A NOVEL NATURAL LANGUAGE PROCESSING (NLP) BASED APPROACH FOR DEVELOPING AUTOMATED SEMANTIC CLAUSE PARSER

Krishnanjan B[1], Swati Mehta[2], Ajai Kumar[3]

[1]*Applied Artificial Intelligence Group, C-DAC, 5th Floor, Westend Centre-III, S.No - 169/1, Sector II, Pune, Maharashtra 411007, India*

[2]*Applied Artificial Intelligence Group, C-DAC, 5th Floor, Westend Centre-III, S.No - 169/1, Sector II, Pune, Maharashtra 411007, India*

[3]*Applied Artificial Intelligence Group, C-DAC, 5th Floor, Westend Centre-III, S.No - 169/1, Sector II, Pune, Maharashtra 411007, India*

Abstract

In traditional syntactic parsing, hierarchical breaking (tree or bracketed way) of sentences into its minimal word-level constituent factor is targeted, whereas there was no effort made to capture meaning of constituent clauses in a parsing process of breaking a sentence in constituents has been done. Hence, this innovative Natural Language Processing (NLP) application parses English Compound and complex sentences which are always major challenges in even traditional syntactic parsing-i.e., to break them into clause level and mark the clause with semantic annotations. This has far-fetched implication, this automated tool for Syntactic-Semantic merger using NLP opens new arena in directly developing a Q-A system, aiding to disambiguation of Machine Translation (MT) systems, Decision Support Systems (DSS) and also developing E-learning for language analysis tool to name a few.

Keywords— *NLP, Semantic, Parsing, Clauses, Semantic Annotation*

----------***----------

1. INTRODUCTION

The thematic area of this application is to develop Natural Language Processing (NLP) automatic semantic clause parser which works upon English language texts of reporting style analysis using rule-based and statistics based hybrid approach by innovatively developing formal rule-sets and algorithms for English sentence processing. This application offers as an output an automatic tree-structured graphical representation where automatically a sentence is parsed into clause-tree and clauses in the tree with nodes are semantically marked/annotated having visual representation of entire process.

2. OVERVIEW

In the area of computational language analysis or Computational Linguistics, parsing or syntactic analysis is the process of analyzing a string of symbols, either in natural language or in computer languages, according to the rules of a formal grammar.

The term has slightly different meanings in different branches of linguistics and computer science. Traditional sentence parsing is often performed as a method of understanding the exact meaning of a sentence, sometimes with the aid of devices such as sentence diagrams. It usually emphasizes the importance of grammatical divisions such as subject and predicate.

The study of structure of sentence is called syntax. It attempts to describe the grammatical order in a particular language in term of rules which in detail explain the underlying structure and a transformational process. Syntax provides rules to put together words to form components of sentences and to put together these components to form meaningful sentences. (1)

As mentioned above, syntactic parsing of a natural language sentence is providing the further pathway to NLP development, but a clause parser with its constituents being defined not only in terms of Part of Speech (POS) tagging but along with semantic roles marked on each node of the constituents has immense benefits hitherto unexplored.

In normal NLP practice, after POS analysis and then sentence representation as syntactic tree or bracketed form, the semantic and other NLP processes continue.

In different types of syntactic parsing that are available as standard linguistic tools like Stanford Parser developed by University of Stanford or Enju Parser developed by University of Tokyo, the emphasis was on structural breaking of sentence into noun-verb-adjective-adverb and its phrasal constituents.

The meaning of the constituent elements of a sentence that construe a parse-tree is not referred to in the traditional parsing systems, only the syntactic category like NP/VP/AdvP/PP etc. are mentioned. However, every element

of the sentence carries a semantic part or meaning that is integral to it and any syntactic analysis ultimately of little use if the understanding of various NLP processes are not achieved through it, like information retrieval or Decision Support System or even Machine Translation where it deals with ambiguity which is essentially a semantic-pragmatic property.

Therefore, to capture meaning and analyze meaning through parsing remained a grey area, as traditional parsing, as mentioned above, targets the structural category-wise breaking.

There was a need for a parser always that breaks sentence constituents into clauses with an aim of capturing meaning of clauses and also with factor of syntactic tree formation with the clause-chunks parsed with syntactic structure of English but marked with <TAG> of semantic nature, hence, leading to hybrid system of syntactic parsing with semantic annotation automatically achieved on constituent clauses of the given sentence. Normal parsers like Phrase Structure Grammar or Transformational Generative Grammar as proposed by Chomsky, in those cases, parsers always targeted upon getting constituents of a sentence in structural terms. As a matter of fact, syntax that traditionally addresses parsing does not merge with semantics as its sole interest remains in structural correlation of constituents in a sentence. It leads to famous Chomskian example of structurally/grammatically perfect English sentence without carrying meaningful idea, i.e., "*Colourless green ideas sleep furiously".*

In this novel approach of syntactic-semantic amalgamation as described in the current paper, the Information-Extraction Retrieval (IE-IR) kind of systems or Machine Translation (MT), Automated Summarizer, Named Entity Recognizer (NER) where semantic annotation are significant for linguistic rule formation and creation of statistical models for supervised learning are made straightway facilitated from this parser tagging containing both syntactic and semantic annotations. The offshoot is, optimization and less processing steps as time-efficiency and minimalism are the key factors for any NLP application development which is interactive and most of the NLP applications need to be interactive, taking input and giving output to user in real-time-let it be IE-IR, MT or Text Summarizer.

3. BENEFICIARIES - IMPACT AND INNOVATIVENESS

There exist many natural language Parsing techniques. These techniques are mainly categorized into three categories: (i) rule based (ii) statistical based and (iii) generalized parsers. All the developed parsers belong to any one of these categories and follow either 'top-down' or 'bottom-up' approach. Statistical parsing techniques are called "data-driven" and rule based parsing techniques are called "grammar-driven" approaches [1].

However, if syntactic-semantic parsing is done, the results can be path-breaking while straight-away Decision-trees of an AI system can be formed from semantic marker of clauses parsed by this clause parser or Question-Answering System can be formed by picking up constituent meanings from a dialogue parse through this semantic parser.

Automatic Text Summarizer can immensely benefit from such semantically annotated trees as the meaningful crux of the text can be picked up by logically following the tree hierarchy and picking up semantic marking on them with the help of well-formulated linguistic and pattern based rules.

There will be significant application of such syntactic-semantic hybrid parser in the field of ambiguity resolution in MT.

Decision Support System (DSS)-a next level of IE-IR, where from a given textual input decision or inference about content can be drawn automatically.

The most important beneficiary of syntactic-semantic merger and co-relative analysis is Natural Language Generation (NLG) which is an emerging field that attempts to replicate human cognition through NLU or Natural Language Understanding and NLG or Natural Language Generation. This emerging field can create true AI as cognition expressed through generated language by machine is key to true AI.

4. OUTCOME

The outcome of this application, as described above, will provide user an application to visualize a sentence of clausal type as input into an output through Graphical User Interface represented as hierarchically broken into constituent clauses and the semantically important clauses like TIME/CAUSE/PLACE/OUTCOME etc. are marked automatically (Semantic Annotations on Nodes) in the automatic tree-formation of clauses out of the sentence that is containing any number of clauses. In the later section a screenshot of actual application and explanation is provided to understand the process.

A simple sentence is not processed as this is a 'Semantic Clause Parser', so clausal sentences are input to it. Clause breaking into simple sentences helps in resolving complex structures of compound and complex/wh sentences in simple dependent string of clauses and each node is annotated further with aforesaid semantic annotations.

5. DETAILED TECHNICAL DESCRIPTION OF THE APPLICATION

This developed system is a full-fledged functional clause parse that depicts automatically clausal sentence into tree-structure and tree-nodes are annotated as per the SEMANTIC TYPE of the clause automatically. However, further rules and variations can be added to cater to more semantic-types of clauses. That is the level of expandability. Here, as a prototype it handles common semantically significant clause

types and also clausal sentences that follow general English grammar.

5.1 Application Overview

This section contains a detailed description of processes and linguistic or NLP oriented resources used to build this application. It involves explaining the theory behind each major functional step as well as resources that came into use.

1) Scope of the Application: This semantic clause parser application parses clause type of sentences of English language written in normal English grammatical way. Since tree-formation with semantic-annotation is shown for each clausal sentence given as input, it is required that sentence by sentence input is given to it to see automatic clausal tree formation with automatic generated semantic annotations on tree-nodes.

The system does not break the constituents of sentences beyond the level of clauses, as semantic-annotation of several of clauses in compound/complex sentences are targeted and are to be input to the application.

2) System Architecture: The system provides a Graphical User Interface which is made web based to run in the intranet for prototype and can be available in WWW as per need.

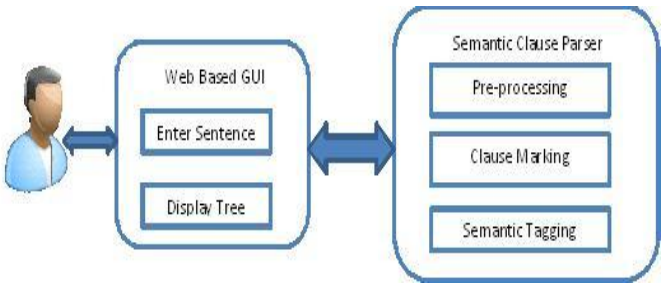

Fig. 1 High Level Architecture Diagram

Functional Steps: As it is stated earlier, the input to the system is a sentence which has to be either complex sentences, i.e. having wh clauses, dependent clauses etc. or compound sentences, for example, sentences connected with 'and' , 'hence', 'but' etc. conjunctions. It is to be noted every clause marker or sentence connector, which connects two simple sentences into a complex or a compound sentence are semantically identified so that they can evoke semantic relevance when they occur in sentence.

The simple sentence, i.e., a sentence that does not contain more than one main verb cannot be input to this clause parser as mentioned earlier. Rather breaking into constituents of simple sentences containing at least one main verb is the aim of the system.

Pre-processing: POS Tagging and Valid Sentence Identification: The first step is to understand a sentence as a valid sentence by linguistic rule sets and models developed for the application where using HMM based Part of Speech (POS) tagger and rules based on the PoS category behaviour in the sentence token continuum the main verb is identified. POS tagging used in the current application follows the University of Pennsylvania developed POS tag-set which is Universal standard for POS taggers.

Use of WordNet: WordNet is the outcome of a research project at Pennsylvania University regarding the creation of a large database of English words [2]. It is available both as an online and offline resource. Any given word in WordNet is tagged as being part of one, or more, parts-of speech (POS); verbs, adjectives, nouns or adverbs. Words of the same POS with similar (conceptual) meaning are grouped together in synonym sets, denoted as *synsets*.

A synset contains a short text description of its conceptual meaning in addition to its list of words (synonyms). More interestingly, the sysnets also contain semantic and lexical references to other synsets. That is, the various synsets reference each other forming complex networks. The references are of a super-subordinate relation. Meaning, general sets link to more specific ones (and vice versa), more specific ones link to very specific, e.g., the set containing the word { *red* } would link to the set with { *crimson* }, as *crimson* is a special kind of *red*. Furthermore, it would also link to color*}*, as red is a *color*.[3] The WordNet with its semantically connected synsets serve along with POS category identification the pillars of linguistic rule formation in identifying semantic roles of sentence constituents like Noun Phrase (NP), Verb Phrase (VP), instrumentals, locatives, causal sentences, statements, conjunctive clause types etc.

Morphological Analyser and Stemmer: The proper noun and its variations are marked and normalized with the help of morphological analyser and lexical stemmer ingeniously built. The Analyzer and Stemmer identifies proper nouns or NPs like single lexical item as NEP (Named Entity-Person): 'Chandran' vs. use of 'Chandran' in a pluralistic fashion. *"Chandrans are a happy family"* where 'Chandrans' and 'Chandran' need to be mapped together and normalized through this Morphological Analyser and Stemmer.

Phrase Marker: After all the above processes, the Proper Noun Phrase chunks like Names, Places which contain more than one word, e.g. 'Saina Nehwal' or place like 'New Delhi' are clubbed together to serve as one entity. Same process is followed for phrasal verbs like 'give up' which actually means 'surrender' hence, a single semantic entity or 'turn up means 'arrive'; both are treated as VP chunks and joined entities for future annotation purpose.

In traditional syntactic parsing the above steps are not followed, and the departure from traditional parsing to this indigenously designed 'Semantic Clause Parser' starts at this pre-processing level itself.

Word Sense Disambiguation (WSD)

Word sense disambiguation (WSD) is the process of determining which sense of a word is used in a text. For instance, consider the sentence "I'm having an old friend for dinner". Does the word having referred to what is being

served, or does it refer to the company? Techniques involve supervised, semi-supervised, unsupervised and cross-lingual evidence methods. Other methods involve the use of dictionaries and knowledge-bases. One such method is the Lesk algorithm. Lesk assumes that the words in a given phrase are likely related to each other and that the relations can be observed through the words definitions [8]. Words can thus be compared—through their definitions—finding the pairs that are most closely related.[3] In the application presented here, WSD is used to disambiguate cases like *'I will bank on you for this job which is important.'* vs. *'I will go to the bank to withdraw money that is located far away from my home.'* In the first case the word 'bank' means 'depend upon' and act as a verb and in the second case it means the 'financial institute'. The word 'bank' is polysemous in nature, hence, WSD technique is applied to derive correct 'meaning in context' in the application elaborated here.

Active/Main Verb Marker with Active-Passive Annotation: The verbs are marked by Penn tagset of POS as 'VB', 'VBG', 'VBZ' etc. as per their category, for example, verb in present continuous form is marked as 'VBG'. But every verb marked by POS tagger is not main verb in the sentence. This module has been developed indigenously to address the issue and based on linguistic rules of English as well as trainable lexical database, out of many verb tagging in a sentence main verb can be identified. For example, system will identify verbs like this: *"Arriving at the Hyderabad Railway Station by train, he went straight to his hotel."* In this sentence 'arriving' is a VBG according to POS tagging process and 'went' is tagged as VBD (verb in past form). This module correctly tags the main verb 'went' and marks even its role as active verb with a mark 'A-verb'.

Clause Marking: Based on the main verb in a sentence and clause markers the clause in marked and represented internally in traditional bracketed forms, breaking them into simple sentences. Later that is depicted through the Graphical User Interface into tree structure for visual purpose of the user.

Semantic Annotations on Clauses: The semantic annotations are marked on clauses based on the clause indicators and their semantic value that is depicted in context. The WSD module along with WordNet and verb identifier play role in determining a clause marker like 'as' will be depicted as 'causal' dependent clause in case of *"(Many people died)* ***as*** *(there were heavy landslide for rain in the Himalayas)<causal clause>."* Whereas, the sentence: *"(He presented himself to interview board)* ***as*** *(he is in his usual self without pretending) <conjunctive clause>."*

Hence, as we see above, 'as' being the clause marker does not have fixed semantic value but changes context and the semantic clause marker application using the above methodologies and developed linguistic rule-sets and models, disambiguate and tag the semantic nature of the clause along with marking the clause itself.

Depiction in Tree Structure: Finally, based on the above processes, a Graphical User Interface is triggered in to depict the clause separation along with semantic marking/annotations on clauses into a visual tree structure (Fig. 3). The parent-child top down order from main sentence to its clausal constituents along with semantic tagging/annotations are depicted. If there is no semantic annotation available for a constituent clause, then the syntactic marker like <S-> is used. In case of semantic marker is determined by the system, then both syntactic and semantic tagging/annotation are done on the constituents. (Fig.3)

The technical flowchart of the application whose components have been described so far is presented in the Fig. 2 for reference.

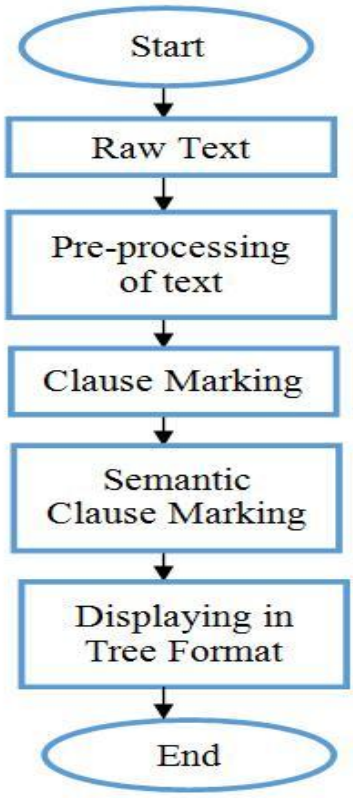

Fig. 2 Flow Chart

3) Details of the semantic annotation and tagging on clauses: As we see in the screenshot of the application below, the main sentence "*While talking to ministry, Ram, minister of external affair, one of the good person told "The blanket shielded the baby from the harsh winter wind but the baby caught the flu anyway."*" is divided into two clauses <s-> and <AUPS->. This annotations or tags are created indigenously to mark semantic process, for example, <AUPS-> means 'Acted Upon Part S-' as second part is not only dependent clause in syntactic term but in semantic view, it depicts the 'statement of the subject'. The <S-> and <AUPS-> are following 'S-' theory of syntactic parsing where clauses are marked with 'S-' in case of complex and compound sentences in dependency grammar parsing.

Moreover, the temporal property of the complex sentence "*While talking to ministry, Ram, minister of external affair, one of the good person told"* has been marked with 'temporal continuity-marker clause' which is a semantic annotation along with <S-> which is syntactic annotation.

In the right hand side, the <AUPS-> is divided into further two clauses "*The blanket shielded the baby from the harsh winter wind" as <S->* without any semantic marker, whereas second clause is marked both syntactically as <S-> and semantic annotation 'constraint-marker clause'. Thus depicting both syntactic and semantic constituents properly marked and hierarchically represented in the tree structure.

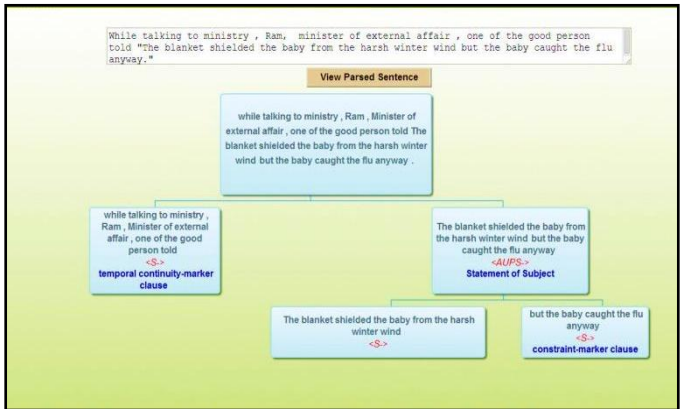

Fig. 3 Actual Screenshot of the Tree annotation & Semantic Tagging

6. CONCLUSION

This application aims at a very innovative approach of bridging gap between traditional syntax where structure is only important, not meaning. In this application semantics or meaning of clauses along with syntax has been targeted. This has far-fetched implication, the Syntactic-Semantic merger opens new arena in directly developing AI based Q-A systems, Text Summarizers, aiding to disambiguation of MT systems, Decision Support Systems and also developing E-learning for language analysis tool to name a few.

In future, Natural Language Generation (NLG) can be done directly out of this kind of syntax-semantics merger as pointed above elaborately. And also apart from English, other Indian languages can be also addressed as extension of this semantic clause parser.

REFERENCES

[1] Monika T. Makwana, Deepak C. Vegda, "Survey: Natural Language Parsing For Indian Languages": Department of Information Technology, Dharmsinh Desai University, Nadiad, India

[2] G. A. Miller, "Wordnet: A Lexical Database for English", *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[3] Björn Dagerman, "Semantic analysis of natural language and definite clause grammar using statistical parsing and thesauri", Bachelor of Science Thesis, School of Innovation, Design and Engineering, Mälardalen University, 2013

[4] Lynn M. Berk, *English Syntax: From Word to Discourse*. New York: Oxford University Press, 1999

[5] Douglas Biber, Susan Conrad and Geoffrey Leech, *Longman Student Grammar of Spoken and Written English*. Harlow, UK: Longman, 2002

[6] Ronald Carter and Michael McCarthy, *Cambridge Grammar of English*. Cambridge, UK: Cambridge University Press, 2006.

[7] H. Alshawi (Ed.), *The Core Language Engine*, Cambridge, MA: MIT Press, 1992.

[8] C. Fellbaum, *WordNet: An electronic lexical database*. Wiley Online Library, 1998.

[9] E. Atwell, *Comparative evaluation of grammatical annotation models.* In R. Sutcliffe, H. Koch & A. McElligott (Eds.), Industrial Parsing of Software Manuals. Amsterdam, The Netherlands: Rodopi.1996.

[10] E. Black., R. Garside and G. Leech (Eds.), *Statistically driven computer grammars of English: The IBM / Lancaster approach*. Amsterdam, The Netherlands: Rodopi, 1993

[11] E. Briscoe and J. Carroll, "Generalised probabilistic LR parsing for unification-based grammars", Computational Linguistics, In Proceedings of the 4th ACL/SIGPARSE International Workshop on Parsing Technologies. Prague, Czech Republic, 1995.

[12] E. Briscoe, C. Grover, B. Boguraev and J. Carroll, "A formalism and environment for the development of a large grammar of English." In Proceedings of the IJCAI-87. Milan, Italy, 1987.

[13] B. Carpenter and C. Manning, "Probabilistic parsing using left corner language models", In Proceedings of the 5th ACL/SIGPARSE International Workshop on Parsing Technologies. MIT, Cambridge, MA, 1997