

Objective Type Question Generation using Natural Language Processing

G. Deena¹, Dr. K. Raja²

Research Scholar, Department of Computer Science and Engineering
Sathyabama Institute of Science and Technology, Rajiv Gandhi Salai, Chennai, 600119, India¹
Department of Computer Science and Engineering, SRM Institute of Science and Technology
Bharathi Salai, Chennai, 600089, India^{1,2}

Abstract—Automatic Question Generation (AQG) is a research trend that enables teachers to create assessments with greater efficiency in right set of questions from the study material. Today's educational institutions require a powerful tool to correctly assess learner's mastery of concepts learned through study materials. Objective type questions are an excellent method of assessing a learner's topic understanding in learning process, based on Information and Communication Technology (ICT) and Intelligent Tutoring Systems (ITS). Creating a set of questions for assessment can take a significant amount of time for teachers, and obtaining questions from external sources such as assessment books or question banks may not be relevant to the content covered by students during their studies. This proposed system involves to generate the familiar objective type questions like True or False, 'Wh', Fill up with double blank space, match the following type question have generated using Natural Language Processing(NLP) techniques from the given study material. Different rules are created to generate T/F and 'Wh' type questions. Dependence parser method has involved in 'Wh' questions. Proposed system is tested with Grade V Computer Science text book as an input. Experimental result shows that the proposed system is quite promising to generate the amount of objective type assessment questions.

Keywords—Intelligent tutoring system; true or false; dependency parser; natural language processing; question generation

I. INTRODUCTION

In academics, an assessment is critical to determine the skill level of a learner. Question generation has emerged as a new area of study in the fields of traditional and in online education and natural language processing [1, 2, 3][4,5]. Research has been done to improve the effectiveness of automatic question generation in recent years. Acquiring the knowledge is the ultimate goal of an educational system, but assessment or evaluation is the final goal in learning [6,7,8]. Evaluation process needs questions to measure the learner's topic knowledge [9, 10,11].

Automatic Question Generation (AQG) is the process of automatically creating syntactically valid, semantically correct, and relevant questions from a variety of inputs in the form of text, a structured database, or even a knowledge base as given in [12,13]. AQG is a perfect fit for a variety of areas, including Massive Open Online Courses (MOOC), Chatbot systems for customer engagement and interaction and healthcare system are a few examples of automated help

systems as given in [14]. Computerized practices can be produced better than standard specialists, while still maintaining quality.

Natural Language Processing (NLP) is a subfield of Artificial Intelligence (AI) that studies how machines and humans communicate using natural language. The goal of NLP is to identify, comprehend, grasp, and interpret the recorded or text data in human languages, whether in English or in another language, in a manner that is beneficial. Subjective and Objective type questions are two primary methods in traditional and digital educational system [15]. Due to the benefit of rapid and real-time examination, objective type test items are gaining great significance in an Intelligent Tutoring System (ITS) and in dynamic classroom learning. Since there are many uses in conversational AI systems and education, interest in the topic has grown.

Objective types questions are termed as Cloze Type (CT) Question which can be answered in short or in one word. They are very useful for learning specific ideas, and they are frequently employed in school level. However, these questions may be manually prepared, and this necessitates significant understanding on the topics being prepared. The individual or the teacher must examine each sentence of the section and determine whether or not each statement is capable of generating a question. A valid question can be derived from a given sentence only if it includes information that contributes to the learner's knowledge gain. Input is a text file and an output is a generated objective type True or False and 'Wh' type questions using NLP techniques. It is necessary to select the proper response from a list of possibilities, or finish a statement with the use of an appropriate sentence fragment or word. Commonly utilized objective exam items include True or False, Multiple-Choice Questions, and Fill-in-the-blank questions [16].

In "Wh" is investigating where someone will be responsible for generating item pools with items like "What," "When," "Where," "Who," and "Which." Researchers have been focused on automatic CQ creation for the past decade [17,18]. AQG can support the teacher and the learner entity. Additional practise can be given as self-assessment, is provided by AQG to learner. Alternatively, AQG can also aid teachers in the laborious process of deciding on the type of questions they will pose to the learner. Also, AQG can assist online learners in creating questions from random text. The

concept of effective questioning is a strategy to help students pay attention to their learning goals. As well, to ensure its effectiveness in integrating Bloom's lower level and high-level taxonomy the questions are required as mentioned in [19].

The proposed system focuses on developing different types of objective questions like Fill-Up with double blank questions, True or False questions, Match the following and short answer questions to assess a learner's learning gap. The majority of phrases in a text do not lend itself to the generation of high-quality queries. As a result, the informative sentence selection assignment has done to generate questions from those sentences. As the continuation of our previous work, the informative sentences are selected using Latent Semantic Analysis given in [20]. Answer key is the top most word which yield the relation between the sentences and those answer keys are replaced by the blank space.

In True or False type questions, the answer key has identified from the Part-of-speech tag. Tag can be the auxiliary verbs, cardinal number, adjective, negative words of the sentence Based on these tag value and predicting the next word involved in falsified sentence is generated as true or false questions using Natural Language Processing. The sentences will be falsified by changing the tag word.

Fill-up with double blank question, the informative sentence and important word of the sentence is identified using Latent semantic analysis. The important word will be the answer key of the sentence.

In terms of short answer questions, the dependency parser and Named Entity Recognizer (NER) are involved to frame the generate questions. NER will give the Person Name, Place, Organization, and Location from the sentence. The dependency parser gives the "SUBJ, NMOD, DOBJ, DET, CASE, NSUBJ, CC" of the sentence. Both the NER and dependency parser combined together to generate the short answer questions.

In match the following type questions, the abbreviated words are selected from the corpus and consider as input in the first column and the randomly the expansions are given at the right side in the second column as an output.

Evaluation of the proposed system is done through grade V Computer science book of Central Board of Secondary Education. The paragraph from the book is given as input to the system. The pre-processing has been applied to clean the input. Later, the system generates the different type of question and the quality of the system is manually verified by the school teacher. The cardinality and validity are the two parameters to measure the performance of the proposed system. The generated questions were tested with the students of grade V to ensure the understandability of the questions. In general, the experiment observed that 92% of validity has been achieved.

II. RELATED WORK

In recent years, development in an automatic question generation system seems to have become a hot research topic. There are a number of different systems and methodologies and some of which use distinct methods and practices to

generate questions automatically. The tasks focus on specific methods that use sentences or phrases out of unstructured material to generate questions for students, which are used to assist in their knowledge assessment.

In numerable question generation systems have been suggested previously like Fill up the blanks, 'Wh' questions to assess the skill of a learner. The whole work has been divided into different phases like sentence identification, extracting the important keyword and question formation. In the existing methods, different methods were used to extract keywords the appropriate keyword in the sentence is replaced by the blank space to generate the Fill-up-based questions. In the proposed system, the informative sentence and keywords are extracted using Latent Semantic Analysis. and more than one keyword in the sentence is replaced by the blank space to generate the fill up type questions.

Sheetal Rakangore et al, [21] suggested the true or false question by adding the "not", "no" and "never" in the affirmative sentences. The affirmative sentence is generally positive statement which gives sufficient information to the reader. Those sentences will not generally carry any negative words in the sentence. According to the author, the part-of-speech for a sentence is identified and add 'no', 'never', 'not' before the auxiliary verbs. This system converts the affirmative sentence into negative sentences. In the proposed system, numbers of rules are created to generate the questions using NLP techniques.

In the short objective type questions, many researchers have focused to develop factual question based on AUTOQUEST as suggested in [22] that was the first method to generate automatic factual question. Syntactic principles were first used to generate questions, and then some type of statistical assessment of goodness was done in [23].

The other works discuss in this chapter are concerned with finding questions in unstructured text sentences or sequences of sentences. Here, let's describe a few approaches to generate the objective type questions generation that are available in the literature. In Section 3, the proposed system true or false type questions were described, Fill up the double blank type question has described in Section 4, Match the following carried in Section 5, short question answer questions mentioned in Section 6, the experimental result shown in Section 7 and conclusion mentioned in Section 8.

III. PROPOSED SYSTEM

A. True or False

Different type of objective type questions is in practice to assess the learner. True or False, Match the following, fill up the blanks with double blank, Wh-type or One word Question Answer are the variety of question types. True or False (T/F) is the quickest process of measuring the learner. T/F can be resolved by returning a binary value of "True" or "False." Once the learner grasps the subject clearly, the T/F questions are answered as part of the lowest level of the bloom taxonomy. Proposed system works on the basis of sentence tokenization, word tokenization and Part-of-Speech (POS) identification. The affirmative sentence has been involved in question generation.

B. Sentence Tokenization

The given corpus consists of sequence of sentences which can be tokenized into individual sentence. Each sentence will be analysed separately to generate the objective questions. In the current system, to generate the True or False(T/F), the sentence tokenizer is used to extract individual sentences using Natural Language Processing(NLP) techniques. Input paragraph is tokenized into individual sentence. Individual sentences are analysed to find any affirmative sentences, holding any cardinal value, any negative sentences.

C. Keyword Identification

Once the passage contains statements about factual information then question can be raised in the form of True of False. In the proposed system, the given corpus is tokenized into sentences using the sentence tokenizer followed by the word tokenizer to tokenize into individual words. The part-of-speech (POS) tag set is applied that has the ability to label the words as noun, adjective, adverb, and verb. Keywords are identified by the POS are also known as tokens.

D. Rules for True or False Questions

In the proposed system, the affirmative sentences are changed into negative sentence, identify the antonyms to adjective words, changing the cardinal value using random function, using GPT-2, replace the last word with options and in terms of the negative sentence remove the negative words to generate True or False questions for the given corpus. A rule has been created for mentioned types of ways to generate the T/F Questions.

A positive or an affirmative form is employed to convey the validity or reality of a fundamental proposition of the sentence. There will be no negative phrases like “not”, “never”, “none” in a positive sentence which represents the truth of the sentence. This sentence is factually correct and gives some information to the learner.

A negative sentence is a statement that something is false or incorrect. To cancel the sentence's credibility, a negative adverb must be added. Affirmative phrases show their falsity through the negative statement. Don't negate a negative sentence rather make it to positive sentence.

Rule 1: Negate the affirmative statement

Affirmative sentences are the sentences that have positive meaning. An auxiliary verb phrases assists another verb in its tense or in voice formation. Auxiliary verbs are also referred as assisting verbs. These are called as helping verbs to the words in the sentence. Auxiliary verbs are like “am, are, is,

was, were, can, could, may, have”. Perform the steps to generate the falsified sentence.

- 1) Check if the sentence in the corpus has any auxiliary verb, then add “Not” before the auxiliary verb to negate the sentence.
- 2) The affirmative sentence has become falsified sentence as True or False questions.
- 3) Print the question.

Let us consider the samples input sentence from the Grade V Computer Science Book as in Table I.

Rule 2: Adjective word replacement

Another possibility is to change the meaning of a sentence is by setting the antonyms to the label adjectives. The antonyms of the adjectives set from WordNet. WordNet is a lexical database built for natural language processing in English, specifically. English Lexical Database (ELD) is a dictionary which consists of different kind of similar meaning and opposite meanings to a single word. Synsets instances are groups of words that can be used to define the same concept. The Nouns, Verbs, Adjectives, and Adverbs are classified according to their cognitive and linguistic relationships of a word. The given steps follow to find the antonyms of the word.

- 1) Apply POS to extract the adjective tag of a word from the sentence.
- 2) Apply the lemmatization to determine its word's root.
- 3) Apply the antonym function to find the opposite meaning of the root word to alter the meaning of a statement.
- 4) Generate the falsified questions.

The adjective of the word is replaced by the antonyms of that word looked up from the Wordnet as given in the Table II.

Rule 3: To replace the cardinal number

This rule is simple to alter the original meaning of a statement by altering its cardinal value. Find whether any the number is present in the sentence using the POS. Falsify the given statement by using the random function to change the cardinal value. The sample was given in the Table III.

- 1) Check if the tag of word is ‘cardinal’ then apply the random function to change the value of the cardinal.
- 2) The new cardinal value makes the sentence into falsified.

TABLE I. NEGATED AFFIRMATIVE SENTENCES

Auxiliary Verb	Input sentence	Falsified Sentence
was	ENIAC was the first fully electronic digital computer	ENIAC was not the first fully electronic digital computer
was	The ENIAC was developed by John Presper Eckert	The ENIAC was developed by John Presper Eckert
were	First Generation computers were hugely expensive to build	First Generation computers were not hugely expensive to build
was	EDSAC was the first electronic computer that used stored programs.	EDSAC was not the first electronic computer that used stored programs.
was	EDSAC was the first electronic computer that used stored programs.	EDSAC was the first electronic computer that used not stored programs.
have	Second Generation computer have used transistors	Second Generation computer have not used transistors

TABLE II. ADJECTIVE WORD REPLACEMENT

Adjectives	Input sentence	Falsified Sentence
large	First Generation very large in size	First Generation very small in size
expensive	First Generation very expensive	First Generation very cheap
slow	First Generation operating speed was quite slow	First Generation operating speed was quite fast
slow	First Generation operating speed was quite slow	First Generation operating speed was quite accelerate.
cheaper	Third generation were cheaper than second generation	Third generation were expensive than second generation

TABLE III. ALTERNATIVE OF CARDINAL NUMBER

Cardinal	Input sentence	Falsified Sentence
360	IBM 360 examples of third generation	IBM 362 examples of third generation
1948	EDVAC was completed in the year 1948	EDVAC was completed in the year 1957
6000	EDSAC weighted approximately 6000 kilograms.	EDSAC weighted approximately 6006 kilograms.
1946	ENIAC was developed by John Mauchly in the year 1946.	ENIAC was developed by John Mauchly in the year 1955.
1401	IBM 1401 are example of second generation	IBM 1403 are example of second generation

Rule 4: To predict the last word

To falsify the statement, another possibility is predicting the last word of the sentence. The last noun labelled word is extracted separately from the sentence. The extracted word is now then termed as keyword. Predicting the subsequent word is one the task in NLP activities. GPT-2 is an open-source Artificial Intelligence with 40GB of text is analysed to train the prediction algorithm. Generative Pre-trained Transformer-2 (GPT-2) is the leveraged transformer in both unsupervised and supervised text representation. To falsify the sentence, perform the steps to predict the alternatives of the keyword. The implemented samples rules were given in Table IV.

- 1) Split the actual sentence into two halves by using `rsplit()`.
- 2) `Rsplitt()` function to split the last word from the sentence.
- 3) Import GPT-2 from `next_word_prediction`.
- 4) Invoke `predict_next()` with an rest of the sentence as argument.

a) It produces the possibilities of the left-out word

b) Select any of those word to generate the falsify statement and print.

Rule 5: Converting Negative Sentence into Positive Sentence

A negative sentence is one that declares something to be false and it indicates that a specific statement or circumstance is incorrect. List of sample negative words are given in Table V. To generate the true or false question lets convert the negative sentence to positive sentence as affirmative by following the steps,

- 1) Check if any of the negativity is present in the sentence, and then eradicate them to convert the sentence as affirmative.
- 2) Affirmative sentence is generated.

The sample of the removing the negative words were given in the Table VI.

TABLE IV. PREDICTING THE LAST WORD BY GPT-2

Last word	Input sentence	Falsified Sentence
time	Microcomputer is a small computer that is used by one person at a time	Microcomputer is a small computer that is used by one person at a desk.
PC	Desktop computers are also called as PC	Desktop computers are also called as windows
2000	ENIAC was developed by John Mauchly in the year 2000	ENIAC was developed by John Mauchly in the year 2000
Slow	First Generation operating speed was quite slow	First Generation operating speed was quite good
expensive	First Generation very expensive	First Generation very early

TABLE V. NEGATIVE WORDS

no	not	none	Never	nothing
don't	can't	won't	wasn't	isn't

TABLE VI. REMOVAL OF NEGATIVE WORDS

Negative words	Input sentence	Falsified Sentence
not	Mano is not a professional singer.	Mano is a professional singer.
never	The coffee shop is never open	The coffee shop is never open
won't	They won't be eating	They won't be eating
don't	Mithul don't play keyboard	Mithul play keyboard
aren't	They aren't coming to school in pandemic situations	They are coming to school in pandemic situations

E. Fill up the Double Blank based Question

In the work given by author Smith [24], Karamaniset al [25] and Mitkov et al [26] pick appropriate keywords by determining frequency of terms and running regular expressions on nouns. Find the sentences with that keyword in them and their methods produce one-word gap-fill questions. In the continuation of our previous work [27], the important sentence and the keyword was identified through Latent Sematic Analysis (LSA), this method produce the hidden relation between the sentences and the words. To select the keywords, the proposed system fixes the threshold value between 1.5 and 2.0 and those above the value were identified as most important keyword involved in generating the Fill up based assessment questions.

Sentences with two keywords involved in generating the double blank based questions, where the keywords are replaced by the blank space. In the existing system, single keyword is replaced by the blank space. Keyword may be Noun related words extracted using TF-IDF or by keyword extractor. In the proposed system, the LSA used to select the keyword it can be Noun, Verb which has hidden relation between the words and sentences. This double blank fill-based question gives promising result to determine the skill level of a learner. The samples were given in Table VII.

F. Match the Following Questions

In the proposed system, to generate the Match the Following type questions, abbreviation and their expansion, definition type words are taken as input from the given input file and maintained in the .csv file, where the abbreviation words are selected as the keyword. Each abbreviation has the expansion. The table is constructed with the row and column. The keywords are the abbreviated words and define type words, the keywords are the first column and the respective expansion of the keywords and the explanation of the define words are shuffled using shuffle method and prearranged at the second column. Now the keywords have wrong abbreviation in the nearby column. This set is finally made as

Menu	Definition
ENAIAC	Universal Automatic Computer
EDSAC	Electronic Delay Storage Automatic Computer
EDVAC	Artificial Intelligence
UNIAC	Electronic Discrete Variable Automatic Computer
AI	Electronic Numeriacal Integrator and computer

Fig. 1. Match the Following Type Questions.

match the following questions. As a result, the match the following type question are generated and maintained in the table. As part of the objective type question, the learner will take up the assessment to test the skill level. The sample of these questions are executed in jupyter notebook and given in Fig. 1.

G. One Word Question and Answer

“Wh” questions are the type of questions to answer about specific qualities, times, places and people. Like ‘Wh’-type question, we generate one word question and answer to determine the learning skill level of a learner. Dependency Parsing is the process of analysing a sentence's grammatical structure and identifying related words, as well as the nature of the link between them using computer programmes. It enables us to construct a parsing tree based on the tags used to determine the relationship between the words in the sentence, rather than on any Grammar rule as with syntactic parsing, which provides a great deal of flexibility even when the order of the words changes.

In the existing system, the Named Entity Recognizer (NER) has used to generate the Who, What, When, Where type questions. Always it's not recommended to generate “Wh” questions. In the existing methods, only one word has been extracted using POS, NER, and dependency parser to generate questions. In the proposed system, the spacy Matcher, Dependency Parser and NER combined together to generate the assessment questions. In some case, two words.

1) *Matcher*: In spacy, rule-based matching enables you to define your own criteria for locating and extracting words and phrases from a text. To match the individual word or text the matcher is the best option and is easy to create. Here, let set the Matcher's vocabulary to the language of the spacy model in Natural Language Processing (NLP) and maintain a list of dictionaries for the pattern of the type which really required for the sentence. Each dictionary contains the match pattern for a token.

TABLE VII. FILL UP WITH DOUBLE BLANK –QUESTIONS

S. No	Fill up the double blank questions	Keyword
1	Expansion of _____ was termed as Electronic _____ Integrator and Computer	ENIAC Numerical
2	Expansion of ENIAC was termed as _____ Numerical _____ and Computer	Electronic Integrator
3	Expansion of ENIAC was termed as Electronic _____ Integrator and _____	Numerical Computer
4	Expansion of _____ was termed as Electronics Numerical _____ and Computer	ENIAC Integrator
5	It was the first _____ electronic _____ computer	Fully Digital
6	It was the first fully _____ digital _____	Electronic Computer
7	It was the first fully electronic _____	Digital Computer
8	_____ was an example of first-generation _____	ENIAC Computer
9	The ___ could add two large numbers just in 200 _____, much higher speed than its predecessors	ENIAC Microseconds
10	The ENIAC could add two _____ number just in 200 microseconds, much higher speed than its predecessors	Large Microseconds
11	The _____ could add two large _____ just in 200 microseconds, much higher speed than its predecessors	ENIAC Numbers
12	The ENIAC could add two large _____ just in 200 _____ much higher speed than its predecessors	Numbbers Microseconds
13	The _____ was developed by John Presper Eckert and John W. _____ in the year 1946	ENIAC Mauchly

User defined patterns are generated to extract the two consecutive “noun” tag from the sentence to frame the questions. As well generate the pattern like, “characteristics of”, “features of”, “benefits of”, “uses of”, “purpose of” to generate the subjective type questions based on blooms taxonomy. Then search whether the pattern appears in the document and once it matches the respective questions will be generated.

2) *Dependency parser*: Dependency parser is the term used to determine the dependencies between the terms of a phrase in order to assess its grammar and syntax [28]. As a result, the sentence is broken into constituent parts. The process is predicated on the premise that each linguistic unit in a phrase is inextricably linked to the next. These connections are referred to as dependencies. The relationships between each linguistic unit, or word, in a phrase are represented by directed arcs in a structured dependency. A dependence tag indicates the relation between different words.

In the proposed system, dependency parsing allows us to construct a parsing tree using tags to determine the relationship between words in a sentence rather than using any grammar rule as in syntactic parsing, which provides a lot of flexibility even when the order of words changes. These tags are used to identify the subject, verb, adverb, object in the

sentence. The various rules have been generated to establish automatic objective type ‘wh’ questions.

The simple sentence has been parsed using dependency parser identification. The relation between the word exists in the sentence are identified through the tags like nsubj, vbd, jj, obj and amod as given in the dependency parser Fig. 2. Rules were created to generate short answer assessment questions.

By applying NER to the input sentence, “Rome won famous medal” where “Rome” is identified as “Location”. and the possibly generated questions will be “When did Rome won?”, but the generated question is not a semantically exact. In existing system, many researchers have developed the assessment questions based on NER for the short answers, but always cannot expected of holding location, person, cardinal number in the corpus. Hence in the proposed system, NER, Dependency parser and spacy matcher put together to produce the assessment questions. The proposed system produces possible number of questions by following the generated rules.

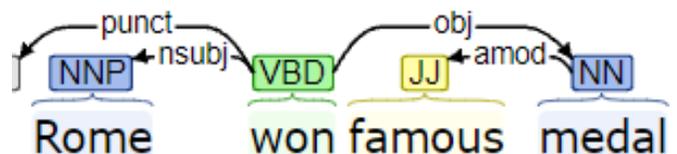


Fig. 2. Dependency Parser of Sample Sentence.

3) *Rules based on Dependency Parser*: The tag like nsubj, vbd, jj, obj, amod and det are determined to frame the rules as mentioned below.

Rule 1:

On the basis of the subject, the question has been generated as given. Use the Part-of-Speech and dependency parser to identify the tag of each word in the sentence. When the subject tag is the "person" then the keyword "Who" is added with the identified root tag and direct object(dobj) of the sentence, ended with question mark.

"Who" +root+ dobj

"Who "+root+ prep+ det+ pobj

Input sentence: Rome won famous medal

Possibly generated questions are,

Output Question: Who won medal?

Rule 2:

The question has been generated based on the Noun (NN) tag. The Noun and object tag of the sentence is determined by part-of-speech and dependency parser. Hence the keyword "What did" or "What" has been added with the subject and root tag of the given sentence.

"what did "+sub+root.

"what" + sub+ root.

As per the given input mentioned in Rule 1, possibly generated questions were,

Output: what did Rome won?

Rule 3:

Open clausal complement (xcomp), a verb or an adjective is a predicative or clausal complement that does not have its own subject. It also involved in question generation using the keyword "who" added with the auxiliary, root and xcomp tag of the components from the sentence.

"Who "+aux+root+"ing to "+xcomp.

The input sentence: "The Sheriff try to eat the apples while the outlaws were fasting", as per the rule 3, the generated question is

Output: Who were trying to eat?

Rule 4:

When any place or location has mentioned in the sentence then the keyword "Where did" or "Where" has added with the subject and root of the sentence to generate the questions.

"Where did "+sub+root.

"Where" + sub+ root.

Rule 5:

Check whether the word is the abbreviated. All the abbreviated words are in uppercase, if so then generate question to give the expansion of the abbreviations.

"Write the expansion of" + abbreviated word

Write the expansion of ENIAC?

Rule 6:

Apply the match pattern to extract the continuous two noun words, so as to generate the semantically correct assessment question. Usually, the single word of the sentence will be extracted either using NER/POS/Dependency Parser. Sometime the two sequential word gives the completeness. Example "data structure", using NER/POS/Dependency parser the term "data" and "structure" tokenized separately. Hence the matcher is defined with NLP,

pattern = [{"POS": "NOUN"}, {"POS": "NOUN"}], so that two continuous "nouns" have been extracted to generate the questions.

Rule 7:

Create a template like "uses of", "examples of", "features of", "characteristics of", "types of", "methods of". Check if the input corpus consists of any of this template, when it holds then use the blooms verbal list to frame subjective type assessment questions. The sample input corpus is given below,

Input: "Array is a kind of data structure store a fixed-size sequential collection of elements of the same type. Array has the characteristics of storing the contiguous element".

The POS of the given is input determined, Array is termed as "NN" as given in Fig. 3, the predefined word combined with NN to frame assessment questions.

The dependency parsers is identified for the same input corpus as given in Fig. 4, and extract the subject of the sentence. Not all the noun(NN) will be fine to involve for questions, so the noun has to be the subject of the sentence and the semantically meaningful questions can be generated as per NER and dependency parser.

Array is the determined as Noun (NN) tag and it is the subject (nsubj) of the input sentence.

Affording to the input corpus, "characteristics of" is present which can be combined with the noun tag to frame assessment questions.

Part-of-Speech:

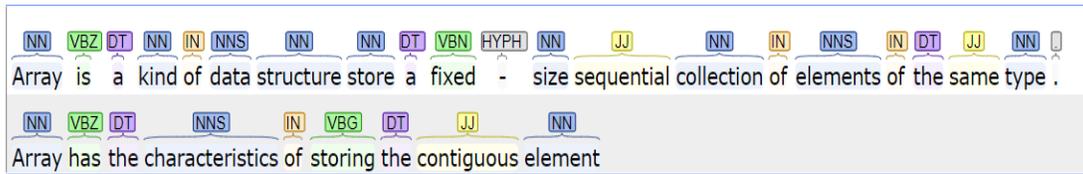


Fig. 3. Part-of-Speech.

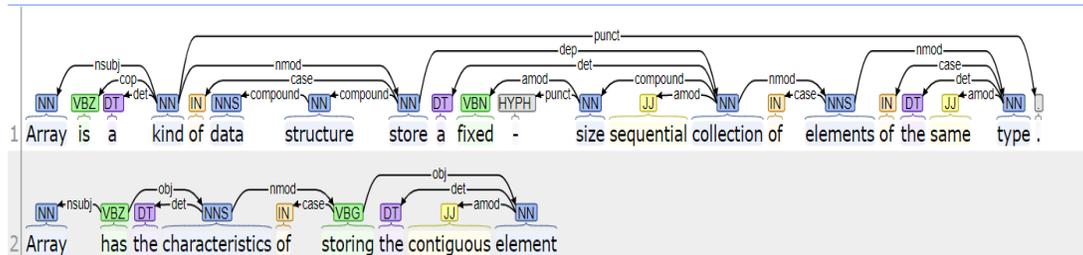


Fig. 4. Dependency Parser.

Blooms verbal list like, “List”, “write”, “State”, “Explain”, “describe”, “Discuss”, “Illustrate”, “Elaborate”, “Mention”.

Rule: Blooms verbal + “the” + ” predefined text” + NN.

Output:

- 1) Mention the characteristics of Array.
- 2) State the characteristics of Array
- 3) List the characteristics of Array

The different set of rules was implemented to the sentence of the input corpus to generate the possible number of assessment questions.

H. Experimental Result

Proposed system has been executed for the sample corpus taken from Grade V computer science book with 50 different text corporuses. Each chapter consists of topics and sub-topics. The system generated a range of objective and subjective questions of various categories. The generated questions have

been manually crisscrossed by domain experts, and the accuracy (A), Recall (R), and Validity (V) of the questions have been measured.

Accuracy (A) is defined as the proportion of retrieved examples that are relevant. It is a metric that measures the number of correct positive predictions that are made during the execution.

$$A = \frac{Q_v}{(Q_v + Q_i)}$$

Recall (R) is a metric that quantifies the proportion of correctly predicted positive predictions out of all possible positive predictions.

$$R = \frac{Q_v}{(Q_v + Q_{ng})}$$

Validity is a metric that represent the number of valid and meaningful questions generated by the proposed system have been compared with the text book questions. Cardinality and validity of each question type is given in Table VIII. Table IX shows the accuracy and recall of the corpus.

TABLE VIII. CARDINALITY AND VALIDITY OF PROPOSED SYSTEM

Corpus Name	Fill up the double blank space		True or False		Short Question Answer	
	Cardinality	Validity	Cardinality	Validity	Cardinality	Validity
ENIAC	12	13	51	48	10	10
EDSAC	13	15	48	45	12	13
EDVAC	11	13	52	50	10	9
UNIVAC	14	15	55	54	11	09
Microcomputers	10	12	46	44	7	8
Minicomputers	13	14	42	43	8	6
Mainframes	16	17	45	41	9	8
Supercomputer	15	17	47	45	10	9
Optical Character Reader	12	13	43	42	9	7
Optical mark Reader	13	14	45	43	8	7
Cardinality	129	143	474	455	94	86

TABLE IX. ACCURACY AND RECALL OF PROPOSED SYSTEM

Corpus Name	Fill up the double blank space		True or False		Short Question Answer	
	Accuracy	Recall	Accuracy	Recall	Accuracy	Recall
ENIAC	0.92	0.86	0.96	0.88	0.88	0.88
EDSAC	0.87	0.93	0.94	0.94	0.93	0.82
EDVAC	0.85	0.85	0.96	0.96	1.0	0.90
UNIVAC	0.93	0.93	0.98	0.95	0.81	0.75
Microcomputers	0.83	0.83	0.96	0.96	0.88	0.80
Minicomputers	0.93	0.87	1.02	0.98	0.83	0.71
Mainframes	0.94	0.94	0.91	0.93	0.86	0.81
Supercomputer	0.88	0.94	0.96	0.96	0.80	0.80
Optical Character Reader	0.92	0.92	0.98	0.95	1.0	0.85
Optical mark Reader	0.93	0.87	0.96	0.93	1.0	0.92
	9.01	8.94	9.63	9.44	9.06	8.24

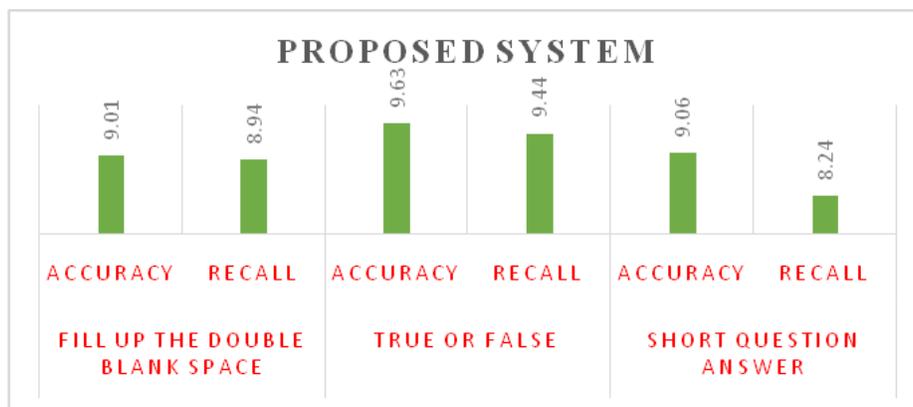


Fig. 5. Accuracy and Precision.

Accuracy and recall of the proposed system is given in Fig. 5. The accuracy of each type is nearly 0.9 % and recall is nearly 0.8%. The overall accuracy and recall are 0.92 % and 0.88 %. The proposed system has been compared with existing system [29][30] and comparatively produces good result. By the existing system[30], the overall precision and recall for the subjective type question were 0.85% and 0.8 %.

IV. CONCLUSION

The proposed system has been designed to verify the learner's knowledge. Its main purpose is to generate objective and subjective type questions from unstructured text. Better result has been achieved with accuracy level of 0.9 and recall as 0.8. Subject expert, recommended the system to assess the learner of grade V student in practice. In future the system can be elaborated with more rules to generate possible number of questions. In match the following question, the important word and their description can be considered to frame questions.

REFERENCES

- [1] Ali, Y. Chali, and S. A. Hasan. Automation of question generation from sentences. 2011.
- [2] Beg and A. Beg, Using open technologies for automatically creating question-and-answer sets for engineering MOOCs, *Comput. Appl. Eng. Educ.* 26 (2018), 617–625.
- [3] L. Bednarik and L. Kovacs, Implementation and assessment of the automatic question generation module, 3rd International IEEE Conference on Cognitive Infocommunications (CogInfoCom), *Inst. Electr. Electron. Eng., Kosice, Slovakia*, 2012, pp. 687–690.
- [4] Ming Liu, A. Calvo Rafael, and V. Rus, Automatic question generation for literature review writing support, *International Conference on Intelligent Tutoring Systems*, Springer, 2010, pp. 45–54.
- [5] P. Pabitha, M. Mohana, S. Suganthi, and B. Sivanandhini, Automatic question generation system, *International Conference on Recent Trends in Information Technology (ICRTIT)*, *Inst. Electr. Electron. Eng., Chennai, India*, 2014, pp. 1–5.
- [6] K. Antonis, T. Daradoumis, S. Papadakis, and C. Simos, Evaluation of the effectiveness of a web-based learning design for adult computer science courses, *IEEE Trans. Educ.* 54 (2011), 374–380.
- [7] J. C. Brown, G. A. Frishkoff, and M. Eskenazi, Automatic Question Generation for Vocabulary Assessment, *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (Vancouver, British Columbia, Canada, 2005)*, *Assoc. Comput. Linguist., Stroudsburg, PA, USA*, 2005, pp. 819–826.
- [8] Maor, and J. K. Currie, The use of technology in postgraduate supervision pedagogy in two Australian universities, *Intl. J. Educ. Technol. Higher Educ.* 14 (2017), 1–12.
- [9] Kurtasov, A system for generating cloze test items from Russian-language text, *Proceedings of the Student Research Workshop associated with RANLP 2013*, *INCOMA Ltd., Shoumen, Bulgaria*, Hissar, Bulgaria, pp. 107–112.
- [10] J. Lee and S. Seneff, Automatic generation of cloze items for prepositions, *Proceedings of Interspeech*, *International Speech*

- Communication Association (ISCA), Antwerp, Belgium, 2007, pp. 2173–2176.
- [11] M. Yildirim, A genetic algorithm for generating test from a question bank, *Comput. Appl. Eng. Educ.* 18 (2010), 298–305.
- [12] Mannem Prashanth, Rashmi Prasad, and Aravind Joshi. (2010). Question generation from paragraphs at UPenn: QGSTEC system description. In *Proceedings of QG2010: The Third Workshop on Question Generation*, pages 84–91.
- [13] Susanti, Y., Iida, R., Tokunaga, T. (2015). Automatic generation of English vocabulary tests. In *Proceedings of the 6th International Conference on Computer Supported Education (CSEDU 2015)*, pp.77-87, 2015.
- [14] Colleen E Crangle and Joyce Brothers Kart. (2015). A questions-based investigation of consumer mental Health information. *PeerJ*, 3:e867.
- [15] Zhe Liu and B. J. Jansen, Subjective versus objective questions: Perception of question subjectivity in social Q&A, *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*, Springer, 2015, pp. 131– 140.
- [16] M. Divate, and A. Salgaonkar, Automatic question generation approaches and evaluation techniques, *Curr. Sci.* 00113891 (2017), 113.
- [17] A. Narendra, M. Agarwal, and R. Shah, Automatic Cloze- Questions Generation, *Proceedings of Recent Advances in Natural Language Processing*, Hissar, Bulgaria 2013, pp. 511–515.
- [18] M. Majumder, and S. K. Saha, Automatic selection of informative sentences: The sentences that can generate multiple choice questions, *Knowl. Manage. E-Learn. Intl. J.* 6 (2014), no. 4, 377–391.
- [19] Swart (2010). Evaluation of Final Examination Papers in Engineering: A Case Study Using Bloom’s Taxonomy. *IEEE Transactions on Education*, (May 2010) Vol. 53, No.2 257-264.
- [20] G.Deena and K.Raja , “Sentence Selection Using Latent Semantic Analysis for Automatic Question Generation in E-Learning System”, *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* ISSN: 2278-3075, Volume-8 Issue-9, July 2019.
- [21] Wolfe, J.H.: Automatic question generation from text-an aid to independent study. *ACM SIGCSE Bull.* 8(1), 104–112 (1976).
- [22] Sheetal Rakangor et al, ” Computer Aided Environment for Drawing (To Set) True or False Objective Questions From Given Paragraph”, *Int. Journal of Engineering Research and Applications*, ISSN : 2248-9622, Vol. 4, Issue 5(Version 3), May 2014, pp.105-108.
- [23] Michael Heilman and Noah, A Smith. 2009. Question generation via over generating transformations and ranking. Technical Report CMU-LTI-09-013, Language Technologies Institute, Carnegie Mellon University.
- [24] Smith, S. ,Avinesh, P.V.S. and Kilgarrieff, Adam (2010) ‘Gap-fill Tests for Language Learners: Corpus-Driven Item Generation’ *Proceedings of ICON-2010: 8th International Conference on Natural Language Processing*. held: 8-11 December 2010, Kharagpur, India. India: Macmillan Publishers.
- [25] NikiforosKaramanis, Le An Ha and Ruslan Mitkov. 2006 Generating Multiple-Choice Test Items from Medical Text: A Pilot Study, In *Proceedings of INLG 2006*, Sydney, Australia.
- [26] Ruslan Mitkov, Le An Ha and NikiforosKaramanis. 2006 A computer-aided environment for generating multiple-choice test items, *Natural Language Engineering* 12(2): 177-194.
- [27] G.Deena and K.Raja , “A Novel Dynamic Self-Assessment Question Preparation Mechanism Using Natural Language Processing”, *Two Day International Virtual Conference on Contemporary Practices of Technology and Management for Economic Growth (ICTMEG’20)*.
- [28] M.C.DeMarneffe and C. D. Manning, Stanford typed dependencies manual. Technical report, StanfordUniversity, 2008.
- [29] Shivank Pandey and KC Rajeswari. Automatic question generation using software agents for technical institutions. *International Journal of Advanced Computer Research*, 3(13):307–311, 2013.
- [30] ShivaliJoshi, Parin Shah , Sahil Shah , “Automatic Question Paper Generation, according to Bloom’s Taxonomy, by generating questions from text using Natural Language Processing”, *International Journal of Innovative Science and Research Technology*, Volume 6, Issue 4, April – 2021.