Memory and Time aware Automated Job Ontology Construction with reduced Ontology Size Based Semantic Similarity

Dr. N. Kumaresh[1*], Dr. S. Oswalt Manoj[2*], Dr. S. Thanga Ramya[3], Dr. V. Gladis Pushparathi[4], Dr. D. Praveena[5]

[1*]Dr. N.G.P Arts and Science College, Coimbatore, India
[2*]Sri Krishna College of Engineering and Technology, Coimbatore, India
[3]RMK Engineering College, Kavaraipettai, India
[4]Velammal Institute of Technology, India
[5]RMD Engineering College, Kavaraipettai, India

Abstract **-** Job ontology is a way of providing details about the tasks and requirements needed for a particular kind of job. Construction and updating of job ontology play a more important role in ensuring the accurate and reliable retrieval of information about the jobs. This is ensured in a research method by introducing Domain Ontology Construction Framework (DOCF) which would support the dynamic and user interactive based ontology construction. However, in the previous method, certain issues related to memory may be raised because of the dynamic updation of the ontology through run time retrieval of the details. By finding the similarity among the ontologies, this issue can be avoided. This is resolved by the introduction of a technique named Semantic Similarity based Job Ontologies Size Reduction (SSJOSR). Here, the semantic similarity among the knowledge ontology and the job ontology is identified and the interconnected information are directly connected together to avoid the memory consumption problems. Semantic distance is the metric that is used to measure the similarity among two entities. The proposed approach is implemented and evaluated from and we can evolve better results when compared to the predominant re- search scheme. This improves the accuracy and the results evidently portray the appropriate collection of the scaling stages and also the likeness measures. This reduces the size of ontologies in a noteworthy scale without misplacing the significant details.

Keywords **-** Semantic Distance, Semantic relatedness, Ontology, DOCF, SSJOSR.

I. INTRODUCTION

In this data driven age, the growth of internet technologies has paved way for in- creased information sharing through various media. The information stored in knowledge repositories is getting multiplied day to day and hence managing them in an efficient way has become a need of the hour. [1] Represents a study on the construction of automated domain ontology in varied domains and contexts. In most industries, hiring a prospective candidate with appropriate skillset catering to the needs of the Indus-try is a great challenge. It is because of the huge number of applications that are received for a single job vacancy. Most of these applications possess the same qualification whereas the real skillsets with respect to the job description may not be easily identified. Though there are a number of solutions today, hiring a right person to the right position is still a challenge. The work mentioned in [2] helps to rectify the issue by proposing a Domain Ontology Construction Framework (DOCF) which constructs an ontology by correlating the skillset and the job description. The documents received from the applicants are clustered based on similarity of content and conceptual similarity. Ontologies have gained popularity in the recent years as a significant tool for knowledge representation since it allows large quantity of data to be represented in a logical and hierarchical fashion [3]. Ontology therefore is a technique which helps to represent knowledge in a logical fashion. A lot of current research works focus on automated construction of ontologies, merging of ontologies and optimal techniques for the construction of ontologies [4]. In this work, the prime focus is over the placement of new knowledge into an existing ontology using an automated system. The biggest challenges encountered with the growth of formally represented knowledge are internal inconsistency and redundancy [6]. One of the most common examples stated with respect to contradictions that arise when a formally represented knowledge grows is "Count Dracula was a vampire" and "vampires does not exist". The statements are contradictory in nature until one statement is aware of the other. This is termed to be implicit context shift [7]. Some statements represented in the knowledge representation are factual, Real-world based whereas the others may be fictional or imaginary. The representation must be built to distinguish between the both [8]. In terms of reasoning context, various axioms are applied to the assertions made in the context of varying representations [9]. In general, most ontologies do not have any mechanism to classify a contextual information under a particular context based on certain assertion and hence is inconsistent in fetching interesting information [10].

This work focusses on the automated construction of ontology, also known as de- pendency graph, from a given job knowledge. This work also emphasizes on the identification of prerequisites and the development of follow-up modules for a given job knowledge requirement, which may also be called as query. In case of domain ontology, the identification of the relationships between different concepts of a domain plays a key role [11]. In

the proposed work, the lecture module represents the concept and labelling a pre-requisite or follow-up of a topic represents the relationship. A dependency graph will be provided as an outcome to the end-user which depicts a concept map which is a abstract representation of the field. The perception of a idea through graphical representation will be high compared to the perception of the same in its equivalent textual form [12]. There exist no systems that can automatically determine the dependencies between topics from a repository of job requirements knowledge. In this work, the Semantic Similarity based Job Ontologies Size Reduction (SSJOSR) is introduced which finds the semantic similarity between job ontology and knowledge ontology and the identified correlation is used to avoid memory consumption problems. The degree of similarity between two entities is usually represented using a measure called semantic distance, which is an inverse of semantic relatedness or semantic similarity.

This work is prearranged based on the following sections. The various literature connected to the automated construction of ontologies are discussed in Section 2. The proposed research methodology and the working mechanism with typical examples are discussed in Section 3. The results of the planned system and the corresponding inferences are deliberated in Section 4. The conclusion part is provided in Section 5.

II. Related Works

A lot of literature is available in the field of ontology with respect to its application in linguistics, software engineering, information retrieval, data mining, machine learning, etc. since these domains yield a vast support due to the application of ontology learning. However, a meagre number of works are available with respect to the construction methods within the ontology learning domain till date.

Maedche and Staab [13] expresses that the ontology learning constitutes four parts namely extraction, pruning, refinement, import or reuse. This work focusses on the ex- traction methods. The construction methods are basically categorized as dictionary-based, association rules, knowledge base and text-clustering. Domain Ontology rapid Development Environment (DODDLE II) [14] is a tool that assists in the identification of taxonomic and non-taxonomic relations. The taxonomic relations extraction is based on Word Net and domain expertise whereas non-taxonomic relations extraction is based on domain-specific texts supplemented with lexical co-occurrence statistical analysis. Maedche and Staab [15] proposed the Text-to-Onto framework which is a semi-automatic ontology learning tool which deploys a wide number of algorithms for various ontology learning subtasks. The framework highlights the application of Natural Language Processing and Data Mining methodologies in the development and maintenance of ontologies. The succeeding system Text2Onto [16] shows an improvement compared to the earlier system. The learned knowledge was represented in the meta-level as instantiated model primitives within the Probabilistic Ontology Model (POM). This can then be translated to communicative knowledge representation languages like OWL and RDFS.

Navigli and Velardi [17] proposed Onto Learn which uses a blend of symbolic and statistical methods. The domain terminology is extracted from the domain corpora. The complex domain terms are arranged in the hierarchical order and also has been understood semantically. The Word Net is clipped and supplemented with the identified domain concepts at the end of the process. The emphasis of the work was to address the disambiguation of word sense. Shamsfard and Barforoush [18] proposed HASTI, which is a system that studies the concepts, the conceptual relations both taxonomic and non-taxonomic, axioms and ontologies can be built on the existing kernel. HASTI is autonomous of domains and helps to build general as well as domain ontologies from the scratch.

Fortuna et.al. [19] Developed a tool Onto Gen to build ontologies by extracting the possible concepts and relations using machine learning and text-mining algorithms. Super- vised learning is used for the discovery of concepts. Dahab et.al [20] created a tool, Text On to ex which uses semantic pattern-based approach for constructing the ontology from usual domain text. The key relationship between the natural domain texts is analysed and mapped to meaningful representation to facilitate the ontology construction. The tool is efficient with respect to the discovery of instances of known relations but fails to discover new relations.

III. Memory and Time-Aware Automated Job Ontology Construction

One of the major challenges faced while working with an ontological representation is the consumption of memory. Though there are a few automated ontology construction techniques, most of them are less aware about the memory consumption problems. In order to overcome this issue, the Semantic Similarity based Job Ontologies Size Reduction (SSJOSR) is introduced in this work. The technique depends on the semantic similarity between job ontologies and interrelated details. This helps to avoid the memory consumption problem.

A. Automated Ontology Construction

This section elaborates the procedure involved in the automated construction of ontology. The first step involves the application of naïve Bayes classification algorithm towards categorization of text and labelling the documents. The Vector Space Model (VSM) algorithm is used to calculate the similarity and cluster the documents which are being imported. Finally, the clustered text is shortened and the essential terms involved in the construction of the domain ontology is achieved using Luhn's summarization algorithm.

The first step involved in the construction of ontology is the proper categorization of the documents. It is also an essential to allocate a tag type for the file since the appropriate association need to specify to link the document to the domain ontology. The classification of text in the document uses the naïve Bayes algorithm.

Bayes' theorem: The naïve Bayes classification is one of the oldest classification method based on Bayesian probability. The formula can be represented as:

$$(X \mid Y) = ((Y \mid X)(X))/(P(Y)) \quad (1)$$

The underlying principle of the naïve Bayesian classification is given like every label would have a description and it clearly implies that the set represents all the entities and it has a tag towards each entity. The likelihood of occurrence or incidence of an entity is given by equation (1).

An entity belonging to N possible tags is represented by a vector $x = (x1, \ldots,)$. The conditional probability based on Bayes theorem is represented below:

$$p(C\_k \mid x) = (p(C\_k)p(x|c\_k))/(p(x)) \quad (2)$$

Since, the entities represented are continuous in nature, equation (2) may not work. This may be due to the circumstance that the classifier can use real features to recognize the entity to which it belongs to. These attributes may be floating-point data. This results in a scenario where the entities follow a normal distribution, known as Gaussian distribution. Therefore the formula can be restated as follows:

$$(x = v \mid c) = 1/\sqrt{(2\pi\sigma\_c\text{^}2)}\ e\text{^}(-(v - \mu\_c)\text{^}2/(2\sigma\_c\text{^}2)) \quad (3)$$

The first run of the naïve Bayes classifier calculates the probability of difference in characteristics of the training data. Once the process is completed, the documents are imported and labeled based on the probability of the feature it contains, which is calculated during the first run. The data marking shall be done based on the data and the maximum probability of occurrence of a label.

Document classification based on Naive Bayes: The main objective of ontology construction is the classification of documents. In case of text classification, if we have a document$d \in X$, where X is the document vector space or the document space, then the labels are represented as classes, which is represented as a set $C = \{c_1, c_2, \cdots, c_j\}$. Document vector spaces are usually high-dimensional in nature. A hit label is assigned to the document collection. Consider $< d, c >$ as training samples such that

$< d, c > \in X \times c.$ Let$< d, c >= \{Delhi\ joins\ the\ World\ Trade\ Organization, India\}$.

This sentence in the given document is classified as India, so that the hit is made over the label 'India'. A supervised training algorithm need to be used such that a training of function Υ is used over the document that is mapped to a category$y: X \to C$. The naïve Bayesian classification is usually done in two modes which may either be multinomial or that of Bernoulli's.

Once the classification is completed, the Vector Space Model (VSM) is used to see the similar files which can be used in-order to construct an ontological structure.

Vector Space Model: It is one of the important tasks to transform the documents into mathematical models, so that they could be acted upon by appropriate algorithms to accomplish a task. One such technique is the deployment of Vector Space Model which transforms the text document into identifier vectors. The transformation may be multi- dimensional where each dimension represents a term. Terms that does not appear in the document will be considered as null vectors. In other words, the terms present in the document will be represented as non-zero vectors. The weight of a term is determined by TF-IDF weighting. The term's definition is based on the context or the application in which it is used. In general, a term may be a word or a long phrase. The count of the different words occurring in the corpus is considered as the dimensionality of the term. The correlation between the documents is computed based on the similarity which is calculated as the angle among the document vector and the query vector. The above calculation is described in equation (4) which is represented as follows:

$$cos\ \theta = (y\_1.\ y\_2)/(\|y\_1\|\|y\_2\|) \quad (4)$$

When the result is 1, the two vectors are non-orthogonal or there exists complete similarity. If the result is 0, the there exists no similarity between the documents.

Luhn's summarization algorithm: Once the documents are converted to vectors and the similarity is computed, then it is required to proceed towards summarization. Once the summarization is done, there will a reduction in the size of the text and the process returns only the important keywords in the document. There are a few summarization algorithms. The Luhn's summarization algorithm is used along with this work. Luhn's summarization algorithm initially categorizes the words in the document into function words and content words. Parts of the literature like pronouns, prepositions, modal verbs, conjunctions, interjections and numerals are considered to be function words. They are also known as structural words since they do not possess a lexical meaning. Nouns, adjectives, verbs and adverbs are considered as content words. The emphasis is over the content words. The words emerging from the same root word are merged to the root word itself. For example, swords will be merged with its root word sword. When the content word frequency exceeds a certain threshold, it is chosen as a significant word in the document. The importance of a sentence is calculated based in the word frequency and the position of the word in the sentences. The calculation is as shown below in equation (5)

$$\frac{(Significant\ words\ in\ the\ cluster)^2}{(Total\ words\ in\ the\ cluster)} \quad (5)$$

The occurrence of above four invalid words among two important words shall be summarily rejected from being considered. This eliminates almost 60% of the original text in the document. The Luhn's summarization algorithm therefore helps in reducing the size of the text that is available without compromising the essence of the file. The two significant advantages of using the summarization algorithms are:

1. Elimination of function words and
2. Establishment of meaningful lexical relations

Word Net is prominently used to establish the lexical relationship between the entities. During the time of establishment of a relation, the following are considered:

a) *Synonyms:* If the meaning of a word is similar to that of the other then the word which is more popular amidst people shall be considered. The other word can be eliminated.

b) *Hyponym:* Hyponym is a improvement of a word. In a concept hierarchy, the hyponym usually occupies the lower levels. It can thus be utilized as a subset of certain significant words.

c) *Hypernym:* Hypernym is a generalization of a word. It's assumed to be the parent in a concept hierarchy. The purpose of a hypernym serves in providing an overall view of the knowledge of words.

XML is the most convenient way of representing an ontology. Once the ontology is constructed, it may even return words synonymous to a given keyword.

B. *Semantic similarity between ontologies*

There are various key terms related to this work such as semantic distance, semantic relatedness and similarity in the semantics. It should be observed as the terms semantic relatedness may alternatively be used in the place of semantic similarity. This calculates the degree in which the similarity is identified between two entities. Semantic distance is considered as the inverse of the similarity in semantics. This work uses semantic similarity as a measure to assess the relationship between entities. Measures based on semantic similarity works with Word Net and other linguistic research tools. However, this has successfully been applied to do- mains like bioinformatics, chem informatics, and biomedical sciences. The measures related to the similarity in the semantics are properly based on the topology of an ontology. The major approaches can be classified into two types in measuring the semantic similarity. The first one is edge-based which uses the properties of the edges as the data source. The other one is node-based that makes use of the properties of the nodes as the data source. Two major approaches are used for the comparison of sets of terms. The first one is the pair-wise approach which computes the similarity by merging the similarity value among the terms present in the sets. The similar approach namely the group-wise approach uses other representations like graphs or vectors to perform the comparison.

In this work, four most prevalent node-based measures are chosen. The similarity is calculated using pairwise strategy. The measures taken are that of Resnik, Jiang-Con-rath, Lin and SimRel. The node-based methods are taken since the edge-based methods are prone to several irregularities like the depth of the variable and the density of the variable etc. These identified measures are clearly based on the Information Content (IC) and this measures the specificity along with the informativeness of a term d using negative log-likelihood:

$$IC\ (\mathrm{d}) = -\ log\ (\mathrm{d}) \quad (6)$$

Where log p(d) is the probability of the term d in a specific corpus.

The Resnik can be used to measure the similarity of the semantics similarity among the two terms $c1$ and $c2$ as the IC of their Most Informative Common Ancestor (MICA)

$$(c1, c2) = (cMICA) = -log\ P(cMICA) \quad (7)$$

The Lin and Jiang-Conrath measures has taken the information content similar to that of Resnik measure and have revised it as follows:

$$sim\_Lin\ (c\_1,c\_2\ ) = (2\times IC\_((C\_MICA)))/(IC\ (c\_1\ )+IC\ (c\_2)) \quad (8)$$

$$sim\_jc\ (c\_1,c\_2\ ) = 1/(IC\ (c\_1\ )+IC\ (c\_2\ )-2\times IC\_((C\_MICA))+1) \quad (9)$$

SimRel measure combines the advantage of Lin measure and Resnik measure. The SimRel measure is proposed as follows:

$$simRel(c1, c2) = simLin(c1, c2) \times [1 - p(cMICA)] \quad (10)$$

IV. Results and Discussion

The experimentation is done with two sample job domains namely the nurse and Information Technology Professional (IT). The proposed SSJOSR is compared with previously existing methods namely DOCF and JKO. The proposed system will get a better performance and the system is related with the existing systems by means of the parameters like recall, precision, accuracy and F- measure. These are commonly deployed to compare with the algorithms which generated ontology. The evaluation is done based on the data sources obtained from the web.

A. *Performance Analysis*

Recall, Precision, Accuracy and F- Measure are some of the key parameters that are used to compare the performance of semantic web query matching which is based on the ontology.

$$Precision = truepos/(truepos+falsepos) \quad (11)$$

$$Recall = truepos/(truepos+falseneg) \quad (12)$$

$$Accuracy = (truepos+trueneg)/(truepos+trueneg+falsepos+falseneg) \quad (13)$$

$$F\text{-}Measure = 2.\ (Precision.recall)/(Precision+recall) \quad (14)$$

Where truepos represents the true positive results, trueneg represents the true negative results, falsepos represents the false positive results, falseneg represents the false negative results

The comparison table represents the performance of the method that is proposed with that of the existing methods. This is represented in terms of two job profiles is illustrated in Table 1.

Table 1 Performance Analysis.

	Nurse			Information Technology		
	JKO	DOCF	SSJOSR	JKO	DOCF	SSJOSR
Precision (%)	57.6	63.8	67.8	59.3	65.2	72.2
Recall (%)	54.7	60.2	68.4	59.5	65.4	77.6
Accuracy (%)	61.7	66.5	74.8	58.1	64.3	75.8
F-measure (%)	56.46	61.94	67.5	59.5	65.5	75.95

It is clearly evident from Table 1 that the SSJOR provides better performance compared to that of DOCF and JKO in the creation of ontologies. The tabular representation is interpreted as a graph in Figure 1.

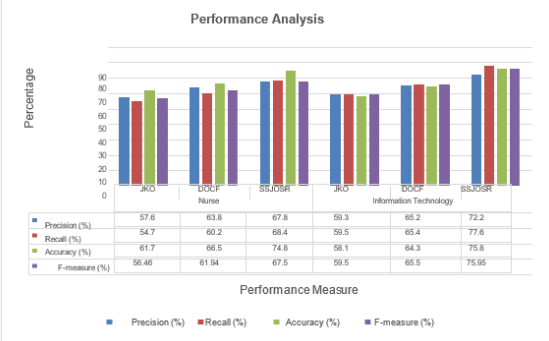

Figure 1: Performance Analysis Chart

V. REFERENCES

[1] N. Kumaresh, P. Senthil Kumar and Dr. J. Abdul Samath, "Certain Investigations on Automated Domain Ontology Construction," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 11, 2018, pp. 581-585.

[2] N. Kumaresh, J. Abdul Samath & M. Mohamed Iqbal, "An automatic construction of domain ontology to analyze competence prerequisites of jobs," *International Journal of Enterprise Network Management*, vol. 9, no. 3/4, 2018, pp. 441–454.

[3] R. Hoehndorf, P. N. Schofield & G. V. Gkoutos, "The role of ontologies in biological and biomedical research: a functional perspective," *Briefings in bioinformatics*, vol. 16, no. 6, 2015, pp. 1069–1080. https://doi.org/10.1093/bib/bbv011.

[4] B. Jou, T. Chen, N. Pappas, M. Redi, M. Topkara & S. F. Chang, "Visual affect around the world: A large-scale multilingual visual sentiment ontology," *In Proceedings of the 23rd ACM international conference on Multimedia, ACM,* 2015, pp. 159-168. https://doi.org/10.1145/2733373.2806246.

[5] Claudia d'Amato, Steffen Staab, G. B. Andrea, Tettamanzi, Minh Tran Duc, Fabien Gandon, " Ontology Enrichment by Discovering Multi-Relational Association Rules from Ontological Knowledge Bases," *In: Proceedings of SAC '16 - 31st ACM Symposium on Applied Computing, Pisa, Italy*, 2016, pp. 333-338. ff10.1145/2851613.2851842ff. ffhal-01322947.

[6] J. Rowley & R. Hartley, "Organizing knowledge - An introduction to managing access to information," *Ashgae Publishing Ltd*., 2008.

[7] F. Pheasant-Kelly & N. Russell, "Revisionist Vampires: Transcoding, Intertextuality, and Neo-Victorianism in the Film Adaptations of Bram Stoker's Dracula," *Neo-Victorian Villains: Adaptations and Transformations in Popular Culture*, 2017, pp. 325-343.

[8] J. Alber, "The social minds in factual and fictional we-narratives of the twentieth century. Narrative," *The Ohio State University Press*, vol. 23, no. 2, 2015, pp. 213-225.

[9] S. C. Shapiro & D. R. Schlegel, "Use of background knowledge in natural language under-standing for information fusion. In Information Fusion (Fusion)," *In: Proceedings of the 18th International Conference on*, 2015, pp. 901-907.

[10] D. Vrandečić & M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 57, no. 10, 2014, pp. 78-85.

[11] T. T. S. Nguyen, H. Y. Lu & J. Lu, "Web-page recommendation based on web usage and domain knowledge," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 10, 2014, pp. 2574- 2587.

[12] Gillian Rose, "Visual methodologies-An introduction to researching with visual materials," *Fourth Edition. Sage Publishing*, 2016.

[13] A. Maedche & S. Staab, "Ontology learning for the semantic web," *IEEE Intelligent Systems*, vol. 16, no. 2, 2001, pp. 72–79.

[14] T. Yamaguchi, "Acquiring conceptual relationships from domain-specific texts," *In: Proceedings of IJCAI Workshop on Ontology Learning (OL), USA: Seattle*, 2001.

[15] A. Maedche and S. Staab, "Ontology Learning for the Semantic Web," *IEEE Intelligent Systems. Special Issue on the Semantic Web*., vol. 16, no. 2, 2001, pp. 72 –79.

[16] P. Cimiano and J. Volker, "Text2Onto – a framework for ontology learning and data driven change discovery," *In: Proceeding of 10th International Conference on Applications of Natural Language to Information Systems, NLDB 2005, Alicante, Spain*, June 15 - 17, 2005, pp. 227– 238.

[17] R. Navigli and P. Velardi, "Learning Domain Ontologies from Document Warehouses and Dedicated Websites," *Computational Linguistics*, vol. 30, no. 2, 2004, pp. 151–179.

[18] M. Shamsfard and A. Barforoush, "Learning ontologies from natural language texts," *Int. J. Human-Computer Studies*, 2004, pp. 17–63.

[19] B. Fortuna, M. Grobelnik, D. Mladenic, "Ontogen: semi-automatic ontology editor," *In: Proceedings of the 2007 conference on Human interface: Part II*, 2007, pp. 309–318.

[20] M. Dahab, H. Hassan and A. Refea, "TextOntoEx: automatic construction form natural English text," *Expert Systems with Applications*, vol. 34, no. 2, 2008, pp. 1474–1480.