

Student Testing and Monitoring System (Stms) Using Nlp

Muhammad Saad

Bs-Software Engineering, University of Gujrat, Sialkot, Pakistan
Email: saadchohan570@gmail.com

Shanzah Aslam

Bs-Software Engineering, University of Gujrat, Sialkot, Pakistan
Email: shanzahaslam2@gmail.com

Warda Yousaf

Bs-Software Engineering, University of Gujrat, Sialkot, Pakistan
Email: wardausaf1997@gmail.com

Moeed Sehnan

Ms-Electrical Engineering, Linnaeus University of Sweden, Sweden
Email: moeed.sehnan@uogsialkot.edu.pk

Sidra Anwar

Bs-Computer Science, Gc Women University, Sialkot, Pakistan
Email: engrsid.es@gmail.com

Danish Rehman

Bs-Software Engineering, University of Gujrat, Sialkot, Pakistan
Email: mrdanishrehman7@gmail.com

Received: 26 June 2019; Accepted: 21 August 2019; Published: 08 September 2019

Abstract—In the domain of knowledge, there is a rising demand for such a System to provide learning support via a platform which can generate any sort of questions automatically from provided source either (PDF) books or simply any keyword against a user needs to perform a test where STMS serves the purpose. Regarding Keyword operation, the System scraps all the text from Wikipedia and converts it into multiple choice questions. Moreover, it summarizes raw text from Wikipedia and parse the text from provided content to generate Multiple-Choice Questions(MCQs). The System also finds all the Named Entities and POS (Parts of speech tags) in the content to create relevant questions. The questions include Multiple-Choice Questions(MCQs), Cloze based questions and WH- questions (why, where, when etc.).

In addition, when users score standard points in the test then they qualify for earning zone where they can earn money (\$ Dollars) for scoring points in each test. The Income comes from AdSense applied on the website and other Local ads, Affiliating marketing and advertisements. All in all, the System would help in educational learning by providing helping material in the lacking knowledge areas after analyzing the tests users have performed while the Web-Traffic is the key to Success for monetary benefits.

Index Terms—Web based Student Testing and Monitoring System (STMS), Natural Language Processing, Artificial Intelligence, Semantic Role Labelling, Machine Learning, Wikipedia Scraping, Text Mining.

I. INTRODUCTION

As technology has always been the most rapidly growing domain out of all others. From clocks to smart watches, calendars, or calculators; everything has renovated and taken a new compact form. The same is happening in the domain of knowledge and learning. There is an observed rising concern with the creation of such a platform that allows a common user, from any field to perform quizzes based on Multiple-Choice Questions(MCQs) questions from whatever domain they choose [1]. Certainly, it would also help those who lack in practical experience of freelancing but have theoretical knowledge.

Web based Student Testing and Monitoring System (STMS) is the business research and industrial support study which will assist the users globally to earn money and enhance their knowledge & skills by solving a

questionnaire (Multiple-Choice Questions (MCQs) or WH- questions) in their domain of interest more specifically on their given outlines of the courses or selected keywords [2]. The golden feature of this study is that the user wouldn't just have to perform tests only on the predefined domains and data only but, they can also upload their very own reference books over this system and the system will generate Multiple-Choice Questions(MCQs) from that runtime uploaded file and even from a single keyword they provide [1,2]. This feature will allow the user to not just stay confined only in pre-defined data and domains but to try vast knowledge areas and explore the most out of their concerned field. These Multiple-Choice Questions (MCQs) quizzes will then be assessed by the system and results will be generated. In addition to this, there is a portal for low achievers as well. It will provide them with certain helping materials that will guide them through and help in gaining more. There is an additional feature of earning as well. Our vigilant users who will reach a certain score level will be given the opportunity to earn through the system.

NLP is being used for the automatic generation of questions. As educationists need to conduct the examination to judge student's calibre and creating questions manually is very time-consuming so developing a system which automatically generates questionnaire will be of quite a huge help.

Furthermore, the automated process of generating questions on the premises of the course given, the system of generating questions are isolated into two units i.e. selection of required content and question production. Questions are formed by finding relevant parts of the text which constitutes the content selection. Researcher has seven discourse connectives in concentration i.e. like because, since, although, as a result, for example, and for instance, on that basis of which Question type will be decided [3]. As, if a sentence consists of "since" then question type would be "When". The System has been evaluated for semantic and syntactic soundness of question by two evaluators. Such a system can also be called as a multi-agent system because of its working.

"Semantic Based Automatic Question Generation system uses two techniques named as Semantic Role Labeling and NER (Named Entity Recognizer) technique that converts the inputted sentence to a semantic pattern while developed and artificial immune system would be able to classify the patterns according to the question type" [4]. Feature extraction, learning, storage memory and associative retrieval are the features utilized by the immune system for the purpose of solving classification and recognition tasks. NER and SRL techniques will be used to parse inputted sentence which will specify whether there are person name, location, date in the sentence and through this identification a pattern for the question will be proposed, for example, if there is a person's name mentioned then pattern of the question would be "Who" [5].

This technique of question generation gives a promising ratio of 87% of truly generated question

patterns [4,6]. This ratio is far beyond other way of generating questions automatically.

It is researcher's job to see whether the questions this system is producing can be used as pre-questions or not. Text-based and supporting image are the types of pre-questions to be investigated.

The system suggested here further has three subsystems: Generator of Questions, Descriptor of Knowledge and Executer of E-Learning. The subsystem of Knowledge Descriptor allows the teacher to explain the learning contents. The subsystem of Questions Generator would receive the learning materials and produce required questions in multiple choice questions. The subsystem of E-learning Executer will utilize the generated questions and will let the students to use these questions in education process. While system generated correct and wrong options question (CWOQ) algorithm would be executed relying on sub-algorithm generating One solution, two Solutions, All of The Above, and none Of The Above algorithms." A question bank have been developed for course General Biology course (Bio110) at Faculty of Science, King Abdul-Aziz University (KAU)" [6,7]. The bank contains 12 sections with 239 sub-points and a sum of 46345 inquiries. The student through this system gets the ability to solve the exercises of each section and the shows the given and correct answer after self-produced multiple-choice questions from domain ontologies [7].

Domain specific ontologies are the basics of the approach shown in this paper and lexicons such as WordNet or other linguistic resources aren't utilized [8]. Semantic Web standard technology OWL (Ontology Web Language) is used by the system to generate multiple-choice questions. The System creates multiple choice question items using the Class-based, property-based, terminology-based strategies. Property-based strategies may produce a large number of multiple-choice questions but are very difficult to manipulate syntactically. The created tests were assessed in three dimensions: Pedagogical quality, phonetic/linguistic accuracy and the quantity of questions delivered.

Another research is performed to generate questions from documents written in the Punjabi language. An NER (Named Entity Recognition) Tool was needed to create which can recognize the nouns from a given phrase or sentence and produce proper questions from it. [9] Moreover, mind the Gap: Learning to Choose Gaps for Question Generation needs to address the problem of generating good questions into two parts: first, select sentences to be queried about, and second, to know which part of the sentence will be turned into a question. To achieve the goal of selecting better gap-fill questions, the author has broken down the task into stages i.e. Sentence selection, Question construction and Distractors generation [9,10].

Features brought into consideration for generating questions are Token count, lexical, syntactic, and semantic and NER feature is used to generate the fill in the blanks question [11]. The system would consist of a straightforward pipeline. First the source text is broken

down into sentences to be processed by SENNA software, SENNA does the tokenizing, POS tagging, syntactic constituency parsing and semantic role labelling. A matcher function returns a list of patterns that match the predicate-argument structure of the sentences from the source. This function is called for each predicate and its linked semantic arguments [12]. Paper focuses on evaluating the produced questions basically in terms of their linguistic quality.

II. RELATED WORKS

CQG (Cloze Question generation) system is the system that produced a list of cloze questions in English article. CQG system is broken down into three main units, i.e. the selection of sentence, key and distractor. Initial stages involve the selection of relevant and informative portions of the text and keywords from the selected text are recognized out in the second stage [13]. Key selection will not be noun or adjective instead it will be found on the basis of NER. The third stage involves the selection of Distractors. First two stages are not bound in a fixed domain while the third one is bound. "Because quality of distractor depends on domain so distractor will be selected on the basis of the key selected and through web, list of distractors will be generated and knowledge-based distractor list will be generated". System is manually evaluated through the following three steps:

- 1). Evaluation of the selected sentence
- 2). Evaluation of selected keyword
- 3). Evaluation of selected distractor.

"Automatic question generation on the basis of the discourse connectives ", content selection and question generation are the two units of this system [14]. In content selection the relevant part of text is found to frame the question and question generation is includes sense disambiguation of the discourse connectives, recognition of the type of question and performing syntactic changes on the content. Two evaluators have been used for the check of semantic and syntactic correctness.

"Automatic Question Generation Using Software Agents for Technical Institutions" produced a system which took text files from the user as an input and the output hence produced by the system is a set of questionnaires based on bloom's taxonomy [15]. Bloom's taxonomy-based questions help in assessing the learning capability of the user. In Document processing tree tagger tool and stemming process is done to eliminate the human process [16]. Data characterization takes a rundown of keyword created by Data Processing and finds the Bloom's classification of those words, via looking through suitable activity verb in store which fits with the given keyword. The output of Information Classification is used as an input of Question Generation to produce questions. The procedure is a format-based methodology, which fits the chosen keywords in the question layout as per the Bloom's levels.

III. METHODOLOGY

Apart from **Stanford** and **NLTK** in **NATURAL LANGUAGE PROCESSING**, there is a lot to enhance which system describes [14].

The methodology behind the Multiple-Choice Questions(MCQs) generation used in this proposed system includes:

A. Text Parsing

Either user would provide text in the form of PFD files or simply give any interesting topic, System has to parse the text especially when it comes from Wikipedia. STMS uses Beautiful Soup Python library to scrap the text from URL for a given keyword. Text parsing includes removal of References[n], special characters, and links in the text that has been scrapped from Wikipedia. Text refining is the basic approach of Pre-processing of the text. An example Python code to scrape text from URL.

```
1) from urllib2 import urlopen
2) from bs4 import BeautifulSoup
3) url = "https://en.wikipedia.org/wiki/NLP"
4) allData = urlopen(url)
5) bs = BeautifulSoup(data, "lxml")
6) rawText = bs.findAll("p") #Find All <p>
   (paragraph tags in the HTML file)
```

B. Text Summarization

In this step, Regular Expression (REGEX) plays an important role to refine text. After refining text can be summarized using SUMY Python Library which includes (LUHN, LexRank, LSA) summarization algorithms to Summarize the text and moreover system selects an average number of sentences to generate question sentences. The importance of the sentences can be analysed using Keyword Frequency Method.

```
1) import re
2) import sumy
3) from sumy.parsers.plaintext import
   PlaintextParser
4) from sumy.nlp.tokenizers import Tokenizer
5) from sumy.summarizers.lex_rank import
   LexRankSummarizer
6) from sumy.summarizers.luhn import
   LuhnSummarizer
7) Removal of References and extra spaces + special
   characters using RegularExpressions
8) article_text = re.sub('[0-9]*J', '', article_text)
9) article_text = re.sub('\s+', '', article_text)
10) Summarization
11) summarizer = LexRankSummarizer()
12) summary1 = summarizer(original.document, 50)
```

C. Question Formation

After Sentences selection or text summarization, the system focuses on the question formation based on NER and SRL. SLR (Semantic Role Labelling), one of the approaches of Natural Language Processing is used to make the patterns of the words in the string of sentences.

The Named Entity Recognition (NER) comes with the same methodology to find Entities like (PERSONS, LOCATIONS, DATE, and ORGANIZATION). The System creates clusters of them and then generates questions accordingly.

D. Wh- Question Approach

In this approach, the system evaluates the sentences according to the NES. STMS classify the sentences like NEP (For PERSON), ORG (Organization) and LOC (Location). And generate questions accordingly.

- 1) *WHO/WHOM functions for NE(Person).*
- 2) *WHERE functions call for LOC (NE-Location)*
- 3) *WHEN functions call for (NE-Date).*

The WH- defined functions will create complete sentences accordingly.

E. Pos Based

Parts-Of-Speech based questions are handled in an informal approach to ask all the relevant Proper Nouns (PNNs), which can be identified using POS tagging in python. Natural Language Tool Kit (NLTK) Python library has a major role in Natural Language Processing of the text to find all the POS tags.

- 1) *from nltk.tag import pos_tag*
- 2) *stopwords = nltk.corpus.stopwords.words('english')*
- 3) *newList = [(sent.text, sent.label_) for sent in nlp(text.ents)]*
- 4) *qtext = word_tokenize(article_text)*
- 5) *tags = pos_tag(ner_Text)*
- 6) *print (tags)*

It is needed to remove all the stop words to get the accurate Proper Nouns from a refined data.

F. Cloze Based

In Natural Language Processing Question Cloze based questions formation is a formal style. First of all, the system has to define the Entity to be blanked. Then **RegEX** can replace a certain string with that specific NE

tag. The accurate answer can be placed as any of the four answers but not always at the first place. Simple **for Loop** can be used to change the placement of the correct answer among four of them. **RegEX** is used in the following code to replace the blank "_____" with certain (NE) that System requires:

- 1) *pattern = " "*
- 2) *replacement_str = "_____"*
- 3) *for k, v in newList:*
- 4) *if v == "PERSON":*
- 5) *pattern = pattern + r'(' + k + r')|'*
- 6) *pattern += r'(' + r"'" + r')|'*
- 7) *compiled_pattern=re.compile(pattern,re.IGNORECASE)*
- 8) *result=re.sub(compiled_pattern,replacement_str,final_summary)*
- 9) *each_sentence = sent_tokenize(result)*

Then system tokenizes all the string of text into tokens of sentences.

G. Distractor Generator

One of the important steps to make an appropriate Multiple-Choice Question is to generate distractors against a correct answer. And like always Python Libraries come with a definite solution. WordNet (lexical database of English) which can be used to find Hyponyms against each correct answer and then hyponyms of each Hyponyms can help the system to find definite distractors.

e.g. Correct answer is (cat)

- a) Cat
- b) Rabbit
- c) Rat
- d) Goat

This is how hyponyms help the System STMS to complete the process of questions along with distractors.

H. General Work Flow Of The Algorithm

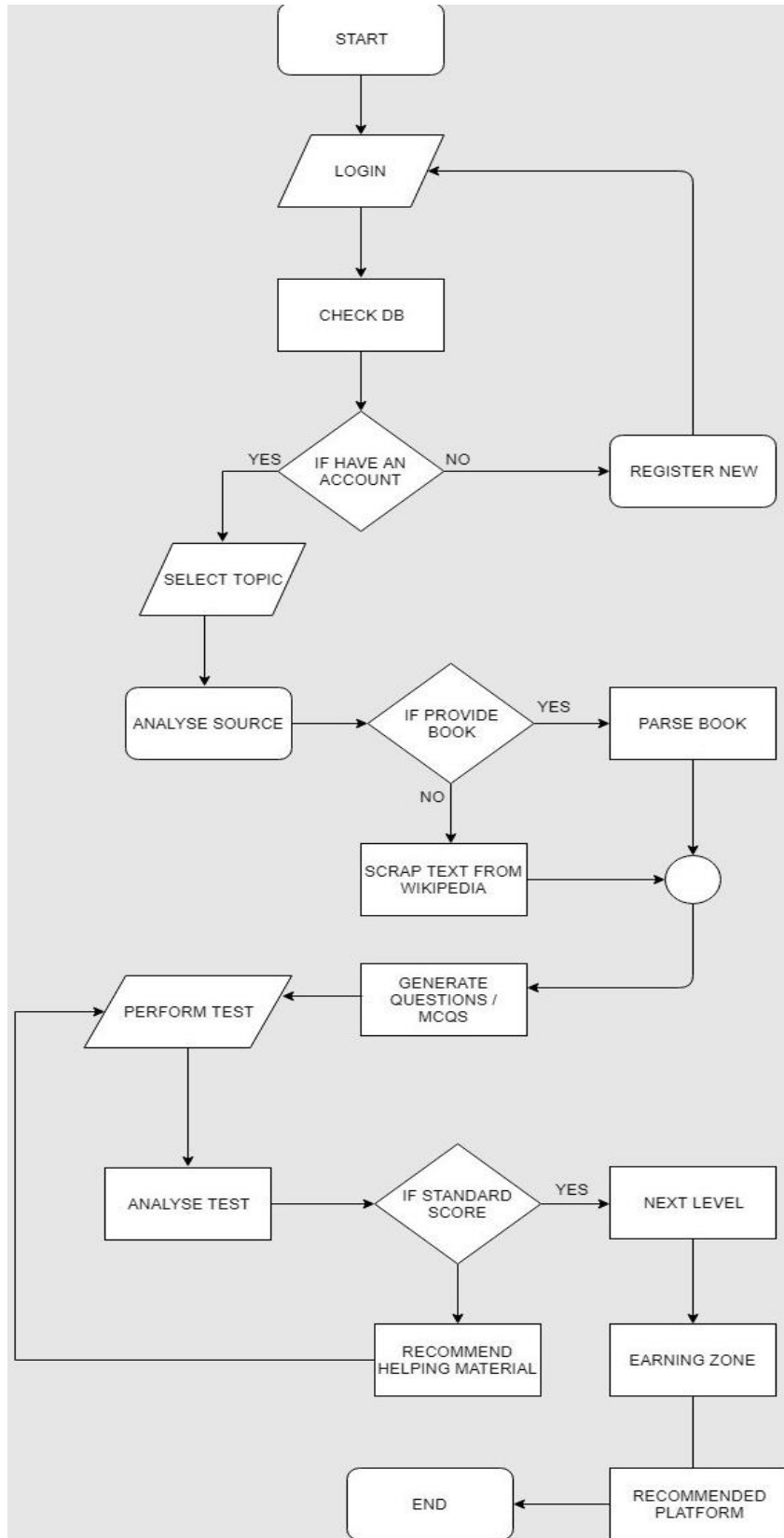


Fig.1. Workflow

I. General Application Architecture

A three-tier architecture is used which will be client side, Web Server and DBMS.

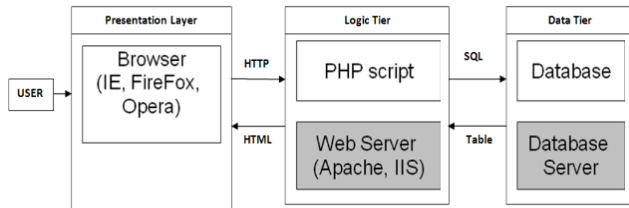


Fig.2. A 3-Tier Architecture

J. Tools And Technologies

1. Languages

- a) **Bootstrap:** Bootstrap is used for front end development of the site which is the latest version of HTML5. This would help to create a modern and responsive site.
- b) **CSS:** CSS is used for the beauty of the site which will be helpful for the modern front end of the site.
- c) **PYTHON:** Python language is preferred because web scrapping for data extraction from different websites is needed and only python language has major support for web scrapping in term of libraries and framework.
- d) **PHP:** PHP is used for the server side of website because it would be easier to integrate with the backend process.
- e) **SQL:** SQL will be used for the database creation and database handling alongside with PHP. The reason again to select SQL is that it operates well with PHP.

2. Tools

- a) **Net Beans:** Net Beans will be used to develop server-side code and the reason to choose Net Beans is that it has more features and provide easier development.
- b) **Anaconda:** Anaconda will be used in the project to develop a web scrapping code because it provides all python features at one platform.
- c) **Photoshop:** Adobe Photoshop is used to develop the logo and other different graphics files for the website.

3. Techniques

- a) **Data Mining:** To refine the data from LEO Brain in the form of Multiple-Choice Questions(MCQs), relevant to the outline’s keywords.
- b) **Web Scrapping:** Web scrapping technology will be used to extract data in the form of Multiple-Choice Questions(MCQs) which will be generated by Python.

IV. ALGORITHM

- 1) *stopwords* = `nltk.corpus.stopwords.words('english')` *url* = `"https://en.wikipedia.org/wiki/Natural_philosophy"`
- 2) *data* = `urlopen(url)`
- 3) *bs* = `BeautifulSoup(data, "lxml")`
- 4) *rawText* = `bs.findAll("p")`
- 5) *article_text* = ""
- 6) *for p in rawText:*
- 7) *article_text* += `p.text`
- 8) *article_text* = `re.sub("[0-9]*J", '', article_text)`
- 9) *article_text* = `re.sub("\s+", '', article_text)`
- 10) *original* = `PlaintextParser.from_string(article_text,Tokenizer("english"))`
- 11) *summarizer* = `LexRankSummarizer()`
- 12) *summary1* = `summarizer(original.document, 50)`
- 13) *nlptext* = `nlp(final_summary)`
- 14) *newList* = `[(sent.text, sent.label_) for sent in nlptext.ents]`
- 15) *pattern* = ""
- 16) *replacement_str* = "_____"
- 17) *for k, v in newList:*
- 18) *if v == "PERSON":*
- 19) *pattern* = `pattern + r'(' + k + r')'`
- 20) *pattern* += `r'(' + r"'" + r')'`
- 21) *compiled_pattern* = `re.compile(pattern, re.IGNORECASE)`
- 22) *result* = `re.sub(compiled_pattern,replacement_str,final_summary)instances = sent_tokenize(result)`

V. VRESULTS

To illustrate, system would generate Multiple Choice Questions(MCQs) from the given URL of Wikipedia:
<https://en.wikipedia.org/wiki/Pakistan>

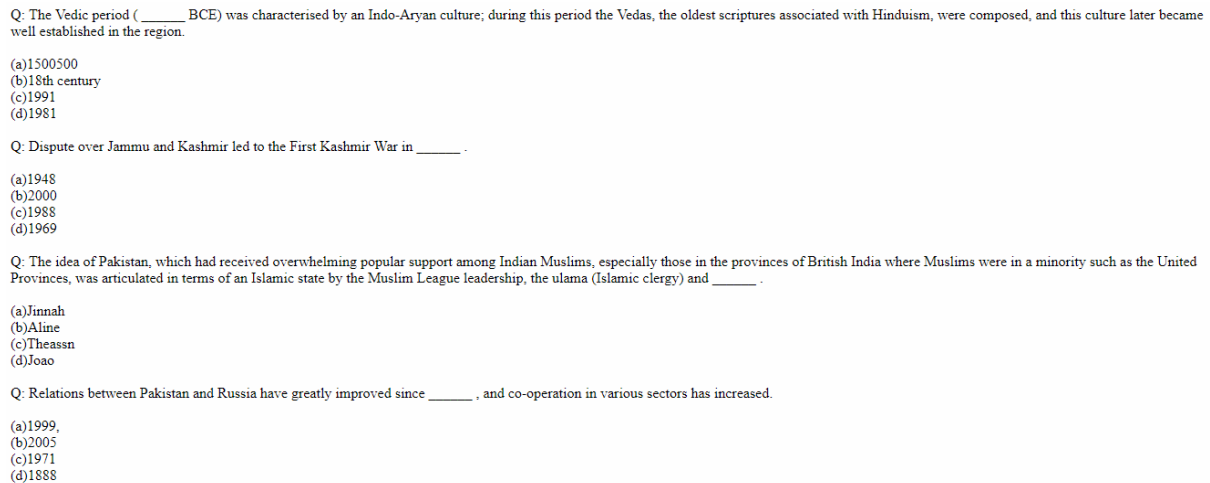


Fig.3. Results Screenshot

VI. DISCUSSION

In present times, unemployment is hitting its highest scores. Technology is taking over and manpower is being kicked out of the companies [17]. Regardless of whatever degrees they carry they are still being surpassed by the technology wave happening around [18]. The thought behind creating this project is to attract users to interact with knowledge domains, to learn through our easy to interact with user platforms and perform timed test sessions and to earn through this interactive learning process.

Apart from this, it will provide immense ease to students who can't find questionnaires made for their bulk of courses. And then they fail to comprehend such huge amount of uniform data and lose their scores in their examinations. This system will provide them with easy to access and easy to generate quizzes and questionnaires.

As indicated by the "Ontology Based Multiple-decision Question generation report 2015" 913 and 428 inquiries were created from the KR and Java ontologies, individually [19]. Among these are 535, 344 questions that have at any rate 3 distractors for the KR and Java ontologies, individually. Among the 913 KR questions, 50 questions were chosen haphazardly to be looked into by area specialists. Java questions were first nourished to a programmed filter that filters out any inquiry where there is a cover, i.e., a string of multiple characters that show up in both the stem and the key. This is to dispose of the odds of give some insight to the understudies to figure out the right answer without really having the expected learning to answer it effectively. These inquiries were not chosen in a simply irregular manner, rather, each inquiry has been chosen arbitrarily, at that point checked physically to ensure that it isn't repetitive, i.e., comparable regarding substance to recently chosen inquiry however extraordinary in introduction.

An inquiry is considered "helpful" in the event that it is evaluated as either "valuable for what it's worth" or "valuable however requires minor enhancements" by a commentator. 46 out of the 50 KR inquiries were viewed as valuable by in any event one commentator. 63 out of

the 65Java inquiries have been evaluated as helpful by at any rate one commentator. A given distractor is considered "valuable" on the off chance that it has been useful (i.e., picked by at any rate one understudy). For the inquiries conveyed in the first round, in any event two out of three distractors were helpful. For the inquiries conveyed in the second round, at any rate one distractor was valuable with the exception of one inquiry which has been addressed accurately by all the seven understudies. We utilized Pearson's coefficient to process thing separation to demonstrate the connection between understudies' exhibition on a given inquiry and the general execution of every understudy on all inquiries. As per commentators' remarks, this is mostly because of the absence of proper articles (i.e., the, an, a).

An inquiry is too difficult for a specific gathering of understudies on the off chance that it is addressed effectively by under 30 % of the understudies and is excessively simple whenever replied by more than 90 % of the understudies. In the two cases, the inquiry should be looked into and improved. As needs be, we believe the inquiry to be difficult on the off chance that it is addressed effectively by 30–60 % and simple on the off chance that it is addressed accurately by 60–90 % of the understudies. Concerning the six inquiries conveyed on paper, two inquiries were sensibly difficult and two were sensibly simple for the understudies. These four inquiries were in accordance with difficulty estimations by the Multiple-Choice Questions (MCQs) Generator apparatus. One out of the six inquiries were too difficult for the understudies. Amazingly, the device and the three commentators have evaluated this thing as simple.

At long last, one inquiry was unreasonably simple for the under studies, be that as it may, it was evaluated as difficult by the device. This is expected to having an idea in the stem. So also, for the inquiries managed on the web, one inquiry was sensibly difficult and one inquiry was sensibly simple for the understudies; just in accordance with apparatus estimations. One out of the six inquiries was unreasonably simple for the understudies (100 % correct answers). This inquiry was appraised as simple by the instrument. Once more, one inquiry was

evaluated as difficult by the device yet was simple for the understudies due to having an idea in the stem. Two inquiries were not in accordance with apparatus estimations but rather were in accordance with estimations of in any event two analysts. For 51 out of the 65 Java inquiries, there has been an understanding between the apparatus and in any event one commentator. The level of understanding is a lot higher with simple inquiries achieving 100 % concurrences with at any rate one analyst. This could imply that the produced distractors for difficult questions were not conceivable enough which can be because of numerous reasons, for example, having lexical or semantic intimations. Figure 2 demonstrates the quantity of inquiries for which there is an understanding between the apparatus and at any rate one, a few analysts for the 50 KR inquiries and the 65 Java inquiries that have been checked on by the space specialists.

VII. CONCLUSION

From the comparative analysis of the question paper generated in the past out in this research, it is evident that Natural Language Processing is still a domain of huge improvements.

The system generated through the extensive research and hard work that we have employed in our project has such techniques that create questions first by scrapping the given text either in pdf or through a keyword given query. Scrapping is done through a library named BeautifulSoup which is a library of python language. Then the scrapped data is cleaned using RegEx. Cleaning basically removes any references, links or any special characters in the text. In addition, the summarization of the cleaned text is carried on using the Summy library of python. This basically summarizes the whole text to present it into a compiled form. Then this summarized text is used for Sentence Tokenization where the text is identified that where the sentences end. Using this text the Named Entity Recognizers (NER) are found. These NERs are found using Natural Language Processing (NLP) which employs the library named Natural Language ToolKit. From here the data is chunked using RegEx. This creates the questions and then distractors are generated using WordNet library.

To sum up, this system closely works on precision and generates cloze based questions.

VIII. FUTURE WORK

As AI is the field of precision and not accuracy. It is intended to improve the percentage of this accuracy making it near to perfect in future. Secondly this algorithm holds the system back in generating questions of subjects like chemistry maths or statistics. In future, we are looking for the solutions of this hurdle and planning to make the system smart enough to recognize the written text and generate questions without any human help.

REFERENCES

- [1] Baumeister, G. J. (2017). Knowledge Engineering and Software Engineering. 36 German Conference on Artificial Intelligence.
- [2] Gopal Sakarkar, S. M. (2012). Intelligent Online e-Learning Systems: A Comparative Study. International Journal of computer Applications, 65(4).
- [3] Divate, M., & Salgaonkar, A. (2017). Automatic Question Generation Approaches and Evaluation Techniques. Current Science, 113(09), 1683. doi: 10.18520/cs/v113/i09/1683-1691
- [4] Mrs. Subitha Sivakumar, M. V. (2015). A User-Intelligent Adaptive Learning Model for Learning Management System Using Data Mining And Artificial Intelligence. International Journal for Innovative Research in Science & Technology, 1(10)
- [5] Padmaja Appalla, V. M. (2017). An efficient educational data mining approach to support e-learning. 24(4).
- [6] Draper, S. (2009). Catalytic assessment: understanding how MCQs and EVS can foster deep learning. British Journal Of Educational Technology, 40(2), 285-293. doi: 10.1111/j.1467-8535.2008.00920.x.
- [7] Bhattacharya, S. a. (2016). Intelligent e-Learning Systems: An Educational Paradigm Shift. International Journal of Interactive Multimedia and Artificial Intelligence, 4.
- [8] Chinaguravaiah Makkena, K. A. (2017). A Study on efficiency of Data Mining Approaches to Online -. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2(4).
- [9] D.Suresh, S. (2013). The Impact of E-learning system Using Rank-based Clustering Algorithm (ESURBCA). International Journal of Computer Applications, 83.
- [10] Gáyk, J. (2015). Using Visual Analytics Tool for Improving Data Comprehension. International conference on Educational Data Mining.
- [11] Guohui Xiao, D. C. (2018). Ontology-Based Data Access: A Survey. Proceedings of the twenty-Seventh International Joint Conference on Artificial Intelligence.
- [12] Jacopo Amidei, P. P. (2018). Evaluation Methodologies in Automatic Question Generation 2013-2018.
- [13] Lauren Fratamico, S. P. (2017). A Visual approach towards knowledge Engineering and Understanding How Students Learn in Complex Environments.
- [14] Shabina Dhuria, S. C. (2014). Ontologies for Personalized E-Learning in the Semantic Web. International Journal of Advanced Engineering and Nano Technology, 1(4).
- [15] Xindong Wu, H. C. (2015). Knowledge Engineering with Big Data. Intelligent Systems, IEEE.
- [16] Zachary A. Pardos, L. H. (2018). Analysis of Student Behaviour in Habitable Worlds Using Continuous Representation Visualization.
- [17] Dahler, D. (2016). IMPROVING WRITING DESCRIPTIVE TEXT BY USING NLP STRATEGY AT VIII GRADE OF SMPN 11 DURI. Lectura : Jurnal Pendidikan, 7(2). doi: 10.31849/lectura.v7i2.247
- [18] Divate, M., & Salgaonkar, A. (2017). Automatic Question Generation Approaches and Evaluation Techniques. Current Science, 113(09), 1683. doi: 10.18520/cs/v113/i09/1683-1691
- [19] Pishghadam, R., & Shayesteh, S. (2014). Neuro-linguistic Programming (NLP) for Language Teachers: Revalidation of an NLP Scale. Theory And Practice In Language Studies, 4(10). doi: 10.4304/tpls.4.10.2096-2104

Authors' Profiles



Muhammad Saad Chohan received his degree of BS in Software Engineering in 2019 from University of Gujrat. He is conducting research for Artificial intelligence.



Shanzah Aslam received her degree of BS in Software Engineering in 2019 from University of Gujrat. She is conducting research for Artificial intelligence.



Warda Yousaf received her degree of BS in Software Engineering in 2019 from University of Gujrat. She is conducting research for Artificial intelligence.



Moeed Sehnan is a Lecturer of software engineering in university of Sialkot. He received master degree in Electrical Engineering from Linnaeus university of Sweden in 2010-2012. He teaches course related to Artificial intelligence, computer networks, Research methodology and professional practices. His primary research interests involve Reinforcement learning, mobile communication, 5G and antenna design.



Sidra Anwar received her degree of BS in Software Engineering in 2014 from Fatima Jinnah Women University, Rawalpindi and MS in Project Management from CIIT, Islamabad in 2017. She has been an interdisciplinary researcher and lecturer of Computer science in GC Women University, Sialkot since Aug 2015.



Danish Rehman received his degree of BS in Software Engineering in 2019 from University of Gujrat. He is conducting research for Artificial intelligence.

How to cite this paper: Muhammad Saad, Shanzah Aslam, Warda Yousaf, Moeed Sehnan, Sidra Anwar, Danish Rehman, " Student Testing and Monitoring System (Stms) Using Nlp ", International Journal of Modern Education and Computer Science(IJMECS), Vol.11, No.9, pp. 26-34, 2019.DOI: 10.5815/ijmeecs.2019.09.03