Journal of Universal Computer Science, vol. 17, no. 7 (2011), 1021-1042
submitted: 30/10/10, accepted: 15/3/11, appeared: 1/4/11 © *J.UCS*

Towards Classification of Web Ontologies for the Emerging Semantic Web

Muhammad Fahad
(DISP Laboratory and CERRAL Center
Université Lumière of Lyon, Bron, France
muhammad.fahad@univ-lyon2.fr)

Nejib Moalla
(DISP Laboratory and CERRAL Center
Université Lumière of Lyon, Bron, France
nejib.moalla@univ-lyon2.fr)

Abdelaziz Bouras
(DISP Laboratory and CERRAL Center
Université Lumière of Lyon, Bron, France
abdelaziz.bouras@univ-lyon2.fr)

Muhammad Abdul Qadir
(Center for Distributed and Semantic Computing
Mohammad Ali Jinnah University, Islamabad, Pakistan
aqadir@jinnah.edu.pk)

Muhammad Farukh
(CERRAL Center
University of Lumière Lyon2, Bron, France
muhammad.farukh@univ-lyon2.fr)

Abstract: The massive growth in ontology development has opened new research challenges such as ontology management, search and retrieval for the entire semantic web community. These results in many recent developments, like *OntoKhoj*, *Swoogle*, *OntoSearch2*, that facilitate tasks user have to perform. These semantic web portals mainly treat ontologies as plain texts and use the traditional text classification algorithms for classifying ontologies in directories and assigning predefined labels rather than using the semantic knowledge hidden within the ontologies. These approaches suffer from many types of classification problems and lack of accuracy, especially in the case of overlapping ontologies that share common vocabularies. In this paper, we define an ontology classification problem and categorize it into many sub-problems. We present a new ontological methodology for the classification of web ontologies, which has been guided by the requirements of the emerging Semantic Web applications and by the lessons learnt from previous systems. The proposed framework, *OntClassifire*, is tested on 34 ontologies with a certain degree of overlapping domain, and effectiveness of the ontological mechanism is verified. It benefits the construction, maintenance or expansion of ontology directories on the semantic web that help to focus on the crawling and improving the quality of search

for the software agents and people. We conclude that the use of a context specific knowledge hidden in the structure of ontologies gives more accurate results for the ontology classification.

Keywords: Ontology classification and retrieval, Semantic matching, Ontology searching, Web page Classification, Semantic web portals
Categories: H.3.2, H.3.3, H.3.7, M.3, M.7

1 Introduction

The Semantic Web provides virtual communities that enable software agents and users to extract, use and share knowledge. It uses the notion of ontologies for the conceptualization and elicitation of the domain knowledge and stores it in terms of concepts and properties in a machine understandable and processable manner [Berners-Lee, 01]. Due to their capacities of decidability and expressiveness, ontologies have played a fundamental role for describing the semantics of data not only in the emerging semantic web but also in traditional knowledge engineering, and act as a backbone in the knowledge-base and semantic-based information processing systems. Several tasks such as information storage, processing, retrieval, decision making, etc., are done on the basis of ontologies by such systems. But, significant increase in the number of ontologies being developed and maintained over the web demands various new techniques for the ontology storage, classification, ranking, search and retrieval. Similar to the current web, searching the relevant knowledge is one of the main problems for the emerging Semantic Web. Thus, for the realization of Semantic Web vision, there have been a lot of more efforts needed to fulfil the promises of high precision by the use of available semantics and reasoning on the ontologies, as well as retrieval of precise results with rank and relationships between them [Berners-Lee, 06]. This dilemma requires the proper classification of web ontologies, which is also essential for many other tasks such as the development of ontology directories on the web [Dmoz, 07], focused crawling for ontology retrieval [Ehrig, 05; Su, 05], concept specific modular ontology analysis [Seidenberg, 06], improving the quality of search [Pan, 06], etc.

Classification is traditionally defined as a supervised learning problem in which a set of labelled data is used to train a classifier that can be used to label future examples [Mitchell, 97]. Ontology classification is a challenging classification problem for the efficient and effective ontology management and retrieval for the Semantic Web and enterprise ontology based business applications. Prior to the ontology classification, much work has been done for the web page classification that aims at assigning a web page to one or more predefined category labels [Chakrabarti, 02]. The current web is a heterogeneous infrastructure containing unstructured or semi-structured data of various types. This opens up a number of other classification research problems like web site classification [Pierre, 01; Ester 02; Qi, 06], web page classification [Peng, 02; Glover, 02], blog classification [Qu, 06; Mishne, 06], multimedia data classification [Bosch, 07; Zhang, 07], etc. In past, search engines have used the flat documents that are without structure on the basis of bag-of-words mechanism for searching and retrieval. The drawback of such mechanism is that the words within a document are considered to have the same relevance or value, without

considering their role, frequency of occurrences or importance within the document. Therefore, keyword matching based on the bag-of-words mechanism compromises accuracy and information retrieval results. The experiences and advancement of technology make realization to use the structure within the documents. Therefore, many different techniques are proposed to compute the significance of phrases within the document (e.g., title, headings, abstract, summary, tags, etc.) [Golub, 05; Nie, 06]. On the basis of computed significance of phrases, ranking algorithms are applied to find the best match. The use of structural analysis has improved the quality of search and brought out higher precision, but has been overlooked for the classification of semantic web documents. Research challenges for the Semantic web document classification can be elaborated as ontology classification, RDF repository classification, lightweight/heavyweight ontology categorization, etc. Now-a-days, for a specific domain, there are several ontologies available which were developed by the different communities according to their requirements. Therefore, multiple ontologies associated with a same domain/concept appear to be quite common on the Semantic Web. For example, as mentioned in one of the research studies about the development of semantic web portal, *Swoogle* searches over *300* distinct terms that appear to stand only for the '*Person*' concept [Ding, 05]. It is likely that large and complex ontologies will require a novel solution and central index of ontologies for fulfilment of the sound Semantic Web vision.

Recent years have seen many semantic web portals, such as *OntoKhoj*, *Swoogle*, *OntoSearch2,* for the ontology searching, ranking and classification. But, these existing approaches exploit keywords, phrases and terms about the ontologies rather than the semantic knowledge hidden within the structure of ontologies for their classification. The consequence is that the semantics of knowledge is not understandable by the machine and becomes a bottleneck in the process of ontology searching and retrieval on the web. This requires new approaches for the ontology classification based on the structural knowledge and semantics analysis to meet the requirements of the emerging Semantic Web. Thus, our main idea behind this work is to replace the plain text classification algorithm in the process of ontology classification with the ontology specific classification algorithm. The proposed approach uses the category ontology rather than bag-of-words for the classification of arbitrary ontologies, and structural analysis of knowledge hidden in the ontologies.

Rest of the paper is organized as follows. We discuss current approaches for the ontology classification, searching, and retrieval in section 2. We define an ontology classification problem and categorize it into many sub problems in section 3. Following, we discuss ontology classification mechanism that fulfils the demands of ontology classification for the emerging Semantic Web. We elaborate the methodology of ontology classification which is exploited by our ONTology CLASSIFIcation and REtrieval (*OntClassifire*) component in section 4. The same section presents our preliminary experiment results about the ontology classification, and usage of our *OntClassifire* in several tasks. Finally, section 5 concludes the paper and shows our future direction.

2 Related Work

There are many applications that make use of ontologies for the classification of web documents, emails, text files and many other tasks for the knowledge management and retrieval. In this regard, Grobelnik and Mladenic (2005) propose a simple approach by exploiting the content of document and information about the web page context which is obtained from the link structure of the web for the classification of web documents into the large topic ontology. For the classification of emails, Taghva et al. (2003) propose an ontology-based system that makes an ontology which later on applies rules for the identification of features to be used for the classification of emails. From the training set of emails, associated probabilities for features are calculated and used as a part of the feature vectors for an underlying Bayesian classifier. For the ontology-based text categorization, Wu et al. (2003) describe a methodology in which the domain ontologies are automatically acquired through the morphological rules and statistical methods. Reich et al. (2002) propose an ontology-based skill management system for the efficient access to people's capabilities and their profiles. The heart of the system is based on three ontologies, each designed for the *skills*, *education*, and *job*, which act as a fundamental technology for exploring the individual's specific skills and competencies. Another, ontology-based semantic match making of skills description is proposed by Colucci et al. (2003).They also formalize the *skills* ontology for the demand and supply of skills between the demanders and sellers. The role of ontologies is magical in classifying the objects and improving the quality of search in various applications, but an ontology classification itself is also a demanding problem which should be addressed in a semantic way for its own efficient management and retrieval for the emerging Semantic Web.

One of the semantic web portals that facilitates the ontology searching and classification is *Ontokhoj* that allows engineers and software agents to retrieve trustworthy ontologies, and expedite the process of ontology engineering through extensive reuse of ontologies [Patel, 03; Supekar, 03]. It is designed for the crawling, classification, ranking and searching ontologies on the Semantic Web. For ranking, it exploits and extends the strategy of ranking based on the citations as used by the Google's PageRank algorithm and employs the semantic links denoted by the instantiations and subsumptions between the ontologies. For the classification of web ontologies, it treats the ontology as a plain text and uses the text classification algorithms for its classification. The use of text classification algorithm for the classification of Semantic Web ontologies is the biggest drawback especially for the overlapping ontologies that share common vocabularies, because the plain text classification algorithms only use keywords that result in the poor performance and hence the classifier's accuracy is compromised.

Another semantic web search engine is *Swoogle* which is based on the metadata engine and retrieval system for the semantic web documents [Ding, 05]. It makes the use of multiple crawlers to find the semantic web documents, index them and provides a keyword based querying facility to its large repository of the semantic data. It is widely used by the semantic web community, but, it does not address the problem of ontology classification. *Ontosearch2* is another ontology search engine developed to address the problem of finding ontologies appropriate for the desired domains [Pan, 06]. It makes the use of semantic entailments for the searching rather

than only using the keywords or metadata like *Swoogle* and *Ontokhoj*. It also provides a restricted query interface with the only keyword search, and uses the citations to an ontology or links to an object within the *Abox* (Assertional box) of the ontology for its ranking. It provides the *Tbox* (Terminological box) and *Abox* searching mechanisms with some other search directives by allowing restrictions on the search query, and performs search on the desired portions. *Watson*, aimed at addressing the limitation of *Swoogle* that adopts web centric approach, provides an entry point to access the huge amount of semantic data in an intelligent way [Aquin, 07]. It exploits multiple strategies which automatically collect, analyse, and index the semantic data and ontologies that are published on the web.

Besides these web portals, there are some developments that aid in an ontology sharing and selection. *OntoSelect* is a dynamic ontology library that facilitates access to the web ontologies by natural language mechanisms [Buitelaar, 04]. *Ontolingua* is another important contribution that provides user a distributed collaborative environment to browse, edit, create, modify and use ontologies [Ontolingua, 10]. It requires user to register and then perform the desired task. Oyster aims at sharing web ontologies to a peer-to-peer network [Palma, 2006]. Developers, first, register and share ontologies with their metadata, which are later on access by the ontologist community over the local registries. There are some other approaches that are developed for finding, collecting and indexing the semantic web documents on the web such as Eberhart's (2002) RDF crawler, DAML Crawler [Dean 2006], and MultiCrawler [Harth, 2006].

From the research literature on the semantic web portals, we analyze that most of the works have not addressed the problem of ontology classification. Only *OntoKhoj* addressed this particular problem and used the traditional algorithms of plain text classification for classifying web ontologies in the directories by assigning predefined labels. It has implemented Naive bayer's text classification algorithm for the classification of web ontologies that treat the ontologies with complex structure and semantics as plain texts. This is the main reason due to which these web portals suffer with many types of classification problems and lack of accuracy in the ontology searching and retrieval, especially in the case of overlapping ontologies. For example, classifying *EE Department* ontology in *Electrical Engineering* domain or *Electronic Engineering* or *University* domain requires an ontology specific keen algorithm and in depth knowledge analysis on the structure and semantics rather than a simple text classification algorithm. As the Semantic Web gains momentum with the explosive number of ontologies, where multiple ontologies associated with a same domain/concept appears to be quite common, it is of immense importance to classify them into respective domain hierarchies. It helps humans and web agents to find the correct and desired ontology (or concept) on the web and supports the ontology engineering processes.

In order to meet the real challenge of ontology searching and retrieval, we built an ontology based approach for the ontology classification that facilitates such tasks [Fahad, 10]. We believe that once the ontologies are properly classified, then they are searched in a sound semantic manner in an ontology based application or on the Semantic Web. For building, *OntClassifire*, we benefit from our existing approach of ontology matching and merging [Fahad, 07] with several modifications. An ontology based approach works better for the overlapping ontologies that come across due to

the semantic heterogeneities and structure requirements during the modeling of domain knowledge. Due to the use of more semantic and structural knowledge within the ontologies, our approach of *OntClassifire* enhances the accuracy of ontology classification and provides an efficient access to the huge amount of knowledge content for the Semantic Web users.

3 Ontology Classification- Background and Problem Description

Many Semantic Web related technologies have been emerging after the introduction of Semantic Web concept by Tim Berners-Lee in 2001. One of the most prominent developments from these efforts is the status of ontology development languages. After intensive work on semantic standards, World Wide Web Consortium (W3C) has standardized an Ontology Web Language (OWL) in 2004. Now, OWL appeared as a mature language for the development of semantic contents and become a key technology for establishing the enterprise semantic applications. OWL, with its sparkling power of decidability, has implied a significant leverage of the Semantic Web from a research level to an industry standard for building next generation applications. OWL Ontology contains a lot of structural and contextual information in terms of different constructs, e.g., classes, datatype properties, object properties, parent-child relationships, description logic (DL) axioms, etc. Due to their expressive nature, they are more than the text documents or HTML web pages. Therefore, the plain text classification techniques [Ghani, 02; Gabrilovich, 04] that benefit the document or web page classification are not much useful for the ontology classification and searching on the Semantic Web. For this reason, an ontology classification is not only important, but also distinguished from the traditional classification techniques, and thus deserves more efforts of research.

As very little work has been done particularly for the ontology classification in the research literature, therefore we define specific terms about the ontology classification for promoting understandability based on the terminology used in the area of web page classification. The general problem of ontology classification can be divided into more specific problems depending upon the number of classes in the problem of interest, the domain knowledge modeled within the ontologies, and the number of classes that can be assigned to an ontology instance. Following various ontology classification sub-problems are defined with examples.

1. Based on the number of classes in the problem, the classification can be divided into binary, ternary, or multiclass ontology classification. Binary ontology classification categorizes the instance ontologies into exactly one of two classes. Figure 1 shows an example of binary ontology classification that determines the type of ontologies whether they are hierarchical ontologies (i.e., basic RDF ontologies) or expressive ontologies having DL axioms (i.e., OWL ontologies). Multiclass ontology classification associates the instance ontologies with more than two classes.
2. Based on the number of classes that can be assigned to an instance ontology, the classification can be divided into a single-label or multi-label ontology classification. Former strategy deals with assigning one and only one class label

to an instance ontology, but the later deals with assigning more than one class to an instance ontology.

3. Based on the type of class assignment, the ontology classification can be divided into a hard or soft ontology classification. Hard ontology classification determines whether an instance can either be or not be in a particular class. In soft ontology classification, an instance can be predicted to be in some classes with some likelihood and often a probability distribution across all the classes. Figure 2 shows an example of soft ontology classification that calculates rank to classify an instance ontology across all the classes of domain, i.e., *Journal*, *Proceeding*, *Magazine* and *Book*.

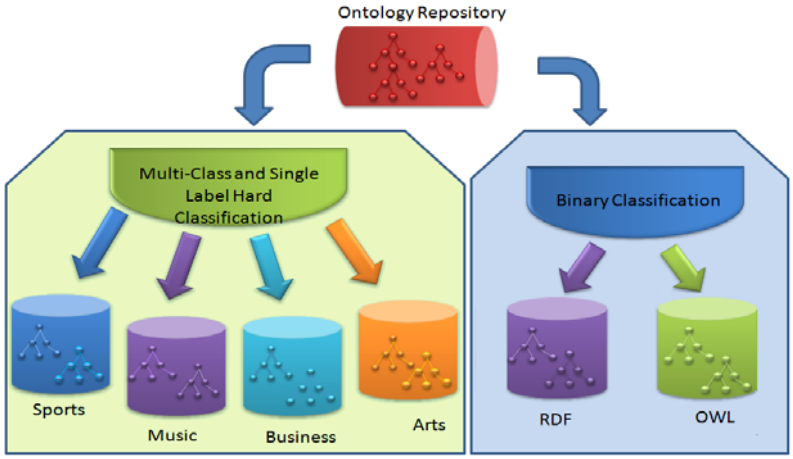

Figure 1: Examples of Ontology Classification (a) Multi-Class, Single Label and Hard Ontology Classification, (b) Binary Ontology Classification

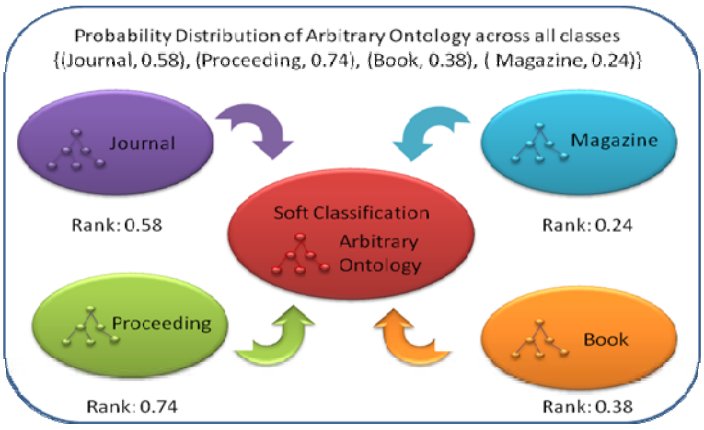

Figure 2: Example of a Soft Ontology Classification

4. Based on the domain knowledge modeled within the ontologies, the classification can be divided into subject, functional and sentimental ontology classification. Subject ontology classification categorizes the ontologies depending on their domain and topic, e.g., art, disease, business, sports, etc. Functional ontology classification determines the role that the ontologies play, e.g., admission ontology, personal home page ontology, patient examination ontology, etc. Sentimental ontology classification determines the messages or opinion that is presented in the ontologies, e.g., message between the business processes or stock exchange conditions, interaction between the multi-vendor semantic systems, author's attitude in the blog ontology, etc.
5. Based on the organization of categories, an ontology classification can be taken as a flat classification scheme or hierarchical classification. In a flat ontology classification, all the categories are considered as parallel. Hierarchical ontology classification deals with the categories that are organized in a hierarchical tree-like structure, in which each category may have a number of subcategories. Figure 3 shows an example of a flat versus hierarchical ontology classification.

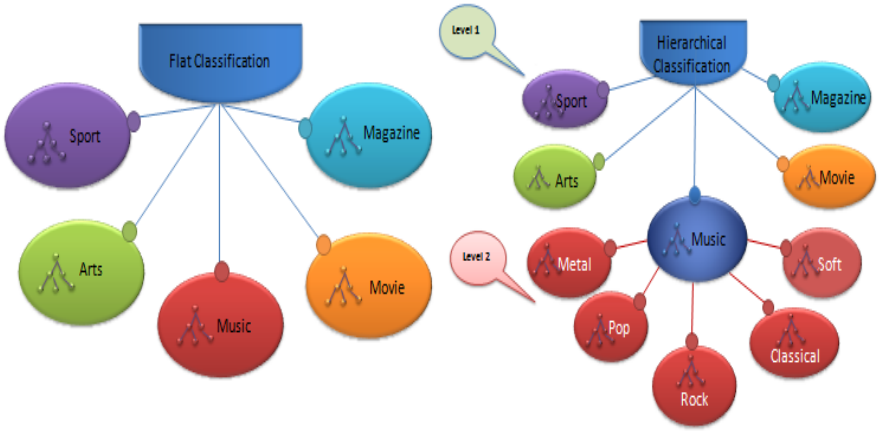

Figure 3: Flat versus Hierarchical Ontology Classification

4 OntClassifire – A Semantic based Ontology Classifier

This section presents our semantic based ontology classifier, *OntClassifire*, and discusses the semantic similarity computation for the ontology classification between the domain and arbitrary ontologies. It aims at classifying the arbitrary ontologies in one or more predefined categories for the efficient ontology management and search. For each predefined category, we assume that there is a representative domain ontology rather than bag-of-words which is used for the classification purpose. *OntClassifire* matches the domain ontology with the arbitrary ontologies and calculates a match rank. As the domain ontologies are specific to the particular domain and henceforth capture most of the common terminologies about that domain, they require a soft classification mechanism for their classification. Therefore, in order to meet the needs, we adopted a soft classification approach that is very much

helpful in the case of overlapping ontologies where an instance ontology is predicted to be in some classes with some likelihood, with a probability distribution across all the classes. For example, if we assume there are only four predefined categories, it specifies *MatchRank{(Thesis_ontology, 0.2), Journal_ontology, 0.3), (ScientificMag_ontology, 0.38), (ConferenceProceeding_ontology, 0.87)}* as *Match Rank* for the multi-class soft ontology classification of an arbitrary instance ontology O_a across all the domain ontologies of interest. When the match rank is found above the threshold value, the specified predefined label is associated with an arbitrary instance ontology. However, the calculated match rank across all the ontologies is stored in the knowledge base for further assistance to the application or human user. It calculates a match rank on the basis of an ontology matching algorithm, and in this way results in more accurate classification of the arbitrary ontologies as the context of concepts, properties and structure of knowledge is matched and analyzed. It exploits the existing schematic matching techniques (i.e., linguistic, synonym and axiomatic) for the calculation of match rank. We are working with the OWL ontologies; however the methodology can be applied for the similarity computation and classification of other ontologies as well. The following sub-sections elaborate the methodology, show its usage, and discuss the experiment results.

4.1 Match Rank Calculations by *OntClassifire*

The *OntClassifire* gets an arbitrary ontology O_a for the classification purpose. It starts the semantic similarity computation between the O_a and the domain ontologies $\{O_{d1}, O_{d2},\ldots,O_{dn}\}$belonging to the predefined categories. For the similarity computation, each of the concepts in ontologies is analysed. For example, a concept '*Book*' is judged on the basis of its label, attributes (e.g., *ISBN*, *Title*), relations (e.g., written by author, published by publisher) and its semantic neighbourhood (e.g., parent and children concepts), as shown in the Figure 4. Therefore, the proposed model employs all the syntactic, structural and semantic knowledge present in the ontologies to compute the match rank so that an arbitrary ontology should be assigned with an accurate pre-defined category label.

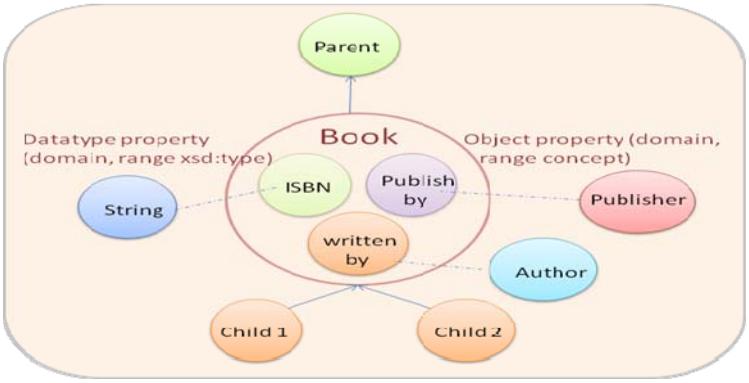

Figure 4: Different Constructs associated with a Single Concept 'Book'

Let two ontologies be O_d and O_a, for matching a concept *c* of O_d with the concept *c'* of O_a, it exploits many inter-ontology semantic similarity parameters to compute whether how much a concept *c* is similar to *c'* as represented by the equation 1. Finally, *OntClassifire* aggregates the concept similarities found between the ontologies, calculates the match rank, and assigns a label to an arbitrary ontology O_a.

$$Sim\ (c, c') = \alpha Lcc' + \beta Dcc' + \gamma Occ' + \mu Pcc' + \Theta Hcc' + \Omega Acc' \qquad (1)$$

Where,

Lcc': Concept label similarity between *c* and *c'*
Dcc': Datatype properties similarity between *c* and *c'*
Occ': Object properties similarity between *c* and *c'*
Pcc': Parent concepts similarity *c* and *c'*
Hcc': Children concepts similarity *c* and *c'*
Acc': DL Axiom similarity between *c* and *c'*

Once the similarities between the concepts of domain ontology O_d and an arbitrary ontology O_a are calculated, *OntClassifire* then calculates a match rank between the ontologies O_d and O_a by aggregating the weights of concepts. Let a category ontology O_d has n concepts then the match rank between the ontologies is calculated by the equation 2.

$$MatchRank(O_d, O_a) = \frac{\sum_{i=1..n} Sim\ (c, c')_i}{n} \qquad (2)$$

As the overlapping categories share the common vocabularies, hence weights with the concepts of domain ontologies dominate the specific attributes of each category. The user can configure these weights (α, β, γ, μ, Θ, Ω) that value the parameters of semantic similarity. Moreover, the user can adjust the weights for the linguistic similarity and synonym similarity accordingly. The parameters involved in the match rank calculation are elaborated as follows.

Concept Label Similarity (Lcc'). Label of a concept is highly significant and comprises the utmost weight in the description of concepts. Lcc' computes linguistic and synonym based correspondences between the labels of concepts *c* and *c'* of ontologies O_a and O_d. Linguistic similarity finds the string based correspondences between labels, which are calculated by the edit distance [Levenshtein, 1966]. Synonym similarity is computed based on the lexical database *Word Net* [Miller, 1995] that helps to detect the concepts which have the same meanings but are linguistically different. For example, concepts that are synonyms (e.g., c_1:*Student*, c_2:*Scholar*) and abbreviations (e.g., c_1:*InformationTechnology*, c_2:*IT*) are determined by this way.

Concept Properties Similarity (Dcc' and Occ'). Datatype properties and Object properties in an ontology represent the context and semantics of concepts. Generally, datatype properties are called the attributes of a concept in the ontology. Object properties or relations make direct and reciprocal links between concepts within an ontology. For example, consider the ontology in Figure 5. Object properties *Contributes*(*Author*, *Paper*) and *isReviewedBy*(*Paper*, *PcMembers*) make associations between the concepts and represent the real descriptions, which help the

classification algorithm to judge the real category of an arbitrary ontology. In OWL-DL ontology, these properties comprise of four things; (i) Domain concept, (ii) Property Label, (iii) Range of property, and (iv) Tags associated with a property. For example, concept *Book* has a datatype property *ISBN* of type string with *Functional* and *Inverse-Functional* tags. Here, domain concept (*Book*) represents the concept to which this property belongs. *OntClassifire* does not match domain concepts of datatype property as it has already computed by concept label similarity. It matches other three things, i.e., concept datatype similarity (DP_{sim}) checks a label of datatype property (Ld) based on linguistic and synonym strategy, Range type (Rd) and Tag (Td) associated. Let *nc* and *nc'* are the number of datatype properties associated with concepts *c* and *c'*, and *d* be the similar datatype properties between them. Total datatype similarity (Dcc') between concept *c* and *c'* is calculated by the equation 3.

$$DP_{sim} = Sim(Ld + Rd + Td)_{Od, Oa}$$
$$Sim\ (DP_1, DP_2) = dcc' = DP_{sim}$$
$$Dcc' = \sum_{i=1..n} dcc' = \frac{(d/nc + d/nc')}{2} \qquad (3)$$

OWL provides four types of tags, i.e., Functional, Inverse Functional, Symmetric and Transitive. With datatype properties only first two are applicable. Similarly, semantic similarity between the object properties (Occ') is computed between the concepts of domain and arbitrary ontologies.

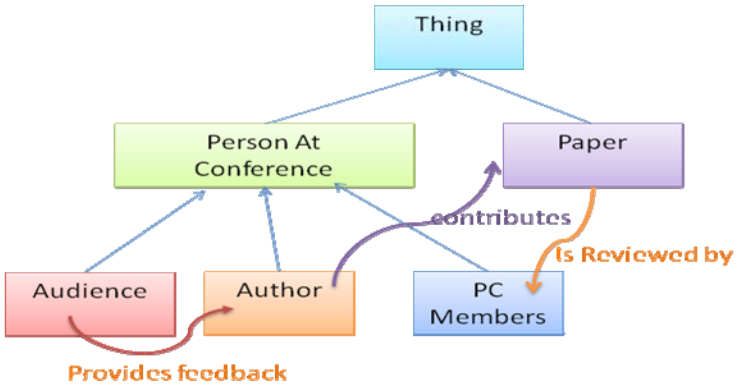

Figure 5: Associations between different Concepts by Object properties

Concept Parent and Children similarity (Pcc' and Hcc'). An OWL ontology starts from a top concept *Thing* that captures everything. It also allows multiple inheritance, therefore parent similarity requires computation of correspondences between all the parent concepts. Pcc' analyses whether the parents of concept *c* and *c'* are semantically similar or not, and Hcc' checks for their children concept similarity. For example, let concept *c* and *c'* has *d* similar parent (or children) concepts, and *nc* and *nc'* be the number of parent (or children) concepts of *c* and *c'* accordingly, Pcc' (or Hcc') similarities are computed by the equation 4.

$$Pcc' = \sum_{i=1..n} Pcc' = \frac{(d/nc + d/nc')}{2} \qquad (4)$$

DL Axiom Similarity (Acc'). OWL classes are described through the so-called class descriptions that enrich the background information of the concepts and represent the constraints of real world situations. OWL ontology also supports unnamed classes that are formed by the set of restrictions on the values for particular properties of the concepts. Such class descriptions are equivalent to the Description Logic (DL) axioms, e.g., a *Publication* concept can be represented as *{Thesis* ∏ ∃ *WrittenBy.Student}* or *{Paper* ∏ ∃ *ReviewedBy.Committee* ∏ ∃ *>8 haslimit.Pages}* accordingly to its context. Figure 6 shows an owl ontology syntax with an equivalent DL axiom of *Publication* concept. DL axioms define the context of concept that helps a lot in finding the accurate semantic similarity between the concepts of ontologies. It links the concept by different means that depict the concept's real semantics. It can be formed from union, complement, intersection and restriction operators applied on the primitive concept or anonymous concept and/or by boolean combinations. Some of the boolean combinations of primitive and anonymous restrictions with different operators are shown in the Figure 7.

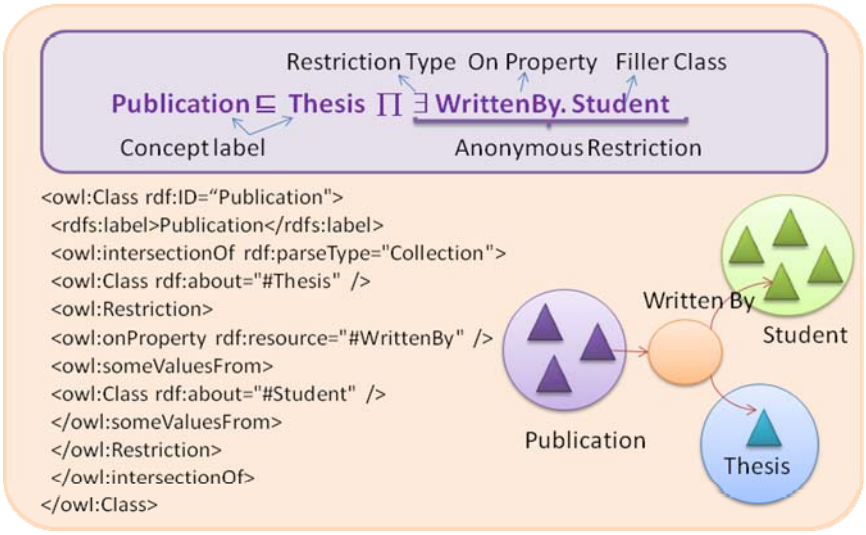

Figure 6: Complex Concept Axiomatic Definition made by intersection between Primitive concept and Anonymous Restriction class

DL axiom similarity checks the expression of concept description formed by other concepts (primitive/anonymous/combination) and operators between them. In case of anonymous concept, similarity constitutes of correspondences between restriction type, object property involved and range concepts that act as filler class. Restriction analysis for semantic similarity between concepts is highly significant as it

defines the necessary conditions or necessary and sufficient conditions for them. The necessary condition of a class makes that class a subclass of the restriction class. In case of necessary and sufficient condition for the class, both the restriction class and the restricted class will be interpreted as equivalent, i.e., they always have exactly the same members. In a case, where class description is in the form of union or intersection between concepts, DL axiom similarity tokenizes the class description into the set and performs matching between the operands, i.e., the elements of the set. Thus, the semantic similarity between DL axioms is calculated from the number of matches between the element of two sets (S1 and S2) that belong to concepts *c* and *c'* respectively, by the equation 5.

$$\mathbf{Acc'} = \frac{(\,|S1| \prod |S2|\,)}{|S1|} \qquad (5)$$

This calculated axiomatic similarity is an asymmetric measure that determines the extent to which the knowledge of a category ontology O_d is covered by the arbitrary ontology O_a. It is obvious that the category ontology has limited knowledge particular to the specific category, but the arbitrary ontology may cover, extend or have plenty of other concepts. This leads to difference in values of Acc' and Ac'c. For example, when *c'* concept contains many primitive concepts in DL axiom but matches all the primitive concepts in DL axiom of *c*, then the value of Acc' is equal to 1, but Ac'c may approach zero. Here, we are interested in knowing how much an arbitrary ontology covers the axioms of the category ontology, so that an arbitrary ontology is assigned an appropriate category label. Thus, the axiomatic analysis of concepts of OWL ontologies increases the ability of classifier to make more accurate reasoning on the concepts for their semantic similarities and then for the ontology classification.

Figure 7: Boolean combination of different operators for class descriptions

All these inter-ontology factors form a similarity computation between the concepts *c* and *c'* and generate the aggregated similarity or match rank between the domain and arbitrary ontologies. But, it requires exhaustive analysis for the similarity computation where each concept *c* of the domain ontology O_d is matched with each concept *c'* of arbitrary ontology O_a. Therefore, there is a tradeoff between efficiency and effectiveness of a matching process that results in a reliable classification. Performing the exhaustive computation with several factors for each concept, increases the effectiveness as this enables to identify the maximum possible similarities between the concepts; which is significant in case of overlapping domains. But, applying these factors phase by phase and selecting the candidate concepts after each phase based on the weighted concepts and properties of domain ontologies reduces the exhaustive computation for each concept, which minimizes the run time complexity of ontology matching and hence for the ontology classification.

4.2 Prioritization of different factors for Match Rank Calculations

Various constructs within the ontology have some inherent semantics that could be exploited to determine the specific category to which it belongs. Therefore, we prioritized such semantics based on the intuitive reasoning. An axiom of the concept depicts the real semantics by connecting different concepts via properties and applies restrictions on primitive concepts, hence given the highest weight. Concept relations or Object properties (with the domain and range concepts) are on second priority as they show the direct associations between the concepts. Weights for the labels of individual concepts and properties are assigned low values, as the labels only can depict any context of the concepts and can be found common in the overlapping domain ontologies but differ in the class descriptions.

4.3 Preliminary Experiment Results

We have performed experiment to determine effectiveness of the ontological approach for the classification of web ontologies with the simple and weighted ontology approaches, and compared these with the popular *Naïve Bayes* classification criterion that was selected as a best algorithm for the *OntoKhoj*. The *Naive Bayes* classifier is a simple probabilistic classifier with the strong independence assumptions. It assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature. From its independent feature model, it is inferred that the words are not dependent on the length of the document, position within the document with relation to other words, or the other document-context. But, our ontological approach differs from its independent feature model and measures the classification mechanism on the basis of combined feature model especially analyzing context. Below, we provide the detail by an example scenario illustrating the working of *OntClassifire*, and comparison with the text classification approach for the classification of web ontologies.

Example Scenario from the Case Study. For conducting the experiment, first, we built the hierarchical category ontology that contains several categories, e.g., *University*, *ComputerScience_Department* and *Publication*. The *Publication* category is further classified into many subcategories, such as, *Book*, *Conference_Proceeding*,

Journal, *Thesis*, *Magazine*, *NewsPaper*, etc. Second, each of the categories is elaborated with the domain ontology that enriches the semantics and differentiates the categories themselves. Figure 8 shows the fragment of the category ontology, and the domain ontologies of two categories, i.e., *Book* and *Conference_Proceeding*.

These categories are overlapping and therefore the domain ontologies share common vocabulary in terms of the concepts (e.g., *Author*, *Publisher*, etc.), properties (*ISBN*, *Title*, *Price*, etc.) and relations (e.g., *collectionOf*, *formatType*, etc.) between them. Therefore, the differentiated axioms, concepts and properties between these categories are assigned weights in the domain ontologies so that the classification can be done more accurately on the basis of specific differentiating aspects of each category. For example, an axiom (*Academic_Papers* subsumed by ∃ *peerReviewedBy*. *PCMembers* ∏ ∃ *PresentedAt.Conference*), concepts (*Academic_Papers*, *Organizing-Committee*, *PCMembers*, *Conference*, etc.) and properties (*presentedAt*, *peerReviewedBy*, *feedback*, etc.) differentiate the category *Conference_Proceeding* from the category *Book*, and hence assigned with some more weights.

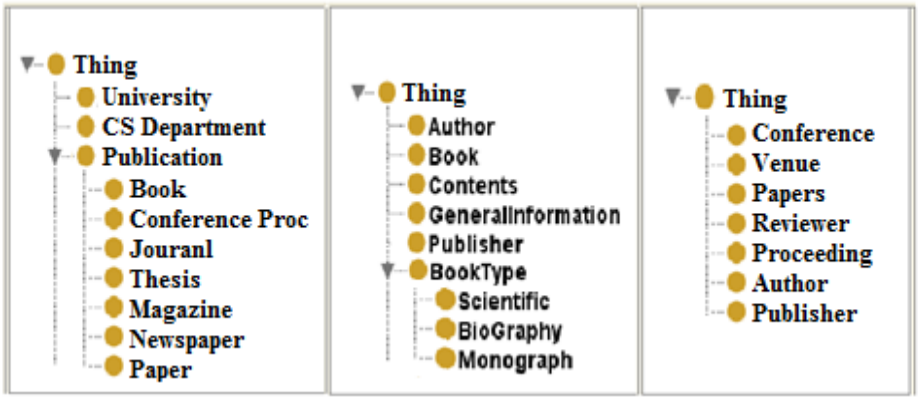

Figure 8: Fragments of ontologies, (a) Category ontology, (b) Book domain ontology and (c) conference proceeding domain ontology

When the arbitrary ontology O_a (as shown in the Figure 9) comes, *OntClassifire* computes the similarities between the domain ontologies and arbitrary ontology, as shown in the Table 1. Finally, on the basis of calculated highest match rank, an arbitrary ontology is assigned a label *Conference_Proceeding (proc.)*, and the match rank is preserved in the knowledge base which would be used for future perspective of query answering for ontology retrieval. From this experiment, we conclude that different parameters contribute different values in judging the correspondences between the ontologies. There is not a single parameter that is regarded as effective when ontologies are developed by different communities with different perspectives and requirements about the domain. Only the hybrid matching based on the combined similarity measures between the labels, attributes, relations and class descriptions can produce the best result.

Sim/Ont	**O_a,O_{pub}**	**O_a,O_{book}**	**O_a,O_{proc}**	**O_a,$O_{journal}$**	**O_a,O_{thesis}**
Lcc'	0.53	0.41	0.96	0.52	0.47
Dcc'	.066	0.46	0.87	0.53	0.60
Occ'	0.54	0.30	0.78	0.38	0.36
Pcc'	0.29	0.21	0.66	0.29	0.26
Hcc'	0.41	0.23	0.76	0.44	0.33
Acc'	0.33	0.11	0.86	0.26	0.21
Aggregated	0.458	0.286	0.815	0.403	0.374

Table 1: MatchRank between an arbitrary ontology O_a and the domain ontologies

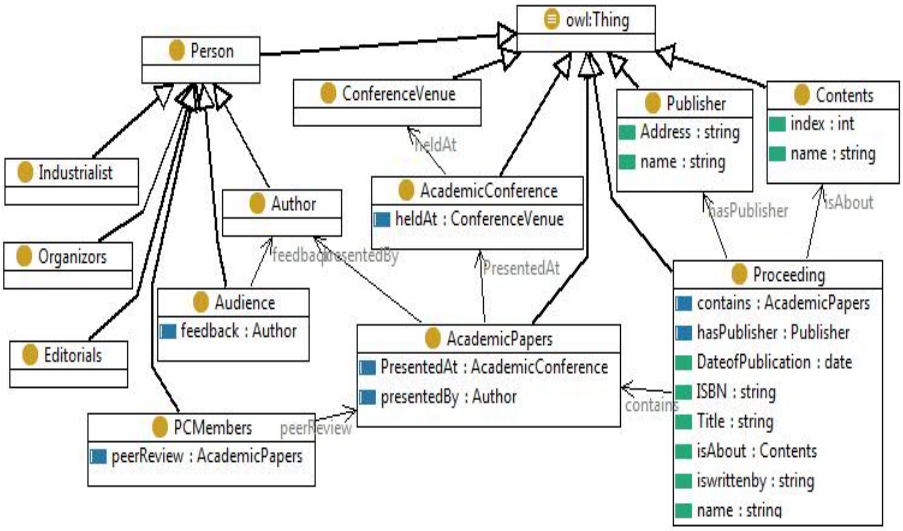

Figure 9: An arbitrary Ontology O_a for the classification

Comparative Analysis. We have tested *OntClassifire* on 34 ontologies that belong to 6 different categories (*University*, *CS_Department*, *Publication*, *Book*, *Conference_Proceeding*, and *Journal*). We have manually downloaded 28 ontologies from the Semantic Web. Some of the ontologies are from the same consortium or developer and constitute versions as 1.0, 1.1, etc., on the basis of extensions made by them. In addition, we selected the ontology from each of the category and provided them to another research group to make changes in the ontological constructs, perform some more extensions, and embed some more degree of overlapping between the categories. The aim behind this is to analyze the performance of *OntClassifire* on the ontologies that share common vocabularies. Therefore, each of the categories in the experiment has several ontologies which share the common vocabularies and

semantic similarities between ontological constructs. The result of classification experiment can be positive or negative depending on the classifier's accuracy. The produced experiment results for the each category may or may not match with the category's actual status, leading to four different cases:

True Positive (TP) or Correctly Classified: An arbitrary ontology O_a is correctly classified in the category *C* by the *Classifier* and Human Expert is agreed with it.

True Negatives (TN) or Correctly Unclassified: An arbitrary ontology O_a is not classified in the category *C* by the *Classifier* and Human Expert is agreed with it.

False Positives (FP) or Incorrectly Classified: An arbitrary ontology O_a is incorrectly classified in the category *C* by the *Classifier* and Human Expert is not agreed with it.

False negative (FN) or Missed Classification: An arbitrary ontology O_a is not classified in the category *C* by the *Classifier* and according to Human Expert it should be classified.

On the basis of these four cases, Precision and Recall values are calculated with the following equations and shown in the Table 2.

Precision (Pre) = True Positive/ (True Positive + False Positive)

Recall (Rec) = True Positive / (True Positive + False Negative)

The experimental results show that the ontological approach for the classification of web ontologies is the best approach based on the comparative analysis between the Naive Bayer's text classification and the ontological approaches, i.e., Simple (Sm) ontology approach and Weighted (Wg) ontology approach of *OntClassifire*. When the precision is equal to 1, it means that there are no False Positives, i.e., the ontologies that should be classified in the actual categories are truly classified into them. But, when the precision is low, it means that the classifier has made some ontology classifications which it should not have made. Similarly, when the recall is equal to 1, it means that there are no False Negatives, i.e., the ontology classification made by the classifier is actually achieved. But, when the recall is low, it means that the classifier has missed some ontology classifications that it should make according to the human expert.

The results with the simple ontology approach are better than the text classification algorithm, and promising when some weights are attached to the domain ontologies which significantly dominate the vocabulary of each category. Moreover, the overlapping ontologies that share the common vocabularies were correctly classified by the weighted ontological approach. Most of the erroneous classifications by the text algorithm are observed with the *Publication* category, which is the basic category and further subdivided into other three subcategories, i.e., *Book*, *Conference_Proceeding* and *Journal*. When the text classification algorithm found commonalities between the keyword terminology of an arbitrary ontology O_a and the category *Publication* ontology, it assigns an arbitrary ontology O_a the most general label. In addition, due to naive bayer's independence assumption, i.e., presence of a feature is unrelated to the presence of other features. But, it is avoided

by the ontological approaches by making the use of combined structural analysis with the help of axiomatic definitions and association between concepts.

Ontologies	**Approaches**	**TP**	**TN**	**FP**	**FN**	**Pre**	**Rec**
Uni	Wg OntClassifire	6	24	0	0	1	1
	Sm OntClassifire	5	22	2	1	0.714	0.833
	Naive bayer clfr	4	21	3	2	0.571	0.666
Cs Dept	Wg OntClassifire	4	26	0	0	1	1
	Sm OntClassifire	4	25	1	0	0.8	1
	Naive bayer clfr	2	22	4	2	0.333	0.5
Publication	Wg OntClassifire	6	23	1	0	0.857	1
	Sm OntClassifire	5	23	1	1	0.833	0.833
	Naive bayer clfr	2	18	6	4	0.25	0.333
Book	Wg OntClassifire	7	22	0	1	1	0.875
	Sm OntClassifire	7	21	1	1	0.875	0.875
	Naive bayer clfr	5	18	4	3	0.555	0.625
Conf Proc	Wg OntClassifire	5	24	0	1	1	0.833
	Sm OntClassifire	4	23	1	2	0.8	0.666
	Naive bayer clfr	4	22	2	2	0.666	0.666
Journal	Wg OntClassifire	4	26	0	0	1	1
	Sm OntClassifire	3	25	1	1	0.75	0.75
	Naive bayer clfr	2	23	3	2	0.4	0.5

Table 2: Comparison between the Ontological Approaches with the Text Classification Approach

4.4 Applications and Usage of the Ontology Classification

In this section, we elaborate a number of tasks which can benefit from the ontology classification with *OntClassifire*. The classification of ontologies is essential for the ontology, concept or information management and retrieval tasks on the Semantic Web. It improves the quality of web search for the specific ontologies and concepts. In addition, the classification of web ontologies is a crucial task to promote more focused crawling for the ontology retrieval and concept specific modular ontology analysis. It also helps knowledge engineers and agents in selecting the right ontology over the web directories, and expedites the ontology reusability. The Ontology classification based on an ontology matching approach exploits the matching of context specific knowledge that would result classification of an arbitrary ontology in a appropriate category, with the probability distribution across all the categories. Such soft classification mechanism exploited by *OntClassifire* could be more useful to end-users, as search results are presented in a ranked list for their assistance. The use of

ontology matching for the ontology classification provides higher accuracy of the classification process especially in the case of overlapping ontologies where the text classification algorithms did not work well within the current semantic web portals. The overlapping categories require analysis on the concepts, properties, structure and semantics hidden with their combinations.

This work can also benefit the construction, maintenance or expansion of the ontology directories on the Semantic Web. Currently, the ontology directories are maintained by the human editors such as those provided by Yahoo! [Yahoo, 07], DAML library [Daml, 06], and the dmoz Open Directory Project (ODP) [Dmoz, 07] that facilitate browsing of ontologies within the predefined set of categories. The ontology classifier does this job automatically replacing the tedious manual effort to help update and expand such directories on the Semantic Web.

5 Conclusion and Future Direction

Classification practice has long been adopted in the digital libraries and information systems to support user in clarifying his information need and structure search results for browsing. For the last decade, it has received great attention in the context of helping users to cope with the vast amount of information on the Web. With the passage of time the Semantic Web has gained much momentum and hence there is a significant growth seen in the ontology development and reuse. This increases the demands for searching the relevant domain ontologies over the web. The Ontologies, especially those developed in the Web Ontology Language (OWL), are significantly complex data structures than mere traditional web pages, as OWL builds up several levels of complexity on top of the XML for the conventional web data. Moreover, by defining terms on the similar concepts, these ontologies often overlap with each other. Therefore, it is of immense importance to classify web ontologies into the respective domain hierarchies for their efficient management and search by the people and autonomous individuals. But, the complex structures of ontologies present additional challenges as compared to the traditional text classification and web page classification. In this research paper, we discuss the state-of-the-art ontology semantic web portals, and analyze that they are not effective for meeting the demands of ontology classification for the emerging Semantic Web. We present *OntClassifire* that makes use of the context specific similarity measures to fit the ontologies into a predefined directory of general categories. We replace the plain text classification algorithm in the process of ontology classification with an ontology specific classification algorithm. Instead of using keyword search with bag-of-words, we use basic domain ontology for each predefined category and benefit from the ontology matching research to find the correspondences between the domain ontology and an arbitrary ontology for the classification purpose. Finally, by computing the weights of the correspondences found, *OntClassifire* calculates the probability whether both ontologies relate, and hence classify an ontology in to one of the predefined categories. We tested our ontology based framework on 34 ontologies with a certain degree of overlapping domain, and compared it with the text classifier to verify the effectiveness of ontological mechanism for the classification of web ontologies. We conclude that the use of ontologies is successful in the classification of objects, text documents, web data and ontologies themselves. We believe that the proposed model

forms a suitable basis for the ontology classification for the upcoming Semantic Web. One of our ongoing researches is to train the *OntClassifire* on an ontology repository with rich dataset such as the dmoz and present the empirical results. At the same time, we are building the retrieval mechanisms of the proposed framework and present the integrated work as a semantic web portal.

References

[Aquin, 07] Aquin, M.D., Sabou, M., Dzbor, M., Baldassarre, C., Gridinoc, L., Angeletou, S., Motta, E.: WATSON: A Gateway for Next Generation Semantic Web Applications, In Proc. 6th International Semantic Web Conference, Korea 2007, 23-24

[Berners-Lee, 01] Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities, Scientific American, May 2001

[Berners-Lee, 06] Berners-Lee, T., Hall,W., Hendler, J., Shadbolt, N.,Weitzner, D.J.: Creating a science of the web, Web Science 313, 2006, 769–771

[Bosch, 07] Bosch, A., Zisserman, A., Munoz, X.: Image classification using random forests and ferns, In Proc. 11th International Conference on Computer Vision, IEEE Press, Brazil 2007, 1-8

[Buitelaar, 04] Buitelaar, P., Eigner, T., Declerck, T.: OntoSelect: A Dynamic Ontology Library with Support for Ontology Selection, In Proc. 3rd International Semantic Web Conference Demo Session, Japan 2004

[Chakrabarti, 02] Chakrabarti, S., Joshi, M.M., Punera, K., Pennock, D.M.: The structure of broad topics on the web, In Proc. 11th International Conference on World Wide Web, ACM Press, New York 2002, 251–262

[Colucci, 03] Colucci, S., Di-Noia, T., Di- Sciascio, E., Donini, F.M., Mongiello, M., Mottola, M., A formal approach to Ontology-Based Semantic Match of Skills Descriptions, Journal of Universal Computer Science, 9, 12 (2003), 1437-1454

[Daml, 06] The DAML ontology library, 2006, http://www.daml.org/ontologies/

[Dean, 06] Dean, M., Barber, K.: Daml crawler, http://www.daml.org/crawler/ (August 2006)

[Ding, 05] Ding, L., Pan, R., Finin, T., Joshi, A., Peng, Y., Kolari, P.: Finding and Ranking Knowledge on the Semantic Web, In Proc. 4th International Semantic Web Conference, Springer LNCS 3729, Ireland 2005, 156–170

[Dmoz, 07] Corporation, N.C., 2007, The Dmoz open Directory Project (ODP), http://www.dmoz.com/

[Eberhart, 02] Eberhart, A.: Survey of RDF data on the web, Technical report, International University in Germany (2002)

[Ester, 02] Ester, M., Kriegel, H.P., Schubert, M.: Web site mining: A new way to spot competitors, customers and suppliers in the World Wide Web, In Proc. 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, Canada 2002, 249–258

[Ehrig, 05] Ehrig, M., and Maedche, A.: Ontology-Focused Crawling of Web Documents, In Proc. ACM symposium on Applied computing, Melbourne, Florida 2005, 1174–1178

[Fahad, 07] Fahad, M., Qadir, M.A., Noshairwan, M.W., Iftakhir, N.: DKP-OM: A Semantic Based Ontology Merger, In Proc. 3rd International Conference I-Semantics, J.UCS, Graz Austria 2007, 313-322

[Fahad, 10] Fahad, M., Moalla, N., Bouras, A., Qadir, M.A., Farukh, M.: A semantic approach for classification of web ontologies, In Proc. 10th International Conference on knowledge management and knowledge technologies, J.UCS, Graz Austria 2010, 69-80

[Gabrilovich, 04] Gabrilovich, E., Markovitch, S.: Text categorization with many redundant features: Using aggressive feature selection to make SVMs competitive with C4.5, In Proc. 21st International Conference on Machine learning, ACM Press, New York 2004, pp. 41

[Ghani, 02] Ghani, R.: Combining labeled and unlabeled data for multiclass text categorization, In Proc. 19th International Conference on Machine Learning, USA 2002, 187–194

[Glover, 02] Glover, E.J., Tsioutsiouliklis, K., Lawrence, S., Pennock, D.M., Flake, G.W.: Using web structure for classifying and describing web pages, In Proc. 11th Intl. Conference on World Wide Web, ACM Press, USA 2002, 562–569

[Golub, 05] Golub, K., Ardo, A.: Importance of HTML structural elements and metadata in automated subject classification, In Proc. 9th European Conference on Research and Advanced Technology for Digital Libraries, Springer LNCS 3652, Austria 2005, 368–378

[Grobelnik, 05] Grobelnik, M., Mladenić, D.: Simple classification into large topic ontology of Web documents, Journal of Computing and Information Technology, vol. 13(4), 2005, 279-285

[Harth, 06] Harth, A., Umbrich, J., Decker, S.: MultiCrawler: A Pipelined Architecture for Crawling and Indexing Semantic Web Data, In Proc. International Semantic Web Conference, Springer LNCS vol. 4273, USA 2006, 368–378

[Levenshtein, 1966] Levenshtein, I.V.: Binary Codes capable of correcting deletions, insertions, and reversals, Cybernetics and Control Theory, 10, 8 (1966), 707–710

[Miller, 1995] Miller, G.: Wordnet: A lexical database for English, Communication of the ACM, 38, 11 (1995), 39–41

[Mitchell, 97] Mitchell, T.M.: Machine Learning, New York, McGraw-Hill, 1997.

[Mishne, 06] Mishne, G., De-Rijke, M.: Capturing global mood levels using blog posts, Computational Approaches to Analyzing Weblogs, Papers from the 2006 Spring Symposium, AAAI Press, California 2006, 145–152

[Nie, 06] Nie, L., Davison, B.D., Qi, X.: Topical link analysis for web search, In Proc. 29th Annual International ACM SIGIR Conference on Research & Development on Information Retrieval, ACM Press, New York 2006, 91–98

[Ontolingua, 10] Ontolingua Website, http://www.ksl.stanford.edu/software/ontolingua/

[Palma, 06] Palma, R., Haase, P.: Oyster - sharing and re-using ontologies in a peer-to-peer community, In Proc. International Semantic Web Conference, Springer LNCS vol. 3729, Ireland 2005, 1059-1062

[Pan, 06] Pan, J.Z., Thomas, E., Sleman, D.: ONTOSEARCH2: Searching and Querying Web Ontologies, In Proc. of WWW/Internet, Spain 2006, 211–218

[Patel, 03] Patel, C., Supekar, K., Lee, Y., Park, E.K.: OntoKhoj: A semantic web portal for ontology searching, ranking and classification, In Proc. 5th ACM Intl. Workshop on Web Information and Data Management, USA 2003, 58–61

[Peng, 02] Peng, X., Choi, B.: Automatic web page classification in a dynamic and hierarchical way, In Proc. IEEE International Conference on Data Mining, Washington, DC 2002, 386–393

[Pierre, 01] Pierre, J. M.: On the automated classification of web sites, Computer and Information Science 6, (2001) http://www.ep.liu.se/ea/cis/2001/001/

[Qi, 06] Qi, X., Davison, B.D.: Knowing a web page by the company it keeps, In Proc. 15th ACM International Conference on Information and Knowledge Management, ACM Press, New York 2006, 228–237

[Qu, 06] Qu, H., Pietra, A.L., Poon, S.: Automated blog classification: Challenges and pitfalls, Computational Approaches to Analyzing Weblogs, AAAI Press, California 2006, 184-186

[Reich, 02] Reich, J.R., Brockhausen, P., Lau, T., Reimer, U.: Ontology-Based Skills Management: Goals, Opportunities and Challenges, Journal of Universal Computer Science, 8, 5 (2002), 506-515

[Seidenberg, 06] Seidenberg, J., Rector, A.: Web ontology segmentation: Analysis, classification and use, In Proc. 15th International Conference on the World Wide Web, ACM, New York 2006, 13–22

[Su, 05] Su, C., Gao, Y., Yang, J., Luo, B.: An Efficient Adaptive Focused Crawler Based on Ontology Learning, In Proc. Fifth International Conference on Hybrid Intelligent Systems, Brazil 2005, 73–78

[Supekar, 03] Supekar, K., Patel, C., Lee, Y.: Characterizing Quality of Knowledge on Semantic Web, In Proc. 17th International Florida AI Research Society Conference, USA, AAAI Press 2004

[Taghva, 03] Taghva, K., Borsack, J., Coombs, J., Condit, A., Lumos, S., Nartker, T.: Ontology-based Classification of Email, In Proc. International Conference on Information Technology: Computers and Communications, Nevada 2003, 194-200

[Wu, 03] Wu, S.H., Tsai, T.H., Hsu, W.L.: Text Categorization Using Automatically Acquired Domain Ontology, In Proc. 6th international workshop on Information retrieval with Asian languages, Japan 2003, 138-145

[Yahoo, 07] Yahoo!, Inc., 2007, http://www.yahoo.com/

[Zhang, 07] Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: A comprehensive study, International Journal of Computer Vision, 73, 2 (2007), 213 - 238