RESEARCH ARTICLE OPEN ACCESS

Probabilistic Relational Concept Extraction in Ontology Learning

S. Suresu [1], M. Elamparithi [2]
Research Scholar [1], Assistant Professor [2]
PG Department of Computer Applications
Sree Saraswathi Thyagaraja College, Pollachi
Tamil Nadu - India

ABSTRACT
Progresses in ranges, for example, characteristic dialect preparing, data recovery, machine learning, information mining, and information representation to understand a constantly developing assemblage of literary data in electronic structures, referred to just as data. The intermixing of systems from these regions has empowered us to remove and speak to realities and examples for enhancing the administration, access and interpretability of data. Notwithstanding, it was not until the turn of the thousand years with the Semantic Web dream and the blast of data due to the "Read/Write" Web that the requirement for a deliberate group of study in extensive scale extraction and representation of certainties and examples turned out to be more self-evident. Cosmology Learning (OL) which expects to turn realities and examples from a regularly developing assortment of data into shareable, abnormal state builds for improving ordinary applications (e.g. Web look) and empowering insightful frameworks. Ontologies are successfully formal and unequivocal determinations, as ideas and relations, of shared conceptualizations. Ontologies may contain sayings for acceptance and authorizing requirements. Probabilistic Relational Of Concept Extraction in Ontology Learning (PROCEOL) depicts the techniques for idea extraction.

Keywords:- Ontology learning, Concept extraction

I. INTRODUCTION

Probabilistic Relational Concept Extraction is a technique for extracting ontology concepts from natural language corpora that uses probabilistic relational learning. Since MLNs work with relational data, natural language corpora must be pre-processed in order to extract relational data. Once the corpus is pre-processed it can be used as input for concept extraction. PROCEOL extracts concepts from the pre-processed corpus. Probabilistic Relational Of Concept Extraction in Ontology Learning (PROCEOL) describes the methods of concept extraction. Concept Extraction is the process that contains the process of Concept Identification and the Concept Extraction. Most of the existing concept extraction is developed from the Markov Logic Network and also use the Probabilistic Latent Semantic Analysis. In our proposed method concept extraction process is based on the Noise removal and Co-occurrence analysis. Co-occurrence analysis is the process for analyze co-occurrence (related data). Thus PROCEOL consists of the four steps: Noise Removal, Co-occurrence Analysis, Concept Identification and Concept Labeling as shown in Figure -1.

II. MARKOV LOGIC NETWORK

In first-order logic, a first-order knowledge base KB can be seen as a set of hard constraints on the set of possible worlds. A formula is false if there exists a world that violates it. Some systems allow a limited violation of constraints but this is more in the sense of some tolerance to noise. The basic idea in MLNs is to soften these constraints so that when a world violates one formula in the KB, this formula becomes less probable, but not impossible. Each formula has an associated weight that reflects how strong a constraint it is: the higher the weight, the greater the difference in log probability between a world that satisfies the formula and one that does not, other things being equal. The two main tasks concerning MLNs are learning and inference. Inference is conducted either to find the most probable state of the world y given some evidence x, where x is a set of literals, or to calculate

the probability of query q given evidence x. Learning can be divided into structure learning and parameter (weight) learning; both generative and discriminative models can be applied to them. Structure learning consists in finding a set of weighted formulas from data while parameter learning focuses on computing weights for a given set of formulas. We detail these tasks in the following subsections.

2.1 Weight Learning

Given a set of formulas and a relational database, weight learning consists in finding formula weights that maximize either the likelihood or pseudo-likelihood measure for generative learning or either the conditional likelihood or max-margin measures for discriminative learning.

2.1.1 Generative Weight Learning

Generative approaches try to induce a global organization of the world and thus, in the case of a statistical approach, optimize the joint probability distribution of all the variables. Concerning MLNs, given a database, which is a vector of n possible ground atoms in a domain or x = (x1 … xl, … xn), where xl is the truth value of the l-th ground atom (xl = 1 if the atom is true, and xl = 0 otherwise), weights can be generatively learned by maximizing the likelihood of a relational database shown in equation-1.

$$P(X = x \mid M_{L,C}) = \frac{1}{z}\exp\left(\sum_{i \varepsilon F}\sum_{j \varepsilon G_i} w_i g_j(x)\right) = \frac{1}{z}\exp\left(\sum_{i \varepsilon F} w_i n_i(x)\right) \quad (1)$$

where $n_i(x)$ is the number of true instantiations of F_i in x.

2.1.2 Discriminative Weight Learning

In discriminative learning, a weight vector w is discriminatively learned by maximizing the conditional log-likelihood (CLL) or the max-margin of a set of query atoms Y given a set of evidence atoms X. We first present the CLL and its optimization methods then the ones related to the max-margin. The CLL of Y given X is defined by:

$$P(Y = y \mid X = x) = \frac{1}{z_x}\exp\left(\sum_i w_i n_i(x, y)\right) \quad (2)$$

where Z_x normalizes over all possible worlds consistent with the evidence x, and $n_i(x, y)$ is the number of true groundings of the i^{th} formula in data. The negative CLL is a convex function in case the truth values of all predicates are known in training data. It may no longer be convex when there is missing data. All methods are based on convex optimization hence complete data is required by using the closed-world assumption.

2.2 Structure Learning

The structure of a MLN is a set of formulas or clauses to which are attached weights. In principle, this structure can be learned or revised using any inductive logic programming (ILP) technique.

2.2.1 Discriminative Structure Learning

In many learning problems, there is a specific target predicate that must be inferred given evidence data; discriminative learning is then preferred. ILS-DSL (Iterated Local Search - Discriminative Structure Learning): It learns discriminatively first-order clauses and their weights. This algorithm shares the same structure with the ILS approach for generative MLN structure learning. It also learns the parameters by maximum pseudo-likelihood but scores the candidate structures using the CLL measure.

2.3 Inference

There are two basic types of inference: finding the most likely state of the world consistent with some evidence, and computing arbitrary conditional probabilities.

III. ONTOLOGY LEARNING USING MARKOV LOGIC NETWORK (OL-MLN) PROCESS

Statistical relational learning combines the expressive power of knowledge representation formalisms with probabilistic learning approaches, thus enabling one to represent syntactic dependencies between words and capturing statistical information of words in text. Many statistical relational learning approaches have been proposed in the literature. Markov Logic Networks (MLNs) constitute an approach for statistical relational learning that combines first order logic with Markov random fields. An MLN is a first order logic knowledge base with weights that can be either positive or negative, associated to each formula. While a traditional first order logic knowledge base is a set of hard constraints on the set of possible worlds, i.e. each world that violates a formula is impossible; an MLN is a set of softened constraints. The higher the weight of a formula, the less probable a world violates it is. Worlds that violate formulas with negative weights are more probable instead.

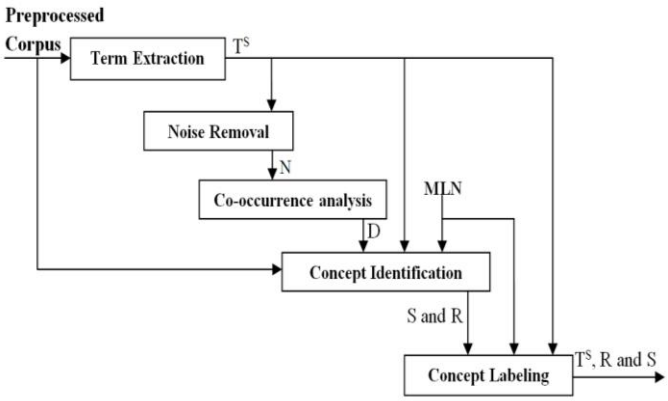

Figure-1: Ontology Learning using Markov Logic Network (OL-MLN) Technique

Statistical relational learning is an approach for machine learning that combines the expressive power of knowledge representation formalisms with probabilistic learning, this enabling one to represent syntactic dependencies between words and capturing statistical information of words in text. Many statistical relational learning approaches have been proposed. Ontology Learning using Markov Logic Network (OL-MLN) is an approach for statistical relational machine learning that combines first order logic with Markov Logic Networks. The OL-MLN technique is shown in Figure-1.

3.3.1 Term Extraction

Term Extraction is a Data Mining technique; in general, it is a Preprocess task in any text process application because of the text dataset converts into individual term dataset so that we can perform further process for learning ontology. Term Extraction process consists Tokenization, Stemming, Stop word removal, Lemmatization and Term Weight. Tokenization is process to split the word and annotated with Part of speech tag for finding Noun or Verb. Part of Speech is software is used to find grammatical meaning of the word. Second process is Stemming is used to reduce the word from their stem. Third process is Stop word filter prepositional words from the text. Lemmatization is the process of grouping together word forms that belong to the same inflectional morphological paradigm and assigning to each paradigm its corresponding canonical form called lemma. Lemmatization process performs the four main processes that are Removal of suffix of length, Addition of new lemma suffix, Removal of prefix of length and Addition of new lemma prefix. Final step of the term extraction is Term weight which calculated by the scores of TF-IDF calculation that is the Term Frequency and the inverse document frequency.

3.3.2 Noise Removal

Noise removal is the task that are used for identify the unidentified words. Noise Removed data N is performed by the Noise removal process that contains the following process.

Spell checking: This is avoiding the error in the spelling and find out the correct spelling of the words.

Abbreviation: This is for identifying the abbreviations for the correct acronyms.

Word Variants: These word variants are used for the related words like synonyms.

Completion of pre-processing, Spell checking process is applied in to the terms. The spell mistake word in this sentence is “diff erent”, is converted into the “different”. This spell checking is the process that performed by the English dictionary as name as Dictionary_English, which is help us to make spell checking. Then the abbreviation is applied in the “LA”, Expansion of “LA” is “Las Angeles” is identified as the country name of the abbreviations, and finally word variants to identify the different name for the particular words. This is also identified in this process and makes the correct sentences into the concept extraction process.

3.3.3 Co-Occurrence Analysis

Co-occurrence analysis is broadly used in different forms of research concerning the domains of ontologies, text mining, data extraction, knowledge extraction and content analysis etc,. Commonly its aim is to find similarities in sense between word pairs and similarities in meaning within word patterns, also in order to discover latent structure of mental and social representations. Co-occurrence data is said to be the D which have the terms that are similar for particular context. Co-occurrence analysis challenges to identify lexical units that tend to occur together for purposes ranging from extracting related terms to discovering implicit relations between concepts.

3.3.4 Concept Identification

The problem here is to learn the set of ontology concepts from a given text corpus D. The approach proposed here learns concepts through their linguistic realizations, i.e. each concept is learned as a set of natural language terms.

3.3.4.1 Lexical Realization

Lexical Realization is the natural language counterpart to a UW. It can be a sub word (a root, an affix), a simple word or a multiword expression (compounds, collocations, idioms). The expression “realization" stands here for a mixture of wording and phrasing, i.e., the manner in which the concept is articulated in a given language. . For instance, the UW 109358358 (= “the natural satellite of the Earth”) is realized, in English, by the word “moon”, in French, by “lune”, in German, by “Mond”, in Russian, by “luna”, in Chinese, and so on. LRs, however, are not simply linguistic realizations; they are lexical realizations. This means that LRs should correspond to the units of the vocabulary of a language, i.e., to a “lexical item”. The differences between “defining” and “naming” a concept, are fairly subjective, and are normally ascribed to the compositionality (or analyticity) of the candidate term: if the meaning of the compound can be reduced to the combination of the meaning of its components, it is said to be simply a definition; otherwise, i.e., if there is a sort of semantic surplus, a supplementary (or even complementary) sense added to the simple combination, the term is considered a lexical item. The expression “the natural satellite of the Earth”, for instance, does not bring any new semantic content to the ones conveyed by its components. This is not the case of “geostationary communications satellite”, which subsumes the idea of “orbit” which is not explicitly present in the compound. Accordingly, “geostationary communications satellite” (208.000 occurrences in Google) should be treated as a LR, whereas “the natural satellite of the Earth”, in spite of its higher frequency, should not.

3.3.4.2 Concept Identification Process

Concept Identification process is performed by the technique of Markov Logic Network. In order to implement the concept identification using Markov Logic Network, system presents the process of Weight Learning and the Inference. Learning of weight contains the three methods, which are Discriminative Learning, Generative Learning and Imitative Learning. Generative learning is at the main process of many approaches to pattern analysis and recognitions, artificial intelligence, and perception and provides a rich framework for imposing structure and prior knowledge on a given problem.

Discriminative Learning is the maximizing of conditional log likelihoods. Imitative Learning is

process that presented as another variation on generative modeling which also learns from exemplars from an observed data source. Here system uses the learning weight of discriminative learning, because the discriminative prediction task outperforms the usual generative approach which maximizes the joint likelihoods of all predicates. For make the discriminative learning method voted Perceptron, Conjugate Gradient, and Newton's Method are used for it. For all the learning weight finally provides the inputs into the Expectation Maximization which is an iterative method used to find maximum likelihood estimates of parameters in probabilistic models.

Completion of learning weight is continuous with the process of probabilistic inference which is performed by the Markov Chain Monte Carlo. In this process, the evidence file is used for the input of inference. So we have to collect the evidence files from the concept identification process. If a given document contains a given word, then the predicate HasDocument(word, documents) is true for that pair; otherwise it is false. Finally this could be described as, given by the predicate Topic(class, documents).

$$\text{Hasword}(+g;p) \Rightarrow \text{topic}(+c,p)$$

In simple MLN model have the formulas of linking word to page classes and page classes to the classes of linked pages. The word-class rules were described as,

$$\text{Has}(p,g) \Rightarrow \text{Class}(p,m)$$
$$\neg\text{Has}(p,g) \Rightarrow \text{Class}(p,m)$$

Here pair of (word, class) is described as p is represent as page, g represents as word, and m represents as class. Linked page classes were related by the formula of

$$\text{Class}(p1,m1)\ \text{LinksTo}(p1, p2) \Rightarrow \text{Class}(p2,m2)$$

Therefore, system needs a rule for each (word, class) pair and a rule stating that, given no evidence, a document does not belong to a class: Topic(c, p). Evidences are collected from the true pairs of the predicates which are the Concept (concepts). Here the concept is described as the Concept (ck) The performance of inference process is to gather the truth values of the possible trainings of the R(concept; word) predicate which means that a word is a lexical realization of a concept, i.e. (wj , ck) Є R, based on the evidence. Here the document is represented as d, and the word or term represented as g. The Term g is collected as {g1,g2,g3,g4, … ,gN}, N be the number of corpus. The evidence is composed of a set of trainings of the HasWord(document; word) predicate, which means that a term is present in a document and the Depends(word, word, dependency) predicate, which means that a word governs another word through the specified syntactic dependency, which is described as,

$$\text{depends}(g3,g1,\ \text{dep}) \wedge \text{depends}(g3,g2,\text{dep}) \wedge R(c,g1) \Rightarrow R(c,g2)$$

To make the complex relations in Markov Logic Network inference are used us probabilistically. There are two basic types of inference: maximum a posteriori (MAP)/most probable explanation (MPE) inference that finds the most probable state of the world consistent with some evidence, as well as conditional/marginal probability inference that finds the conditional/marginal distribution of a formula or a predicate. In this PROCEOL technique, system uses the MC-SAT algorithm; it is the process combines the MCMC and SampleSAT. SampleSAT combines the MaxWalkSAT and the simulated annealing. WalkSAT is repeatedly flipping a variable in a random unsatisfied clause. It is described by with probability p and with probability 1-p. Like that MaxWalkSAT is the highly non-uniform sampling that is extended to the algorithm, which is also been used in algorithms for approximate counting of large data sets. This process is mostly applied to optimization problems. A Markov Chain Monte Carlo process is used for computing marginal and conditional probabilities in Markov Logic Networks. This technique is performed by the combination of Markov Chain Monte Carlo with the MC-SAT algorithm. MC-SAT selects a

random gratifying assignment of a random subset of the currently satisfied phrases.

MC-SAT algorithm:

Y(0) ← Hard clauses random solution

For k ← 1 to total number of samples that are taken in the processing (whole data sets)

R ← ϕ for probability of 1-exp(-w_i) add C_i to R

Sample Y(k)

Uniformly random solution satisfying R

Hard and soft constraints which are often present in MLNs. Here Y(0) describes the hard constraints. k is the samples that we are taken for the process. R is the random subset of clauses. It can be shown that the set of samples from MC-SAT combines to the correct distribution as long as the satisfying assignments are selected uniformly at random. MC-SAT provides the probabilistic inference. Then completion of this process it performs the process of Probabilistic Latent Semantic Analysis (PLSA).

IV. RESULTS AND DISCUSSIONS

The researcher conducts an extensive set of experiments to examine the concept extraction performance of the proposed PROCEOL framework by comparing with a state-of-the-art technique. The proposed PROCEOL consists of the process of noise removal with the concept extraction. That provides the accurate extraction results. This extraction results and other dataset results are shown in below.

The PROCEOL technique was evaluated by comparing its output with a gold standard. For this purpose we use the data set of LonelyPlanet corpus for performing the concept extraction task. Here we evaluate three concept extraction techniques were used in order to extract the concepts from the LonelyPlanet corpus. This three extracted techniques are CPROCEOL, CPROCEOLdep, and CPLSA. CPROCEOL is the concept extraction using the PROCEOL technique. This PROCEOL technique is based on the Markov Logic Networks, which is performed by the Alchemy software packages.

Technique	PRECE			PROCEOL		
	CPRECE	CPRECEdep	CPLSA	CPROCE-OL	CPROCE-OLdep	CPLSA
Precision	5.1726	2.2708	5.3415	5.4291	2.4291	5.4291
Recall	0.3708	0.1	0.6	0.4291	0.1	0.6
F1 Measure	0.6919	0.1915	1.0788	0.7953	0.1920	1.0805

Table-1: Dataset

PROCEOL technique consists of the steps of Noise removal, Co-occurrence Analysis, Weight Learning and Inference calculations. Here the pre-processing steps like tokenization, parser, POS tags are performed by the GATE tool and also Stemming, Stop words, Lemmatization are performed by the java code. Finally we calculate the Term weight for the pre-processing process. This all process is covered by the PROCEOL techniques. CPROCEOLdep is the concept extraction using the PROCEOL technique using Markov Logic Network. This PROCEOL system extract the concept without consider the syntactic dependencies between terms. CPLSA is the concept extraction using traditional Probabilistic Latent Semantic Analysis. Topic discovery and topic extraction process is performed using this Probabilistic Latent Semantic Analysis. Concepts are labeled with the term with highest probability given the particular topic.

These three processes are calculated by the Precision, Recall and F1-Measures. [36] defined the calculation of the F1 measures. F1 measures is calculated as

$$\text{F1-measures} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

This is the basic formula for the F1-Measures. Here Precision and Recall is calculated by the formula of

$$\text{Precision} = \frac{|\text{Relevant documents} \cap \text{Retrieved documents}|}{|\text{Retrieved documents}|}$$

$$\text{Recall} = \frac{|\text{Relevant documents} \cap \text{Retrieved documents}|}{|\text{Retrieved documents}|}$$

Here the Relevant documents is said to be the Reference retrieval and the Retrieved documents is said to be the computed retrieval. So the final F1 measure is process of reference retrieval with the computed retrieval. Retrieved documents (i.e. computed retrieval) are the results of the gold standard; Table 3.1 shows the results of the precision, recall and F measures. The relevant documents (i.e. Reference retrieval) are the results of the concept of ontology learning.

V. CONCLUSION

Probabilistic Relational Of Concept Extraction in Ontology Learning (PROCEOL) is a concept extraction technique, which applies Markov Logic Network to learn ontologies with extraction. Probabilistic Relational Of Concept Extraction in Ontology Learning (PROCEOL) technique is used for Term Extraction and Concept Extraction. Term Extraction is used for extract the word which contains the process of Tokenization, Parser, Part of Speech analysis, stemming, Stop word, Lemmatization and finally the Term Weight. Concept extraction process is the new tasks of noise removal and co-occurrence analysis in to the Markov Logic Networks. Concept extraction process extracts the word with the spell mistakes, problem of analyzing abbreviations and issue of identifying word variants.

REFERENCES

[1] Faure, D., Nedellec C., and Rouveirol, C., Acquisition of Semantic Knowledge using Machine Learning Methods: The System ASIUM, Technical Report number ICS-TR-88-16, Laboratoire de Recherche en Informatique, Inference and Learning Group, Universite Paris Sud, 1998.

[2] Yamaguchi, T., Acquiring Conceptual Relations from domain-Specific Texts, Proceedings of the IJCAI 2001 Workshop on Ontology Learning (OL'2001), Seattle, USA, 2001.

[3] Shamsfard M., and Barforoush, A. A., (a) An Introduction to HASTI: An Ontology Learning System, Proceedings of 6th Conference on Artificial Intelligence and Soft Computing (ASC'2002), Banff, Canada, June, 2002.

[4] Hahn U., Romacker, The SYNDIKATE Text Knowledge Base Generator, Proceedings of the 1st International Conference on Human Language Technology Research, San Diego, USA, 2001.

[5] Chalendar, G., Grau, B., SVETLAN' A System to Classify Nouns in Context, Proceedings of the ECAI 2000 Workshop on Ontology Learning (OL'2000), 2000

[6] Ferret O., and Grau, B., A Thematic Segmentation Procedure for Extracting Semantic Domains from Text, Proceedings of ECAI'98, Brighton, 1998.

[7] Maedche, A., and Staab, S., (b) Semi-Automatic Engineering of Ontologies from

Text, Proceedings of the 12th International Conference on Software Engineering and Knowledge Engineering (SEKE'2000), Chicago, 2000.

[8] Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., Slattery, S., Learning to construct knowledge bases from the World Wide Web, Artificial Intelligence, 118: 69-113, 2000.

[9] Wagner A., Enriching a Lexical semantic Net with selectional Preferences by Means of Statistical Corpus Analysis, Proceedings of the ECAI 2000 Workshop on Ontology Learning (OL'2000), Berlin, Germany, 2000.

[10] Li H., Abe N., Learning Word Association Norms Using Tree Cut Pair Models, Proceedings of the 13th International Conference on Machine Learning, 1996.

[11] Heyer G., Läuter, M., Quasthoff, U., Wittig, T., Wolff, C., Learning Relations using Collocations, Proceedings of the IJCAI 2001 Workshop on Ontology Learning (OL'2001), Seattle, USA, 2001.

[12] Bikel, D. A., Schwartz, R., and Weischedel, R., An Algorithm that Learns What's in a Name, Machine Learning, 34, 211-231, 1999.

[13] Cherfi, H., and Toussaint, Y., How far Association Rules and Statistical Indices help Structure Terminology? Proceedings of the ECAI 2002 Workshop on Machine Learning and Natural Language Processing for Ontology Engineering (OLT'2002), Lyon, France, 2002.

[14] Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., Slattery, S., Learning to extract Symbolic Knowledge from the World Wide Web, AAAI'98, 1998.

[15] Maedche, A., and Staab, S., Measuring Similarity between Ontologies, Proceedings of EKAW'02, Spain, 2002.

[16] Zelle, J. M., Mooney, R. J., Learning Semantic Grammars with Constructive Inductive Logic Programming, Proceedings of the 11th National Conference on Artificial Intelligence (AAAI'93), 817-822, Washington D.C., 1993.

[17] Bisson, G., Learning in FOL with a Similarity Measure, Proceedings of 10th National Conference on Artificial Intelligence (AAAI'92), 82-87, San Jose, California, 1992.