Automatic Question Generation Using NLP Techniques

Sandhya.P[1], Vidhya.V[2], R.K.Kapilavani[3]
[1, 2] Dept of CSE
[3]Assistant Professor, Dept of CSE
[1, 2, 3] Prince Shri Venkateshwara Padmavathy Engineering College, Chennai, India

Abstract- *Most of the teachers face problem of setting questions for various number of grades. So, it is very much difficult for a person to generate the maximum number of possible questions from the larger sets of data. Natural Language processing (NLP) acts as an important tool for processing the large amount of data. Here we use the various NLP techniques like tokenization; stemming and lemmatization are used to pre-process the data. Named Entity Recognition extracts the features of data by finding the nature of each and every sentence from a given paragraph. From the NER, the required information's are identified easily. The rule based algorithm acts as a barrier for generating the unique questions for a given paragraph. The questions that is been generated are of three levels-easy, medium and difficult. Hence Automatic Question Generator is very much efficient for the question setter, to set the different possible questions.*

Keywords- Natural Language Processing, Parts of Speech, Bag of Words, Question Generation, Information Retrieval, Question Identification, Information Reuse and Integration, Web Intelligence, Content Management System, Graphical User Interface, Named Entity Recognition.

I. INTRODUCTION

Question Paper Generator is special and unique software, which is used in school, institutions, colleges and test paper setters who want to have a huge database of questions for a frequent generation of question. This software can be implemented in various fields of studies such as medical, engineering and coaching institutes for theory paper. We can create random question paper with this software anytime and anywhere within seconds. Here, we can enter the unlimited units and chapter depending upon the system storage, capacity as per the requirement. For retrieving questions we have to first specify the subject and we can get unlimited questions in a subject.

Question generation (QG) aims to create natural questions from a given a paragraph or a comprehension. One key application of question generation can be implemented in the area of education to generate questions for reading comprehension materials.

We proposed it in real time application. We use the project in all schools, colleges and companies. It helps us to get all the questions within fraction of seconds.

Type of information

Here, we generally focus on QG about explicit factual information. We can precisely define the types of questions that we want to generate. There are many types of questions, and the researchers have proposed various ways for organizing them. One particular useful and concise discussion of the dimensions by which the questions can be classified was provided by Graesser and Person. They discussed how the questions can be organized by the following characteristics such as the purpose, the type of information they seek, the sources of information, the length of the expected answer and the cognitive processes involved by them. This work addresses a small area as follows.

Purpose

The main purpose includes the correction of knowledge deficits, the monitoring of common ground, the social coordination of action and the control of conversation and attention. This work focuses on monitoring the common ground by assessing the knowledge from a student's area of interest about a peculiar topic.

Types of question generated

There are 16 categories for questions based on the type of information's involved, ranging from simple to complex questions. These categories were derived from the previous 5 works by Lehnert, Graesser and Person. We now focus on the simple end of the spectrum, with a goal of achieving a system from the scalable to large data sets and new domains.

A system for generating more complex questions is possible, but it requires complex encoding significant human knowledge about the targeted domain and the types of questions. Specifically, this work aims to generate two types of questions and they are concept completion questions and verification questions.

A concept completion question elicits particular information that completes a given partial concept or a proposition (e.g., was Thomas a secretary of state?).

Verification questions invites a yes-no answer that verifies the given information (e.g., what is an example of an important document written by Thomas?)

Judgmental questions prefer mostly on the reasonable statements (e.g., Were Thomas's domestic policies successful?).

Goal orientation questions depends on the various types of information that has been provided by the user (e.g., Why did Thomas support the Embargo Act of 1808?),

Source of information

This work aims to generate questions for which the source of answer is literal information of text. It is mainly focused on the sentence or a paragraph level rather than spread across the whole document. The questions generated will not address the world knowledge, common sense, opinions of the answering. It purely depends on the paragraph that we give.

Length of expected answer

This work mainly generates the questions whose expected answers are short. It is usually a single word or a short phrase. The expected answers are not summaries and descriptions.

Cognitive process

The questions generated by this work mainly assess the recognition and recalls the information. It involves a little inference, application, synthesis and other complex processing methods. Thus, in terms of Bloom's taxonomy of education objectives, we mainly focus on "knowledge" level. Higher levels of that taxonomy includes "comprehension" (e.g., restating something by one's own words), "application" (e.g., using an equation to solve a practical problems), "analysis" (e.g., recognizing the errors), "synthesis" (e.g., writing an essay or paragraph), and "evaluation" (e.g., selecting the best design for a developing task). The challenging task is that there should be a connection to existing work in parsing, entity recognition, co reference and other research areas in computational linguistics.

II. RELATED WORK

We incorporate the applications like production rules, LSTM neural network models, and other intelligent techniques. We generate multiple choice questions, true and false questions as well as "Wh"-type (What? When? How?) Questions. Questions have been poorly reviewed due to the clumsy handling of sentence complexities in previous models. In here, we focus on developing a question generation technique without any complexities. For the training and testing dataset, approximately one hundred articles from the website, (http://www.yfes.tn.edu.tw/yfesvi/story.htm) have been used. We come up with a novel task for Question Generation systems like thematic question generation. Inspired by a famous trivia board game, we aim at solving the problem of generating meaningful questions and their distracters for common knowledge topics. In here, we develop an end-to-end system that tackles these issues. The system is trained on a Wikipedia-based dataset consisting of URLs of Wikipedia articles and the important words (keywords) which consist of both bigrams and unigrams are extracted and stored in a dictionary along with many other components of the knowledge base. We have also used Inverse Document Frequency (IDF) measure for ranking the extracted keywords and Context-Based Similarity approach using Paradigmatic Relation discovery techniques for the generation of distracters. The question generation process is done by the rule that has been generated from the mind map. The mind map will help us to determine the interrelationship between the learning materials.

III. PROPOSED WORK

Our proposed work contains annotated data sets for natural language processing (NLP) research in reading comprehension and answering the questions.

The solutions for Natural Language Processing (NLP) can be accomplished by computerized systems in an effective manner. In here we use the various NLP techniques like tokenization; stemming and lemmatization are used to preprocess the data. Named Entity Recognition extracts the features of data by finding the nature of each and every sentence in a given paragraph. From the NER, the required informations are identified easily. The rule based algorithm acts as a barrier for generating the unique questions for a given paragraph. It extracts the knowledge in the form of rules from the classification model and this algorithm is most suitable for analyzing the data containing a mixture of numerical and

qualitative attributes and focuses on sentence's semantic and syntactic structure. The wide portion coverage of comprehension and efficient question paper generation are covered in here. Our system provides an unbiased result for each and every sentence.

IV. SYSTEM ARCHITECTURE

The system architecture has input dataset from which the first proposed module preprocessing is carried out where the procedures like tokenization, parts of speech, n-gram and stemming are conducted and extracted. The feature extraction module makes use of NER which categorizes the words into noun, pronoun, and verb and then the rule based algorithm is applied where the user will get to choose the type of questions to be generated. After choosing the types the system is trained and tested and finally the desired question is generated.

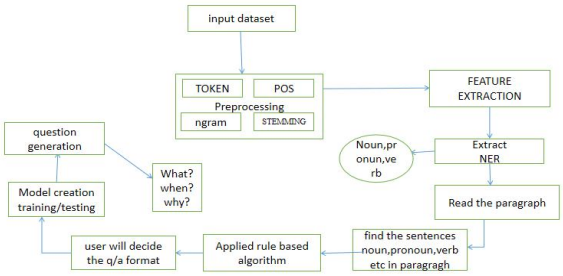

Fig: 1

V. MODULES

In this pre-processing, we first load the metadata into this and then this metadata will be attached to the data and replace the converted data with metadata. Then this data will be moved further and removes the unwanted data in the list and it will divide the data into the train and the test data.

NLP techniques used in here are as follows: tokenization, Pos tagging, N-grams, Stemming and Lemmatization. Tokenization is the first step in text analytics.

It is the process of splitting a phrase, sentence, paragraph, or an entire text document into smaller units. Each of these smaller units is known as tokens. The tokens could be words, numbers, or punctuations. In Token is a single entity that acts building blocks for framing a sentence or paragraph.

After Tokenization, it enters into the second phase and that is Part of Speech Tagging (POS tagging).It means identifying each token's part of speech (noun, adverb, adjective, etc.) and then tagging them into a particular syntax. POS tagging acts as a basis for Natural Language Processing. In order to recognize entities, extract themes, Parts of Speech has to be properly identified.

Whenever the POS is identified, N-grams are used to find the combinations of adjacent words or letters of length N. N-grams with n=1 are known as unigrams. Similarly, bigrams (n=2), trigrams (n=3) and so on. When compared to bigrams and trigrams, unigrams does not contain much information. The basic principle used in N-grams is that they capture the letter or word in a given paragraph.

The next phase includes Stemming where it is a process of linguistic normalization where it reduces words to their root word or chops off the derivational affixes. For example, action, acted, acting word reduce to a common word "act".

In order to provide appropriate root word, Lemmatization is used. Lemmatization is also similar to the stemming but it transforms root word with the use of vocabulary and morphological analysis. Lemmatization is usually more sophisticated than stemming since it provides the accurate results. For example, Stemming may not able to produce the appropriate result for the word "better" but lemmatization provides "good" as its lemma. This strategy will be missed by stemming because it requires a dictionary look-up.

After predicting the root words it is extremely important to extract the required data from the Bag-of-words. Bag-of-words model (BOW) is the simplest way of extracting features from the text or data. BOW converts text into the matrix of word occurrence within a given document. This model concerns about the number of occurrences held in that document.

The next phase is the Named Entity Recognition where the named entities are people, places, and things (products) mentioned in a text document.

Here, the entities can also be of hash tags, emails, mailing addresses, phone numbers and Twitter handles.

At Lexalytics, we've trained supervised machine learning models with the billions of pre-tagged entities and this approach helps us to optimize the accuracy and flexibility.

We've also trained NLP algorithms to recognize the non-standard entities (like species of tree, or types of cancer) in the document. It's also important to note that the Named Entity Recognition models rely on accurate POS tagging models.

Finally the rule based algorithm is used for generating the questions. The rule based algorithm acts as a barrier for generating the unique questions for a given paragraph. It extracts the knowledge in the form of rules from the classification model. The rule based algorithm is most suitable for analyzing data that contains a mixture of numerical and qualitative attributes and focuses on both the syntactic and semantic structure of a sentence.

The questions that has been generated are as follows:

Who? (Asking what or which person or people (subject) e.g., who opened the door?)
Whom? (Asking what or which person or people (object) e.g., whom did you see?)
Whose? (Asking about ownership e.g., whose are these keys?)
Why? (Asking for reason, asking for what e.g., why do you say that?)

Therefore the rule-based machine learning methods typically comprises of a set of rules, or a knowledge base that collectively makes up the prediction model.

VI. CONCLUSION

The proposed work describes An automated system that progresses from the traditional method of paper generation to an automated process, by providing controlled access to the resources.Our system was deployed by an efficient algorithm which is totally randomized and avoids the repetition of questions in a consequent question papers, making it impossible to derive any pattern from the papers. The Question Paper Generation provides an improvement in terms of controlled access to the resources and random generation of the question papers.

REFERENCES

[1] Computational Intelligence Framework for Automatic Quiz Question Generation, Akhil Killawala ; Igor Khokhlov ; Leon Reznik, 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)
[2] Automatic Question Generation from Children's Stories for Companion Chatbot, Che-Hao Lee ; Tzu-Yu Chen ; Liang-Pu Chen ; Ping-Che Yang ; Richard Tzong-Han Tsai, 2018 IEEE International Conference on Information Reuse and Integration (IRI)
[3] Thematic Question Generation over Knowledge Bases, Tanguy Raynaud ; Julien Subercaze ; Frédérique Laforest, 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)
[4] Automatic Question Generation with Classification Based on Mind Map, Selvia Ferdiana Kusuma ; Daniel Oranova Siahaan ; Chastine Fatichah ; Mohammad Farid Naufal, 2019 16th International Joint Conference on Computer Science and Software Engineering (JCSSE)
[5] Dual Learning for Visual Question Generation, Xing Xu ; Jingkuan Song ; Huimin Lu ; Li He ; Yang Yang ; Fumin Shen, 2018 IEEE International Conference on Multimedia and Expo (ICME)
[6] Knowledge-based Questions Generation with Seq2Seq Learning, Xiangru Tang ; Hanning Gao ; Junjie Gao, 2018 IEEE International Conference on Progress in Informatics and Computing (PIC)
[7] Automatic question generation for intelligent tutoring systems, Riken Shah ; Deesha Shah ; Lakshmi Kurup, 2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA)
[8] Bilingual Ontology-Based Automatic Question Generation, Baboucar Diatta ; Adrien Basse ; Samuel Ouya, 2019 IEEE Global Engineering Education Conference (EDUCON)
[9] Outcome based predictive analysis of automatic question paper using data mining, Simranjeet Kour Bindra ; Akshay Girdhar ; Inderjeet Singh Bamrah2nd International Conference on Communication and Electronics Systems (ICCES)
[10] Knowledge Acquisition for Visual Question Answering via Iterative Querying, Yuke Zhu ; Joseph J. Lim ; Li Fei-Fei, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).