An Ontology Enhancement Framework to Accommodate Imprecise Concepts and Relations

Muhammad Abulaish
Department of Computer Science, Jamia Millia Islamia (Central University), New Delhi, India
Email: abulaish@ieee.org

Abstract — In this paper, we propose an ontology enhancement framework to accommodate imprecise concepts and their inter-relations mined from text documents. The proposed framework is modeled as a fuzzy ontology structure to represent concept descriptor as a fuzzy relation which encodes the degree of a property value using a fuzzy membership function. In our application, the fuzzy membership function is determined through text mining. Other than concept descriptors, the inter-concept relations in the ontology are also associated a fuzzy strength. The strength of association between two concepts determines the degree of association between the concepts. The fuzzy ontology with fuzzy concepts and fuzzy relations is an extension of the domain ontology with crisp concepts and relations which is more suitable to describe the domain knowledge for solving uncertainty reasoning problems. The applicability of the fuzzy ontology structure in mining and managing imprecise knowledge from biomedical text documents has been thoroughly experimented. The fuzzy ontology can later be used for information curation to answer imprecise queries posed at multiple levels of specificities along the underlying ontology.

Index Terms — Ontology engineering, ontology enhancement, Fuzzy ontology, imprecise concept, text mining, knowledge management.

I. INTRODUCTION

As envisage by Berners-Lee, Semantic Web (SW) [23] promise to make the Web a meaningful experience for which ontology is increasingly being accepted as a knowledge-management structure to represent domain knowledge in a structured and machine-interpretable form. Due to the vision of the SW, a large body of research is being moving around ontologies, and contributions have been produced regarding methods and tools for covering the entire ontology life cycle, from design to deployment and reuse [4], and ontology languages, such as OIL or OWL [3]. An ontology is a formal conceptualization of a real world, and it can share a common understanding of this real world [18]. Ontology represents a method of formally expressing a shared understanding of information, and has been seen by many authors as a prerequisite for the SW. With the support of the ontology, both user and system can communicate with each other by the shared and common understanding of a domain. Ontologies are emerging as the main area of interest for the success of the SW paradigm. There are many ontological applications that have been presented in various domains [7,8,9,15,17,19,21].

Though ontology plays a key role by defining concepts and relationships in an unambiguous way and it is gaining popularity for domain-specific applications, researchers are actively engaged in tackling some of the chief bottlenecks that still hinders the use of ontology for general-purpose applications. Some of these may be identified as follows:

- Absence of reliable and exhaustive ontologies for most of the domains. Since ontologies are meant to provide shared conceptualization of a domain, building and maintaining ontologies is an expensive task which requires a substantial involvement of domain experts. Acquisition of relevant knowledge for a domain and structuring it are both non-trivial tasks. Besides, experts may disagree. Automatic knowledge acquisition from text documents for ontology creation and/or enhancement may provide an effective solution to this problem.

- Though an ontology stores concepts and relationships in a definitive framework, it is unreal to expect that there exists a unique, unambiguous way of defining every concept and relationship which all authors and users will adhere to. Besides, for most of the domains, other than the strictly technical ones like the medical domain, it is found that knowledge modeling experts differ in their conceptualization of a domain. Moreover, since most of the ontologies essentially stores only structural semantic relations like is-a, part-of etc. among concepts, it is possible to enhance the ontology structure with the generic semantic relations extracted from text documents. What is ideally required is that within the rigid structure of the ontology, which is dictated by the application, there should be the flexibility to adapt new or modified concept descriptors and relationships as novel use of concepts and relations are encountered. This approach preserves the basic structured knowledge format for storing domain
knowledge, but at the same time allows for update of information.

- An ontology is generally designed to be a pre-defined structure with crisp concept descriptions and inter-concept relations. While a crisp definition is sufficient for information retrieval tasks from structured documents, the role of ontologies becomes severely restricted when intended to be used for information retrieval from unstructured text documents. Since web documents are not fully structured sources of information and in Internet almost everything, especially in the realm of search, is approximate in nature, it is not possible to utilize the benefits of a domain ontology straight away to extract information from such a document.

For example, from given snippet of a text document from tourism domain “Food and drink may be supplied by a mini-bar (which often includes a small refrigerator) containing snacks and drinks ...” the generic relation “includes” can be extracted and used to represent a relation includes (mini-bar, refrigerator) between the entities mini-bar and refrigerator. After further analysis, we found that the adverbial word “often” can be mined and associated with the relation “includes” to represent the degree of association between the entities mini-bar and refrigerator. Similarly, the qualifier “small” can also be mined and associated to represent the size of the refrigerator. This necessitates that concepts, their descriptions and inter-concept relations should be associated with a degree of fuzziness that will indicate the support for the extracted knowledge according to the currently available resources. Supports may be revised with more knowledge coming in future.

One way of overcoming this problem is the postulation of a “fuzzy ontology” by adding a value for degree of membership to each term (concepts and relations) that is imprecise in nature. A fuzzy ontology membership value can therefore be used to identify the most likely location in the ontology of a particular term. Each user would have their own values for the membership assigned to terms in the ontology, reflecting their likely information need and world view. Incorporating imprecision into the ontology structure itself can help in resolving ambiguities arising due to differences in user requirement specification and concept descriptions embedded in text documents.

In this paper we have proposed an ontology enhancement framework as a tool that assists domain experts in modeling imprecise domain knowledge. The proposed framework exploits fuzzy logic technique to incorporate fuzzy membership functions into rigid ontology structure. The enhanced ontology, termed as fuzzy ontology structure, is created as an extension of the standard ontology structure. In the proposed design of a fuzzy ontology, a concept descriptor is represented as a fuzzy relation, which encodes the degree of a property value using a fuzzy membership function. Other than concept descriptors, generic semantic relations and their strengths are learned from text documents and represented as a fuzzy relation.

The novelty of the proposed fuzzy ontology structure lies in describing both concepts and relations as a fuzzy relation. In case of concept descriptions, qualifiers help in defining the value of the property to varying degree of precision. Qualifiers can be linguistic qualifiers or fuzzy quantities. Linguistic qualifiers are particularly useful for developing a variable precision concept description for text processing applications, since these qualifiers are responsible for altering the property value of a concept within text documents. Fuzzy numeric values can either reflect varying precision for a property value, or can be easily adapted to reflect strength of association of a property descriptor to the concept. This property provides a generalized nature to the proposed fuzzy ontology structure and makes it ideally suited to handle imprecise concept descriptions of all kinds, including ambiguous or conflicting descriptions. In case of relations, qualifiers are linguistic variables that are either mined from the texts or defined as a function of frequency of association \( \forall(C_i, C_j) \), where \( \forall \) is a relation and \( C_i \) and \( C_j \) are ontology concepts. The relation qualifier represents the strength of associations of ontology concepts and thereby importance of the relations within a corpus and hence reflects the focus of research at a given point in time.

The proposed model can be used for intelligent information and knowledge retrieval through conceptual matching of text. The selected query does not need to match the decision criteria exactly, which gives the system a more human-like behavior. The model can also be used for constructing ontology or terms related to the context of search or query to resolve the ambiguity. The new model can execute conceptual matching dealing with context-dependent word ambiguity and produce results in a format that permits the user to interact dynamically to customize and personalized its search strategy. Last but not least, the proposed model can be used to handle queries at multiple levels of specificity along an ontology.

A system to create fuzzy ontology structure and its applicability in retrieving and curating information from text documents have been thoroughly experimented and reported in [14]. The curated information is used for answering user queries. In this paper we describe a more general ontology-based text information processing system to create fuzzy ontology structure in which both concepts and relations are modeled as fuzzy concepts and fuzzy relations respectively. The proposed system aims at alleviating some of the problems, discussed earlier, through the following mechanisms:

- A fuzzy ontology framework is proposed which models both concept descriptors and inter-concept relations as a fuzzy relation. This will allow for a structured conceptualization to still have the flexibility of variable definitions.
- Starting with a seed ontology, an ontology-based text-information processing system is presented that is equipped with a knowledge acquisition and ontology learning mechanism to facilitates the
enhancement of the underlying ontology with newly acquired information. The facility to enhance the ontology using mined information from texts allows the system to be tuned to answer queries intelligently from any corpus, rather than restricting it to a predefined fixed conceptualization. The proposed ontology-guided text processing system exploits natural language processing techniques to mine imprecise concept descriptors and inter-concept relations along with their strengths from text documents and then utilizes them for enhancement of the domain ontology.

- A fuzzy inference mechanism is proposed to generate the membership degrees for every fuzzy concept and relation of the fuzzy ontology. Every fuzzy relation has a set of membership degrees associated with various concept-pairs of the domain ontology.

We have shown an application of the proposed fuzzy ontology framework to two different domain - tourism (a general-purpose domain) and bio-medicine (a technical domain) to enhance a seed concept ontology into fuzzy ontology. The enhanced fuzzy ontology structure can later used for database curation from text documents to answer user queries intelligently.

Rest of the paper is organized as follows: section II presents a brief review on the existing fuzzy ontology structures. In section III, we give the modeling details of the proposed fuzzy ontology framework to accommodate fuzzy concepts and relations. Section IV presents an ontology-based text processing system to mine ontology concepts and their inter-relations. An AND-OR tree based method to fuzzify the relation strengths is presented in section V. Finally, section VI concludes the paper with future works.

II. RELATED WORK

In this section we present an overview of some of the recent research efforts that have been directed towards the problems of generation of fuzzy ontology structures and its applications to design text processing systems. Though ontology is meant to represent knowledge in an unambiguous structured format, it is practically impossible to assume that all application developers will agree to any such unique structure amicably. Enhancement of crisp ontology structures to a fuzzy ontology structure is viewed as a potential solution to this problem and received a lot of attention in recent times.

Widyantoro and Yen [5] have shown how fuzzy membership values associated to ontology concepts, along with a concept hierarchy, can be used for intelligent text information retrieval. Starting with a set of manually tagged abstracts of papers from several IEEE Transactions, a fuzzy ontology is built on the collection of keywords. The abstracts are tagged based on their title, authors, publication date, abstract body, and author supplied keywords. The hierarchical arrangement of the terms in the newly generated ontology is dependent on their co-occurrence measures. The drawback of this system is its dependence on user judgment about the relevance of articles to user queries which is provided manually.

Wallace and Avrithis [16] have extended the idea of ontology-based knowledge representation to include fuzzy degrees of membership for a set of inter-concept relations defined in an ontology. The membership of these relations are used to judge the context of a set of entities, the context of a user and the context of the query for the purpose of intelligent information retrieval. A fixed set of commonly encountered semantic relations have been identified and their combinations are used to generate fuzzy, quasi-taxonomic relations.

Quan et al. [24] have proposed an automatic fuzzy ontology generation framework – FOGA. They have incorporated fuzzy logic into formal concept analysis to handle uncertainty information for conceptual clustering and concept hierarchy generation. However, the quality of clustering is dependent on assignment of meaningful labels to initial class names, attributes and relations. This is done manually and requires domain expertise. This system is also not designed to extract fuzzy relational concepts from unstructured or semi-structured text documents.

Parry [6] proposes a fuzzy ontology structure in which each overloaded term in the MeSH ontology is associated with a fuzzy membership value given by the user to indicate the relative importance of the term and its associated concepts in the context of information retrieval.

Lee et al. [1] have proposed a fuzzy ontology structure as an extension of the domain ontology with crisp concepts for the purpose of Chinese news summarization. Their system starts with a domain ontology with various events of news which is predefined by the domain experts. The document preprocessing mechanism generates the meaningful terms based on the news corpus and the Chinese news dictionary defined by the domain expert. Then, the meaningful terms is classified according to the events of the news by the term classifier. The fuzzy inference mechanism generates the membership degrees for each fuzzy concept of the fuzzy ontology. Every fuzzy concept has a set of membership degrees associated with various events of the domain ontology.

The proposed fuzzy ontology structure is a novel structure that is created as an extension of traditional ontology structures. The novelty lies in representing both concept descriptions and inter-concept relations as a fuzzy relation in which strengths are represented through linguistic variables. The structure can be easily adapted to reflect strength of association in terms of numeric values. Hence this structure is more general than the fuzzy ontology structures defined earlier since this can accommodate both linguistic variables and numeric values.

Since ontology describes a domain of interest in an unambiguous way, ontology-based text information processing schemes can help in alleviating a wide variety of natural language ambiguities present in a given domain. Ontologies have frequently been incorporated in information retrieval systems as a tool for the recognition
III. PROPOSED FUZZY ONTOLOGY MODEL

Traditionally, as discussed in the previous section, concepts are described in an ontology using a \(<\text{property},\ \text{value},\ \text{constraints}>\) framework and that of relations are described using \(<\text{concept},\ \text{relation},\ \text{concept}>\) framework. In this section we propose a fuzzy ontology which is created as an extension to the standard ontology by embedding a set of membership degrees in each concept and relation of the domain ontology. The concepts and relations with the membership degrees are called fuzzy concepts and fuzzy relations respectively. In fuzzy ontology the property descriptors are accompanied by qualifiers along with values for defining a concept in a \(<\text{property},\ \text{value},\ \text{qualifier},\ \text{constraints}>\) framework, where the value and the qualifier are both defined as a fuzzy set. This framework allows defining the property-value of a concept with differing degrees of fuzziness, without actually changing the concept description paradigm. Such concept descriptions can be termed as imprecise concept descriptions. Similarly, the fuzzy ontology stores inter-concept relations in a \(<\text{concept},\ \text{relation},\ \text{concept},\ \text{relation_strength}>\) framework where relation_strength represents the degree of association between the concepts and is defined through fuzzy inferencing mechanism.

Now, we give the definitions of fuzzy concept, fuzzy relationship, and fuzzy ontology as follows.

**Definition-1 (Fuzzy concept)** – A fuzzy concept is the refinement of the ontology concept by embedding a qualifier set associated with the set of concept values. If a domain ontology has a concept \(C_i\) and the corresponding value set \(\{V_1, V_2, ..., V_n\}\) then we can refine \(C_i\) into the fuzzy concept and denote the fuzzy concept as \(\{C_i : \mu_{C_{i1}}, \mu_{C_{i2}}, ..., \mu_{C_{in}}\}\) where, \(\{Q_1, Q_2, ..., Q_n\}\) is the qualifier set associated with the value set of the concept \(C_i\).

**Definition-2 (Fuzzy relationship)** – A fuzzy relationship between a pair of ontology concepts is defined by associating a fuzzy strength to the underlying relation. If \(C_i\) and \(C_j\) are two ontology concepts and \(R\) is a relation between them, then we can refine \(R\) into the fuzzy relationship and denote the fuzzy relationship as \(<C_i, R, C_j, \mu_{(C_i,C_j)}(R)>\), where \(\mu_{(C_i,C_j)}(R)\) is the strength of association of \(R\) and is determined through fuzzy inferencing mechanism.

**Definition-3 (Fuzzy Ontology)** – A fuzzy ontology \(\Theta_F\) is an extended domain ontology with fuzzy concepts and fuzzy relationships and can be defined as a quadruple of the form:

\[
\Theta_F = (C, P_F, \text{fv}_F, M)
\]

- \(C\) is the set of ontology concepts.
- \(P_F\) is a set of fuzzy property sets. A fuzzy property \(p_r \in P_F\) is defined as a quadruple of the form \(p_r(c, v_r, q_r, f)\), where \(c \in C\) is an ontology concept, ‘\(v_r\)’ represents fuzzy attribute values and could be either fuzzy numbers or fuzzy quantifiers, ‘\(q_r\)’ models linguistic qualifiers and are hedges, which can control or alter the strength of an attribute value and \(f\) is the restriction facets on \(v_r\).
- \(\text{fv}_F\) is a set of inter-concept relations between concepts. Like fuzzy concept properties, \(\text{fv}_F\) is defined as a quadruple of the form \(\text{fv}_F(c_i, c_p, t, q)\), where \(c_i, c_p \in C\) are ontology concepts, ‘\(t\)’ represents relation type, and ‘\(q\)’ models relation strengths and are linguistic variables, which can represent the strength of association between concept-pairs \(<c_i, c_p>\).
- The choice of fuzzy numbers or fuzzy quantifiers for values is dictated by the nature of the underlying attribute and also its restriction facets. The complete range of values over which an attribute can take values defines the universe of discourse \(\text{M}\). The universe of discourse is decomposed into a collection of fuzzy sets. Each fuzzy set is defined over a domain that overlays part of the universe of discourse.

An interesting aspect of modeling attributes as fuzzy sets is that with an underlying set of numeric values, one can associate different fuzzy quantifier sets to represent different aspects of the same value. For example, a single price value can be interpreted as being “close to” or “far away” from another value of price, and at the same time can also be interpreted as “cheap” or “expensive.” Moreover, hedges can also be applied to create new fuzzy sets with different meanings. Thus modeling an attribute as a fuzzy set allows a single attribute to contribute to different types of imprecision in concept description.

Fuzzy qualifiers are used in fuzzy models to dynamically create new fuzzy sets and change the meaning of linguistic variables. This enables the modification of existing fuzzy sets temporarily to provide different meaning to the underlying linguistic variable. Most of the applications consider linguistic qualifiers as those elements that modify the value of a fuzzy number. However, modeling qualifiers become more complex when the fuzzy quantifier set is itself graded. For example, the weather domain uses three values hot, cold, and cool to model the weather condition in terms of temperature. In this case, fuzzy modeling of the temperature can be achieved by the membership table shown in Fig. 1. As we can see, the weather value “cool”...
can be interpreted to be as “cold” to some extent, and vice versa, where the extent is defined by the fuzzy membership values. An interesting thing to observe over is that since “cool” and “cold” are basically intensity variations of the same temperature, where “cool” is an intensified version of “cold”, thus the weather which is “very cold” can be considered to be “cool” with a higher membership value than the weather which is simply “cold”. Thus in this case we want that rather than working as an intensifier, which hardens or reduces the membership value, the intensifier “very” should increase the membership value of “cold” to “cool”. Obviously, this is a special situation occurring due to the gradation among the fuzzy quantifiers themselves.

![Fuzzy membership functions for temperature values](image)

**Fig. 1. Fuzzy membership functions for temperature values**

To take care of all such situations, we have adopted a generalized approach to model fuzzy quantifier and qualifier sets. In this scheme both fuzzy quantifiers and fuzzy qualifiers can be modeled as graded sets, with the similarity between two variables defined as a function of their relative positions in the set. This allows us to control and combine the effects of qualifiers over quantifiers in a more context dependent way. The next section presents detailed description of the modeling scheme with specific references to domains indicating the types of values for which a particular modeling is suitable.

A. Encoding Domain Knowledge using Fuzzy Ontology Structure

Since the essence of fuzzy sets is to be able to control the degree of imprecision rather than bind a single membership function to a definition, we propose the use of application-specific fuzzy-membership functions for fuzzy quantifiers and qualifiers. Though the membership functions themselves change depending on the nature of the domains, their role in modifying fuzzy attribute values remains unchanged across applications. For appropriate fuzzy-fication of concept descriptions, each attribute is also associated with a qualifier set which is a collection of hedges. Since the qualifiers associated to different properties are usually different, hence the hedge sets are also different though may be overlapping. To maintain uniformity of using concept descriptions, every value is always assumed to be accompanied by a qualifier. Hence to model values without a qualifier, we have used the qualifier “null”. For every qualifier set, we have included the value “null” to indicate the absence of any qualifier. Since the proposed fuzzy ontology structure was motivated by text information retrieval applications, we look at the qualifier set as a set of hedges which are to be designed in an application specific way. As we have discussed earlier, we perceive that the role of a modifier for a domain does not remain static. Rather it is defined as a function of both the qualifier and the value it is trying to modify. In case of matching a pair of <value, qualifier> tuples, the overall effect is to be determined as a function of the distance between the qualifiers, and the value pairs. When values match, but qualifiers do not match the overall aim is to always decrease the precision of an associated value.

Qualifier sets are modeled as graded sets. The similarity between two objects in the graded set is defined as a function of their relative positions within the set. The position of “null” is selected depending on the nature of qualifiers used. For most of the qualifier sets, “null” occupies a central position, with dilution hedges occurring towards its left and intensification hedges occurring towards its right. However, if a domain includes only intensification hedges then “null” is located as the first element in an ascending ordered set. Similarly, for a set of only dilution hedges, “null” occupies the extreme right position in an ascending ordered set.

We now show how the fuzzy memberships are computed for qualified variables. Fuzzy memberships for qualified variables are computed using composition of the fuzzy membership values for the variables and the qualifiers. The similarity between two qualified variables <q_i, v_i> and <q_j, v_j> is expressed as a fuzzy membership function denoted by \( \mu_{q_iq_j}(q_i, v_i) \).

Since qualifiers are modeled as graded sets, fuzzy membership functions for these sets can be designed using their relative positions within the set. The distance between two qualifiers in the collection reflects their degree of dissimilarity. The distance between the qualifier \( q_i \) at position \( i \) and the qualifier \( q_j \) at position \( j \) within a set is defined by using equation 1.

\[
d(q_i, q_j) = |i - j|
\]  

The fuzzy membership function for the qualifier set is then defined as given in equation 2.

\[
S = f(q_i, q_j) = 1 - \frac{d(q_i, q_j)}{MAX + 1}
\]

where, \( MAX = \max \{d(q_i, q_j) \mid q_i, q_j \in Q, q_i \neq q_j \} \).  

In order to compute the fuzzy membership of compositions, we have taken the dilution or intensification aspects of both the qualifiers and values.
An element $t_i$ is a dilution with respect to another element $t_j$ in the graded set if $i < j$ in the ordered set \{ $t_i$, $t_j$ \}. Conversely, $t_j$ is an intensifier with respect to $t_i$. This information is encoded in terms of a function as given in equation 3.

$$Sgn (t_i, t_j) = \begin{cases} +1, & \text{if } i < j \\ -1, & \text{if } i > j \\ 0, & \text{if } i = j \end{cases} \quad (3)$$

The elements $t_i$ and $t_j$ can represent a pair of qualifiers $q_i$ and $q_j$ or a pair of values $v_i$ and $v_j$. The composite fuzzy membership function is defined as shown in equation 4.

$$\mu_{q_i, q_j}(v_i, v_j) = \begin{cases} f(q_i, v_j), & \text{if } Sgn (q_i, q_j) \times Sgn (v_i, v_j) = -1 \\ f(q_j, v_i), & \text{if } Sgn (q_i, q_j) \times Sgn (v_i, v_j) = +1 \\ f(q_i, v_j) \times f(q_j, v_i), & \text{if } Sgn (q_i, q_j) \times Sgn (v_i, v_j) = 0 \end{cases} \quad (4)$$

Moreover, the numeric attributes can also be expressed as fuzzy numbers which simple represent fuzzy numeric intervals over the domain of particular variable. Fuzzy numbers are generally represented using bell-shape, triangular or trapezoidal membership function along with a fuzzy quantifier defined over the numeric domain with appropriate fuzzy functions. A subset of hedges known in the domain of fuzzy set theory like, few, somewhat, small, average, more or less, many, very, high etc. can also be used directly on crisp numbers to convert them into fuzzy sets through the process called approximation [11].

IV. ONTOLOGY-BASED TEXT PROCESSING SYSTEM

In this section, we propose an ontology-based text information processing system, shown in figure 5, to extract fuzzy concept descriptors and fuzzy relationships from text documents to enhance the underlying domain ontology into fuzzy ontology. In figure 5, the domain ontology with various concepts and structural semantic relations is predefined by the domain experts. The system consists of four main modules – retrieval agent, document processor and parser, concept descriptors and relation extractor, and Fuzzy Inference Engine. The functionalities of the modules are stated here briefly.

- The retrieval agent uses the ontology concepts and retrieves relevant web pages from Web to create a text corpus.
- The document processor and parser module accepts free-form text documents and identifies information components by dividing them into individual record-size chunks after cleaning the Meta Language (ML) tags. This also uses a Parts-Of-Speech (POS) tagger that assigns POS tags to individual words. Finally, it
creates a semi-structured intermediate representation of the texts on the basis of POS analysis.

- The concept descriptors and relation extractor module uses the semi-structured texts as input and applies a combination of the natural language processing and text-mining approach to learn new concept descriptors and relations from them.
- The fuzzy inference engine generates the membership degrees for each fuzzy concept and fuzzy relations of the fuzzy ontology.

The functional details of these modules are given in the following sub-sections.

A. Retrieval Agent, Document Processor and Parser

First, the retrieval agent, document processor and parser chunks, which in our case is a sentence identified by the tags, divides the document into individual record-size segments. These segments are stored into a text corpus for further processing by the retrieved web pages from Web. The retrieved web pages may be assumed to be a valid value, provided occurrence of a full stop. The document processing mechanism also consists of a part of speech (POS) Tagger, an integral part of the Stanford Java NLP parser, which assigns POS tags to individual words. The POS plays an important role in information extraction. Concept names are usually nouns, relations are verbs, concept descriptors are adjectives and description qualifiers mostly consist of adverbs. Thus words with these parts-of-speech are to be extracted from sentences while mining imprecise concept descriptions and relations from the texts. The document parser performs a POS tag analysis and creates a parse tree to represent the grammatical structure of a sentence. For this we have used the Stanford Java NLP parser. An example sentence, its tagged version and the corresponding parse tree is shown in figure 6.

B. Concept Descriptors and Relation Extractor

In this section we will discuss the working principle of the concept descriptors and relation extractor module that uses the parse tree generated by the document processor and parser module along with parent domain-concepts present in the underlying ontology to learn fuzzy concepts and relations to create fuzzy ontology. The purpose of this module is two-fold – concept descriptor extraction and relation extraction, which are discussed separately in the remainder of this section.

Concept Descriptor Extraction - In order to extract fuzzy concept descriptions, the extraction process has employed a two-pronged approach which exploits the description of the parent domain concepts, their inter-relationships and constraints derived from the ontology structure to extract relevant information from the intermediate form of the text documents. Given a property name, the module looks for possible values that are likely to occur as adjectives to fill up the object description. Hence any adjective retrieved can be assumed to be a valid value, provided positional constraints are satisfied. Obviously, this method allows accommodating object descriptions with property values that are not present in the underlying ontology. The ontology descriptor set can be appropriately enhanced. In the absence of a property name, property value from the underlying ontology is used as a pointer to fill up the particular property slot. For an identified property value, the associated adverbial words are extracted as a fuzzy qualifier for the property under consideration.

A formal knowledge-distillation algorithm is presented in [14]. Starting with a seed ontology which contains a small set of property values, the knowledge-distillation algorithm is applied iteratively on the document collection to learn new property values and qualifiers from it. In order to decide the correct class for new qualifiers and values extracted, we have applied statistical analysis on the learned value and qualifier sets independently. For all different values in the set, frequencies of their occurrences with different properties

---

2 http://nlp.stanford.edu/software/lex-parser.shtml
are computed and a value is assigned to the property with which it has maximum number of occurrences. The same is done for the assignment of qualifiers to different properties.

While building exhaustive unambiguous ontologies are prohibitively complex, this mechanism can be employed for building good ontologies over time. Hence this mechanism is ideally suited for building ontology-based text information retrieval systems for any domain, where the chief bottleneck is that of building the ontologies. An application of the fuzzy ontology structure in database curation for the purpose of answering user imprecise queries over text documents is presented in [14].

Relation Extraction - A relation is assumed to be binary in nature, which defines a specific association between an ordered pair of ontology concepts or entities. The ontology concepts or entities generally appear as a noun phrase in text documents. The process of identifying relations is accomplished in two stages. During the first stage, prospective information components which might embed relations within them are identified from the sentences. During the second stage, a feasibility analysis is employed to identify correct relations.

A relation is usually manifested in a document as a relational verb which may occur in a sentence in its root form or as a variant of it. Different classes of variants of a relational verb are recognized by our system. The first of this class comprises of morphological variants of the root verb, which are essentially modifications of the root verb itself. In the context of technical domain, we also observe that the occurrence of a verb in conjunction with a preposition very often changes the nature of the verb. For example, in biomedical domain, the functions associated to the verb activates may be quite different from the ones that can be associated to the verb form activates in, in which the verb activates is followed by the preposition in. Thus our system also considers relations represented by a combination of root verbs or their morphological variants, and prepositions that follow these. Typical examples of relations identified from biomedical domain in this category include “activated in”, “binds to”, “stimulated with” etc. To recognize relations correctly, all prepositions at distance one or two from a relational verb are considered. This increases the accuracy of the system in identifying relations, since it has been found that very often the text is interjected with adverbs following the main verb. Using the proposed approach, the adverbs are eliminated from consideration since they simply used by the author to emphasize on the strength of the associated relational verb. One such sample sentence is shown below, in which the relation to be identified is expressed in, though the words occur in the text separated by the adverb exclusively.

MEDLINE:95016436 - A family of <cons sem="G#protein_family_or_group">serine proteases</cons> specifically processes the <cons sem="G#protein_subunit">nuclear factor-kappa B subunit</cons> p65 in vitro and may impair human <cons sem="G#other_name">immunodeficiency virus</cons> replication in these cells.

The arguments associated to a relation can be inferred from the entities located in the proximity of the relational verb. Initially all triplets of the form <noun phrase, verb phrase, noun phrase>, are retrieved by traversing the parse tree built earlier. The working principle of the information component extraction process is explained by the following steps:

- List of information components \( L_{IC} \) is initialized to null.
- The parse tree is traversed to locate relational verb for creating information components. Starting with the node containing relational verb, if both left and right sub-trees contain noun phrases, the required verb is located as follows:
  - The verb represented at the parent of these sub-trees is assumed to represent a relation.
  - If the right sub-tree of the node contains a preposition within distance 1 or 2 from the node verb, then the preposition is associated to the verb, and the verb-preposition pair is identified as a possible relational verb.
- A unique combination of the left and right noun phrases along with the possible relational verb identified as above is added to the list of information components \( L_{IC} \).

A partial list of relational verbs with their associated entity-pairs extracted from a small set of web pages describing tourism domain is shown in Table I.

<table>
<thead>
<tr>
<th>Left Entity</th>
<th>Relation Qualifier</th>
<th>Relation</th>
<th>Entity Qualifier</th>
<th>Right Entity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hotel</td>
<td>null</td>
<td>is</td>
<td>null</td>
<td>establishment</td>
</tr>
<tr>
<td>Hotel</td>
<td>null</td>
<td>serves</td>
<td>usually</td>
<td>meals</td>
</tr>
<tr>
<td>Hotel</td>
<td>null</td>
<td>provides</td>
<td>null</td>
<td>paid lodging</td>
</tr>
<tr>
<td>Hotel</td>
<td>null</td>
<td>have</td>
<td>null</td>
<td>conference services</td>
</tr>
<tr>
<td>Hotel</td>
<td>necessarily</td>
<td>provides</td>
<td>null</td>
<td>accommodation</td>
</tr>
<tr>
<td>hotel</td>
<td>usually</td>
<td>synonymous</td>
<td>null</td>
<td>pub</td>
</tr>
<tr>
<td>mini-bar</td>
<td>often</td>
<td>includes</td>
<td>small</td>
<td>refrigerator</td>
</tr>
<tr>
<td>capsule hotel</td>
<td>null</td>
<td>supplies</td>
<td>minimal</td>
<td>facilities</td>
</tr>
<tr>
<td>capsule hotel</td>
<td>null</td>
<td>supplies</td>
<td>minimal</td>
<td>room space</td>
</tr>
<tr>
<td>Boutique hotel</td>
<td>null</td>
<td>describes</td>
<td>luxurious</td>
<td>hotel</td>
</tr>
<tr>
<td>Boutique hotels</td>
<td>generally</td>
<td>fitted with</td>
<td>null</td>
<td>telephone and Wi-Fi Internet connections</td>
</tr>
<tr>
<td>Boutique hotels</td>
<td>null</td>
<td>have</td>
<td>null</td>
<td>on site dining facilities</td>
</tr>
<tr>
<td>Boutique hotels</td>
<td>null</td>
<td>offers</td>
<td>null</td>
<td>bars</td>
</tr>
</tbody>
</table>

### Table I

RELATIONAL VERBS AND ASSOCIATED ENTITY-PAIRS EXTRACTED FROM TEXT DOCUMENTS OF TOURISM DOMAIN
Since the above process considers only those verbs which co-occur with noun phrases in their vicinities, a large number of irrelevant verbs are eliminated from being considered as relations. However, our aim is not just to identify possible relational verbs but to identify feasible relation triplets. Hence we engage in further statistical analysis to identify feasible relation triplets.

To consolidate the final list of relations we take care of two things. Firstly, since various forms of the same verb represent a basic relation in different forms, the feasible collection is extracted by considering only the unique root forms after analyzing the complete list of information components. Again, each relation can occur in conjunction with multiple concept-pairs, while some concept-pairs may not ever co-occur. Hence, in the second phase of feasibility study, all feasible ordered triplet combinations are compiled. The core functionalities of the feasible relation finding module are summed up in the following steps.

- Let \( L_V \) be the collection of verbs or verb-preposition pairs, which are extracted as part of information components. Since each verb can occur in more than one form in the list \( L_V \), the frequency of occurrence of each root verb is the sum-total of its occurrence frequencies in each form. All root verbs with frequency less than a user-given threshold are eliminated from further consideration. The surviving verbs are termed as most-frequently occurring root verbs and represent feasible relations.

- Once the frequent root verb list is determined, the list \( L_V \) is further analyzed to identify the complete list of all relational verbs including frequent root verbs, their morphological variants and their co-occurrence with prepositions.

- For each inferred relation verb form \( \mathcal{R} \), the frequency of occurrence of each unique ordered triplet \( <E_i, \mathcal{R}, E_j> \) is computed where \( E_i \) and \( E_j \) are the entities located in the proximity of \( \mathcal{R} \). Obviously, two entities \( E_i \) and \( E_j \) may define two different tuples even with the same relation \( \mathcal{R} \) between them, where their roles will be reversed. Thus \( <E_i, \mathcal{R}, E_j> \) and \( <E_j, \mathcal{R}, E_i> \) denote two different relation triplets.

Hence a particular relation may occur in conjunction with multiple entity-pairs and a particular entity-pair may be related through multiple relations. Each relation is assigned a strength, where strength of a relation reflects the frequency of co-occurrence of a relational verb in conjunction with an ordered pair of entities. The strength of the relation \( \mathcal{R} \) is computed as a fuzzy membership value \( \mu_{<E_i, \mathcal{R}, E_j>} (\mathcal{R}) \) indicating the degree of co-occurrence of the triplet \( <E_i, \mathcal{R}, E_j> \). The frequency of the relation \( \mathcal{R} \) occurring in conjunction with the ordered entity-pair \( <E_i, E_j> \) against all occurrences of \( \mathcal{R} \) and the ratio of frequency of the ordered concept pair \( <E_i, E_j> \) occurring in conjunction with \( \mathcal{R} \) against all occurrences of the pair. This is shown in equation 5, where \( |<E_i, \mathcal{R}, E_j>| \) represents the frequency count of the relation triplet \( <E_i, \mathcal{R}, E_j> \).

\[
\mu_{<E_i, \mathcal{R}, E_j>} (\mathcal{R}) = \frac{1}{2} \left[ \frac{|<E_i, \mathcal{R}, E_j>|}{\sum_{a,b} |E_a, \mathcal{R}, E_b|} + \frac{|<E_j, \mathcal{R}, E_i>|}{\sum_{a,b} |E_a, \mathcal{R}, E_b|} \right] \quad \text{(5)}
\]

The strength of a relation reflects the significance of a particular type of association between an entity-pair.

C. Fuzzy Inference Engine

In this section, we will describe the working principle of the fuzzy inference engine. The fuzzy inference engine is responsible to generate the membership degrees for each fuzzy concept and fuzzy relation of the fuzzy ontology. In case of fuzzy concept descriptors, either the value or qualifier sets are mined from the text documents or determined through proper fuzzification process, as mentioned in section V. For example, in tourism domain the values used to describe the type of a hotel room is luxury, cheap, etc. and that of qualifier values are often, very etc. Similarly, the room rent values of a hotel are generally given as numeric values which are first modeled as linguistic variables low, medium, high etc. before embedding into the fuzzy ontology.

In case of fuzzy relations, if a relation is associated with a fuzzy qualifier in text documents it is extracted and used by the system to represent the degree of association of the relation. For example, in tourism domain, we found a relation triplet \(<\text{mini-bar}, \text{often includes}, \text{small refrigerator}>\) in which the qualifier often is associated with the relation include to represent its degree of association. It may be the case that relations are not associated with fuzzy qualifiers but their association is many-to-many. For example, in biomedical domain, the activation relation can be defined between different biological substance-pairs. In such cases, an appropriate fuzzy membership generation mechanism, discussed in the following section, is proposed to assign the relation strength a proper linguistic qualifier.

V. Calculating Degree of Association Between Ontology Concepts

In this section, we have presented a process to calculate the degree of associations, in terms of linguistic qualifiers, between the ontological concepts mined from text documents. For experiment purpose, we have considered the GENIA corpus [10] in which biological entities are tagged with the GENIA ontology concepts. The GENIA ontology [10] is a taxonomy of 47 biologically relevant nominal categories in which the top three concepts are biological source, biological substance and other_name. The other_name refers to all biological concepts that are not identified with any other known concept in the ontology. The sub-tree rooted at source
contains 13 nominal categories and the other rooted at 
substance, contains 34 nominal categories. The GENIA 
corpus contains 2000 tagged MEDLINE abstracts. Tags 
are leaf concepts in GENIA ontology. A biological 
relation is expressed as a binary relation between two 
biological concepts. Following this definition, while 
mining for biological relations, we define a relation as an 
activity co-occurring with a pair of tags within the 
GENIA corpus. In [13] we had identified a set of 24 root 
verbs and their 246 variants, which represent biological 
relations occurring in the GENIA corpus. A complete list 
of all feasible biological relations and their morphological 
variants extracted from the GENIA corpus is available on 
http://www.geocities.com/mdabulaish/BIEQA/. We can 
enhance the GENIA ontology with these relations.

Since the GENIA corpus is tagged with leaf-level 
concepts, all relations are defined between entities or 
between leaf-level concept pairs. However keeping track 
of all instances may not be useful from the aspect of 
domain knowledge consolidation. Hence our aim is to 
generalize a relation at an appropriate level of specificity 
before including it in the ontology. This reduces over- 
specialization and noise.

All molecular biology concepts in the GENIA 
ontology are classified into two broad categories, source 
and substance. Hence the entity pairs occurring with each 
relation can be broadly classified as belonging to one of 
the following four categories: (i) <source, source> (ii) 
<source, substance> (iii) <substance, source>, and (iv) 
<substance, substance>.

Every instance of a relation belongs to one of these 
categories and the total number of instances associated to 
any category can be obtained with appropriate 
summation. Since a generic concept can represent 
multiple specific concepts, hence the first step towards 
characterizing relations is to consolidate the total number 
of relations belonging to each category, identify the 
pathways through which they are assigned to a category 
and then find the most appropriate generalization of the 
relation in that category.

In order to achieve this, we define a concept-pair tree 
to represent each category. The root node of a concept- 
pair tree denoted by (L<sub>1</sub>, R<sub>1</sub>) contains one of the four 
generic concept-pairs defined earlier. Each node N in a 
concept-pair tree has two constituent concepts <C<sub>1</sub>, C<sub>2</sub>> 
denoted as the LEFT and the RIGHT concepts. The 
LEFT and RIGHT concepts are specializations of L<sub>1</sub> and 
R<sub>1</sub> respectively, as obtained from the underlying 
ontology. Each concept-pair tree stores all possible 
ordered concept-pairs that match the root concept-pair 
(L<sub>1</sub>, R<sub>1</sub>) and is generated using a recursive algorithm, 
described in the next section.

A. Generating Concept-Pair Trees

The concept-pair tree is represented as an AND-OR 
tree, where each node has links to two sets of children, 
denoted by L<sub>1</sub> and L<sub>2</sub>. L<sub>1</sub> and L<sub>2</sub> each contain a set of 
concept-pair nodes. The two sets L<sub>1</sub> and L<sub>2</sub> are 
themselves connected by the OR operator, while the 

nodes within each of them are connected with each other 
through an AND operator. For every node N, the two sets 
of child nodes L<sub>1</sub> and L<sub>2</sub> are created as follows:

1. L<sub>1</sub> consists of concept pairs created by expanding the 
LEFT concept to consider all its child nodes in the 
concept ontology, while keeping the RIGHT concept 
unchanged.

2. L<sub>2</sub> is created by keeping the LEFT concept 
unchanged while considering all children of the 
RIGHT concept in the concept ontology.

3. When any of the concepts LEFT or RIGHT is a 
leaf-level ontology concept, the corresponding set L<sub>1</sub> 
or L<sub>2</sub> respectively is NULL.

Starting from a root concept pair <L<sub>0</sub>, R<sub>0</sub>>, the 
complete concept-pair tree is created recursively as 
follows:

\[
\text{OR[AND [<children of L<sub>0</sub>, R<sub>0</sub>]], AND [<L<sub>1</sub>, children of R<sub>0</sub>]]}
\]

In order to exemplify the process, let ‘a’ and ‘d’ 
represent two root concepts in a concept ontology, at 
each of which an ontology sub-tree is rooted, as shown in 
upper stub of Fig. 6. In order to create an AND-OR 
concept-pair tree, the root is the concept pair <a, d>. And, 
the sets L<sub>1</sub> and L<sub>2</sub> for the root node <a, d> are determined 
as L<sub>1</sub>: <b, d>, <c, d>; L<sub>2</sub>: <a, e>, <a, f>. Fig. 6 shows the 
resulting AND-OR tree in which “AND” is represented by 
‘\(\land\)’, and “OR” is represented using the symbol ‘\(\lor\)’. It 
may be noted that leaf-level pairs occur more than once 
in the tree. Each occurrence defines a path through which 
relations between that pair may be propagated up for 
generalization. Two sets of relations converging at a 
parent node, could be viewed as alternative models for 
generalization or could be viewed as complementing each 
other to form the total set at the parent level, depending 
on whether they are coming via the AND path or the OR 
path. This is further explained in the next section.

B. Mapping Relation Instances over a Concept-Pair Tree

After creating the four different concept-pair trees for the 
GENIA ontology, the most feasible representation of a 
relation for each of these categories is obtained using 
these. Suppose there are N instances of a relation R<sub>g</sub> 
observed over the corpus. Each of these instances is 
defined for a pair of leaf-level concepts. Based on the 
generic category of the leaf-level concepts, each relation 

Fig. 6. Sample AND-OR concept-pair tree
instance can be mapped to a leaf node in one of the four concept-pair trees.

For each concept-pair tree $$T^G$$, all instances that can be mapped to leaf-level nodes of $$T^G$$ are mapped at the appropriate nodes. These counts are propagated up in the tree exploiting its AND-OR property. Since each leaf-level node has multiple occurrences in a concept-pair tree, each relation instance is mapped to all such leaf-level nodes. For each non-leaf node in the concept-pair tree, the total number of relations is equal to the number of instances propagated up through all its children in either $$L_1$$ or $$L_2$$. In order to derive the most appropriate levels for describing a relation, the concept-pair tree is traversed top-down. Starting from the most generic level description at the root level, an information loss function based on set-theoretic approach is applied at each node to determine the appropriateness of defining the relation at that level.

C. Characterizing Relations at Appropriate Levels of Specificity

The process of determining the most specific concept pairs for relations follows a top-down scanning of the AND-OR tree. Starting from the root node, the aim is to determine those branches and thereby those nodes which can account for sufficiently large number of relation instances. When the frequency of a relation drops to an insignificant value at a node the node and all its descendent need not be considered for the relation conceptualization, and may be pruned off without further consideration. The lowest unpruned node becomes a leaf and is labeled as the most specific concept-pair for defining a relation.

Information Loss ($$N$$) = \[ \frac{|IC_N - IC_P|}{|IC_P + IC_N|} \] .......(6)

where, $$IC_N$$ = Count of instances of relation $$r_g$$ at N, $$IC_P$$ = count of instances of $$r_g$$ at parent P of N. Equation 6 defines a loss-function that is applied at every node N to determine the loss of information incurred if this node is pruned off. The loss function is computed as a symmetric difference between the number of instances that reach the node and the number of relation instances that were defined at its parent. Equation 6 states that if the information loss at a node N is above a threshold, it is obvious that the node N accounts for a very small percentage of the relation instances that are defined for its parent. Hence any sub-tree rooted at this node may be pruned off from further consideration while deciding the appropriate level of concept pair association for a relation. For our implementation this threshold has been kept at 10%.

Since a parent node has two alternative paths denoted by the expansion of LEFT and RIGHT respectively, along which a relation may be further specialized, the choice of appropriate level is based on the collective significance of the path composed of retained nodes. For each ANDed set of retained nodes, total information loss for the set is computed as the average information loss for each retained child. The decision to prune off a set of nodes rooted at N is taken as follows: Let information loss for nodes retained at $$L_1$$ is $$E_1$$ and that for nodes retained at $$L_2$$ is $$E_2$$.

- If $$E_1 = 0$$, then $$L_1$$ is retained and $$L_2$$ is pruned off, otherwise, if $$E_2 = 0$$ then $$L_2$$ is retained and $$L_1$$ is pruned off.
- Otherwise, if $$E_1 \approx E_2$$, i.e., $$\frac{\min(E_1, E_2)}{\max(E_1, E_2)} \geq 0.995$$ then both the sub-trees are pruned off, and the node N serves as the appropriate level of specification.
- Otherwise, if $$E_1 < E_2$$, then $$L_1$$ is retained and $$L_2$$ is pruned off. If $$E_2 < E_1$$ then $$L_2$$ is retained while $$L_1$$ is pruned off.

The set of concept-pairs retained are used for conceptualizing the relations.

D. Mapping Relation Strengths to Linguistic Variables

Since all relations are not equally frequent in the corpus, hence we associate with each conceptualization a strength $$S$$ which is computed in terms of relative frequency of occurrence of the generic relation in the corpus. Equation 7 computes this strength, where G denotes the category of concept-pairs: source-substance, source-source, substance-substance and substance-source. $$|T^G|$$ denotes the total count of all relations that are defined between ordered concept pairs defined in the tree $$T^G$$, and $$\hat{N}_{rg}^G$$ denotes the total number of relation instances of type $$r_g$$ mapped to $$T^G$$.

\[ \mu_{(C_i, C_j)}(r_g) = \frac{1}{2} \left\{ \frac{\# <C_i, r_g, C_j>}{\hat{N}_{rg}^G} + \frac{\# <C_i, r_g, C_j>}{\hat{N}_{rg}^G} \right\} \] .......(7)

Fig. 7. A plot of relation strengths and their %age counts for all four categories of trees
The basic task in designing the fuzzy membership functions is to identify the nature of the membership functions and the parameters for defining those functions. These parameters are derived from the graphs shown in figure 8. Figure 8 is obtained by normalizing the relation percentages. Each curve shows only one valley, and this common valley for all trees is observed at strength 0.4. Hence 0.4 is selected for defining the intermediate class “moderate”. The membership functions for the categories “weak”, and “strong” for each category are obtained through curve-fitting on different sides of the valley, while the membership function for class “moderate” is obtained by using the values surrounding 0.4. The fuzzy membership functions for categories “moderate” and “strong” are always characterized by Gaussian functions, whereas for the category “weak”, different types of functions are derived. The parameters for each type of tree are presented below.

The distribution of relation strengths for this tree is represented by the blue curve in figure 8. The membership function for fuzzy set “weak” and is represented by a linear curve whose common valley for all trees is observed at strength 0.4. Hence 0.4 is selected for defining the intermediate class “moderate”. The membership functions for the categories “weak”, and “strong” for each category are obtained through curve-fitting on different sides of the valley, while the membership function for class “moderate” is obtained by using the values surrounding 0.4. The fuzzy membership functions for categories “moderate” and “strong” are always characterized by Gaussian functions, whereas for the category “weak”, different types of functions are derived. The parameters for each type of tree are presented below.

The distribution of relation strengths for this tree is represented by the blue curve in figure 8. The membership function for fuzzy set “weak” is obtained as a quadratic equation given in 11. The membership function for the fuzzy set “moderate” and “strong” are represented by Gaussian functions defined in 9 and 10 respectively.

\[ \mu_{\text{weak}}(x) = a + bx + cx^2, \text{ where } a = 1.623, b = -6.987, c = 7.813 \ldots (8) \]

\[ \mu_{\text{moderate}}(x) = ae^{\frac{(x-b)^2}{2c}}, \text{ where } a = 1.013, b = 0.4, c = 0.082 \ldots (9) \]

\[ \mu_{\text{strong}}(x) = ae^{\frac{(x-b)^2}{2c}}, \text{ where } a = 0.315, b = 0.486, c = 0.062 \ldots (10) \]

The basic task in designing the fuzzy membership functions is to identify the nature of the membership functions and the parameters for defining those functions. These parameters are derived from the graphs shown in figure 8. Figure 8 is obtained by normalizing the relation percentages. Each curve shows only one valley, and this common valley for all trees is observed at strength 0.4. Hence 0.4 is selected for defining the intermediate class “moderate”. The membership functions for the categories “weak”, and “strong” for each category are obtained through curve-fitting on different sides of the valley, while the membership function for class “moderate” is obtained by using the values surrounding 0.4. The fuzzy membership functions for categories “moderate” and “strong” are always characterized by Gaussian functions, whereas for the category “weak”, different types of functions are derived. The parameters for each type of tree are presented below.

The distribution of relation strengths for this tree is represented by the blue curve in figure 8. The membership function for fuzzy set “weak” is obtained as a quadratic equation given in 11. The membership function for the fuzzy set “moderate” and “strong” are represented by Gaussian functions defined in 9 and 10 respectively.

\[ \mu_{\text{weak}}(x) = a + bx + cx^2, \text{ where } a = -0.013, b = 4.132, c = -9.211 \ldots (11) \]

\[ \mu_{\text{moderate}}(x) = ae^{\frac{(x-b)^2}{2c}}, \text{ where } a = 1.621, b = 0.359, c = 0.041 \ldots (12) \]

\[ \mu_{\text{strong}}(x) = ae^{\frac{(x-b)^2}{2c}}, \text{ where } a = 1.078, b = 0.524, c = 0.063 \ldots (13) \]
equation is given in 14. The membership functions for the fuzzy set “moderate” and “strong” are defined as Gaussian functions defined in equations 15 and 16 respectively.

\[ \mu_{\text{weak}}(x) = a + bx, \text{ where } a = 1.194, b = -2.194 \quad \ldots (14) \]

\[ \mu_{\text{moderate}}(x) = ae^{-\frac{(x-b)^2}{2c^2}}, \text{ where } a = 2.506, b = 0.357, c = 0.032 \quad \ldots (15) \]

\[ \mu_{\text{strong}}(x) = ae^{-\frac{(x-b)^2}{2c^2}}, \text{ where } a = 1.131, b = 0.476, c = 0.049 \quad \ldots (16) \]

Substance-Source Tree: The distribution of relation strengths for this tree is shown in cyan color in figure 8. Like source-substance and source-source categories in this case too, the member function for “weak” is derived as a quadratic equation and both the membership functions for “moderate” and “strong” are obtained as Gaussian functions. The membership functions of the fuzzy quantifiers “weak”, “moderate” and “strong” are given in equations 17, 18, and 19 respectively.

\[ \mu_{\text{weak}}(x) = a + bx + cx^2, \text{ where } a = -0.256, b = 5.987, c = -12.179 \quad \ldots (17) \]

\[ \mu_{\text{moderate}}(x) = ae^{-\frac{(x-b)^2}{2c^2}}, \text{ where } a = 1.629, b = 0.356, c = 0.037 \quad \ldots (18) \]

\[ \mu_{\text{strong}}(x) = ae^{-\frac{(x-b)^2}{2c^2}}, \text{ where } a = 1.061, b = 0.485, c = 0.045 \quad \ldots (19) \]

Table II shows top 5 relations mined from GENIA corpus and the associated generic concept-pairs along with fuzzy strength to reflect the degree of associations.

E. Enhancing Domain Ontology to a Fuzzy Relational Ontology

Since GENIA ontology stores information about biological concepts only, it cannot be exploited for representing biological interactions. Hence, we consider extending this ontology by adding the generic relations to this. It has been established earlier that generic relations are fuzzy in the sense that a relation can be defined between different concept-pairs with varying degrees of strength and vice-versa. This is best done through the use of linguistic qualifiers that express the strength of a relation to a varying degree. Thus rather than using a <concept-relation-concept> structure, we use the fuzzy relational ontology model described earlier which expresses a relation as <Ci, rG, Cj, \mu_{\text{Ci,j}}(r_G)>, where Ci and Cj are generic concept-pairs associated through rG and \mu_{\text{Ci,j}}(r_G) \in S represents the degree of association between concepts Ci and Cj. We have already shown how these strengths are derived and mapped to fuzzy quantifiers.

To accommodate generic relations and their strengths, in addition to existing GENIA ontology classes, the fuzzy GENIA relational ontology structure contains three generic classes - a “ConceptPair” class, a “GenericRelation” class and a “FuzzyStrength” class. The ConceptPair class consists of HasLeftConcept and HasRightConcept properties whose values are the instances of the GENIA concept classes. FuzzyStrength class has been defined to store the fuzzy quantifiers that can be associated with the generic relations to represent their strength. This class consists of a single property TermSet which is defined as a symbol and contains the fuzzy quantifiers “weak”, “moderate” and “strong”. The GenericRelation class has two properties – LeftRightActors and Strength. The LeftRightActors property is a kind of OWL object property whose range is bound to the ConceptPair class. This is also restricted to store exactly one value, an instance of the ConceptPair class, for every instance of a generic relation. The Strength property is also a kind of OWL object property for which the range is bound to the FuzzyStrength class. This property is also restricted to store exactly one value for every instance of the generic relations. All mined generic relations are defined as instances of the class GenericRelation. Figure 9 shows a snapshot of a portion of the enhanced Fuzzy GENIA relational ontology structure implemented by using Protege3.1.

VII. CONCLUSION AND FUTURE WORK

In this paper an ontology-based text information processing system is proposed to create a fuzzy ontology structure. The fuzzy ontology with fuzzy concepts and fuzzy relations is an extension of the domain ontology with crisp concepts and relations that is more suitable to describe the domain knowledge for solving the uncertainty reasoning problems. Though relations in a text co-occur with entities, the proposed system characterizes mined relations at generic concept level rather than at the entity level. Thus the mined set of relations is not likely to reflect any chance co-occurrences.

In this paper, we have also proposed a methodology to generate generic representation for inter-concept relations and enhance domain knowledge in terms of a fuzzy relational ontology structure. The generalization task is framed as an optimization problem over a AND-OR concept-pair tree. Since an ontology is not a database,
hence it should not be a store-house for relation instances. The proposed fuzzy relational ontology adheres to this principle and stores knowledge about the various categories of relations occurring in the corpus at appropriate levels of conceptualization rather than every instance of relation mined. The strengths of the relations are expressed as fuzzy membership values to categories WEAK, MODERATE and STRONG, where the membership value reflects likelihood of observing a particular association in a corpus. The mined relations can be used to formulate context-based queries at multiple levels of specificities and answer them intelligently. A glimpse of the experimental results for both general-purpose as well as technical domains has been provided. Presently, we are developing a query answering module in line with [12] to answer fuzzy queries over text documents. Extension of the ontology structure into a rough-fuzzy ontology is also being studied.

REFERENCES


Muhammad Abulaish was born in India on August 4, 1971. He has obtained his Master degree in computer science and applications from Motilal Nehru National Institute of Technology, Allahabad, India. Later, he did Ph.D. in computer science from Indian Institute of Technology Delhi, India.
He is working as a Reader in the Department of Computer Science, Jamia Millia Islamia, New Delhi, India. He has more than 10 years Computer Science teaching experience at undergraduate and postgraduate levels. He has published over 24 research articles for International Journals, Books and Conference Proceedings. His research interest spans over the areas of Data Mining, Text Mining, Information Retrieval, Web Intelligence, and NLP.

Dr. Abulaish is a member of IEEE and its Computer society. He is also a life member of Computer Society of India, Indian Society for Technical Education, Indian Science Congress Association, and the Institutions of Electronics and Telecommunication Engineers. He is an IPC member of several notable conferences. He also serves as a reviewer for various journals including IEEE Transactions on Knowledge & Data Engineering, IEEE Transactions on Computational Biology and Bioinformatics, Web Intelligence and Agent Systems – an International Journal, and Journal of Emerging Technologies in Web Intelligence.