let us consider a recommendation site that provides the following information:

- hotel's star category
- hotel's cost per night
- a list of statistics considering a hotel's star category and its facilities, e.g *40 per cent of 3-star hotels provide a swimming pool*

In addition, let us consider the following query:

I'm looking for a low cost hotel that provides a swimming pool

The aforementioned query is described by the following:

- Vagueness: A membership function describes the low cost criterion
- Uncertainty: The "swimming pool" criterion is annotated with an uncertainty degree (e.g 40%)

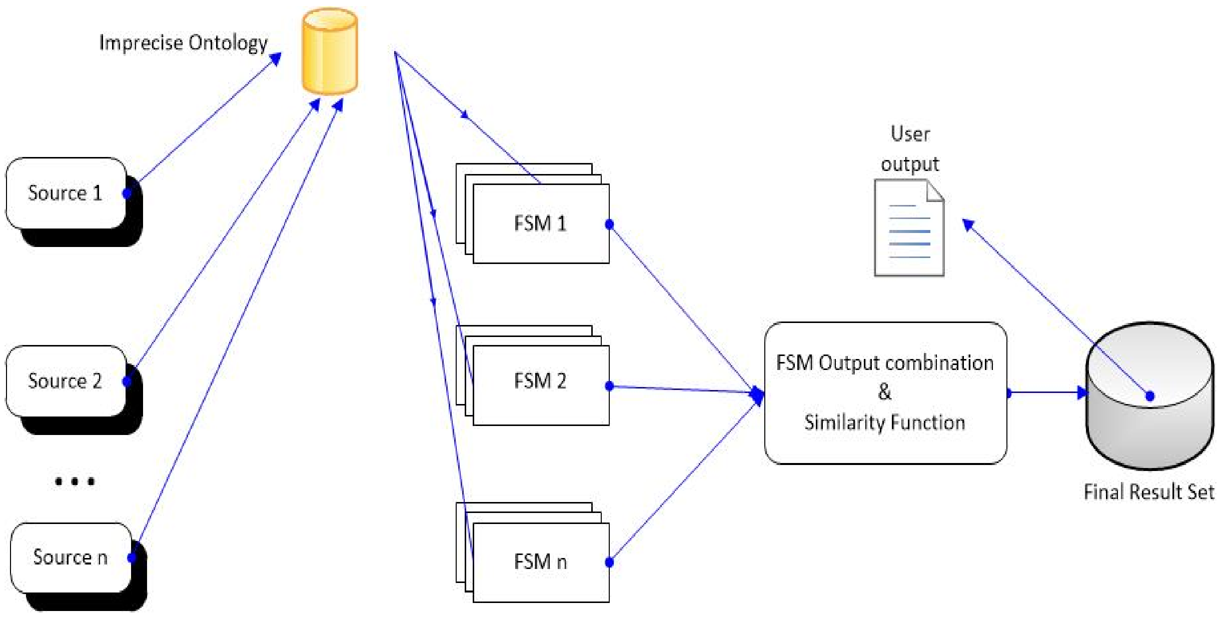

Figure 3: System Architecture

In our case study, we employ FSM modelling by considering uncertainty (except from vagueness) as part of an FSM rule. More precisely, each source (recommendation site) i, $i = \{1, 2, \ldots, n\}$ is assigned an FSM_i, in order to classify the sources of information (i.e the elements), based on the criteria defined by the user. Each FSM contains the same set of criteria, with any other FSM. The criteria are formalized in order to take a rule format. Following this, they are inserted into each FSM_i. Each FSM_i processes the input from $source_i$ based on the user defined criteria. The output of each FSM_i is inserted in our combination module, in order to derive a combined result. The combination is preformed based on Dempster's Rule. More precisely, if D is a fuzzy concept of interest (for example *proposed hotel*) and d is the output of each FSM, i.e the degree d to which a hotel is proposed, then the following frame of discernment is defined:

$$\mathcal{W} = \{D, \overline{D}\}$$

where $\overline{D}$ represents *not D*. Following, a basic probability assignment, m is assigned to the powerset of the frame of discernment as follows:

$$m(\{D\}) = d$$
$$m(\{D, \overline{D}\}) = 1 - d$$

Following, the Combined Firing Level of two FSMs with degrees d_1, d_2 and basic probability assignments m_1, m_2, respectively, is defined through Dempster's Rule of Combination as follows:

$$combined_{1,2}(\{D\}) = \frac{\sum_{x,y \in \mathcal{W}: x \cap y = D} m_1(x) \times m_2(y)}{1 - \sum_{x,y \in \mathcal{W}: x \cap y = \emptyset} m_1(x) \times m_2(y)}$$

A key component of the whole approach is the imprecision ontology. Our imprecision ontology is composed of a set of terms suitable for representing imprecise information. These terms actually represent statements of the form a : C = b, where a is a named individual, C is a Concept name and b is the combined confidence degree of a is C. These statements can be regarded as a special case of Dempster-Shafer Fuzzy OWL axioms, defined in 5.5, where the condition is always an equality (rather than the general case $\geq$). Moreover, our ontology serves as a repository for constraints and rules definition. This ontology contains a class named $Element$. Each element from the data set is an individual of this class with a confidence degree value. Moreover, a data property $hasConfidenceDegree$ is defined, having domain the class $Element$ and range in $[0, 1]$. In addition, there exist a class named $Constraint$, that contains the imprecise criteria defined by the user, e.g close to metro station, expensive etc. Our system architecture is depicted in Fig 3.

When dealing with different sources, before applying our combined degree function, we have to define the concept of identical individuals. For example, suppose that two sources contain data about hotels. Each hotel is identified by its name. So, a method for defining that two hotels are identical should exist, i.e. if hotel with name *h1* is the same with hotel with name *h2*. This means that a *similarity function* should be applied. In our method, the similarity of an individual with another individual, both defined as sets of characters, is defined as follows:

Element e:it is defined as a set of characters

Element $e1$ equals $e2$ iff $e1 = e2$

or $e1 \subseteq e2$

or $e2 \subseteq e1$

As we have previously stated, in our case study, we have considered a set of recommendation sites as different sources and as sources' elements the set of hotels existing in the site. Each element is described by a set of attributes. The attribute values come from different sources of information. The attributes may or may not exist in all sources. Our goal is to define a confidence degree for proposed for each element based on these attributes. These attributes appear in the FSM criteria. The attributes are defined either as vague concepts, for example *low rating* or *expensive hotel*, or as crisp values, for example *3-star hotel*. A membership function is used to define *vagueness factor* and *uncertainty factor* of each concept. The criteria are used in order to define a *hotel score*. We have considered that *hotel score* can take two values, namely *proposed* and *not proposed*, which actually can be seen a two different vague concepts. In Fig 4, the membership functions of these two concepts are depicted. Our implementation uses JFuzzyLogic, a Java library for defining fuzzy inference systems [31].

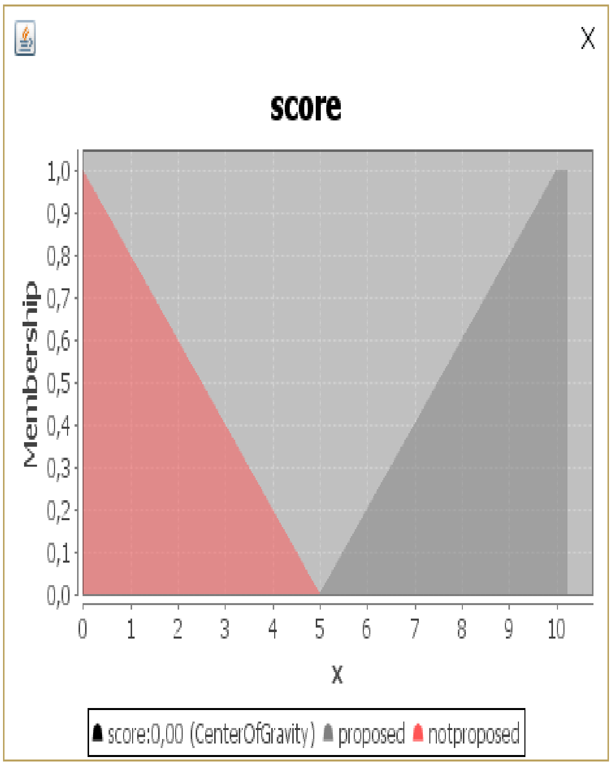

Figure 4: Score Membership Functions

In order to test our system we have selected three hotel recommendation sites, namely, *www.booking.com* (*source1*), *www.hotelclub.com* (*source2*) and *www.easytobook.com* (*source3*). The initial set for the first source included $2,381$ hotels. Similarly, for the second source we have a set of 834 hotels, whereas for the third a set of 546. Each site provides us with different kind of information, more precisely:

www.booking.com:

- *Hotel Name*: The hotel's name
- *Hotel Review*: A numerical value as a result of how proposed the hotel is, based on a set of reviews
- *Hotel Price*: The cost per night
- *Hotel Likes*: The number of people that have added this hotel in a wish list

www.hotelclub.com

- *Hotel Name*: The hotel's name
- *Hotel Lat*: Latitude of the hotel
- *Hotel Lng*: Longitude of the hotel

- *Hotel Stars*:The categorization of the hotel based on *stars* classification
- *Hotel Review*:A numerical value as a result of how proposed the hotel is, based on a set of reviews
- *Hotel Price*: The cost per night

www.easytobook.com

- *Hotel Name*: The hotel's name
- *Hotel Review*:A numerical value as a result of how proposed the hotel is, based on a set of reviews
- *Hotel Price*: The cost per night

In our implementation, we have defined the vague concepts *close*[1], *many likes*, *good review* and *cheap* by applying a set of membership functions.

Our goal is to test our system with different queries in order to evaluate the results. The queries defined for the evaluation are the following:

- **Query 1**→ *Hotel that is close to a metro station and is cheap*: This query demands the *Hotel Lat* and *Hotel Lng* information, existed in *www.hotelclub.com* and *Hotel Price*.
- **Query 2**→ *Hotel that provides a swimming pool and is cheap*: This query demands the *Hotel Stars* existed in the second site information and *Hotel Price*. The *Hotel Stars* provided the *swimming pool* information with a *degree of uncertainty*.
- **Query 3**→ *Hotel that has been classified as favourite and is cheap*: This query demands the *Hotel Likes* information existed in *www.booking.com* and *Hotel Price*.
- **Query 4**→ *Hotel that has good review score and is cheap*: This query demands the *Hotel Review* and *Hotel Price* information existed in both sites.

Our approach adopts an ontology for representing imprecise data. In Fig 5 the classes of our ontology are depicted. As we can see in this figure our ontology serves as a way to store and manage the information considering hotel elements as well as the set of queries.

After classifying the hotels using the fuzzy criteria, we reduced this set, as we will see next, eliminating the hotels that have been assigned confidence degree equal to 0.0. The reduction of the initial set can be proved to be very useful in Big Data situations, a very

[1] For calculating distance based on longitude and latitude metrics a *distance function* has been applied.

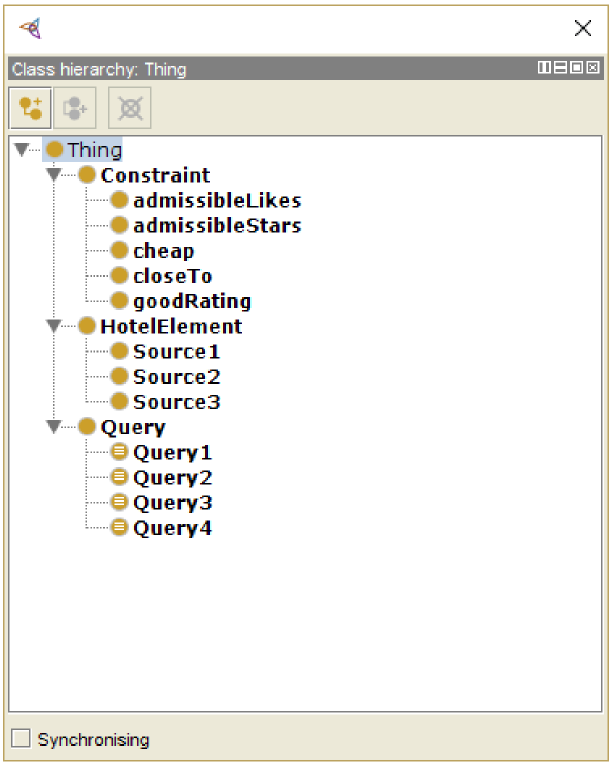

Figure 5: Hotel Ontology

common issue in World Wide Web. This means that our classifier can be used in Big Data environments, in order to build the consistent set of data, as this is defined in [29] and [30].

The reduced set is derived as follows. At first, the hotels that have been assigned a degree of 0.0 are excluded immediately. Next, a *cutoff* value, for example a degree > 7.0 or > 8.0, is used to do a further reduction. The choice is regarded as a tradeoff between *data volume* and *precision results*.

Using the queries described above, we have the following results:

1. Query 1: In this query the first source was reduced into $1,381$ (58% of the initial set) data elements, the second into 267 (32% of the initial set), and the third derived 251 (45% of the initial set) elements.

2. Query 2: In this query the first and third source derive the same result, as they do not provide the *stars* information, whereas the second derives a set of 215 (26% of the initial set) hotels.

3. Query 3: In this query the first source derives a result of 238 (10% of the initial set) , whereas the second gives a result of 280 (34% of the initial data set) hotels. The third derives the same result as in *Query 1* and *Query 2*.

4. Query 4: In this query the first source derives a result of 504 (21% of the initial data set), the second a result of 263 (31% of the initial data set) hotels and the third a

result of 335 (61% of the initial data set) hotels.

Following, we proceeded by combining *source1* and *source2*. These two sites can be considered as independent sources of information and, as such, the *Dempster's rule of Combination* can be applied. The *combined confidence degrees*, which constitute the output of this phase, serve as an input to be combined with the values of *source3*. As we have stated, we apply Dempster's rule of Combination, in order to combine the various confidence degrees. The output of our system, i.e. the *combined confidence degrees*, can be regarded as membership values, denoting *how much proposed the hotel is*.

Our approach provides a means to classify web data based on user-defined criteria. Moreover, the criteria defined have the flexibility to be vague. We employ well-founded Fuzzy Logic techniques, through Fuzzy System Modelling, in order to represent vagueness. We also consider uncertainty as a result of missing information or, in other words, ignorance. In our method, we tackle uncertainty and vagueness issues under a common framework, by employing fuzzy sets for this purpose. A necessary precondition of our method is data independence, when combination through Dempster's Rule is considered. This is achieved through the selection of different data sources which we consider as independent.

6.3 A Matchmaking Case Study

Matchmaking problems [67] can be considered as ontology applications for the Semantic Web. As already stated in 3.3, in its typical form, a matchmaking problem consists of two groups, denoted as "sellers" and "buyers". Each seller and buyer defines a set of constraints, as *requirements* and *preferences*. A very common situation in constraint setting is the vagueness that describe them [19, 68]. As an example let us consider a job recruitment process, with the following constraints:

- *Job Seeker Constraints*:
  - Job with salary no less than $25,000$ per annum
  - Ideal job salary $30,000$ per annum
- *Job Advertisement Constraints*:
  - Job with salary no more than $26,000$ per annum
  - Ideal job salary $23,000$ per annum

The constraints are defined in a way that an ideal value exists and as the value increases or decreases the satisfaction of the seeker/recruiter goes down. In a formal way, as already stated in 3.3, the constraints are defined through the following membership functions:

$$\mu_{Seeker}(x) = \left\{ \begin{array}{ll} 0, & \text{for } 0 \leq x \leq 25000 \\ \frac{x-25000}{5000}, & \text{for } 25000 \leq x \leq 30000 \\ 1, & \text{for } 30000 \leq x \end{array} \right\}$$

$$\mu_{Advertisement}(x) = \left\{ \begin{array}{ll} 1, & \text{for } 0 \leq x \leq 23000 \\ \frac{26000-x}{3000}, & \text{for } 23000 \leq x \leq 26000 \\ 0, & \text{for } 26000 \leq x \end{array} \right\}$$

The first function is called a *right-shoulder membership function*, whereas the second is called a *left-shoulder membership function*. So, the Job Advertisement salary constraint is represented through the *left-shoulder* membership function, with its value denoted as f_1, whereas the the Job Seeker constraint is represented through the *right-shoulder* membership function, with its value denoted as f_2.

This means that for a job individual, j, we have the following axioms:

$$\tau 1_{fuzzy} :< j \quad f_1 >$$
$$\tau 2_{fuzzy} < j \quad f_2 >$$

Also, we define a set of weights considering Seeker and Advertisement requirements, in a way similar to [19], denoting the *credibility* of the Seeker and Advertisement. This is defined by regarding the constraints as a set $\mathcal{C}$ of the following form:

$$\mathcal{C} = \{s_1, \ldots, s_k, a_1, \ldots, a_l\}$$

with s_i denoting a Seeker constraint and a_i denoting an Advertisement constraint. Then, weights are defined through a basic probability assignment m_{weight}:

$$m_{weight} : 2^{\mathcal{C}} \rightarrow [0, 1]$$

In our case study, we have the following set:

$$\mathcal{C} = \{s_1, a_1\}$$

as we have one Seeker and one Advertisement constraint.

The fuzziness describing constraints along with weights definitions pave the way for the application of our Dempster-Shafer Fuzzy DL in the matchmaking procedure.

If we consider Seeker and Advertisement as two fuzzy interpretations, $\mathcal{I}_{Seeker}$ and $\mathcal{I}_{Adverstisement}$, each of them being a model of $\tau 1_{fuzzy}$ and $\tau 2_{fuzzy}$, as follows:

$$\mathcal{I}_{Seeker} \models \tau 1_{fuzzy}$$
$$\mathcal{I}_{Advertisement} \models \tau 2_{fuzzy}$$

Formally, these two interpretations are represented as:

$$\mathcal{I}_{Seeker} = \{\Delta^{\mathcal{I}_{Seeker}}, \cdot^{\mathcal{I}_{Seeker}}\}$$
$$\mathcal{I}_{Advertisement} = \{\Delta^{\mathcal{I}_{Advertisement}}, \cdot^{\mathcal{I}_{Advertisement}}\}$$

where

$$\Delta^{\mathcal{I}_{Seeker}} \equiv \Delta^{\mathcal{I}_{Advertisement}} \equiv \mathbb{N}$$

and $\cdot^{\mathcal{I}_{Seeker}}$, $\cdot^{\mathcal{I}_{Advertisement}}$ are defined based on the membership functions μ_{Seeker} and $\mu_{Advertisement}$. Also, $\mathbb{N}$ is the Salary value domain.

Now, let us suppose that we have a job posting, with salary $25,000$. This, in an ontology context, is represented as an individual $j : Job \sqcap (hasSalary.\{25000\})$.

Also, let have weights of 0.8 for the Seeker and 0.2 for the Advertisement formally represented as $m_{weight}(\{s_1\}) = 0.8$ and $m_{weight}(\{a_1\}) = 0.2$. These weights have been arbitrarily chosen in order to describe our case example.

Our goal is to compute a *matchmaking degree* that depicts the satisfaction value of the job individual, based on fuzzy constraints and uncertainty. In order to do this, we consider the following fuzzy axiom:

$$\tau_{fuzzy} :< j \quad f >$$

where f is defined as $min\{f_1, f_2\}$.

Then, $\mathcal{I}_{Seeker}$ and $\mathcal{I}_{Adverstisement}$ are models of

$$\tau_{fuzzy} :< j \quad f >$$

i.e, $\mathcal{I}_{Seeker} \models \tau_{fuzzy}$ and $\mathcal{I}_{Advertisement} \models \tau_{fuzzy}$.

To sum up, each job individual is related to two fuzzy constraint degrees:

- Fuzzy constraint degree of Seeker
- Fuzzy constraint degree of Advertisement

Also, by considering a basic probability assignment modelled as weights on Seeker and Advertisement constraints, we have each fuzzy degree associated to a mass degree. This, paves the way for the application of our *Dempster-Shafer DL*.

Following, we overview our matchmaking ontology, depicted in Fig 6. Our ontology is based on the one defined in [68].

In order to represent our world, we consider the following classes:

- *Job Seeker*: It is defined as an OWL class and represents the part who searches for a job position
- *Job Advertisement*: It is defined as an OWL class and represents the part who posts a job position
- *Job*: It is defined as an OWL class and represents the jobs of interest
- *Crisp*: It is defined as OWL class and relates a Job individual with a salary value and an uncertainty value
- *LeftShoulderConstraint*: It is defined as an OWL class and defines a Left-Shoulder membership function
- *RightShoulderConstraint*: It is defined as an OWL class and defines a Right-Shoulder membership function

The constraints are defined in the following way:

$$JobSeeker \equiv Job \sqcap$$
$$hasSalary.RightShoulderConstraint$$

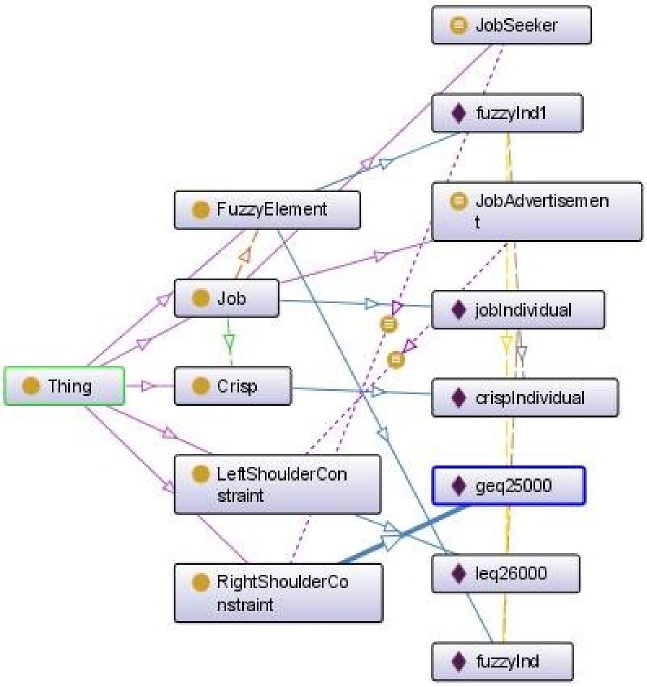

Figure 6: Matchmaking Ontology

$$JobAdvertisement \equiv Job \sqcap$$
$$hasSalary.LeftShoulderConstraint$$

Each LeftShoulderConstraint and RightShoulderConstraint is related to a fuzzy degree through the *hasFuzzy* data property, whereas each job individual is related to an uncertainty degree through the *hasUncertainty* data property.

The matchmaking processing derives a *matchmaking degree factor*, defined as:

$$Matchmaking \equiv JobSeeker \sqcap JobAdvertisement$$

This *degree factor* is a combination of *fuzzy constraints* and *uncertainty* of salary crisp value.

Considering the constraints, we define for each salary crisp value a membership degree for each constraint. A rule variable is defined by the symbol $?x$, where x is a user defined variable. The Left Shoulder Constraint is defined through the following set of rules:

$$
\begin{aligned}
&LeftShoulderConstraint(?l), hasCrisp(?j, ?c),\\
&hasElement(?f, ?l),\\
&hasFuzzy(?j, ?f), hasIdealValue(?l, ?i),\\
&hasThresholdValue(?l, ?t),\\
&hasValue(?c, ?v), lessThanOrEqual(?v, ?i)\\
&- > hasFuzzyFactor(?f, 1.0)
\end{aligned}
$$

$$
\begin{aligned}
&LeftShoulderConstraint(?l), hasCrisp(?j, ?c),\\
&hasElement(?f, ?l),\\
&hasFuzzy(?j, ?f), hasIdealValue(?l, ?i),\\
&hasThresholdValue(?l, ?t),\\
&hasValue(?c, ?v), greaterThanOrEqual(?v, ?t)\\
&- > hasFuzzyFactor(?f, 0.0)
\end{aligned}
$$

$$
\begin{aligned}
&LeftShoulderConstraint(?l), hasCrisp(?j, ?c),\\
&hasElement(?f, ?l),\\
&hasFuzzy(?j, ?f), hasIdealValue(?l, ?i),\\
&hasThresholdValue(?l, ?t),\\
&hasValue(?c, ?v), divide(?d, ?s1, ?s2),\\
&greaterThan(?v, ?i),\\
&lessThan(?v, ?t), subtract(?s1, ?t, ?v),\\
&subtract(?s2, ?t, ?i)\\
&- > hasFuzzyFactor(?f, ?d)
\end{aligned}
$$

A Right Shoulder Constraint is defined in an analogous way.

We model the Belief Degree of the job individual through the following rule:

$$\begin{aligned}
&FuzzyElement(?f1), FuzzyElement(?f2),\\
&LeftShoulderConstraint(?l),\\
&RightShoulderConstraint(?r), hasCrisp(?j, ?c),\\
&hasElement(?f1, ?l),\\
&hasElement(?f2, ?r), hasFuzzy(?j, ?f1),\\
&hasFuzzy(?j, ?f2), hasFuzzyFactor(?f1, ?fa1),\\
&hasFuzzyFactor(?f2, ?fa2), hasWeight(?f1, ?u1),\\
&hasWeight(?f2, ?u2), add(?s, ?m1, ?m2),\\
&multiply(?m1, ?fa1, ?u1),\\
&multiply(?m2, ?fa2, ?u2) - > hasBel(?j, ?s)
\end{aligned}$$

In our case example, we derive a matchmaking degree of 0.12 for the job posting. This value is the *belief degree* of *Job* $?j$ being a job that matches Seeker and Advertisement constraints. The derivation of the belief degree value comes as a result of our rules definition.

The matchmaking ontology is implemented in Protégé. In addition, the rules were defined by employing the *rules plugin* provided. So,the whole process, i.e ontology representation and reasoning is performed through an integrated tool, in an efficient way.

In addition, we have developed a matchmaking application for job recruitment, integrating fuzzy logic, Dempster-Shafer and ontologies, which is presented in [68]. In this application, the *Seeker* and *Advertisement preferences* are represented as concepts and roles in our ontological model. A set of data (job postings) is considered as a real-world case example of our method. In order to draw a matchmaking degree, a set of rules has also been defined.

7. CONCLUSIONS AND FUTURE WORK

In our thesis, we defined an approach for representing uncertainty and vagueness under a common framework in a Semantic Web environment. In order to represent uncertainty we employed Dempster-Shafer model. Vagueness has been represented through Fuzzy Logic and Fuzzy Sets. At first, we examined our problem though an ontological point of view. Thus, we implemented suitable semantic web ontologies for capturing imperfect concepts. Following, for establishing our theoretical framework, we combined the classical crisp DL $\mathcal{ALC}$ with a Dempster-Shafer module. Next, we have proceeded by adding fuzziness in this model. Throughout our work, we formally defined the syntax and the semantics and examined decidability and complexity issues. At the same time, the framework has been applied to some case studies and a real-world application of our method, producing very satisfactory results.

The main advantage of our method resides on the fact that we do not tackle uncertainty and vagueness as independent notions. This representation is in accordance with real-world applications, since very often uncertainty and vagueness coexist. The Dempster-Shafer model has been proven to be an ideal framework for representing estimations, since it models a world in a way similar to human thinking, in cases of reasoning.

In addition, our theoretical framework has been built upon $\mathcal{ALC}$, a well-established DL. Our syntax has been defined as an extension of $\mathcal{ALC}$ syntax. Vagueness is represented through Zadeh's Fuzzy Logic, by considering membership degree conditions on crisp $\mathcal{ALC}$ axioms. In addition, we employ Dempster-Shafer theory for representing the uncertainty part. In order to employ this theory, we have defined belief degree conditions. The notion of *possible world* has an important role in defining the semantics of our framework. More precisely, we have regarded the set of possible worlds as a frame of discernment and defined mass functions on subsets of this set. As a final step, we have considered the combination of statements from different Knowledge Bases, by employing our Combined Dempster-Shafer entailment, an entailment method based on Dempster's Rule of Combination.

More precisely, in 1.2.2 the main achievements of our dissertation are summarized:

- *We defined ontologies that capture uncertain and vague concepts:* The idea was employed, firstly, in a methodology that operates on probabilistic knowledge bases and employs Dempster's rule of Combination. Dempster-Shafer Theory is an ideal framework for representing information incompleteness. Towards this, the combination of various results derived from probabilistic knowledge bases can be represented through a Dempster-Shafer frame of discernment and next, we proceeded by performing combination based on Dempster's rule. In another application of ontological representation, we considered uncertain and vague concepts by representing them through an *imperfection factor*. These statements can be either a rule or an event statement. Next, we proceeded by combining these statements for deriving a new statement annotated with a combined imperfection factor. The approach has also been employed in Big Data environments.

- *We defined an extension of a crisp DL with Belief Plausibility Degrees:* To delve into the theoretical framework of our thesis, we considered the simple and widely known DL $\mathcal{ALC}$. Based on this DL, we extended thoroughly the syntax and semantics in order to define a DL for representing uncertain information, named Dempster-Shafer DL ($\mathcal{DS} - \mathcal{ALC}$). Since we employed the Dempster-Shafer framework, a set of constraints that stem from this theory should be preserved. More precisely, the possible world concept ensured evidence independency and the set of possible worlds served as a frame of discernment. Next, we defined a new notion of entailment, based on Dempster's rule of Combination, denoted as $\models_{DScombined}$. In addition, as in the crisp case, we considered decidability and complexity issues.

- *We defined an extension of a fuzzy DL with Belief Degrees:* Finally, we defined a framework that combined Dempster-Shafer and Fuzzy Logic for handling information imperfection aiming at the Semantic Web. In order to do this, we defined suitable logic representation formalisms. More precisely, firstly, we defined a Dempster-Shafer Fuzzy Knowledge Base, in order to represent uncertainty in a fuzzy logical framework. We extended a crisp Knowledge Base along with Fuzziness, providing a model for representing vague information annotated with incomplete information. Our model, denoted as a Dempster-Shafer Fuzzy Description Logic, is actually a DL for that described uncertainty and vagueness under a common framework. In order to do this, we considered fuzzy $\mathcal{ALC}$ assertions that can be entailed by a set of DL interpretations, using a basic probability assignment defined on these sets. Decidability and reasoning issues have also been addressed as a way to show the practical feasibility of our model.

Regarding applications, we tested our method in two fields:

- Recommender systems: These environments have been proven to be an ideal framework of our method, since they combine vagueness with uncertainty. In addition, the various recommendation sites serve as independent sources of information. Thus, the Dempster-Shafer framework along with Dempster's Rule of Combination were efficiently employed. In order to represent information, we employed well-founded ontological tools combined with Fuzzy Systems Modelling procedures. Finally, recall and precision measurements have shown that our method produced satisfactory results.

- Matchmaking systems: We developed a matchmaking application for job recruitment situations, integrating fuzzy logic, Dempster-Shafer and ontologies,. In this application, the Seeker and Advertisement preferences were represented as concepts and roles in our ontological model. In order to draw a matchmaking degree, a set of rules has also been defined

Future work The Dempster-Shafer framework was proven to be an ideal one for representing ignorance. Although it has many advantages, the complexity of the rule of Combination along with conflicts' modelling remains an issue to be tackled for representing real world case studies. As a future work, we will consider complexity and decidability issues more thoroughly, mostly aiming at Dempster's rule evaluation performance. In [111], other formulas for combining evidence are outlined. These formulas provide for lower complexity. Thus, the adaptation of these formulas in a DL environment can serve as a way to gain better complexity.

We shall also consider Big Data environments in a more thorough framework. Although we examine some Big Data issues through our dissertation, we do not consider some well known algorithms such as the one defined in [33]. As a future work, we will focus on the application of our model in a Big Data environment.

Another area of future work resides in the expressiveness level. As our dissertation has been defined upon DL $\mathcal{ALC}$, we may consider the extension of other DLs. Apart from fuzzy $\mathcal{ALC}$, other fuzzy extensions are described in [118], [84], [96], [121], [119],[120]. Moreover, although $\mathcal{ALC}$ is the basic DL, in cases of Semantic Web, a set of other DLs is usually employed, namely, $\mathcal{SROIQ}(D)$ [59], $\mathcal{SHOIN}$ [63] and $\mathcal{SHIF}$ [65]. So it will be useful to extend our framework to these DLs. In addition, for representing vagueness, we employed Zadeh's Fuzzy Logic. In future, we will consider other Fuzzy Logics as described in Table 4.

In cases of strongly conflicting evidence, Dempster's Rule produces counter-intuitive examples. Towards this, other rules have been proposed [111]:

- The Discount and Combine method
- Yager's modified Dempster's Rule
- Inagaki's modified Dempster's Rule
- Zhang's Center Combination Rule

As a future work, we may consider the combination of evidence based on some of these rules.

In the area of applicability, case studies other than recommender systems and matchmaking environments can be examined. Some of them are:

- Semantic annotation
- Information extraction
- Ontology alignment
- Representation of background knowledge

These fields are described in [84] as some of the most representative ones of Semantic Web applications.

ABBREVIATIONS - ACRONYMS

RDF	Resource Description Framework
SPARQL	SPARQL Protocol and RDF Query Language
OWL	Web Ontology Language

REFERENCES

[1] Paulo Armando Cavalcante Aguilar, Jerome Boudy, Dan Istrate, Hamid Medjahed, Bernadette Dorizzi, Joao Cesar Moura Mota, Jean Louis Baldinger, Toufik Guettari, and Imad Belfeki. Evidential Network-Based Multimodal Fusion for Fall Detection. *International Journal of E-Health and Medical Communications (IJEHMC)*, 4(1):46–60, 2013.

[2] M. R. Ahmadzadeh and Maria Petrou. Use of Dempster-Shafer Theory to Combine Classifiers which Use Different Class Boundaries. *Pattern Analysis Application*, 6:41–46, 2003.

[3] Jesus Alcalá-Fdez and Jose M. Alonso. A Survey of Fuzzy Systems Software: Taxonomy, Current Research Trends, and Prospects. *IEEE Transactions on Fuzzy Systems*, 24(1):40–56, Feb 2016.

[4] Grigoris Antoniou, Sotiris Batsakis, and Ilias Tachmazidis. Large-Scale Reasoning with (Semantic) Data. In *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14)*, WIMS '14, pages 1:1–1:3, New York, NY, USA, 2014. ACM.

[5] Grigoris Antoniou and Frank van Harmelen. *A Semantic Web Primer, 2nd Edition*. The MIT Press, 2008.

[6] Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyaschev. The DL-Lite Family and Relations. *Journal of Artificial Intelligence Research*, 36(1):1–69, September 2009.

[7] Lofti A.Zadeh. *The Concept of a Linguistic Variable and its Application to Approximate Reasoning*, pages 1–10. Springer US, Boston, MA, 1974.

[8] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the EL Envelope. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, IJCAI'05, pages 364–369, San Francisco, CA, USA, 2005. Morgan Kaufmann Publishers Inc.

[9] Franz Baader, Ian Horrocks, and Ulrike Sattler. Description Logics as Ontology Languages for the Semantic Web. In *Festschrift in honor of Jörg Siekmann, Lecture Notes in Artificial Intelligence*, pages 228–248. Springer-Verlag, 2003.

[10] Franz Baader, Ian Horrocks, and Ulrike Sattler. Description Logics. In Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, editors, *Handbook of Knowledge Representation*, chapter 3, pages 135–180. Elsevier, 2008.

[11] Ying Bai and Dali Wang. *Fundamentals of Fuzzy Logic Control — Fuzzy Sets, Fuzzy Rules and Defuzzifications*, pages 17–36. Springer London, London, 2006.

[12] J. Douglas Barret and William H. Woodall. A Probabilistic Alternative to Fuzzy Logic Controllers. *IIE Transactions*, 29(6):459–467, June 1997.

[13] Mathias Bauer. Approximations for Decision Making in the Dempster-Shafer Theory of Evidence. In *UAI'96*, pages 73–80, San Francisco, CA, USA, 1996. Morgan Kaufmann Publishers Inc.

[14] Amandine Bellenger and Sylvain Gatepaille. Uncertainty in Ontologies: Dempster-Shafer Theory for Data Fusion Applications. *in Workshop on the Theory of Belief Functions, Brest-France*, 2010.

[15] Tim Berners-Lee and Mark Fischetti. *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. Harper San Francisco, 1st edition, 1999.

[16] Fernando Bobillo, Miguel Delgado, Juan Gómez-Romero, and Umberto Straccia. Fuzzy Decription Logics Under Gödel Semantics. *International Journal Approximate Reasoning*, 50:494–514, 2009.

[17] Fernando Bobillo and Umberto Straccia. An OWL Ontology for Fuzzy OWL 2. In *Proceedings of the 18th International Symposium on Foundations of Intelligent Systems*, ISMIS '09, pages 151–160, Berlin, Heidelberg, 2009. Springer-Verlag.

[18] Fernando Bobillo and Umberto Straccia. Representing Fuzzy Ontologies in OWL 2. In *WCCI 2010 IEEE World Congress on Computational Intelligence*, WCCI 2010, pages 2695–2700, 2010.

[19] Fernando Bobillo and Umberto Straccia. Fuzzy Ontology Representation Using OWL 2. *International Journal Approximate Reasoning*, 52(7):1073–1094, October 2011.

[20] Fernando Bobillo and Umberto Straccia. Generalized Fuzzy Rough Description Logics. *Information Sciences*, 189:43 – 62, 2012.

[21] Fernando Bobillo and Umberto Straccia. The Fuzzy Ontology Reasoner fuzzyDL. *Knowledge Based Systems*, 95(C):12–34, March 2016.

[22] Ronald J. Brachman and Hector J. Levesque. The Tractability of Subsumption in Frame-Based Description Languages. In *Proceedings of the Fourth AAAI Conference on Artificial Intelligence*, AAAI'84, pages 34–37. AAAI Press, 1984.
[23] Ronald J. Brachman and James G. Schmolze. An Overview of the KL-ONE Knowledge Representation System. *Cognitive Science*, 9(2):171 – 216, 1985.
[24] Martin Buchheit, Francesco M. Donini, and Andrea Schaerf. Decidable Reasoning in Terminological Knowledge Representation Systems. Technical report, DFKI, 1993.
[25] Silvia Calegari and Elie Sanchez. A Fuzzy Ontology-Approach to Improve Semantic Information Retrieval. In *Proceedings of the Third ISWC Workshop on Uncertainty Reasoning for the Semantic Web Busan, Korea, November 12, 2007*, 2007.
[26] Andrea Calì and Thomas Lukasiewicz. An Approach to Probabilistic Data Integration for the Semantic Web. In *Uncertainty Reasoning for the Semantic Web I*, pages 52–65, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
[27] Andrea Calì, Thomas Lukasiewicz, Livia Predoiu, and Heiner Stuckenschmidt. Rule-Based Approaches for Representing Probabilistic Ontology Mappings. In *Uncertainty Reasoning for the Semantic Web I*, pages 66–87, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
[28] Jianhua Chen and Sukhamay Kundu. A Sound and Complete Fuzzy Logic System Using Zadeh's Implication Operator. In *Foundations of Intelligent Systems*, pages 233–242, Berlin, Heidelberg, 1996. Springer Berlin Heidelberg.
[29] Alexey Cheptsov. Semantic Web Reasoning on the Internet Scale With Large Knowldege Collider. *International Journal of Computer Science and Applications, ©Technomathematics Research Foundation*, 2011.
[30] Alexey Cheptsov and Zhisheng Huang. Making Web-Scale Semantic Reasoning More Service-Oriented: The Large Knowledge Collider. In *Web Information Systems Engineering - WISE 2011 and 2012 Workshops - Combined WISE 2011 and WISE 2012 Workshops, Sydney Australia, October 12-14, 2011 and Paphos, Cyprus, November 28-30, 2012, Revised Selected Papers*, pages 13–26, 2012.
[31] Pablo Cingolani and Jesús Alcalá-Fdez. jFuzzyLogic: A Robust and Flexible Fuzzy-Logic Inference System Language Implementation. In *Proceedings of the IEEE International Conference on Fuzzy Systems*, pages 1–8. IEEE, 2012.
[32] 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, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
[33] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM*, 51(1):107–113, January 2008.
[34] Steven Decker, Sergey Melnik, and Frank van Harmelen. The Semantic Web: The Roles of XML and RDF. *IEEE Internet Computing*, 15(3):63–74, 2000.
[35] Arthur P. Dempster. *A Generalization of Bayesian Inference*, pages 73–104. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
[36] Zhongli Ding and Yun Peng. A Probabilistic Extension to Ontology Language OWL. In *Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences - Track 4 - Volume 4*, HICSS '04, Washington, DC, USA, 2004. IEEE Computer Society.
[37] Didier Dubois. Uncertainty Theories: A Unified View. `www.bris.ac.uk/engineering/media/engineering-mathematics/bridging-the-gap/uncertainty/dubois.pdf`,last checked=2018-08-27, 2007.
[38] Didier Dubois, Lluís Godo, and Henri Prade. Weighted Logics For Artificial Intelligence – An Introductory Discussion. *International Journal of Approximate Reasoning*, 55(9):1819 – 1829, 2014.
[39] Didier Dubois, Jérôme Lang, and Henri Prade. A Brief Overview of Possibilistic Logic. In *Symbolic and Quantitative Approaches to Uncertainty*, pages 53–57. Springer Berlin Heidelberg, 1991.
[40] Didier Dubois, Jérôme Mengin, and Henri Prade. Possibilistic Uncertainty And Fuzzy Features In Description Logic. A Preliminary Discussion. In *Fuzzy Logic and the Semantic Web*, volume 1, pages 101 – 113. Elsevier, 2006.
[41] Didier Dubois and Henri Prade. Fuzzy Sets and Probability : Misunderstandings, Bridges and Gaps. In *In Proceedings of the Second IEEE Conference on Fuzzy Systems*, pages 1059–1068. IEEE, 1993.

[42] Didier Dubois and Henri Prade. Possibility Theory, Probability Theory and Multiple-Valued Logics: A Clarification. *Annals of Mathematics and Artificial Intelligence*, 32(1):35–66, Aug 2001.
[43] Didier Dubois and Henri Prade. Possibilistic Logic: A Retrospective And Prospective View. *Fuzzy Sets and Systems*, 144(1):3 – 23, 2004.
[44] Didier Dubois and Henry Prade. *Possibility Theory and Its Applications: Where Do We Stand?*, pages 31–60. Springer Handbook of Computational Intelligence, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
[45] Ronald Fagin, Joseph Y. Halpern, and Nimrod Megiddo. A Logic For Reasoning About Probabilities. *Information and Computation*, 87(1):78 – 128, 1990. Special Issue: Selections from 1988 IEEE Symposium on Logic in Computer Science.
[46] Brian R. Gaines. Fuzzy And Probability Uncertainty Logics. *Information and Control*, 38(2):154 – 169, 1978.
[47] Rosalba Giugno and Thomas Lukasiewicz. P-$\mathcal{SHOQ}(D)$ for Probabilistic Ontologies in the Semantic Web. In *Logics in Artificial Intelligence*, pages 86–97, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.
[48] Benjamin N. Grosof, Ian Horrocks, Raphael Volz, and Stefan Decker. Description Logic Programs: Combining Logic Programs with Description Logic. In *Proceedings of the 12th International Conference on World Wide Web*, WWW, pages 48–57, New York, NY, USA, 2003. ACM.
[49] Víctor Gutiérrez-Basulto, Jean Christoph Jung, Carsten Lutz, and Lutz Schröder. A Closer Look at the Probabilistic Description Logic Prob-EL. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI, San Francisco, California, USA, August 7-11*, 2011.
[50] Volker Haarslev and Ralf Möller. RACER System Description. In Rajeev Goré, Alexander Leitsch, and Tobias Nipkow, editors, *Automated Reasoning*, pages 701–705, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.
[51] Joseph Y. Halpern. *Reasoning about uncertainty*. MIT Press, Cambridge, Mass., London, 2003.
[52] Jochen Heinsohn. Probabilistic Description Logics. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, UAI'94, pages 311–318, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers Inc.
[53] Jon C. Helton. Uncertainty and Sensitivity Analysis in the Presence of Stochastic and Subjective Uncertainty. *Journal of Statistical Computation and Simulation*, 57(1-4):3–76, 1997.
[54] Bernhard Hollunder. An Alternative Proof Method for Possibilistic Logic and Its Application to Terminological Logics. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, UAI'94, pages 327–335, San Francisco, CA, USA, 1994. Morgan Kaufmann Publishers Inc.
[55] Xin Hong, Chris Nugent, Maurice Mulvenna, Sally McClean, Bryan Scotney, and Steven Devlin. Evidential Fusion of Sensor Data for Activity Recognition in Smart Homes. *Pervasive Mobile Computing*, 5(3):236–252, June 2009.
[56] Matthew Horridge. *A Practical Guide To Building OWL Ontologies With The Protege-OWL Plugin*. University of Manchester, 1 edition, 2004.
[57] Matthew Horridge, Dmitry Tsarkov, and Timothy Redmond. Supporting Early Adoption of OWL 1.1 with Protege-OWL and FaCT. In *Proceedings of the Second OWL Experiences and Directions Workshop (OWLED)*, 2006.
[58] Ian Horrocks. OWL: A Description Logic Based Ontology Language. In *Principles and Practice of Constraint Programming - CP 2005*, pages 5–8, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.
[59] Ian Horrocks, Oliver Kutz, and Ulrike Sattler. The Even More Irresistible $\mathcal{SROIQ}$. In *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'06, pages 57–67. AAAI Press, 2006.
[60] Ian Horrocks, Bijan Parsia, Peter Patel-Schneider, and James Hendler. Semantic Web Architecture: Stack or Two Towers? In *Principles and Practice of Semantic Web Reasoning*, pages 37–41. 2005.
[61] Ian Horrocks and Peter Patel-Schneider. Reducing OWL Entailment to Description Logic Satisfiability. *Web Semantics*, 1(4):345–357, October 2004.
[62] Ian Horrocks and Peter F. Patel-Schneider. *KR and Reasoning on the Semantic Web: OWL*, pages 365–398. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[63] Ian Horrocks, Peter F. Patel-Schneider, and Frank van Harmelen. From $\mathcal{SHIQ}$ and RDF to OWL: The Making of a Web Ontology Language. *Web Semantics: Science, Services and Agents on the World Wide Web*, 1(1):7 – 26, 2003.

[64] Franz Huber and Christoph Schmidt-Petri. *Degrees of Belief*. Springer Publishing Company, Inc., 1st edition, 2009.

[65] Horrocks Ian and Sattler Uwe. A Description Logic with Transitive and Converse Roles Role Hierarchies and Qualifying Number Restrictions. Technical report, 1999.

[66] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender Systems: An Introduction*. Cambridge University Press, New York, NY, USA, 1st edition, 2010.

[67] Manish Joshi, Virendrakumar Bhavsar, and Harold Boley. Knowledge Representation in Matchmaking Applications. In *Advance Knowledge Based Systems, Model, Applications and Research*, pages 29–49. 2010.

[68] Loukia Karanikola and Isambo Karali. A Fuzzy Logic Approach for Reasoning Under Uncertainty and Vagueness - A Matchmaking Case Study. In *2016 2nd International Conference on Information Management (ICIM)*, pages 52–56, May 2016.

[69] Loukia Karanikola and Isambo Karali. Semantic Web and Ignorance: Dempster-Shafer Description Logics. In *Proceedings of the Thirtieth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2017, Marco Island, Florida, USA, May 22-24, 2017.*, pages 68–73, 2017.

[70] Loukia Karanikola and Isambo Karali. Towards a Dempster-Shafer Fuzzy Description Logic - Handling Imprecision in the Semantic Web. *IEEE Transactions on Fuzzy Systems*, Early Access(99):1–1, 2018.

[71] Loukia Karanikola, Isambo Karali, and Sally I. McClean. Uncertainty Reasoning for the Semantic Web based on Dempster-Shafer model. In *4th International Conference on Information, Intelligence, Systems and Applications, IISA 2013, Piraeus, Greece, July 10-12, 2013*, pages 1–4, 2013.

[72] Loukia Karanikola, Isambo Karali, and Sally I. McClean. Uncertainty Reasoning for the ”big data” Semantic Web. In *Proceedings of the 15th IEEE International Conference on Information Reuse and Integration, IRI 2014, Redwood City, CA, USA, August 13-15, 2014*, pages 147–154, 2014.

[73] Vladik Kreinovich, Claude Langrand, and Hung T. Nguyen. Combining Fuzzy and Probabilistic Knowledge Using Belief Functions. `https://pdfs.semanticscholar.org/535d/ea9d2d5fd503392b24f65c2a47d5f55caff0.pdf`,last checked=2018-08-27, 2001.

[74] Sebastian Ryszard. Kruk and W. D. McDaniel. *Semantic Digital Libraries*. Springer Berlin, 2009.

[75] Ralf Küsters and Ralf Molitor. Approximating Most Specific Concepts in Description Logics with Existential Restrictions. In *KI 2001: Advances in Artificial Intelligence*, pages 33–47, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[76] Douglas Laney. 3D Data Management: Controlling Data Volume, Velocity, and Variety. Technical report, META Group, February 2001.

[77] Kenneth J. Laskey and Kathryn Blackmond Laskey. Uncertainty Reasoning for the World Wide Web: Report on the URW3-XG Incubator Group. URSW’08, pages 108–116, Aachen, Germany, Germany, 2008. CEUR-WS.org.

[78] Francesca Alessandra Lisi and Corrado Mencar. Towards Fuzzy Granulation in OWL Ontologies. In *Proceedings of the 30th Italian Conference on Computational Logic, Genova, Italy, July 1-3, 2015.*, pages 144–158, 2015.

[79] Liping Liu and Ronald R. Yager. *Classic Works of the Dempster-Shafer Theory of Belief Functions: An Introduction*, pages 1–34. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[80] V.J. Loginov. Probability Treatment of Zadeh Membership Functions and Their use in Pattern Recognition. *Engineering Cybernetics*, pages 66–69, 1966.

[81] Vincenzo Loia. Fuzzy Ontologies and Fuzzy Markup Language: A Novel Vision in Web Intelligence. In *Advances in Intelligent Web Mastering – 3*, pages 3–10, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[82] Thomas Lukasiewicz. Probabilistic Default Reasoning with Strict and Defeasible Conditional Constraints. `https://pdfs.semanticscholar.org/5c06/be49b83d88e69fd1a77b816aa116790ae17e.pdf?_ga=2.43210341.1535585336.1535320078-13683457.1520182461`,last checked=2018-08-27, 2000.

[83] Thomas Lukasiewicz. Expressive Probabilistic Description Logics. *Artificial Intelligence*, 172(6):852 – 883, 2008.

[84] Thomas Lukasiewicz and Umberto Straccia. Managing Uncertainty and Vagueness in Description Logics for the Semantic Web. *Web Semantics*, 6(4):291–308, November 2008.
[85] Thomas Lukasiewicz and Umberto Straccia. Description Logic Programs Under Probabilistic Uncertainty and Fuzzy Vagueness. *International Journal of Approximate Reasoning*, 50(6):837–853, June 2009.
[86] Carsten Lutz and Lutz Schröder. Probabilistic Description Logics for Subjective Uncertainty. In *Principles of Knowledge Representation and Reasoning (KR 2010)*, pages 393–403. AAAI Press; Menlo Park, CA, 2010.
[87] María J. Martín-Bautista, Daniel Sánchez, María-Amparo Vila, and Henrik. L. Larsen. *Measuring Effectiveness in Fuzzy Information Retrieval*, pages 396–402. Physica-Verlag HD, Heidelberg, 2001.
[88] Serafín Moral and Nic Wilson. Markov Chain Monte-Carlo Algorithms for the Calculation of Dempster-Shafer Belief. In *Proceedings of the Twelfth AAAI National Conference on Artificial Intelligence*, AAAI'94, pages 269–274. AAAI Press, 1994.
[89] John N. Mordeson, Mark J. Wierman, Terry D. Clark, Alex Pham, and Michael A. Redmond. *Yen's Method*, pages 155–162. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
[90] Miklos Nagy, Maria Vargas-Vera, and Enrico Motta. DSSim: Managing Uncertainty on the Semantic Web. Ontology Matching Conference'07, pages 160–169, Aachen, Germany, Germany, 2007. CEUR-WS.org.
[91] Andriy Nikolov, Victoria Uren, Enrico Motta, and Anne de Roeck. Using the Dempster-Shafer Theory of Evidence to Resolve ABox Inconsistencies. In *Uncertainty Reasoning for the Semantic Web I*, volume 5327 of *Lecture Notes in Computer Science*, chapter 9, pages 143–160. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2008.
[92] Nils J. Nilsson. Probabilistic Logic. *Artificial Intelligence*, 28(1):71 – 87, 1986.
[93] Werner Nutt, Francesco M. Donini, Maurizio Lenzerini, and Daniele Nardi. The Complexity of Concept Languages. *Information and Computation*, 134(1):1–58, April 1997.
[94] Rafael P Nyssen. Fuzzy Description Logics. `https://www.inf.unibz.it/~penaloza/FDL.pdf`,last checked=2018-08-27, 2012.
[95] Pekka Orponen. Dempster's Rule of Combination is #P-Complete. *Artificial Intelligence*, 44(1):245 – 253, 1990.
[96] Fernado Bobillo Ortega. *Managing Vagueness in Ontologies*. PhD thesis, Universidad de Granada, 2008.
[97] Jeff Z. Pan. *Description Logics: Reasoning Support for the Semantic Web*. PhD thesis, School of Computer Science, The University of Manchester, 2004.
[98] Judea Pearl. Reasoning with Belief Functions: An Analysis of Compatibility. *International Journal of Approximate Reasoning*, 4:363–389, 1990.
[99] Witold Pedrycz and Fernando Gomide. *Fuzzy Systems Engineering: Toward Human-Centric Computing*. Wiley-IEEE Press, 2007.
[100] Gregory Provan. A Logical Interpretation of Dempster-Shafer Theory, with Application to Visual Recognition. In *Proceedings of the Fifth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-89)*, pages 287–294, Corvallis, Oregon, 1989. AUAI Press.
[101] Gregory M. Provan. A Logic-Based Analysis of Dempster-Shafer Theory. *International Journal of Approximate Reasoning*, 4(5):451 – 495, 1990.
[102] Guilin Qi, Qiu Ji, Jeff Pan, and Jianfeng Du. PossDL – A Possibilistic DL Reasoner for Uncertainty Reasoning and Inconsistency Handling. In *Proc. of the Extended Semantic Web Conference (ESWC2010)*, 2010.
[103] Guilin Qi, Jeff Z. Pan, and Qiu Ji. A Possibilistic Extension of Description Logics. In *Proceedings of DL'07, 2007*.
[104] Victor Raskin and Julia M. Taylor. Fuzziness, Uncertainty, Vagueness, Possibility, and Probability in Natural Language. In *2014 IEEE Conference on Norbert Wiener in the 21st Century (21CW)*, pages 1–6, June 2014.
[105] Andri Riid and Ennu Rüstern. *Interpretability Issues in Fuzzy Modeling*, chapter Transparent Fuzzy Systems in Modelling and Control, pages 452–476. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[106] Fagin Romald and Halpern Joseph Y. Uncertainty, Belief, and Probability. *Computational Intelligence*, 7(3):160–173.

[107] Alessandro Saffiotti. The Knowledge Engineering Review - An AI View Of The Treatment of Uncertainty. `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.412.5398&rep=rep1&type=pdf`,last checked=2018-08-27.

[108] Alessandro Saffiotti. A Belief-Function Logic. In *AAAI*, pages 642–647. AAAI Press / The MIT Press, 1992.

[109] Klaus Schild. A Correspondence Theory for Terminological Logics: Preliminary Report. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence - Volume 1*, IJCAI'91, pages 466–471, San Francisco, CA, USA, 1991. Morgan Kaufmann Publishers Inc.

[110] Manfred Schmidt-Schaub and Gert Smolka. Attributive Concept Descriptions with Complements. *Artificial Intelligence*, 48(1):1–26, February 1991.

[111] Kari Sentz and Scott Ferson. Combination of Evidence in Dempster-Shafer Theory. Technical report, 2002.

[112] Magnes Setnes and Hans Roubos. GA-fuzzy Modeling and Classification: Complexity and Performance. *Trans. Fuz Sys.*, 8(5):509–522, October 2000.

[113] Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, 1976.

[114] Glenn Shafer. Perspectives on the Theory and Practice of Belief Functions. *International Journal of Approximate Reasoning*, 4(5):323 – 362, 1990.

[115] Glenn Shafer and Roger Logan. Implementing Dempster's Rule for Hierarchical Evidence. *Artificial Intelligence*, 33(3):271 – 298, 1987.

[116] Giorgos Stoilos, Giorgos Stamou, and Jeff .Z. Pan. Fuzzy Extensions of OWL: Logical Properties and Reduction to Fuzzy Description Logics. *International Journal of Approximate Reasoning*, 51(6):656 – 679, 2010.

[117] Giorgos Stoilos, Giorgos B. Stamou, Vassilis Tzouvaras, Jeff Z. Pan, and Ian Horrocks. Fuzzy OWL: Uncertainty and the Semantic Web. In *Proceedings of the OWLED*05 Workshop on OWL: Experiences and Directions, Galway, Ireland, November 11-12, 2005*, 2005.

[118] Umberto Straccia. A Fuzzy Description Logic. In *Proccedings of 15th National Conference on Artificial Intelligence*, pages 594–599, Madison,USA, 1998. AAAI-98.

[119] Umberto Straccia. Reasoning Within Fuzzy Description Logics. *Journal of Artificial Intellignece Research*, 14(1):137–166, April 2001.

[120] Umberto Straccia. Transforming Fuzzy Description Logics into Classical Description Logics. In *Logics in Artificial Intelligence*, pages 385–399, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[121] Umberto Straccia. A Fuzzy Description Logic for the Semantic Web. In *Fuzzy Logic and the Semantic Web, Capturing Intelligence, chapter 4*, pages 167–181. Elsevier, 2005.

[122] Umberto Straccia. *Foundations of Fuzzy Logic and Semantic Web Languages*. Chapman & Hall/CRC, 2013.

[123] Umberto Straccia. All About Fuzzy Description Logics and Applications. In *Reasoning Web. Web Logic Rules: 11th International Summer School 2015, Berlin, Germany, July 31- August 4, 2015, Tutorial Lectures.*, pages 1–31. Springer International Publishing, Cham, 2015.

[124] Umberto Straccia. Fuzzy Semantic Web Languages and Beyond. In *Advances in Artificial Intelligence: From Theory to Practice*, pages 3–8, Cham, 2017. Springer International Publishing.

[125] Umberto Straccia and Fernando Bobillo. From Fuzzy to Annotated Semantic Web Languages. In *Reasoning Web: Logical Foundation of Knowledge Graph Construction and Query Answering: 12th International Summer School 2016, Aberdeen, UK, September 5-9, 2016, Tutorial Lectures*, pages 203–240. Springer International Publishing, Cham, 2017.

[126] Thomas M. Strat. Continuous Belief Functions for Evidential Reasoning. In *Proceedings of the Fourth AAAI Conference on Artificial Intelligence*, AAAI'84, pages 308–313. AAAI Press, 1984.

[127] Martin Unold and Christophe Cruz. How to Enrich Description Logics With Fuzziness. *2017 Computing Conference*, pages 51–57, 2017.

[128] Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter. *Handbook of Knowledge Representation*. Elsevier Science, San Diego, USA, 2007.

[129] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview, October 2009.

[130] Hui Wan. *Reasoning About Uncertainty and Correlated Beliefs*. PhD thesis, Stony Brook, NY, USA, 2010. AAI3426527.
[131] Nic Wilson. *Handbook of Defeasible Reasoning and Uncertainty Management Systems: Algorithms for Uncertainty and Defeasible Reasoning*, chapter Algorithms for Dempster-Shafer Theory, pages 421–475. Springer Netherlands, Dordrecht, 2000.
[132] Nic Wilson and Serafin Moral. Fast Markov Chain Algorithms for Calculating Dempster Shafer Belief. In *Proceedings of the 12th European Conference on Artificial Intelligence*, ECAI 1996, 1996.
[133] Pornpit Wongthongtham and Bilal Abu-Salih. Ontology-Based Approach for Semantic Data Extraction from Social Big Data: State-of-the-art and Research Directions. *CoRR*, abs/1801.01624, 2018.
[134] Huadong Wu, M. Siegel, R. Stiefelhagen, and Jie Yang. Sensor Fusion Using Dempster-Shafer Theory. In *IMTC/2002. Proceedings of the 19th IEEE Instrumentation and Measurement Technology Conference (IEEE Cat. No.00CH37276)*, volume 1, pages 7–12 vol.1, 2002.
[135] Ronald R. Yager. Including Semantics and Probabilistic Uncertainty in Business Rules Using Fuzzy Modeling and Dempster-Shafer Theory. In *AAAI Spring Symposium: AI Meets Business Rules and Process Management*, pages 111–116. AAAI, 2008.
[136] Cristiane A. Yaguinuma, Marilde T. P. Santos, Heloisa A. Camargo, M. C. Nicoletti, and Tatiane M. Nogueira. A Meta-Ontology for Modeling Fuzzy Ontologies and its Use in Classification Tasks based on Fuzzy Rules. *International Journal of Computer Information Systems and Industrial Management Applications*, 2014.
[137] John Yen. *Evidential Reasoning in Expert Systems*. PhD thesis, University of California,Berkeley, CA, Jan 1986.
[138] John Yen. Computing Generalized Belief Functions for Continuous Fuzzy Sets. *International Journal of Approximate Reasoning*, 6(1):1 – 31, 1992.
[139] John Yen. *Generalizing the Dempster-Shafer Theory to Fuzzy Sets*, pages 529–554. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
[140] John Yen. Implementing Evidential Reasoning in Expert Systems. *CoRR*, abs/1304.2731, 2013.
[141] Lofti Zadeh and Anca Ralescu. On the Compatibility of Evidence in Dempster-Shafer Theory. `https://arxiv.org/ftp/arxiv/papers/1304/1304.3119.pdf`,last checked=2018-08-27.
[142] Lofti A. Zadeh. Fuzzy Sets. *Information and Control*, 8(3):338 – 353, 1965.
[143] Lofti A Zadeh. Probability Measures of Fuzzy Events. *Journal of Mathematical Analysis and Applications*, 23(2):421 – 427, 1968.
[144] Lofti A. Zadeh. The Concept of a Linguistic Variable and Its Application to Approximate Reasoning—i. *Information Sciences*, 8(3):199 – 249, 1975.
[145] Lofti A. Zadeh. Fuzzy Sets, Fuzzy logic, and Fuzzy Systems. chapter Fuzzy Sets and Information Granularity, pages 433–448. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1996.
[146] Lofti A Zadeh. *Fuzzy Set Theory and Probability Theory: What is the Relationship?* International Encyclopedia of Statistical Science, 2011.
[147] Lotfi A. Zadeh. A Computational Approach to Fuzzy Quantifiers in Natural Languages. *Computers and Mathematics with Applications*, 9(1):149 – 184, 1983.
[148] Lotfi A Zadeh. A Simple View of the Dempster-Shafer Theory of Evidence and its Implication for the Rule of Combination. *AI Magazine*, 7(2):85–90, July 1986.
[149] Lotfi A. Zadeh. Fuzzy Sets Fuzzy Logic and Fuzzy Systems. chapter Discussion: Probability Theory and Fuzzy Logic Are Complementary Rather Than Competitive, pages 805–810. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 1996.
[150] Lotfi A. Zadeh. Is There a Need for Fuzzy Logic? *Information Sciences*, 178(13):2751–2779, July 2008.
[151] Lotfi A. Zadeh. A Note on Z-numbers. *Information Sciences*, 181(14):2923 – 2932, 2011.
[152] Bernard P. Zegler. Some Properties of Modified Dempster-Shafer Operators in Rule Based Inference Systems. *International Journal of General Systems*, 14(4):345–356, 1988.
[153] Fu Zhang, Jingwei Cheng, and Zongmin Ma. A Survey On Fuzzy Ontologies For The Semantic Web. *The Knowledge Engineering Review*, 31(3):278–321, 2016.
[154] Yan-cai Zhao, Zu-hua Liao, Teng Lu, and Juan Tong. Generalized Fuzzy Sets and Fuzzy Relations. In Bing-Yuan Cao, Pei-Zhuang Wang, Zeng-Liang Liu, and Yu-Bin Zhong, editors, *International Conference*

on Oriental Thinking and Fuzzy Logic, pages 11–16, Cham, 2016. Springer International Publishing.

[155] Qing Zhu and E. S. Lee. Dempster-shafer Approach in Propositional Logic. *International Journal of Intelligent Systems*, 8(3):341–349, 1993.

[156] Enrico Zio and Nicola Pedroni. *Literature Review of Methods for Representing Uncertainty*. Cahiers de la Sécurité Industrielle, 2013.