A Question Generator System Using Stanford Parsing

Virendra Kumar[1], Imran Khan[2], Vikas Choudhary[3]

[1]M. Tech scholar, Bhagwant University, Ajmer
[2] Assistant Professor, AIT, Ajmer
[3] Assistant Professor, AIT, Ajmer

ABSTRACT: Today, when online tutoring is in its growing phase. There are lots of problems arising for students. Problems regarding questions and their answers, what questions they have to know in the reference of the material which has been taught to them. Not only in online tutoring, in all the fields where there is a need of some questions, are facing lots of difficulties. The solution for these types of problems is a question generation system (Qgs). The question generation system will take a sentence as an input and produce an output contains all the possible questions for that sentence. The parser which will be going to use is 'Stanford parser'. This Qgs will work on the basis of entities present in the sentence (input). All the questions (output) which will be produced by the Qgs will be grammatically correct.

The input sentence is parsed by the Stanford parser where the semantic and syntactic analysis will be done and the sentence is parsed on the basis of its grammatical structure. Then the entities present in the sentence are identified by name entity recognition technique. The output of all these phases is integratedinto transformation process and the sentence is transformed in interrogative sentence and passed to the grammar checker where the output is checked under grammatical structure and correct interrogative sentence is present as an output of the question generation system.

Keywords: Stanford parser, semantic &syntactic analysis, entity recognition technique.

I. INTRODUCTION

The Question Generation System is a system that can generate a number of questions related to a given sentence. In this system a sentence is used as an input and sort of questions are the output. This system is implemented in *Perl* language. In this system basically there are five modules that are: syntactic and semantic analysis, sentence analyzer, entity recognition, transformation process and grammar checker.

Syntactic and Semantic analysis are done by Stanford parser where the input sentence is parsed and each token is assigned a symbol on the basis of its grammatical structure. Assigning a symbol to a token comes under tagging. Both parsing of sentences and tagging of each word come under syntactic analysis. While in Semantic analysis the dependency is find out among the tokens.

Sentence analyzer is used to analyze the sentence and determine the referential relation among words. Anaphora resolution technologies are used in this module.

Entity recognition is used to identify the entities present in the sentence. Entities like person, location, place, time, duration and others.

The transformation process is the heart of this project. It is the integral process where the outputs of all the modules will integrate with some logic and algorithm to generate the question.

Grammar checker phase is used to make the output correct according to the grammar.

The objective of question generation system is to generate all the possible questions or related questions of a given sentence. The main objective of question generation system is to efficiently evaluate the possible questions through a fully automated system of classifying the sentence based on their subject, verb, object, and the preposition for determining the possible type of questions to be generated.

II. SYSTEM DESIGN

II.2 Activity Diagram
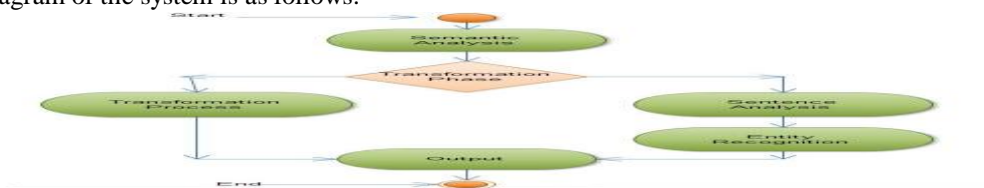
The activity diagram of the system is as follows:

Fig 1: Activity Diagram Of Question Generation System

This diagram shows the flow of activity of question generation system where the input sentence first goes to the syntactic and semantic analysis phase where parsing, tagging and dependency tagging is performed. Then the output of this phase and the input sentence will pass to sentence analyzer phase and the referential relation is determined between the tokens by using anaphora resolution techniques and the combined output will become the input the of entity recognition phase with the input sentence and all the entities present in the sentence are identified by using Stanford named entity recognizer. Then in transformation process all phases will be merged and by some logic and algorithm the final output will be generated.

II.2   Activity Diagram for Syntactic and Semantic Analysis

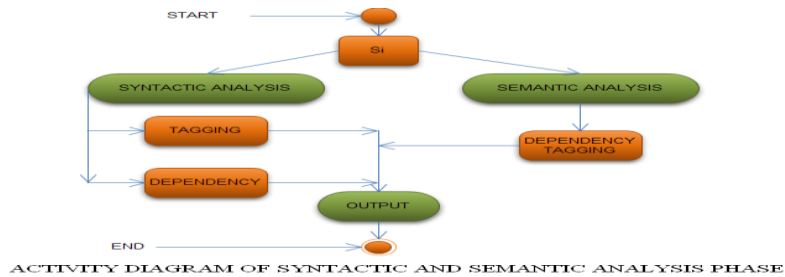The activity diagram for syntactic and semantic analysis is as follows:

Fig 2: Activity Diadram Of Syntactic And Semantic Analysis Phase

The input sentence is first come to this module where the sentence is passed to Stanford parser and the parser will perform the syntactic and semantic analysis. In syntactic analysis, tagging and parsing of sentences will perform and a parse tree is generated with the assignment of symbols to each word and in semantic analysis, dependency tagging is done to determine the relation among the tokens on the basis of their meanings.

II.3   Activity Diagram for Grammar Checker

The activity diagram for grammar checker phase is as follows:
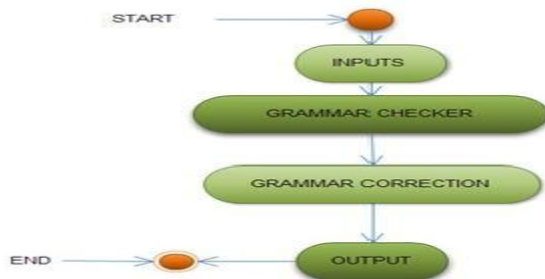

Fig. 3: Activity Diagram of Grammar Checker

The grammar checker module is used to check the grammar of transforming sentences and made the appropriate changes according to the defined grammar syntax.

II.4   Sequence Diagram

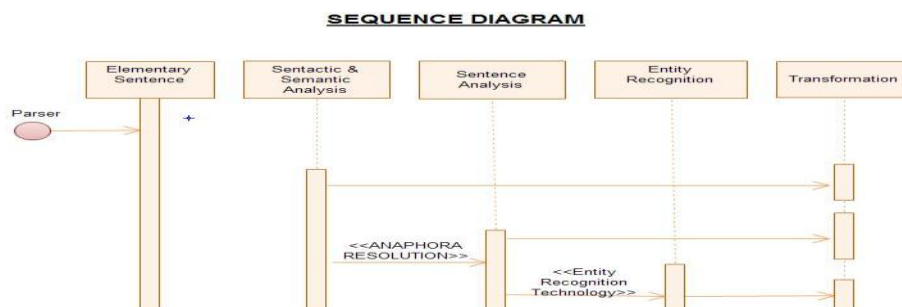The sequence diagram for the system is as follows:

Fig. 4 sequence diagram of the question generation system.

III   METHODOLOGY

In methodology, three modules are going to be discussed together; the modules are 'Entity Recognition', 'Transformation Process' and 'Grammar Checker'.

The discussion will be in a manner that each and every entity or say every Question word will be discussed individually.

III.1   Facts:

The questions words come under this entity are started with auxiliary verb present in the sentence.

There are twelve tenses in English and in that ten contain an auxiliary verb or say helping verb, only indefinite forms of present and past tense does not contain helping verb in their affirmative form.

For making the question of 'facts', there is a need to determine theauxiliary verb from the input sentence. There are few steps which are needed to generate this question, the steps are followed:

Step 1: Determine all the verbs present in the input sentence and match them with the auxiliary file, a file which contains all the auxiliary verbs. Verbs are to be determined by checking the output of the tagger. For eg: Shvet has gone to Delhi.

The output of the tagger is this: "Shvet_NNPhas_VBZgone_VBNto_TODelhi_NNP"

Verbs present in the sentence are 'has' and 'gone'.And when we match these verbs with the auxiliary file we get 'has' as an auxiliary verb.

Step 2: If there is no auxiliary verb, then check the tagging of the verb

i)   If the tagging of the verb is 'VBD' then that means 'Past Indefinite' and the auxiliary verb will be 'did'.

"Shvet_NNPwent_VBDto_TODelhi_NNP"

Here as you can see that the tagging of the verb went is 'VBD'.

ii)   If the tagging of the verb is 'VBZ' that means 'Present Indefinite' and the auxiliary verb will be 'does' because the verb is singular.

"Shvet_NNPgoes_VBZto_TODelhi_NNP"

Here as you can see that the tagging of the verb went is 'VBZ'.

iii)   If the tagging of the verb is 'VBP' that means 'Past Indefinite' and the auxiliary verb will be 'do' because the verb is plural.

"They_NNPgo_VBPto_TODelhi_NNP"

Here as you can see that the tagging of the verb went is 'VBP'.

Step3: This is the final step; in this the auxiliary verb is shifted to the start of the sentence.

For eg: Has Shvet gone to Delhi.

Does Shvet go to Delhi?

IV   RESULTS AND PROJECT CONSTRAINTS

A. Outputs:

a. Facts

Input: - Shyam and Ravi go to college together.

Output: - Do Shyam and Ravi go to college together?

Fig. 5: Facts

III.1.1   Who

Input: - Shvet Chakra is going to attend a seminar.
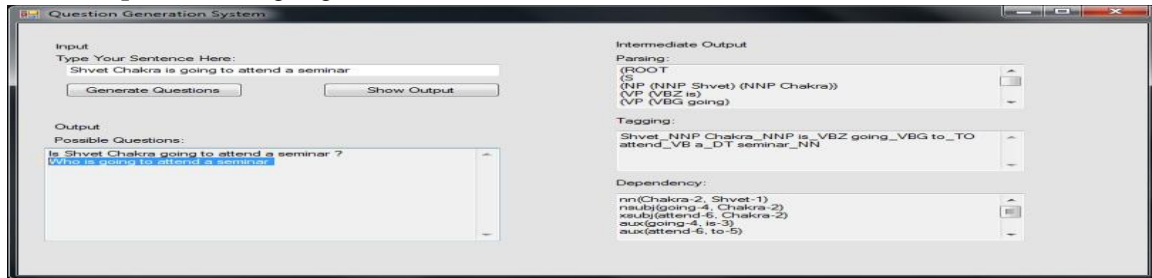**Output**: - Who is going to attend a seminar?

Fig 6: Who

V. CONCLUSION& FUTURE SCOPE

A. Summary:

After successful completion of this project we come out with the better understanding of the Perl scripting language. Along with this the concepts regarding the English language are also improved. We also get to know that how parsing, tagging, and generating dependencies are to be done. During this project we got to know about 'Natural Language Processing' and also came to understand how to use it in human life.

B. FutureWork:

Lots of work was done in this project but there are some points on which still   some work is left to make this project more efficient and practical. So, this project can be used on a large scale. The possible growth areas in this project are listed below:

i) The concept of question 'what' is remaining so that can be added in future?

ii) This system is not working efficiently for more than a single sentence, not even for those sentences which contains conjunctive words. So on that some workcan also be done.

iii) Stemming of verbs can also be performed in the future by using some scripting techniques; this time it was done by using a large set of verbs.

iv) Referential Sentence Analysis is also left for future work.

v) If an Indian 'Named Entity Recognizer' is available then the efforts will get reduced and the performance of thesystem will also be improved.

vi) In future this project can be taken at a stage where it will generate the for a questions paragraph

REFERENCES

Books:

[1] Foundations of Statistical Natural Language Processing by Christopher D. Manning and HinrichSchiitze.

[2] Natural Language Processing APaninian Perspective by AksharBharati, VineetChaitanya and Rajeev Sangal.

[3] PERL Workbook by Lisa Pearl.

[4] Learning Perl the Definitive Guide by Jason Brittain, Ian F. Darwin Randal L. Schwartz, Tom Phoenix, and brian d foy.

Journal Articles / Papers:

[5] Husan Ali, YlliasChali, Sadid A; Hasan University of LethbridgeLethbridge, AB, Canada "Automatic Question Generation from Sentences".

[6] Min-kyoung Kim, Han-joon Kim; Department of Electrical and Computer Engineering, University of Seoul, Korea "Design of Question Answering System with Automated Question Generation".

[7] David Elworthy, Microsoft research Limited, St. George House, 1 Guildhall Street, Cambridge CB2 3NH, UK, "Question Answering using a large NLP System".

[8] K. Beulen, H. Nay, Lehrstuhl fur Informatik VI, RWTH Aachen, University of Technology, D-52056 Aachen, "Automatic Question Generation for Decision Tree based State Tying".

[9] Chia-Yin Chen, Inst. Of info. Systems & Applications National TsingHua University, Hsien-Chin Liou, Dep. Of Foreign Lang. & Lit. National Tsing University, Jason S. Chang, Dep. Of Computer Science National TsingHua University, "FAST- An Automatic Generation System for Grammar Tests".

Michael Heilman, CMU-LT1-11-004, Language technologies Institute School of Computer Science Carnegie Mellon University 5000 Forbes Ave., Pittsburgh, PA 15213,"Automatic Factual Question Generation from Text".

Learning Named Entity Recognition from Wikipedia by JOEL NOTHMAN.

Language Independent Named Entity Recognition in Indian Languages by AsifEkbal, RejwanulHaque, Amitava Das, VenkateswarluPoka and Sivaji Bandy opadhyay, Department of Computer Science and Engineering, Jadavpur University.

A Survey on Named Entity Recognition in Indian Languages   with particular reference to Telugu in IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 2, March 2011, ISSN (Online): 1694-0814.

Web References:

http://nlp.stanford.edu

www.perl.com/pub/2004/08/09/commandline.html

www.tutorialized.com/tutorial/Perl-Commands-Summary/3894

www.perlhowto.com/executing_external_commands

www.englishgrammarsecrets.com/

www.englishclub.com/grammar/prepositions-list.htm

www.wikipedia.org

http://www.google.com