www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume - 3 Issue - 8 August, 2014 Page No. 7615-7621

Concept Identification Using Markov Logic Network

K. Karthikeyan[1], Dr. V. Karthikeyani[2]

[1]Research and Development Centre, Bharathiar University, Coimbatore.
[2]Thiruvalluvar Govt. Arts College, Rasipuram.
*kk_karthikeyan2007@rediff.com
#drvkarthikeyani@gmail.com

ABSTRACT: Ontology is a most essential technology in Data and Knowledge Engineering. Because of Ontology provide many advantages over Object Oriented Concepts, like Knowledge Sharing, reusability, Interoperability and Knowledge Level Validation and Verification. Ontology is a collection of concepts that represent knowledge in the domain and there exist common terminology to provide types, methods and relationship between those concepts in the domain. Ontology used in the form of structural framework in many field like Artificial Intelligence, Information Science, Semantic Web and etc., this concept identification presents an ontology building through the automatic and semi-automatic process. Most of the ontology learning technique developed using the Classifiers, NLP, probabilistic and statistical learning. For the concept identification it uses the process of statistical learning with the combination of text
.

Keywords - Concept, Concept Identification, Ontology learning, Markov Logic Netowrk

1. INTRODUCTION

Ontology is an explicit formal specification of a shared conceptualization of a domain of interest, where formal implies that the ontology should be machine readable and shared that it is accepted by a group or community. Web abstracts are themselves a web of pages which are accumulating of affiliated alone pages, anniversary with alternative scrolling area, text, images and added media embedded. Ontologies play a crucial role in many net and net related applications as they are suggests that by these can be model and share info throughout a selected domain. Ontology learning, within the linguistics internet context, is primarily involved with knowledge acquisition from and for online page and is so moving far away from small and homogeneous knowledge collections to tackle the huge knowledge non-uniformity of the planet Wide internet instead [1]. Typical human language users have a noteworthy ability to investigate sounds and different gestures in a bound very subtle approach. One of our main goals in finding out language is to know however this is often done, [2] and the way that ability arises within the human mind. Conceptual structures that outline Associate nursing underlying metaphysics are relate to the thought of machine processable data on the Semantic Web.

Ontologies [3] are (Meta) data schemas, providing a controlled vocabulary f ideas, every with Associate in Nursing expressly outlined and machine processable linguistics. By process hared and common domain theories, ontologies facilitate each folks and machines to speak in brief, supporting the exchange of linguistics and not solely syntax. Hence, a budget and quick construction of domain-specific ontologies is crucial for the success and therefore the proliferation of the linguistics net [4]. Ontologies are the formal specification of concept in a domain.

Ontology is an explicit specification of a naturalistic vocabulary for a domain, definitions of categories, relations, functions, constraints and alternative objects. [5], [6], [7], [8], [9], [1] Pragmatically, a typical metaphysics defines the vocabulary with that queries and assertions square measure changed among package entities. Ontologies are not restricted to conservative definitions that within the ancient logic sense solely introduce nomenclature and don't add any information concerning the globe. To specify a conceptualization we had like to state axioms that place constraints on the attainable interpretations for the outlined terms. Here, retrieval of web data from the web based on ontology learning. The web documents are playing a vital role for decision purpose whether the concept relation is to be present in ontology or not. The mechanism is based on the Term, Synonyms, Concepts, Concept Hierarchies, Relations and Rules.

In concept identification we have to use the Markov Logic Networks for processing the learning weight and inference. In our concept we use the discriminative learning process for finding the learning weight [7], [8].

This learning weight maximizes the conditional likelihoods of the query predicates of given evidence and the atoms of unknown truth values handled with EM (Expectation maximization). The Markov logic network [9], [10], [11], [12], [13], [14] is used in our concept. We use the MCMC (Markov chain and Monte Carlo) combination with MC-SAT algorithm to find the probabilistic inferences [15]. Deterministic dependency produce disconnected regions, with out support of probability distribution, it seems to be complicated in design Markov chains for MCMC inference.

An MCMC algorithm using SampleSAT procedure to solve deterministic and near-deterministic dependency in proper way and switch over between isolated or near-isolated region with non-zero probability. This MC-SAT get input from Markov logic, this Markov logic has Markov network and first-order logic and used to calculating conditional probability in graphical model using Markov Chain Monte Carlo. So, MC-SAT used to process the sample into Markov logic using SampleSAT to generate new state for given variables.

2. RELATED WORK

For providing the best result in concept identification, [6] described that Probabilistic Relational Concept Extraction combines both the statistical analysis with the probabilistic learning approach. For identifying ontology concepts from the natural language corpus, the method Markov Logic Networks used. A Markov logic network is a first-order knowledge base with a weight attached to each formula. Here Markov Logic Network method is solved by the MCMC (Markov Chain Monte Carlo) inference with the combination of MC-SAT algorithm. MC-SAT [25] described that it is an inference procedure that combines ideas from MCMC. Also paper describes the MC-SAT procedure is derived from Markov logic, consists Markov networks using weighted clauses in first-order logic.

3. MARKOV LOGIC NETWORK

Markov Logic [17] is indistinguishable from first-order logic, except that every formula includes a weight connected. Markov logic may be a straightforward nonetheless powerful combination of markov networks and first-order logic. Markov logic raises the quality of markov networks to encompass first-order logic. Recall that a first-order domain is defined by a collection of constants representing objects within the domain and a collection of predicates representing properties of these objects and relations between them. A predicate are often grounded by replacement its variables with constants. A first-order Knowledge Base (KB) may be a set of formulas in first-order logic, created from predicates victimization logical connectives and quantifiers. A formula in markov logic may be a formula in first-order logic with associate degree associated weight. The basic plan in markov logic is to melt these constraints. When a world violates one formula within the K it's less probable, however not possible. Less formulas a world violates, a lot of probable it's. Associate degree MLN [1] is often viewed as a template for constructing markov networks. In different worlds it will manufacture different networks and this also has wide varied size, however all can have certain regularities in structure and parameters, given by the MLN.

Combining chance and first-order logic in a very single illustration has long been a goal of Artificial Intelligence. Probabilistic graphical models change us to efficiently handle uncertainty. First-order logic permits us to succinctly represent a good variety of information.

A Markov Logic network [21] may be a first-order knowledge domain with a weight attached to every formula, and may be viewed as a temple for constructing Markov Networks. MLNs give a compact language to specify terribly giant a

Markov Networks, and therefore the ability to flexibly and modularly incorporate a large vary of domain information into them. Several necessary tasks in applied mathematics relative learning, like collective classification, link prediction, link-based cluster, social network modeling, and object identification, area unit naturally developed as instances of MLN learning and illation.

Markov Logic, linguistics internet languages can be created probabilistic just by adding weights to statements and linguistics web illation engines may be extended to perform probabilistic reasoning merely by passing the proof of Directed Acyclic Graph (DAG) [26], [27], [28], [29], [30] with weights connected, to a probabilistic illation system. Weights is also set by hand inferred varied sources or learned mechanically from information. MLN acts as a template for a markov network. We have extended and adapted several of these standard methods to take particular advantage of the logical structure in a markov logic network, yielding tremendous savings in memory and time. Markov logic combines first-order logic and probabilistic graphical models in a unifying representation. The main idea behind Markov Logic is that, unlike first-order logic, a world that violates a formula is not invalid, but only less probable.

Statistical Relative learning combines [2] the communicatory power of data representation formalisms with probabilistic learning approaches, therefore enabling one to represent grammar dependencies between words and capturing applied mathematics information of words in text. The markov logic network represent an approach for applied mathematics relative learning that mixes first order logic with markov random fields. Associate MLN could be a first order logic mental object weights, which may be either positive or negative, associated to every formula.

The main idea behind Markov logic [23] is that, unlike first-order logic, a world that violates a formula is not invalid, but only less probable. This is done by attaching weights to first-order logic formulas: the higher the weight, the bigger is the difference between a world that satisfies the formula and one that does not, other things been equal.

Two common inference tasks in Markov Logic are the maximum a posteriori (MAP) and probabilistic inference. MAP inference aims at finding the most probable state of the world given some evidence. In Markov Logic this task is the same as finding the truth assignment that maximizes the sum of the weights of satisfied formulas. This can be done by any weighted satisfiability solver. Probabilistic inference aims at determining the probability of a formula given a set of constants and, maybe, other formulas as evidence. The probability of a formula is the sum of the probabilities of the worlds where it holds. There are two approaches for learning the weights of a given set of formulas. There are generative and discriminative learning. Generative learning aims at maximizing the joint likelihood of all predicates while discriminative, at maximizing the conditional likelihood of the query predicates given the evidence ones. Maximum-likelihood or MAP estimates of Markov network weights cannot be computed in closed form, but, because the log-likelihood is a concave function of the weights, they can be found efficiently using standard gradient based or quasi-Newton optimization methods

A term is any expression representing an object in the domain. It can be a constant, a variable, or a function applied to a tuple of terms. For example, Anna, x, and GreatestCommonDivisor(x, y) are terms. An atomic formula or atom is a predicate symbol applied to a tuple of terms (e.g., Friends(x, Mother of (Anna))).

Parentheses may be used to enforce precedence. A positive literal is an atomic formula; a negative literal is a negated atomic formula. The formulas in a KB are implicitly conjoined, and thus a KB can be viewed as a single large formula. A ground term is a term containing no variables. A ground atom or ground predicate is an atomic formula all of whose arguments are ground terms. A possible world or her brand interpretation assigns a truth value to each possible ground atom.

The syntax of the formulas in an MLN is the standard syntax of first-order logic. An MLN can be viewed as a template for constructing Markov networks. Given different sets of constants, it will produce different networks, and these may be of widely varying size, but all will have certain regularities in structure and parameters, given by the MLN. The Probability distribution over possible worlds x specified by the ground markov network. It can be represented by

$$P(X=x) = \frac{1}{Z}\exp\left(\sum_i w_i n_i(x)\right) = \frac{1}{Z}\prod_i \phi_i(x_{\{i\}})^{n_i(x)}$$

We defined MLNs as log-linear models, they could equally well be defined as products of potential functions, as the second equality above shows. This will be the most convenient approach in domains with a mixture of hard and soft constraints. The graphical structure of markov network follows from there is an edge between two nodes of markov network iff the corresponding ground atoms appear together in at least one grounding of one formula in L. Thus, the atoms in each ground formula form a (not necessarily maximal) clique in markov network.

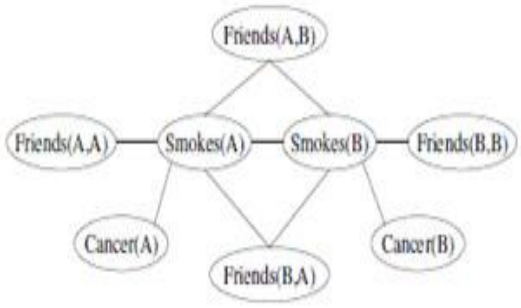

Figure 1 Ground Markov Network

A first-order KB can be seen as a set of hard constraints on the set of possible worlds, if a world violates even one formula, it has zero probability. The basic idea in MLNs is to soften these constraints: when a world violates one formula in the KB it is less probable, but not impossible. The fewer formulas a world violates, the more probable it is. 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. In an MLN, the derivative of the negative conditional log-likelihood (CLL) with respect to a weight is the difference of the expected number of true groundings of the corresponding clause and the actual number according to the data.

This last assumption allows us to replace functions by their values when grounding formulas. Thus the only ground atoms that need to be considered are those having constants as arguments. Features can also be learned from data, for example by greedily constructing conjunctions of atomic features.

4. CONCEPT IDENTIFICATION

4.1. Preprocess

Preprc (1) is the task perform before the Concept Hierarchy Extract. In Preprocess the following tasks are perform, Tokenization, POS, Chunking, Syntactic Analysis, Stemming, Lemmatization, Stop Word, Term Weighting, Hypernym Extraction.

Tokenization is a task to split the word from the sentence and mark the starting and ending of the words. The tokenized words are send to input for Part Of Speech, it is one of the annotated tag process in which each tokenized word are identified as Subject, Verb or Noun etc., after that the annotated process data are under going for Chunking, it is divide the text according to syntactically correlated word. Then, the system perform Syntactic Analysis, is the method to analysis the words according to the concern language grammar, here apply the English grammar, it construct parse tree, it show that semantic relation between the words.

Stemming is process to reduce the word to their root. For example, the fishing, fisher, fished reduces to fish. In contrast to Stemming, Lemmatization is algorithmic process to group the derived or stemming words so that analyses as single item according to the domain context. Then the system perform Stop Word algorithm to remove or filter the function words like a, an, the, which, what, etc. Term Weighting is process to find frequency of an each in the corpus and calculate inverse document frequency also. Finally in preprocess task, the system perform Hypernym Extraction, is process to group the words according to their semantic relation. For example, rose, jasmine, are hyponyms of flowers.

4.2. Concept Identification

Concept Identification is an important portion covered in our proposed system. Concept identification is performed by the technique of MLN. Using MLN we have to perform the process of learning weight and inference. Figure 2 describes the process of concept identification.

For performing the learning weight we have to use the method of MLN. To find the weights in a database we have to use the Maximum a Posteriori (MAP) weight method. This means the weights that maximize the product of their prior probability and the data likelihood. Pseudo-likelihood is that the product of the conditional chance of every variable given the values of its neighbors within the data. Whereas economical for learning, it will offer poor results once long chains of inference are needed at enlarging time.

Pseudo-likelihood is systematically outperformed by discriminative coaching, it minimizes the negative conditional probability of the question predicates given the evidence ones. This learning weight can be performed by four methods. First, progress based on voted Perceptron. Here, using gradient descent algorithm use the gradient named as g, scaling based learning rate η, and to update the weight vector w, it can be represented by,

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \mathbf{g}$$

The spinoff of the negative conditional log-likelihood (CLL) with relevancy a weight is that the distinction of the expected range of true groundings of the corresponding clause and therefore the actual range in step with the information.

$$\frac{\partial}{\partial w_i} - \log P(Y=y|X=x) = E_w[n_i] - n_i$$

Where y is the state of the non-evidence atoms in the data, and x is the state of the evidence.

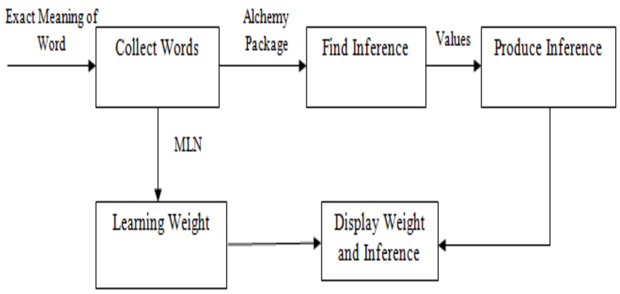

Fig. 1 Concept Identification

The second process is the contrastive divergence. In contrastive divergence we use MCMC algorithm. The MCMC algorithmic program usually used with contrastive divergence is Josiah Willard gibbs sampling, expect for MLNs a lot of quicker various method MC-SAT is offered. Because ordered sample in MC-SAT square measure a lot of less related to than ordered sweeps in Josiah Willard gibbs sampling, they carry additional data and square measure doubtless to yield a better descent direction. Specially, the various samples square measure doubtless to be from completely different modes, reducing the error and potential instability related to choosing one mode.

The third progress is per-weight learning rates. To modify each algorithms to own a distinct learning rate for each weight. Since standardization of each learning rate individually is impractical, we use an easy heuristic to assign a learning rate to every weight.

$$\eta_i = \frac{\eta}{n_i}$$

Where η is the user-specified global learning rate and ni is the number of true groundings of the ith formula. These values are being fixed, so it cannot be contribute to the variance.

The final process in the series is Diagonal Newton. In diagonal newton we just multiplying the gradient, g, by the inverse Hessian, H inverse.

(5)

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \mathbf{H}^{-1}\mathbf{g}$$

In Diagonal Newton (DN) methodology, this uses the inverse of the diagonoized jackboot insitu of the inverse jackboot. DN typically uses a smaller step size than the total Newton methodology. The main aim of this method is to found the step size. In each iteration, we take a step in the diagonalized Newton direction

(6)

$$w_i = w_i - \alpha \frac{E_w[n_i] - n_i}{E_w[n_i^2] - (E_w[n_i])^2}$$

Then we compute the step size,

$$\alpha = \frac{-\mathbf{d}^T\mathbf{g}}{\mathbf{d}^T\mathbf{H}\mathbf{d} + \lambda \mathbf{d}^T\mathbf{d}}$$

Where d is the search direction. For a quadratic function and λ= 0, this step size would move to the minimum function value along d.

Regarding inference we have to perform the task of finding inference using alchemy software we have to finalize the inference values of each word in the schema.

5. CONCLUSION

Pre-processing consists of many activities. These are all placed in ontology learning progress. Pre-processing is used for extracting meaningful words from the corpus. The pre-processing activities are could be performed by tools and languages. In our process we use GATE tool for performing operations of tokenization, POS tagging, chunking and syntactic analysis. After that we had to do the activities of stop-word removal, stemming, lemmatization, term weighting and also hypernym extraction. These all are done by using Java language. Second process is concept identification. In concept identification we use MLN method to learning the weight of words. For that purpose we can use the simple weight learning method to produce the good results. The main progress include in that is to find the inference values. For finding the inference we could use alchemy process. It may be the software to produce the optimized values of every word in the corpus. Alchemy Packages also used for implement the concept identification process. Alchemy packages are used for make the perfect inference process.

6. FUTURE ENHANCEMENT

In this paper we present efficient concept identification technique, this technique could provide the best concept identification process with more accuracy and also the efficiency. Thus our proposed technique gives better process. But to improve the relationship in ontology learning we can move onto the process of semantic relation. In future, the idea to make the relationship in semantic web use association rule mining for joining the relationship. To identify the non useful words we want to implement the semantic relation. Then we need to implement another process named as axiom learning. Axiom learning is an important process in learning ontology. The entire final step of the process is to implement the ontology population. Ontology population is used to analyze the population in semi automatically. This idea is decided to implement in future.

REFERENCES

[1] Philipp Cimiano, Andreas Hotho and Steffen Staab, "Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis," in Journal of Artificial Intelligence Research, vol. 24, 2005, pp. 305–339.

[2] Hassan Khosravi," Discriminative Strucure and parameter learning for Markov Logic Networks", In Proceedings of the 25th International Conference on Machine Learning (ICML), Helsinki, Finland, July 2008.

[3] Dellschaft, K., Staab, S.: On how to perform a gold standard based evaluation of ontology learning. In: Proceedings of ISWC-2006 International Semantic Web Conference, 2006.

[4] K.Karthikeyan and Dr.V.Karthikeyani, "Migrate Web Documents into Web Data," Electronics Computer Technology (ICECT) 3rd International Conference, 2011, Vol. 5, pp. 249 - 253.

[5] Thomas Hofmann, "Probabilistic Latent Semantic Analysis," In Proc. of Uncertainty in Artificial Intelligence, UAI'99, 1999, pp.289-296.

[6] Karthikeyan.K and Dr.V.Karthikeyani, "Understanding text using Anaphora Resolution", Pattern Recognition, Informatics and Mobile Engineering (PRIME) 2013 International Conference, 2013, pp- 346 – 350.

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[8] Wilson wong, wei liu and mohammed bennamoun, "Ontology Learning from Text: A Look Back and into the Future," ACM Comput. Surv. 44, 4, August 2012, 36 pages.

[9] Paul Buitelaar, Philipp Cimiano and Bernardo Magnini, "Ontology Learning from Text: An Overview," in Applications and Evaluation, 2005, pp. 3--12.

[10] Sharon A. Caraballo, "Automatic Construction of a hypernym-labeled noun hierarchy from text", 99 Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, pp-120-126, 1999.

[11] Marie-Catherine de Marneffe, Bill MacCartney, Christopher D. Manning, "Generating Typed Dependency Parses from Phrase Structure Parses", Department of Computing Science, Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation, pp- 1-8, 2006.

[12] Lucas Drumond and Rosario Girardi, "A Survey of Ontology Learning Procedures," WONTO, volume 427 of CEUR Workshop Proceedings, CEUR-WS.org, 2008.

[13] Lucas Drumond and Rosario Girardi, "An Experiment Using Markov Logic Networks to Extract Ontology Concepts from Text," in ACM Special Interest Group on Applied Computing, 2010, pp. 1354-1358.

[14] Marti A. Hearst, "Automatic Acquisition of Hyponyms from Large Text Corpora," COLING '92 Proceedings of the 14th conference on Computational linguistics, vol. 2, Jan. 1992, pp. 539-545.

[15] Martin KAVALEC and Vojtech SVATEK, "A Study on Automated Relation Labelling in Ontology Learning," in Ontology Learning from Text: Methods, Evaluation and Applications, 2005, pp. 44--58.

[16] Stanley Kok, Parag Singla, Matthew Richardson and Pedro Domingos, "The Alchemy System for Statistical Relational AI: User Manual", proceeding at University of Washington, Department of Computer Science and Engineering, Aug 3, 2007.

[17] A.Matthew Richardson and Pedro Domingos, “Markov Logic Networks,” 2006, vol. 6, pp. 1–44.
[18] GerardSalton, Christopher Buckley, “Term Weighting Approaches in Automatic Text Retreival”, Information Processing and Management: an International Journal, vol.24, 1998, pp- 513-523.
[19] Rion Snow, Daniel Jurafsky and Andrew Y. Ng, “Learning Syntactic Patterns for Automatic Hypernym Discovery”, In: Advances in Neural Information Processing Systems (NIPS 2004), December 13-18, 2004.
[20] Fei Wu, Daniel S. Weld, “Automatically Refining the Wikipedia Infobox Ontology”, 17th international conference on World Wide Web, pp- 635-644, 2008.
[21] Quang-Thang DINH, Christel Vrain and Matthieu Exbrayat, ”Generative Structure Learning for Markov Logic Network Based on Graph of Predicates”, IJCAI'11 Proceedings of the Twenty-Second international joint conference on Artificial Intelligence – Vol. 2, pp. 1249-1254.
[22] Sandeepkumar Satpal , Sahely Bhadra , S Sundararajan , Rajeev Rastogi, Prithviraj Sen “Web Information Extraction Using Markov Logic Networks,” 2011.
[23] Ting Wang, Yaoyong Li, Kalina Bontcheva, Ji Wang and Hamish Cunningham, Automatic Extraction of Hierarchical Relations from Text”, 3rd European conference on The Semantic Web: research and application, pp- 215-229, 2006.
[24] Kaustubh Beedkar, Luciano Del Corro, Rainer Gemulla, “Fully Parallel Inference in Markov Logic Networks”, proceeding at Max-Planck-Institut für Informatik, vol.2, 2011, pp- 373-384 .
[25] Hilda Koopman, Dominique Sportiche and Edward Stabler, “An Introduction to Syntactic Analysis and Theory”, 2013.
[26] Sandeepkumar Satpal, Sahely Bhadra, S Sundararajan, Rajeev Rastogi and Prithviraj Sen, “Web Information Extraction Using Markov Logic Networks”, 17th ACM SIGKDD international conference on Knowledge discovery and data mining, 2011, pp- 1406-1414.
[27] David-Liben-Nowell, Jon Kleinberg, “The Link Prediction Problem for Social Networks”, CIKM '03 Proceedings of the twelfth international conference on Information and knowledge management, 2003, pp- 556-559.
[28] Hoifung Poon Pedro Domingos, “Sound and Efficient Inference with Probabilistic and Deterministic Dependencies”, AAAI'06 Proceedings of the 21st national conference on Artificial intelligence - Volume 1,pp. 458-463.
[29] K.Karthikeyan and Dr.V.Karthikeyani, “PROCEOL: Probabilistic Relational of Concept Extraction in Ontology Learning”, International Review on Computers and software, Vol.9, No.4, 2014.
[30] Lucas Drumond and Rosario Girardi, “Extraction Only Concept Hierarchies from text using Markov Logic”, proceeding at federal university of Maranhao, Proceedings of the 2010 ACM Symposium on Applied Computing 2010, pp- 1354-1358.