High-Performance Linguistics Scheme for Cognitive Information Processing

D. Suryanarayana, Prathyusha Kanakam, S. Mahaboob Hussain and Sumit Gupta

Abstract Natural language understanding is a principal segment of natural language processing in semantic analysis to the use of pragmatics to originate meaning from context. Information retrieval (IR) is one of the emerging areas to deal with enormous amounts of data, which are in the form of natural language. Content of the query posed will affect both volume of data and design of IR applications. This paper presents a cognition-applied methodology termed as High-Performance Linguistics (HPL), which is a question-answering system for interpreting a natural language sentence/query. It constitutes three phases of computations: parsing, triplet generation and triplet mapping/matching. The generation of the triplets for the knowledge base is to create new data and compare them with that of stored triplets in the database. Thus, the generation of the cognitive question-answering system can make easy using this machine learning techniques on the generated triplet database.

Keywords Pragmatics · RDF · Triplets · Ontology · Information retrieval · Linguistics · Semantics · Indexing

D. Suryanarayana (✉) · P. Kanakam · S.M. Hussain · S. Gupta
Department of Computer Science & Engineering, Vishnu Institute of Technology, Vishnupur, Bhimavaram, Andhra Pradesh, India
e-mail: suryanarayanadasika@gmail.com

P. Kanakam
e-mail: prathyusha.kanakam@gmail.com

S.M. Hussain
e-mail: mahaboobhussain.smh@gmail.com

S. Gupta
e-mail: sumit108@hotmail.com

P.K. Sa et al. (eds.), *Progress in Intelligent Computing Techniques: Theory, Practice, and Applications*, Advances in Intelligent Systems and Computing 518,
DOI 10.1007/978-981-10-3373-5_37

1 Introduction

Nowadays, computation on the Web is a critical task to retrieve the accurate information because every minute the World Wide Web (WWW) is becoming big and big with lot of information and resources. Web crawlers handle this critical job to retrieve related information from the Web documents. Knowledge engineering is the major task to execute to achieve the information semantically from the current semantic Web. Semantic Web technologies initiate a huge impression to work on the semantic Web and semantic search make promising. To step into enhanced progress of information retrieval from the semantic web documents, some of the authors offered various practices such as Probabilistic Model and Vector Space Model [1]. For the advancement of semantic search and retrieving process, a variety of implements is put into practice and broadens by means of latent semantic indexing [2], machine learning [3] and probabilistic latent semantic analysis [4]. Semantic information helps computers to understand what we put on the web, and it was the current research issue of World Wide Web (WWW) to provide semantic data according to the query. The intent of the users to query the search engine in natural language interrelated to the human cognition. Thus, semantics are related to the intent and meaning of the users query. Most of the search engines try to provide the results as per the query posted from the huge repository of databases depending/according to the terms located in the query even though for the direct questions/query some search engines failed to answer. Some of the current search engines, especially semantic search engines, are trying to understand the intent/semantics of the user and their queries. They can provide the better results for any type of natural language queries. [5]. Interpreting the formal languages in the web content is more effortless using ontologies. Resource Description Framework (RDF) imparts to add semantic information to web pages.

2 Related Work

Estimating the cognition of user or their natural language queries (NLQ) is a tricky task for the system to retrieve the expected results. Most users are not satisfying with the results retrieved by a question-answering system. To facilitate the best results, every system needs to undergo a technique for understanding the content and semantics of the query.

2.1 Question-Answering System

In 1993, Boris Katz and his team developed the web-based question-answering system called START. It is not just like a search engines to retrieve the information

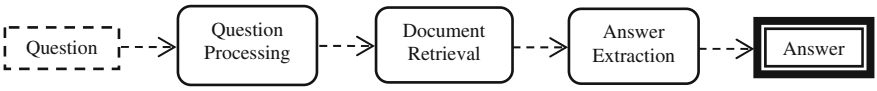

Fig. 1 Conventional question-answering system

depending upon the hits, but it simply supply the right information from the default database [6]. Richard J Cooper introduces a question-answering system in Perl that uses CMU Link Parser [7]. The main goal of the proposed question-answering system is to provide accurate content for the natural language query by the user to the system. Depending upon the semantic structure of the data, it is easy to build up a question-answering system with precise information for the posed queries and satisfies the user needs. The proposed system processes the database for the query and searches for the relative information, which is in triplet form constructed by the resource description framework. Question processing performs for the question classification of the trained machine learning algorithms. Figure 1 shows the three main segments of the question-answering system query processing, retrieval of documents and extraction of answers.

Question processing involves query posed by user through making it ready by changing its form to triplet for interpreting semantically in order to give better results by the machine. Document retrieval includes inquiring ontology database to fetch corresponding ontology for that query. Answer extraction implicates is to finding the property value of that particular triplet generated during question processing phase.

2.2 Machine Learning Using RDF and SPARQL

The general issue of machine learning is to look at a typically extensive space of potential theories to decide the one that will best fit the information and any earlier knowledge. The learner makes an expectation of the property of best, the right answer exhibited and the learner changes its theory appropriately. In supervised realizing, there is essentially the supposition that the descriptors accessible are identified with an amount of significance. The machine that learns to infer a function using trained data is supervised machine learning. Cognition applied to experience or to study whether result retrieved suits the query posed or not. Many training examples will be in the training data, and each example classified into pairs of input vector object and a supervisory signal as an output that actually desired.

Parsing of the sentence involves two main process-text lemmatization and text categorization. Semantic interpretation of natural language query obtained after text categorization. To retrieve the actual content of a Natural Language Query given by a user to a search engine, it must change its query form to a variety of forms, i.e., the search process will be done in the microlevel of the database. To grab semantics from the natural language sentence, Lambda Calculus need to apply. Always prefer disambiguated language (a language without ambiguity) while dealing with the

semantics of the sentence in order to avoid ambiguities. There should be a compositional relation between syntax and semantics from the side of formal semantics. Principle of compositionality is defined as the significance of the sentence is a set of semantics of its elements and the process of that way of syntactically united [8]. In order to provide syntax and semantics for language *L*, every well-formed sentence in it must represent in a compositional way.

To retrieve exact results from the semantic web documents which are in the form of RDF format, a unique query language is used called SPARQL. This specification defines the semantics as well as syntax of the SPARQL to RDF. Finally, the outcome of the queries in SPARQL syntax will be in triplet or in graphical representation called RDF graphs. Mostly, the syntax of the query of SPARQL represents conjunctions, disjunctions and some optional patterns. Therefore, the entire semantic web documents are in <*subject, object, predicate*> triples [9]. SPARQL endpoint is RDF triple database on server usually, which is available on web and top of web transfer protocol, there is a SPARQL protocol layer means via http SPARQL query transfers to server and server gives its results to client. It is like SQL but works on RDF graphs not on tables. Graph pattern is RDF triple that contains some patterns of RDF variables. These patterns combined to get different patterns of more complex results.

3 High-Performance Linguistic Scheme

Information retrieval is one of the emerging areas to deal with massive amounts of data that presented in natural language. Content of the query posed will affects both volume of data and design of IR applications. Text Lemmatization is critical process involved in question-answering system, which is the process of finding lemmas from the natural language sentence as well to assign some categories to those particular lemmas. It gives the best solution to solve the problem of grasping enormous amounts of data and handle it more efficiently. High-Performance Linguistics, a question-answering system, is a cognition-applied machine to learn how to infer the content of the natural language sentence. The path to give output from input is a trivial task to done. For that, a systematic procedure will give a clue to machine to interpret natural language sentence. Most of the supervised machine learning algorithms uses a model to project known outputs from known inputs as shown in Fig. 2. However, applying cognition to machine to comprehend various categories of the text and mapping of text to document is a complicated task.

To overcome this, HPL algorithm applied to query to infer the content of query. Here, the forms of query/natural language sentence will changes. In earlier work, Palazzo Matrix Model (PMM) gives the occurrence of the term in the document and handle whether the document is appropriate to the query posed by the user [10]. At parsing level in Fig. 3, each natural language sentence is categorized as lemmas and allotted respective categories.

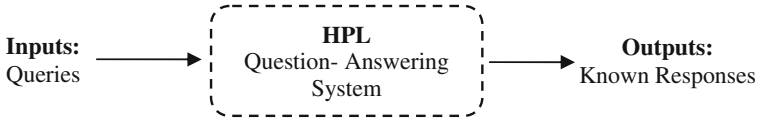

Fig. 2 Typical structure of a question-answering system

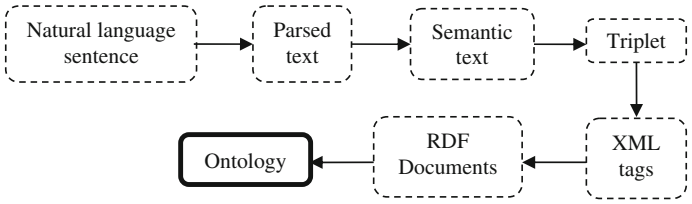

Fig. 3 Various states of a NLQ

Table 1 Category rules for semantic representations of NLP

Input trigger	Logical forms
A constant *c*	`NP : λ c`
Arity one predicate *p*	`N : λx.p(x)`
Arity one predicate *p*	`S\NP : λx.p(x)`
Arity two predicate *p*	`(S\NP)/NP : λx.λy.p(y, x)`
Arity two predicate *p*	`(S\NP)/NP : λx.λy.p(x, y)`
Arity one predicate p	`N/N : λg.λx.p(x) Λ g(x)`
Arity two predicate *p* and constant *c*	`N/N : λg.λx.p(x, c) Λ g(x)`
Arity two predicate *p*	`(N\N)/NP : λx.λg.λy.p(y, x) Λ g(x)`
Arity one function *f*	`NP/N : λg.argmax/min(g(x), λx.f(x))`
Arity one function *f*	`S/NP : λx. f(x)`

There are *425* lexical categories such as noun, pronoun and determiner. For semantic representation, directionalities (forward/backward) are applied, and there by combinatory rules are generated. Finally, from XML to RDF, documents are used to create respective ontology. Using the rules and classical programming language (λ-Calculus), the sentence will be explicated as parts as shown in Table 1.

3.1 HPL Algorithm

This algorithm contains three fragments—parsing, triplet generation and triplet matching for accurate generation of results from the ontology database. Before applying the algorithm, NLQ undergo preprocessing (removal of stop words).

Begin

$Q <n_1, n_2, ..., n_p, ..., n_m>$ /* Query with 'm' words*/

Step 1: $L<Q>$ /* Apply Lemmatization by removing stop words*/

Step 2: Derive categories for each word

$S = |n_1|, |n_2|, ... |n_p|,... |n_m|$

N/N NP/N... N/N ... NP

Step 3: Apply parsing process.

/* applying forward/ backward directionalities along with λ-calculus*/

N/N NP/N ... N/NP NP

N/NP N/NP

S

Apply λ-calculus then,

λ[P(x)] <=> P(x), where P is predicate and x is subject

Step 4: Apply triplet generation process

Using λ- notation, obtain triplet form

Q = <S> <O=?> <P>

where S= subject, P= predicate, O=object

Step 5: Apply matching process

/*Compare this triplet set with that of stored triplet sets in database*/

$Q = \{<S>, <O=?>, <P>$

$Q_1 = \{<RS_1>, <RO_1>, <RP_1>\}$

where $Q_1 \rightarrow$ RDF Triplet for Query Q

$<S> = <RS_1>$ /* Resource*/

$<O> = <RO_i>$ /* Property Value*/

$<P> = <RP_i>$ /* Property*/

Here, $<RO_1>$ gives the value of <P>. Therefore, it will search in resource description table:

Resource	Property	Property value
<Subject>	<Predicate>	<Object>
$<RS_i>$	$<RP_i>$	$<RO_i=?>$

The value of $<RO_1>$ is answer to query 'Q'

Step 6: Ontology related to output value from matching process is derived.

End

The link between λ-notations and triplet generation as <Subject> <Object> <Predicate> is simple through single transformation and single function definition scheme. Here, the object value is unknown, and output is the object value or property value. Then, derived triplet compared with existed triplets in the database. K-Nearest Neighbor is applied to find the similarity between test triplet which is generated with that of training triplets stored in database by using <Subject> and <Predicate>. Euclidean distance measure along with K-NN is employed to find similarity distance D_s between nearest triplet (t_n) with that of k-triplet (derived triplet). Consider, triplet t_n = <t_n(S)> <t_n(P)> and triplet t_k = <t_k(S)> <t_k(P)>, then

$$D_s(t_n, t_k) = \sqrt{\sum_{w=1}^{m} ((t_n(S)) - t_k(S))^2} \quad (1)$$

if $D_s(t_n, t_k) = 0$, then t_n is the matching triplet for t_k otherwise not.

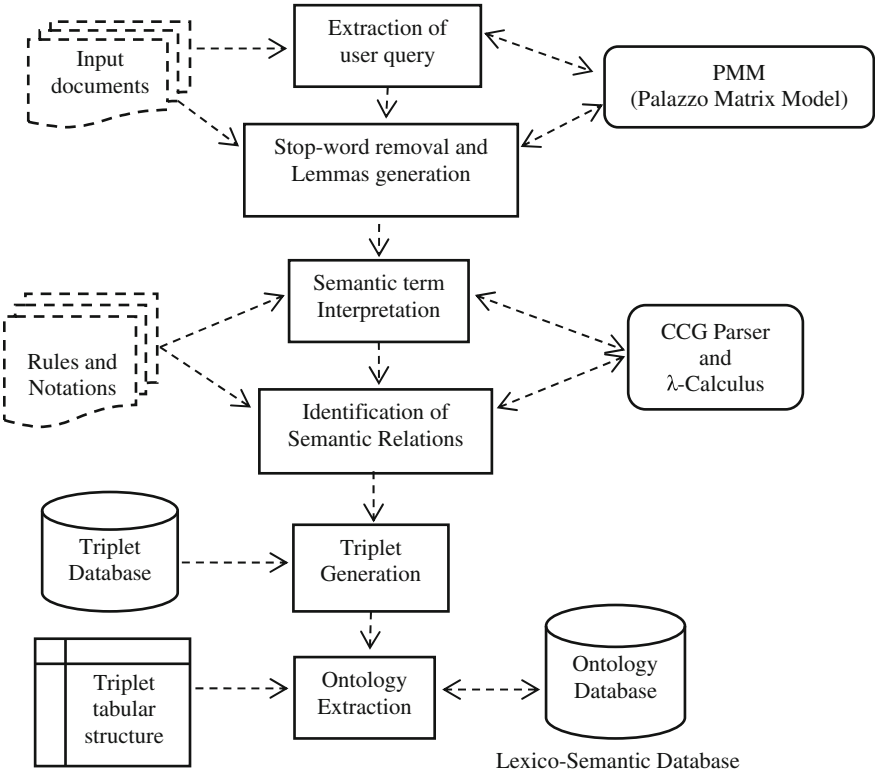

Fig. 4 Internal anatomy of HPL

Then, obtained triplet and associated ontology inferred from database by indexing it with from triplet tabular structures. Then, the machine projects the corresponding object or property value. It depicts a system that automatically learns ontologies. As shown in Fig. 4, the entire collection of ontology is termed as semantic knowledge bases enabling exciting applications such as question answering on open domain collections. This system automatically learns ontology from texts in a given domain. The domain-specific ontology that results from these knowledge acquisition methods incorporated into lexico-semantic database that various natural language processing systems may employ. This system helps to extract specific knowledge and for searching that knowledge from unstructured text on the web. It uses ontology-based domain knowledge base known as lexico-semantic database. Ontology conceptualizes a domain into a machine-readable format. Mostly, information on web represented as natural language documents. Knowledge extraction process involves reducing the documents into tabular structures (which indexed easily) for grabbing the context from the document i.e., answering to user's queries. HPL system mainly depends on triplet generated and domain ontology mapping to that triplet for extracting exact content or semantics from natural language queries posed by the user.

4 Experiments and Results

Observations were made on trial set of *10* queries in natural language, randomly collected from students. All the queries posed belong to single domain called educational ontology. Triplet generation for three queries among the ten is in Table 2.

Authors conducted several methods to find the exactness of HPL system on triplet database that contains 20 overall triplets on educational ontology and *10* sample natural language queries applied to that of triplets to match which are in the database. The correct value to give the exactness is precision, defined as,

Table 2 Triplet generation for the natural language queries

Query	Subject	Predicate	Object
What is the exam fee for JNTUK III–II semester external exams?	JNTUK III–II semester external exams	Has exam fee	# value (~650)
How to install python in PC?	Python in PC	Has install process	# value (URL)
Who is the author of social networks and the semantic web?	social networks and the semantic web	Has author	# value (Peter Mika)

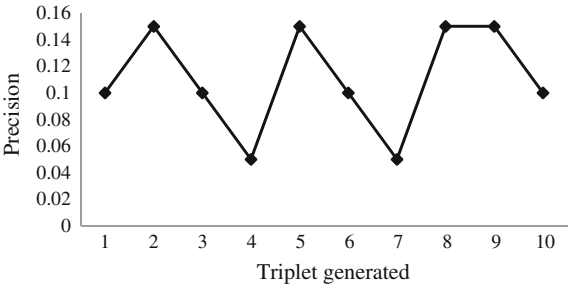

Fig. 5 Triplet generated versus precision calculated

$$\text{Precision} = \frac{\text{Correct triplets matched}}{\text{total triplets in database}} \tag{2}$$

Thus, *k* (generated triplet) values ranging from *1* to *10* and the precision value for each *k* value is computed. Mostly, the precision value is range of *0–0.1* as shown in Fig. 5. The values produced are giving evidence to show the truthfulness of HPL system.

5 Conclusion

Most of the question-answering systems developed based on text retrieval or web documents retrieval methodology where users may retrieve embedded answers from the systems. The idea behind this paper is to create the database with the relationships among the subject, object and predicate and make it accurate to answer the questions.

By simply parse the natural language sentence using CCG and λ-Calculus, generation of triplet made easy. Thus, matching algorithm actively searches for the matching contents in the database and generates the accurate and coherent answer for the question in the same triplet form without any long and embedded sentences. This cognitive informative processing mechanism is helpful to the users for their desired information by providing relevance and factually correct data from the database. In future, this retrieval process employed images to retrieve the relevant information in the semantic manner.

Acknowledgements This work has been funded by the Department of Science and Technology (DST), Govt. of India, under the grants No. SRC/CSI/153/2011.

References

1. Baeza-Yates, R, A., Ribeiro-Neto, B, A.: Modern Information Retrieval. ACM Press/Addison-Wesley, (1999).
2. Deerwester, S, C,. et al.: Indexing by latent semantic analysis. Journal of the American Society for Information Science and Technology – JASIS, Vol. 41(6), 391–407 (1990).
3. H. Chen.: Machine learning for information retrieval: Neural Networks, Symbolic learning, and genetic algorithms. Journal of the American Society for Information Science and Technology – JASIS, Vol. 46(3), 194–216 (1995).
4. Thomas Hofmann.: Probabilistic latent semantic indexing. In: International conference SIGIR '99, ACM, New York, NY, USA, 50–57 (1999).
5. Dumais, Susan, Michele Banko, Eric Brill, Jimmy Lin, and Andrew Ng.: Web question answering: Is more always better?. In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 291–298 (2002).
6. Boris Katz.: START: Natural Language Question Answering System. (1993), http://start.csail.mit.edu/index.php.
7. Boris Katz, Gary Borchardt and Sue Felshin.: Natural Language Annotations for Question Answering. In: 19th International FLAIRS Conference (FLAIRS 2006), Melbourne Beach, FL, (2006).
8. Partee, B, H.: Introduction to Formal Semantics and Compositionality. (2013).
9. Suryanarayana, D., Hussain, S, M,. Kanakam, P., Gupta, S.: Stepping towards a semantic web search engine for accurate outcomes in favor of user queries: Using RDF and ontology technologies. 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Madurai, India, 1–6 (2015).
10. Mahaboob Hussain, S., Suryanarayana, D., Kanakam, P., Gupta, S.: Palazzo Matrix Model: An approach to simulate the efficient semantic results in search engines. In: IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2015, Coimbatore, 1–6 (2015).