ISSN 2395-8618

Unsupervised Domain Ontology Learning from Text

V. Sree Harissh, M. Vignesh, U. Kodaikkaavirinaadan, and T. V. Geetha

***Abstract*—Construction of Ontology is indispensable with rapid increase in textual information. Much research in learning Ontology are supervised and require manually annotated resources. Also, quality of Ontology is dependent on quality of corpus which may not be readily available. To tackle these problems, we present an iterative focused web crawler for building corpus and an unsupervised framework for construction of Domain Ontology. The proposed framework consists of five phases, Corpus Collection using Iterative Focused crawling with novel weighting measure, Term Extraction using HITS algorithm, Taxonomic Relation Extraction using Hearst and Morpho-Syntactic Patterns, Non Taxonomic relation extraction using association rule mining and Domain Ontology Building. Evaluation results show that proposed crawler outweighs traditional crawling techniques, domain terms showed higher precision when compared to statistical techniques and learnt ontology has rich knowledge representation.**

***Index Terms*—Iterative focused crawling, domain ontology, domain terms extraction, taxonomy, non taxonomy.**

I. Introduction

ONTOLOGY in computer science can be viewed as formal representation of knowledge pertaining to particular domain [1]. In simpler terms ontology provides concepts and relationship among concepts in a domain. Machines perceive contents of documents(blogs, articles, web pages, forums, scientific research papers, e-books, etc.) as sequence of character. Much of the semantic information are already encoded in some form or other in these documents. There is an increasing demand to convert these unstructured information into structured information. Ontology plays a key role in representing the knowledge hidden in these texts and make it human and computer understandable.

Construction of Domain Ontology provides various semantic solutions including: (1) Knowledge Management, (2) Knowledge Sharing, (3) Knowledge Organization, and (4) Knowledge Enrichment.

It can be effectively used in semantic computing applications ranging from Expert Systems [2], Search Engines [3], Question and Answering System [4], etc. to solve day to day problems. For example, if the search engine is aware that "prokaryote" is a type of organism, better search results can be obtained and recall of the system will be improved subsequently.

Ontology is generally built under the supervision of domain experts and are time intensive process. Corpus required for building Ontology are not always readily available. Therefore, it is important to build corpus from web through crawling. Very few work is available that have incorporated crawling as a phase for collecting corpus in building Ontology. Since general crawling does not always provide domain related pages, lot of irrelevant pages are downloaded and filtering is required. Terms extracted using statistical measure or linguistic patterns are prone to noise and require additional level of filtering using machine learning techniques. Also, most systems rely on manually annotated resources for obtaining terms and also for relation discovery. These resources however mostly contain domain generic concepts and lack domain specific concepts and relations [1]. Ontology extracted using lexico-syntactic patterns are limited to certain patterns and require enrichment.

In this work we propose a framework for crawling websites relevant to the domain of interest and also build Domain Ontology without use of any annotated resource in an unsupervised manner. The crawling framework uses a novel weighting measure to rank the domain terms. The proposed framework consists of five phases Corpus Collection, Term Extraction, Taxonomic Relation Extraction, Non-taxonomic relation extraction and Domain Ontology building. Corpus is crawled using iterative focused web crawler which downloads the content which are pertinent to the domain by selectively rejecting URL's based on link, anchor text and link context. Terms are extracted by feeding graph based algorithm HITS with Shallow Semantic Relations and proposed use of adjective modifiers to obtain fine grained domain terms. Hearst pattern and Morpho-Syntactic patterns are extracted to build taxonomies. Non-taxonomic relation extraction is obtained through Association Rule Mining on Triples.

The organization of the paper is as follows: section two describes Related Work, section three describes the System Design, section four describes the Results and Evaluation, section five describes Conclusion and section six describes Future Work.

II. Related Work

In this section, we discuss the literature survey in Corpus Collection, Term Extraction, Taxonomic Relation Extraction and Non Taxonomic Relation Extraction.

Manuscript received on December 21, 2016, accepted for publication on June 18, 2017, published on June 30, 2018.

The authors are with Department of Computer Science and Engineering, College of Engineering, Guindy, India (e-mail: {vharissh14,vigneshmohanceg,naadan.uk}@gmail.com, tv_g@hotmail.com).

ISSN 2395-8618

A. Domain Corpus Collection

Domain Corpus is a coherent collection of domain text. It requires the usage of iterative focused or topical web crawler to fetch the pages that are pertinent to the domain of interest. In the work proposed by [5], a heuristic based approach is used to locate anchor text by using DOM tree instead of using the entire HTML Page. A statistical based term weighing measure based on TF-IDF called TFIPNDF (Term Frequency Inverse Positive Negative Document Frequency) was proposed for weighing anchor text and link context. The pages are classified as relevant or not relevant on the basis of trained classifier and is entirely supervised. The work however lacks iterative learning of terms to classify pages [6].

B. Domain Term Extraction

Domain Terms are the elementary components used to represent concepts of a domain. Example of domain terms pertaining to agricultural domain are "farming", "crops", "plants", "fertilizers", etc. Term Extraction is generally performed from collection of domain documents using any of the following methods: Statistical Measure, Linguistic Measure, Machine Learning and Graph-based Measure.

1) Statistical Measure: Most common Statistical Measure make use of TF (Term Frequency) and IDF (Inverse Document Frequency). Meijer et al. [7], proposed four measures namely Domain Pertinence, Lexical Cohesion, Domain Consensus and Structural Relevance to compute the importance of terms in a domain. Drymonas et al. [8], used C/NC values to calculate the relevance of multiword terms in corpus. These measures however fail to consider the context of terms and fails to capture the importance of infrequent domain terms.

2) Linguistic Measure: Linguistic Measures traditionally acquire terms by using syntactic patterns such as Noun-Noun, Adjective-Noun, etc. For example, the POS tagging of the sentence "Western Rajasthan and northern Gujarat are included in this region" tags "Western" as an adjective and "Rajasthan" as Noun. Lexico-Syntactic patterns makes use of predefined patterns such as "including", "like", "such as", etc., to extract terms. It is however tedious and time consuming to pre-define patterns.

3) Machine Learning: Machine Learning is either supervised or unsupervised. Supervised learning require the algorithm to be trained before usage and target variable is known. Some famous and commonly used supervised algorithms include Naive Bayes, Support Vector Machines and Decision Tree. In unsupervised learning training is not required and hidden patterns are found using unlabeled data. Uzun [9] work considers training features are independent and therefore used TF-IDF, distance of the word to the beginning of the paragraph, word position with respect to whole text and sentence and probability features from Naive Bayes Classifier to classify whether a term is relevant. The drawback of using machine learning is that training incurs overhead and data may not be available in abundance for training.

4) Graph Based Measure: Graph Based Measure is used to model the importance of a term and the relationship between the terms in an effective way. Survey on Graph Methods by Beliga et al. [10], suggest that graphs can be used to represent co-occurrence relations, semantic relations, syntactic relations and other relations (intersecting words from sentence, paragraph, etc.). Work by Ventura et al. [11] used novel graph based ranking method called "Terminology Ranking Based on Graph Information" to rank the terms and dice coefficient was used to measure the co-occurrence between two terms. Mukherjee et al. [12] used HITS index with hubs as Shallow Semantic Relations and authorities as nouns. Terms are filtered based on hubs and authority scores.

C. Taxonomic Relation Extraction

Taxonomy construction involves building a concept hierarchy in which broader-narrower relations are stored and can be visualized as a hierarchy of concepts. For example "rice", "wheat", "maize" come under "crop". They are commonly built using predefined patterns such as the work by Hearst [13] and Ochoa et al. [14]. Meijer et al. [7] proposed construction of taxonomy using subsumption method. This method calculates co-occurrence relations between different concepts. Knijff et al. [15], compared two methods subsumption method and hierarchical agglomerative clustering to construct taxonomy. They concluded that subsumption method is suitable for shallow taxonomies and hierarchical agglomerative clustering is suitable for building deep taxonomies.

D. Non Taxonomic Relation Extraction

Non Taxonomic Relations best describe the non-hierarchical attributes of concept. For example, in the non taxonomic relation "predators eat plants", eat is a feature of predator. Nabila et al. [16] proposed an automatic construction of non-taxonomic relation extraction by finding the non-taxonomic relations between the concepts in the same sentence and non-taxonomic relations between concepts in different sentences. Serra and Girardri [17] proposed a semi-automatic construction of non-taxonomic relations from text corpus. Association between two concepts are found by calculating the support and the confidence scores between the two concepts.

a) : To build a Domain Ontology from Text, the existing methods for Domain Term Extraction deprive from identification of low frequent terms, identification of all syntactic-patterns and require annotated re-sources for machine learning approaches. Graph based methods for identification can be used to solve the above problems as they can represent the meaning as well as composition of text. They also do not require manually annotated data unlike machine learning approaches. General Non-Taxonomic Relation Extraction methods are based on extraction of predicates between two concepts and as all predicates are not domain specific the use of Data Mining Techniques can be helpful in identifying the Domain Relations effectively.

ISSN 2395-8618

III. System Design

In this section we discuss the design of our system. Figure 1 shows the overall architecture diagram of the proposed framework. The system consists of five major phases: (1) Domain Corpus Collection, (2) Domain Term Extraction 3) Taxonomic Relation Extraction, (4) Non Taxonomic Relation Extraction, and (5) Domain Ontology Building.

A. Domain Corpus Collection

Corpus required for construction of Ontology may not be readily available for every domain. Since the quality of the corpus plays a vital role in deciding the quality of Ontology, Iterative Focused Crawling is performed to download web pages relevant to the domain. List of Seed URLs are given as input to the Iterative Focused Crawler. The web pages whose URL, anchor text or link context satisfy the relevance score are added to the URL queue. The depth of the pages to be crawled is specified. The output of the focused crawler is used as corpus for construction of Ontology. Crawling is terminated when the relevance of URL to the context vector decreases drastically. The architecture of crawler is depicted in Figure 2.

Nouns are considered as candidate terms for finding keywords in the domain. Therefore, the nouns are extracted from the corpus using the Stanford parts-of-speech tagger. The context vector of a noun is computed by using proposed weighted co-occurrence score. Weighted co-occurrence ($WCO(w_i, w_j)$) of two words w_i and w_j is given by :

$$WCO(w_i, w_j) = CO(w_i, w_j) X idf(w_i) X idf(w_j) \quad (1)$$

In Equation 1, $idf(w_i)$ and $idf(w_j)$ are the inverse document frequency of words w_i and w_j. $CO(w_i, w_j)$ is the co-occurrence frequency of the two words w_i and w_j. The proposed equation considers the inverse document frequencies of the terms in order to consider the importance of terms which occur rarely and may of importance to the domain. Unit Normalization of the context vector is performed to have a specific range of score between 0 and 1. The normalized context vector of each term is summed along the column and sorted in descending order. The top ranked terms are extracted as concepts based on percentage.

Relevance of the web pages are calculated by computing the average of the Cosine Similarity Score of the test domain vectors and each of the domain vectors of the training document. The relevance of the URL is checked without scanning the pages. It is done by computing relevance of HREF, Anchor Text and/or Link Context. Appropriate threshold are set for HREF, Anchor Text and Link Context. If HREF is not relevant (i.e Relevance Score), Anchor Text will be checked for relevance. If Anchor Text is not relevant, finally, Link Context will be checked.

B. Domain Term Extraction

Domain corpus, which contains a rich collection of text documents is pre-processed to identify the domain terms. Numbers, special characters, etc. which do not play a significant role in ontology construction are removed.

1) Shallow Semantic Relation Extraction: Domain text documents are tokenized into sentences. These sentences are parsed using Stanford Dependency Parser to identify the Shallow Semantic Relations between the words. Shallow Semantic Relations represent the syntactic contextual relations within the sentences. In addition to the Shallow Semantic Relations extracted in [12] we have also extracted and used adjective modifiers obtained through Dependency Parsing. Since, significant amount of domain terms are composed as adjective modifier, it is important to consider these dependencies. For example, in the sentence "Biological research into soil and soil organisms has proven beneficial to organic farming.", "organic farming" and "biological research" are tagged as adjective modifiers.

2) Domain Term Induction Using HITS: HITS algorithm [12], [18] is applied to identify the most important domain terms. It is composed of two major components—Hubs and Authorities. Hubs are represented by Shallow Semantic Relations and authorities are represented by nouns. Hub score is calculated as the sum of authority scores and authority score is calculated as the sum of hub scores. Hub and Authority score are calculated recursively until hub and authority score converges. The Shallow Semantic Relation which has high hub score are selected as multi-grams and nouns which has high authority score are selected as unigrams. These unigrams and multi-grams constitute the domain terms.

C. Taxonomic Relation Extraction

Taxonomic Relations represent hypernym-hyponym relation. A hypernym represents the specific semantic field of a hyponym and a hyponym represents the generic semantic field of the hyponym. The three steps involved in building a taxonomy involves (i) Hearst Pattern Extraction and (ii) Morpho-syntactic Pattern Extraction

1) Hearst Pattern Extraction: Hearst Patterns [13] are commonly used to extract taxonomic relations from text. Sentences containing the domain terms are selected for identification of Hearst Patterns. Sentences are tagged using parts-of-speech tagger to find six types of hearst patterns. Six types of hearst patterns are as listed below: NP_i is considered as a hypernym and NP_j is considered as a hyponym.

1) NP_i such as NP_j

Example : agrochemicals such as pesticides and fertilizers where agrochemicals is a hypernym and fertilizers and pesticides is a hyponym.

2) NP_j or other NP_i

Example : iron, magnesium, zinc, or other nutrients where nutrients is a hypernym and iron, magnesium and zinc are hyponyms.

ISSN 2395-8618

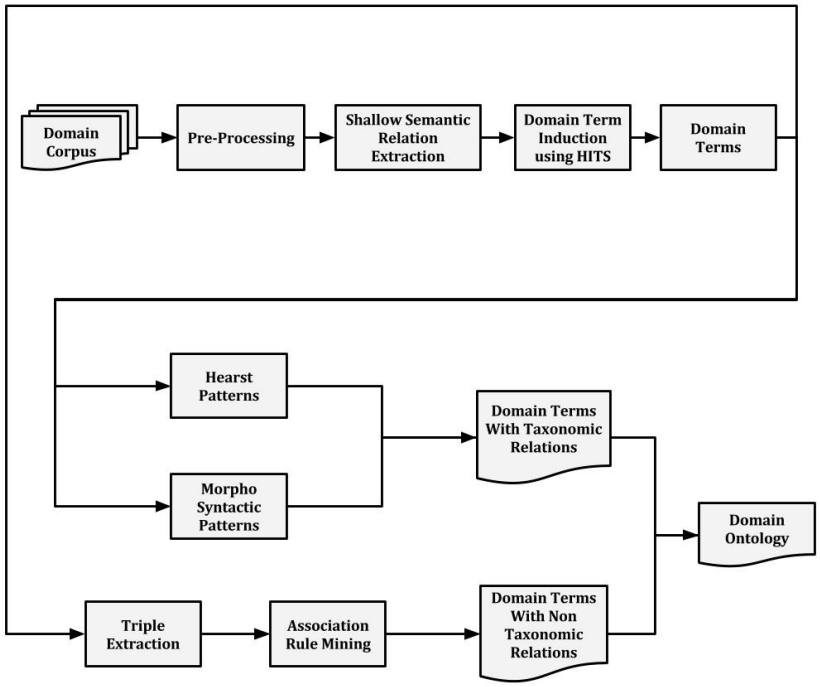

Fig. 1. Architecture of Proposed Framework: Unsupervised Domain Ontology Construction From Text

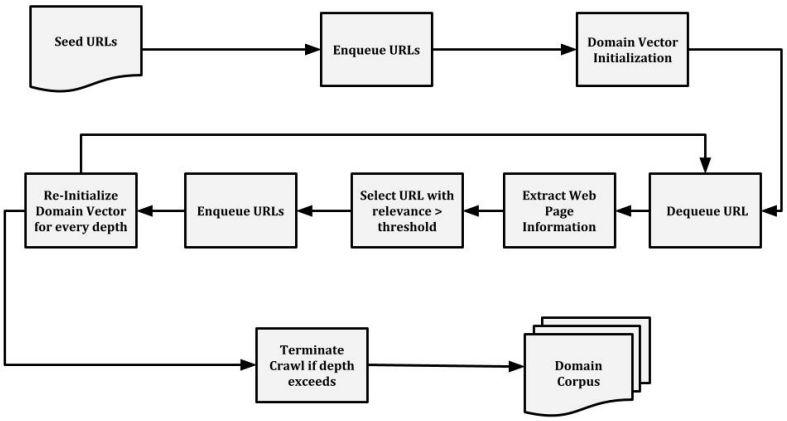

Fig. 2. Flow Diagram of Iterative Focused Crawler

3) NP_j and other NP_i
Example : barley, wheat, and other cereals where cereals is a hypernym and barley and wheat is a hyponym.
4) such NP_j as NP_j
Example : such foods as bread, porridge, crackers, biscuits where foods is a hypernym and bread, porridge, crackers and biscuits is a hyponym.
5) NP_i including NP_j
Example : political issues including water pollution where political issues is a hypernym and water pollution is a hypernym.
6) NP_i especially NP_j
Example : Tropical fruits especially bananas grows in South India where Tropical fruits is a hypernym and bananas is a hyponym.

2) Morpho Syntactic Pattern Extraction: In our work we have also extracted Morpho Syntactic Patterns [14] to extract additional Hypernym-Hyponym relations. There are two rules followed to extract morpho-syntactic patterns.

Rule 1 : If the term t_1 contains a suffix string t_0, then the term t_0 is the hypernym of the term t_1, provided the term t_0 or t_1 is a domain term. For example, “polysaccharide” is considered as the hypernym of the term “homopolysaccharide”.

Rule 2 : If the term t_0 is the head term of the term t_1 , then t_0 is considered as the hypernym of the term t_1, provided term t_0 or t_1 is the domain term. Example: “sweet corn” is the

ISSN 2395-8618

hyponym of the word “corn”.

D. Non-Taxonomic Relation Extraction

Non-Taxonomic Relations represent the properties of the object. It has no class-subclass relationship.

1) Triplet Extraction: A sentence is composed of three components - subject, predicate and object. A triplet in a sentence is defined as the relation between the subject and the object, with the relation being the predicate. Parsed documents using Stanford Parser are input to the triplet extraction process. Subject, predicate and object from the sentences is extracted using Russu’s Triple Algorithm [19].

2) Association Rule Mining: Association Rule Mining [20] is performed to find the non-taxonomic relations between the domain terms. Apriori Algorithm is used for frequent itemset generation and association rule mining. Frequent itemset whose support crosses a suitable threshold are selected for mining association rules. Association rules are filtered from frequent itemsets and association rules which satisfy a suitable confidence score are selected.

E. Domain Ontology Building

The concepts with the taxonomic and non-taxonomic relations are represented in a Resource Description Framework format. The concepts consists of a concept id, a broader relation, a narrower relation and a non-taxonomic relation associated with it. The broader/narrower relation are represented by class/subclass relations. Non-taxonomic relations consists of a property, domain and range. The domain of a property represent the subject whose predicate is that property. The range of a property represent the object whose predicate is that property. Example : “rice” is a concept with concept id “12143”, narrower relations “long-grain rice”,broader relation “crops”, “medium-grain rice”, “short-grain rice”, property “grows in”, domain “rice”, range “South India”.

IV. RESULTS AND EVALUATION

A. Domain Corpus Collection

Domain Corpus Collection consists of implementing an iterative focused web crawler that crawls pages relevant to the domain. 22 seed URLS pertaining to agriculture domain were given as input to the focused crawler. 20,632 documents were obtained at the end of crawling a depth of 3. Table I shows the number of relevant links crawled by the crawler.

TABLE I
NUMBER OF LINKS CRAWLED AT DIFFERENT DEPTHS

Depth	Number of Links Crawled
0	22
1	134
2	816
3	19732
Total	20632

Table II shows the Number of Links crawled through HREF, Anchor Text and Link Context. It is observed that most of the links were found to be relevant through HREF and Link Context. HREF usually contain the text present in the Anchor Text. So, if the relevance fails through HREF there is a high probability of checking the Link Context.

TABLE II
NUMBER OF LINKS CRAWLED THROUGH HREF, ANCHOR TEXT AND LINK CONTEXT

Mode	Count
HREF	606
Anchor Text	2256
Link Context	17842
Total	20632

Table III shows the Number of documents in different similarity range compared to SeedURL pages. It can be seen that most of the pages similarity were in the range of 0.5 to 0.6.

TABLE III
NUMBER OF LINKS CRAWLED THROUGH HREF, ANCHOR TEXT AND LINK CONTEXT

Similarity	Count
0.6 - 0.7	787
0.5 - 0.6	11624
0.4 - 0.5	5782
0.3 - 0.4	2156
0.2 - 0.3	251
0.1 - 0.2	60
0.0 - 0.1	22
Total	20632

Figure 3 shows Histogram analysis of document count to similarity of documents at various depths and Median of similarity score for a particular depth w.r.t seed documents. Histogram analysis strongly suggest that most of the documents crawled belongs to the similarity range of 0.5 to 0.6. It was also observed that the median of relevance score follows a decreasing trend and the number of irrelevant links crawled increased after a depth of 3.

In our work, Convergence Score [21] was used to evaluate the Iterative Focused Crawler. It is defined as the number of concepts present in the final crawl to the number of concepts present in initial seed page set. From Figure 4, we infer that the convergence of Focused Crawler is better than Base Line Crawler since the former crawls the page that are semantically relevant.

B. Domain Term Extraction

In our work, HITS algorithm was used to extract the top quality domain terms. The algorithm took nearly 3000 iterations to rank top quality domain terms.

The precision scores of Graph Based Domain Term Extraction using HITS algorithm used in our work is evaluated against statistical measures such as Linguistic Patterns, Inverse Document Frequency , C-value(LIDF score) and Graph Based

ISSN 2395-8618

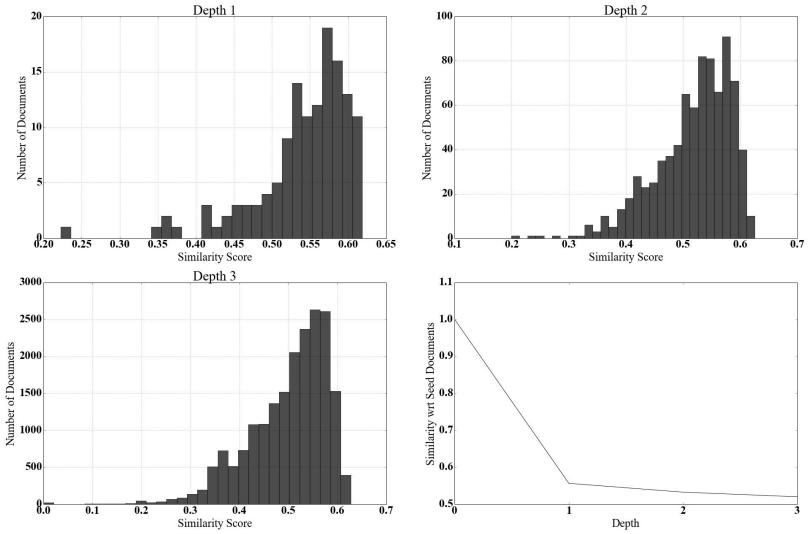

Fig. 3. Histograms analysis of Similarity Scores w.r.t Depth and Median of Similarity Score for a particular Depth w.r.t seed documents

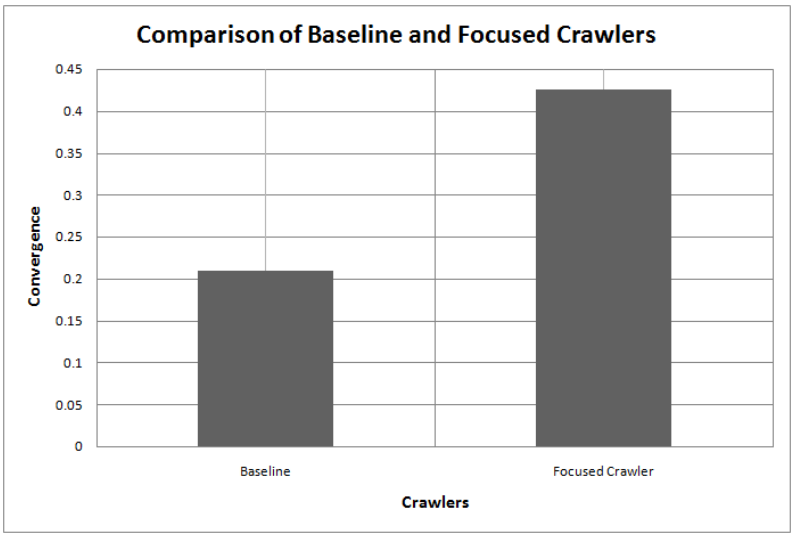

Fig. 4. Comparison of Baseline crawler to Focused Crawler using Convergence Score

TABLE IV
PRECISION SCORES OF TERM EXTRACTION USING HITS, LIDF, TERGRAPH AND DP+DC+LC+SR

Total Terms	Term Extraction using HITS	LIDF	TeRGraph	DP+DC+LC+SR
1000	0.772	0.697	0.769	0.751
2000	0.749	0.662	0.694	0.687
3000	0.733	0.627	0.644	0.657
4000	0.703	0.608	0.593	0.612
5000	0.676	0.575	0.562	0.583
6000	0.662	0.550	0.561	0.561
7000	0.651	0.547	0.552	0.550
8000	0.633	0.546	0.546	0.538

Algorithm Terminology Ranking Based on Graph Information - TeRGraph proposed by [11] and sum of statistical scores obtained from Domain Pertinence (DP), Domain Consensus (DC), Lexical Cohesion (LC) and Structural Relevance (SR) proposed in [7] is shown in Table IV. GENIA corpus used in [11] was used for evaluation purpose. The measures shows that graph based HITS algorithm shows better precision compared to statistical measures and Graph Based algorithm TeRGraph.

ISSN 2395-8618

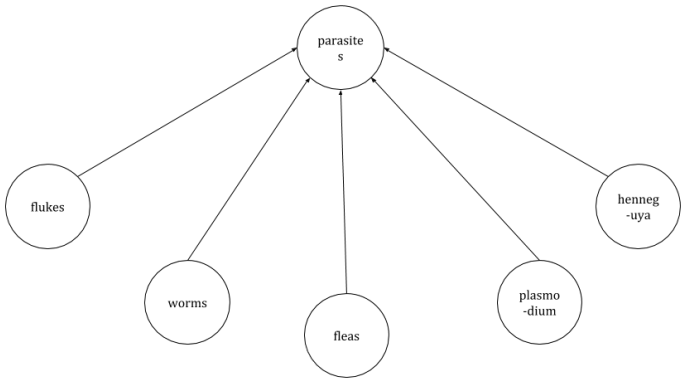

Fig. 5. Taxonomy of Parasites

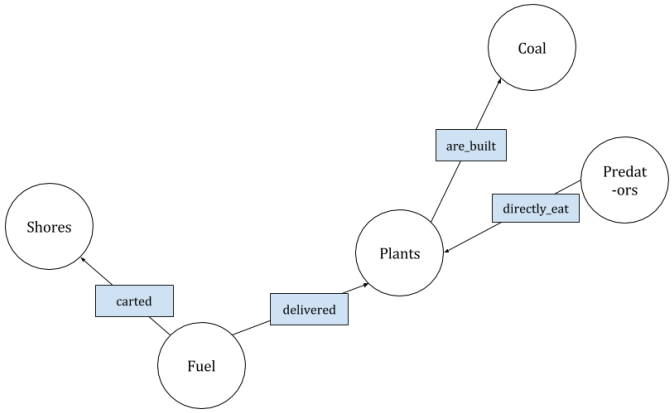

Fig. 6. Non Taxonomy of Plants

C. Domain Ontology

Hearst Patterns and Morpho-Syntactic patterns were used to induce Taxonomy. Total of 6539 Hearst Patterns and 2149 Morpho-Syntactic patterns were extracted to construct the Taxonomy. 5216 triples were extracted and 357 Non Taxonomic Relations were identified using Association Rule Mining. Figure 5 shows the snippet of Parasite Taxonomy and Figure 6 shows the snippet of Non Taxonomic Relations associated with Plants.

In our work, Domain Ontology was evaluated using Metic Based Evaluation techiques Inheritance Richness and Class Richness [22].

1) Class Richness: This metric is related to how instances are distributed across classes. The number of classes that have instances in the KB is compared with the total number of classes, giving a general idea of how well the KB utilizes the knowledge modeled by the schema classes. Thus, if the KB has a very low Class Richness, then the KB does not have data that exemplifies all the class knowledge that exists in the schema. On the other hand, a KB that has a very high CR would indicate that the data in the KB represents most of the knowledge in the schema. Table V shows the Class Richness score for Taxonomy and Non Taxonomy learning methods.

2) Inheritance Richness: Inheritance Richness measure describes the distribution of information across different levels of the ontology's inheritance tree or the fan-out of parent classes. This is a good indication of how well knowledge is grouped into different categories and subcategories in the ontology. This measure can distinguish a horizontal ontology (where classes have a large number of direct subclasses) from a vertical ontology (where classes have a small number of direct subclasses). An ontology with a low inheritance richness would be of a deep (or vertical) ontology, which indicates that the ontology covers a specific domain in a detailed manner, while an ontology with a high IR would be a shallow (or horizontal) ontology, which indicates that the ontology represents a wide range of general knowledge with a low level of detail. Table V shows the Class Richness score for Taxonomy and Non Taxonomy learning methods.

From the results of the evaluation metrics(class richness and inheritance richness), it is evident that the constructed ontology has a good density depicting that the concepts extracted represents a wider knowledge in the domain.

ISSN 2395-8618

TABLE V
INHERITANCE AND CLASS RICHNESS SCORES

Method	Inheritance Richness	Class Richness
Hearst	4.004	0.367
Morpho-Syntactic	2.671	0.068
Hearst + Morpho-Syntactic	3.967	0.41
Non-Taxonomic Relation	1.81	0.21

V. CONCLUSION AND FUTURE WORK

In our work, we have developed an iterative focused crawler for collection of domain corpora, with each element in the co-occurrence matrix weighted as product of co-occurrence frequency and IDF of row and column. The generic terms extracted as concepts are removed using statistical measure. The relevance of the page is checked in the following levels: URL, Anchor Text and Link Context. Domain terms were extracted without any manual annotated resource unsupervised using HITS algorithm with Hubs as Shallow Semantic Relation and Authority as Nouns. The ranked terms were removed of noise using Domain Pertinence. In this work, taxonomy was induced using Hearst Patterns and Morpho-Syntactic Patterns. The Ontology was built automatically without supervision from scratch. In the future, we intend to exploit deep learning methods for building Domain Ontology to make it meaningful and useful.

REFERENCES

[1] Y. Sure, S. Staab, and R. Studer, "Ontology engineering methodology," in *Handbook on ontologies*. Springer, 2009, pp. 135–152.

[2] L.-Y. Shue, C.-W. Chen, and W. Shiue, "The development of an ontology-based expert system for corporate financial rating," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2130–2142, 2009.

[3] Y. Zhang, W. Vasconcelos, and D. Sleeman, "Ontosearch: An ontology search engine," in *Research and Development in Intelligent Systems XXI*. Springer, 2005, pp. 58–69.

[4] V. Lopez, M. Pasin, and E. Motta, "Aqualog: An ontology-portable question answering system for the semantic web," in *European Semantic Web Conference*. Springer, 2005, pp. 546–562.

[12] S. Mukherjee, J. Ajmera, and S. Joshi, "Domain cartridge: Unsupervised framework for shallow domain ontology construction from corpus," in *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM, 2014, pp. 929–938.

[5] L. Liu, T. Peng, and W. Zuo, "Topical web crawling for domain-specific resource discovery enhanced by selectively using link-context," *Proc. The International Arab Journal of Information Technology*, vol. 12, no. 2, 2015.

[6] R. Sheikh, "A review of focused web crawling strategies."

[7] K. Meijer, F. Frasincar, and F. Hogenboom, "A semantic approach for extracting domain taxonomies from text," *Decision Support Systems*, vol. 62, pp. 78–93, 2014.

[8] E. Drymonas, K. Zervanou, and E. G. Petrakis, "Unsupervised ontology acquisition from plain texts: The OntoGain system," in *International Conference on Application of Natural Language to Information Systems*. Springer, 2010, pp. 277–287.

[9] Y. Uzun, "Keyword extraction using naïve bayes," in *Bilkent University, Department of Computer Science, Turkey www. cs. bilkent. edu. tr/˜ guvenir/courses/CS550/Workshop/Yasin_Uzun. pdf*, 2005.

[10] S. Beliga, A. Meštrović, and S. Martinčić-Ipšić, "An overview of graph-based keyword extraction methods and approaches," *Journal of Information and Organizational Sciences*, vol. 39, no. 1, pp. 1–20, 2015.

[11] J. A. Lossio-Ventura, C. Jonquet, M. Roche, and M. Teisseire, "Yet another ranking function for automatic multiword term extraction," in *International Conference on Natural Language Processing*. Springer, 2014, pp. 52–64.

[13] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *Proceedings of the 14th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics, 1992, pp. 539–545.

[14] J. L. Ochoa, Á. Almela, M. L. Hernández-Alcaraz, and R. Valencia-García, "Learning morphosyntactic patterns for multiword term extraction," *Scientific Research and Essays*, vol. 6, no. 26, pp. 5563–5578, 2011.

[15] J. De Knijff, F. Frasincar, and F. Hogenboom, "Domain taxonomy learning from text: The subsumption method versus hierarchical clustering," *Data & Knowledge Engineering*, vol. 83, pp. 54–69, 2013.

[16] N. Nabila, A. Mamat, M. Azmi-Murad, and N. Mustapha, "Enriching non-taxonomic relations extracted from domain texts," in *2011 International Conference on Semantic Technology and Information Retrieval*. IEEE, 2011, pp. 99–105.

[17] I. Serra and R. Girardi, "A process for extracting non-taxonomic relationships of ontologies from text," 2011.

[18] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[19] D. Rusu, L. Dali, B. Fortuna, M. Grobelnik, and D. Mladenic, "Triplet extraction from sentences," in *Proceedings of the 10th International Multiconference" Information Society-IS*, 2007, pp. 8–12.

[20] R. Srikant and R. Agrawal, *Mining generalized association rules*. IBM Research Division, 1995.

[21] S. Thenmalar and T. Geetha, "The modified concept based focused crawling using ontology," *Journal of Web Engineering*, vol. 13, no. 5-6, pp. 525–538, 2014.

[22] S. Tartir, I. B. Arpinar, M. Moore, A. P. Sheth, and B. Aleman-Meza, "OntoQA: Metric-based ontology quality analysis," 2005.