An Automated Questions Bank Generation and Assessment Framework for Post Graduate Software Engineering Courses

By

Nayab Gull

Registration Number: 00000318789

Supervised By

Dr. Farooque Azam

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING

COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING (E&ME),

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY (NUST),

ISLAMABAD

June 2022

An Automated Questions Bank Generation and Assessment Framework for Post Graduate Software Engineering Courses

By

Nayab Gull

Registration Number: 00000318789

A thesis report submitted in partial fulfillment of the requirements for the degree of MS in Software Engineering

Thesis Supervisor:

Dr. Farooque Azam

Supervisor's Signature:

____

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING

COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING (E&ME),

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY (NUST),

ISLAMABAD

June 2022

DECLARATION

I declare that this research work titled, “An Automated Questions Bank Generation and Assessment Framework for Post Graduate Software Engineering Courses” is my work and it has not been submitted for evaluation anywhere else. All the material from other sources used in this report has been appropriately cited.

Student’s Signature:

Nayab Gull-318789

MS-19-CSE

____

PLAGIARISM REPORT

This thesis report has been checked for Plagiarism. Attached is the Turnitin report checked by Supervisor.

Student's Signature:

Nayab Gull

00000318789

Signature of Supervisor:

COPYRIGHT STATEMENT

ACKNOWLEDGEMENT

I am extremely thankful to Allah Almighty for his bountiful blessings throughout this work. Indeed this would not have been possible without his substantial guidance through every step, and for putting me across people who could drive me through this work in a superlative manner. Indeed none be worthy of praise but the Almighty.

I am profusely grateful to my beloved parents for their love, prayers, support, and sacrifices in educating and preparing me for my future. I also thank my siblings who encouraged me and prayed for me throughout the time of my research.

I would also like to express my gratitude to my supervisor **Dr. Farooque Azam** for his constant motivation, patience, enthusiasm, and immense knowledge. His guidance helped me throughout my research and writing of this thesis.

I would like to pay special thanks to **Muhammad Waseem Anwar** for his incredible cooperation and for providing help at every phase of this thesis. He has guided me and encouraged me to carry on and has contributed to this thesis with a major impact. I could not have imagined having a better advisor and mentor for my MS study.

I would also like to thank my Guidance Committee Members **Dr. Wasi Haider Butt** and **Dr. Arsalan Shaukat** for being on my thesis guidance and evaluation committee. Their recommendations are very valued for improvement of the work.

Last but not the least, I would like to express my gratitude to all the individuals who have rendered valuable assistance to my study.

Thanks for all your encouragement!

I am dedicated to my beloved parents whose tremendous support and cooperation led me to this wonderful accomplishment.

Table of Contents

Abstract

The evolution in E-Learning technologies establishes the significance of automation in students' assessment process. Particularly, researchers and practitioners are devising new methods to automatically perform students' assessments effectively during courses. The development of a questions bank for students' evaluation processes like quizzes, exams, etc. is highly important. It is usually performed manually by instructors and therefore, the development of effective questions for comprehensive assessment is a very time-consuming activity. Consequently, artificial intelligence techniques are employed to generate questions automatically. Although there exist several state-of-the-art studies for the automated generation of questions bank, these are mostly content-based techniques where rich contents (e.g., paragraphs, book chapters, etc.) are considered for question bank generation. However, in current e-learning or face-to-face educational practices, lectures are usually conducted through PowerPoint slides where limited content or contextual information is available. A detailed literature review of 50 research papers on question generation indicates the lack of a framework for automated question bank generation from PowerPoint slides. Consequently, the existing question bank generation approaches are hard to apply in a real educational environment where slides are the primary mode of lecture.

To tackle this issue, An Automated Questions Bank Generation and Assessment Framework are proposed in this MS thesis. The proposed framework will employ Artificial Intelligence (AI) techniques like Natural Language Processing (NLP) etc. to enable the automatic generation of questions bank from PowerPoint slides. The effectiveness of the proposed framework will be established through different Postgraduate level software engineering courses where questions bank will be generated from lecture slides automatically. Our results show the effectiveness of the proposed algorithm for the text i.e., PowerPoint slides.

Keywords: Student evaluation assessments, Question bank generation, Natural Language Processing (NLP), Parts of Speech (POS) tagging

Chapter 1

Introduction

Chapter 1
Introduction

This section provides a detailed introduction to the important concepts related to our research, the current problem, and an overview of our solution. It is organized into five sub-sections. **Section 1.1** describes the background study; **Section 1.2** provides the goals and objectives of the thesis. **Section 1.3** discusses the motivation of thesis work. **Section 1.4** gives the problem statement of the research, **Section 1.5** discusses the proposed methodology, and thesis organization is presented in **Section 1.7**.

1.1 Background Study

The purpose of this section is to introduce the background study of multiple important concepts which has been used in this research. These concepts include:

- Students' assessments system
- Current methodologies of questions generation
- Importance of automation in question generation
- Natural Language processing
- Existing automation methodologies

1.1.1 Students' Assessments

In the current competing era, education has become the most important part of everyday life. Instructors impart knowledge to the students by delivering lectures on the education learning process worldwide. Assessments are being taken to assess the students' knowledge and to identify their weak points. Assessment items are mainly categorized into two: objective and subjective type questions from a specific content used in quizzes and papers. The notion of such types of assessment items is being used from lower levels to higher levels of the education system. Objective type questions mainly involve cloze questions i.e., multiple-choice questions (MCQs), and open cloze questions i.e., fill in the blanks type questions. An MCQ contains one or more blanks in a sentence and four options are provided to fill the blanks to complete the sentence. Following is the example of an MCQ question type. It contains a statement of a question proceeding with the four options. One is the key option, and the remaining three options are distractors.

The transformation from the UML model to Java requires:

1. Source model
2. Well-defined language
3. Target model
4. Transformation definition

The correct option here is a Well-defined language. The rest of the three are wrong options also known as distractors [52]. Distractors are basically to distract the learner and check his knowledge. Multiple-choice questions have broad usage in educational assessments mostly at higher grades. They are quick and straightforward to evaluate; they are simple to score objectively; they may be used to sample a wide range of content, and they only take a few minutes to administer. This is the reason such types of assessment items are broadly used in educational assessment. Despite some possible drawbacks, such as the impact of guessing and unintentionally exposing students to incorrect information. Another type of objective type question is fill in the blanks. Fill in the blanks type questions are open cloze questions and provide no options. Here, students need to fill the blank themselves to complete the sentence. The following example shows the fill-in-the-blanks question type.

Question: The goal of ....is to improve the performance of frequently occurring queries and transactions? Answer: Denormalization

The aforementioned types of objective-type questions are widely used to check the students' knowledge of a broad range of domains in a short time. The education system at a higher level is mainly relying on assessments comprising objective-type questions. These types of assessment questions play a major role in educational assessments as well as in active learning. They have a variety of advantages including quick evaluation, less testing time, consistent scoring, and the possibility of an electronic evaluation. Many examinations use an objective type of question papers through a computerized environment. Generate questions are the most important part of the learning process. It is clearly understood that generating the assessment questions is the toughest part. In the manual preparation, questions are not assured in excellence and fairness to assess the students' skills.

1.1.2 Current Methodologies of Questions Generation

Current academic infrastructure relies on the manual generation of questions for students' assessments. Question generation is the task of generating questions from various text inputs and having prospective learning content. Manual preparation of assessment questions is time-consuming and costly. Teachers are concerned about this duty regularly. Generation of such types of objective questions is a very challenging task [35] and gave birth to the idea to automate the question generation task for assessments. To overcome the challenge of creating assessmen items manually, researchers and practitioners have explored and implemented different strategies. The research community devoted substantial effort to find the ttechniques for the automatic generation of assessment questions. The research on the automatic generation of questions started at least 20 years ago [9].

1.1.3 Importance of Automation in Question Generation

Automation in student assessment tasks involves the generation of assessment questions automatically from an intended corpus to examine the content knowledge of the students. To automate the question generation, artificial intelligence techniques aided a lot. Artificial intelligence techniques i.e., Natural language processing NLP has the main role in generating automated questions generation. This technique helps in understanding the text structure of the language.

1.1.4 Natural Language Processing

NLP is the ability of computer software to interpret spoken and written human language, often known as natural language [54]. It's a part of AI (artificial intelligence) (AI). Natural language processing is divided into two stages: data preprocessing and algorithm development. Data preprocessing is the process of preparing and "cleaning" text data so that machines can examine it. Preprocessing transforms data into a usable format and highlights text features that an algorithm can use.

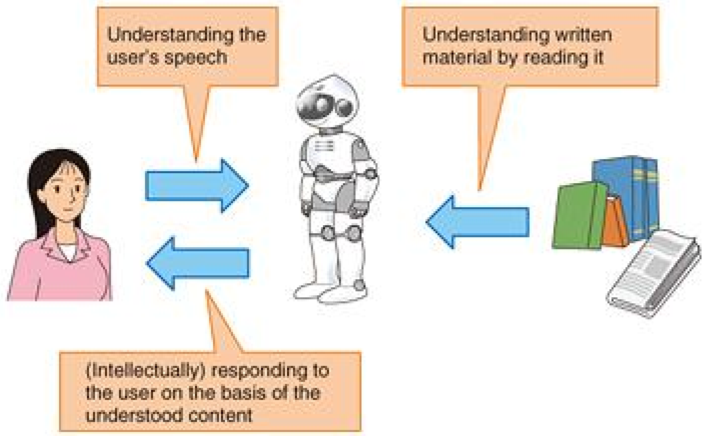

Figure 1- 1- Natural Language Processing

After the data has undergone preprocessing, an algorithm is developed to process it. There are numerous approaches to natural language processing, but the following two are the most popular:

1) Rule-based system: This system makes use of well-established linguistic rules. This technique, which is still in use today, was used in the early phases of natural language processing. 2) System based on machine learning: Machine learning algorithms use statistical methods. They are taught how to perform tasks using training data, and when more data is processed, they modify their methods. Using a combination of machine learning, deep learning, and neural networks, natural language processing algorithms improve their own rules through repeated processing and learning.

Using artificial intelligence and machine learning techniques, the existing state of the art has generated questions automatically. NLP is a technology used by researchers and practitioners who have developed useful algorithms for the development of automatic assessments. To automate question generation and learning, neural network classification models and a variety of other techniques based on the method of inputting text are used. Different types of algorithms are used according to the different types of questions. MCQs and fill-in-the-blank questions are predominantly generated by the researchers in literature. Text from different sources i.e., Wikipedia pages, and books is used as a dataset for question generation tasks. Text is taken from these sources as input to the proposed algorithms and generates questions from them.

1.1.5 Existing Automation Methodologies

Different types of algorithms are applied based on the generation of questions and type of the input taken from the specific domain. For example, if we talk about the medical domain that contains the concept in such a way that there are relations present between entities and events that coreferences over multiple sentences. Now algorithms provided for generating questions for this type of text input would be different compared to the other fields. In literature, many authors [16] [42] [43] [44] focus on the medical field and provide very graceful methodologies for the automatic generation of questions. Some researchers used the techniques for question generation for general domains other than education i.e., for news and sports information, etc. Now for this type of domain, researchers are generating factual questions. Factual questions such as Who? Where? Which Country? When? What?, How much?, and What Organization? are among the questions generated. They can ask specific questions regarding certain facts using factual questions. This type of data can usually be found in the form of information about people, places, dates, events, monetary values, or even specific organizations/institutions. Wikipedia pages [39] are used as datasets for such types of domains to generate questions. Some [11] [17] [19] are generating automatic questions from the datasets of other educational fields i.e., the computer science domain. It is concluded that some authors have provided the algorithms for domain-specific only. Some researchers also proposed algorithms for the open domain. The Open-domain covers many domains of knowledge and encompasses a wide range of topics.

It has been concluded that researchers offer valuable approaches for question generation for different fields. Existing methodologies are found beneficial, but the majority of the researchers utilized the dataset from undergraduate books [18] [44]. Question generation in multiple languages [25] is also found in the literature. This shows the high attention of researchers towards the generation of automatic questions and answers. However, it has also been noticed from the literature that nobody is using the educational dataset for real-time educational scenarios at the postgraduate level where the mode of delivering the lectures is slides. Learning from books is not the current notion of the education system, especially at a higher level. The current mode of delivering lectures is slides and the text present in slides is very different compared to the books where the concept is written in detail. In the lecture slides, important points and clues are present related to a concept while existing methodologies are not giving promising results to the real-time educational scenario. Therefore, an automated question bank generation system is needed to address the higher level of the education system. Therefore, the main aim of this research is to make an automatic question bank generation system that will aid the instructors at the postgrad level.

1.2 Goals and Objectives

The primary goal of this study is to reduce human effort by implementing advanced computing technology. This study aims to offer a method for automatically creating questions from slide content. Although the reduced text content and structure in slides results in poorer annotation quality when compared to long, well-articulated texts, we found that the automatically generated questions and answers are still applicable and appropriate in many cases. A novel methodology is proposed in this study to aid postgraduate instructors by focusing on the real-time education environment. The subject of postgraduate courses is centered on this research. This study employs artificial intelligence techniques such as natural language processing NLP to smoothly tailor the text of PowerPoint slides. To accommodate the text structure that is employed in PowerPoint lecture presentations, the proposed methodology uses a rule-based approach. To create questions from the text, a variety of methodologies and technologies have been used. Therefore, this study caters to the text used in lecture slides where sentences are not in the appropriate order and only the major hint is given.

1.3 Motivation

The imparting of knowledge at the Postgraduates level in universities of Pakistan is not through books but through PowerPoint slides. Instructors are delivering the lectures through slides and generate assessment questions from slides as well. They use an assessment strategy to check their knowledge daily in the form of quizzes. To make questions from every lecture daily is quite a hard task. As automatic question generation tools are available in the literature, there is still a lack of MCQ generators for education subjects, especially in Computer Science. Most of the MCQ generator evaluates general domain questions.

The automatic question generation features benefit online learning twofold. First, as an instructor, it reduces the workload to create a quiz, tutorial, or exercise questions for students to work on. Therefore, the proposed system will help academia greatly where students' assessments can be performed automatically. This allows national universities to streamline their educational activities more efficiently.

1.4 Problem

Education is becoming an essential aspect of daily life. Instructors adhere to a thorough assessment technique to improve their students' abilities and strictly follow the assessment strategy regularly. As a result, it is the primary concern of instructors to make assessment questions daily. Making complete assessments regularly is a time-consuming and labor-intensive task that could greatly benefit from computational assistance. As previously stated, researchers offer useful ways for the automatic production of assessment questions. There are various state-of-the-art studies for automated question bank generation, they are generally content-based strategies that consider rich information (e.g., paragraphs, book chapters, etc.) for question bank generation. However, in today's e-learning or face-to-face educational methods, lectures are typically delivered through PowerPoint slides with limited content or background information. As a result, existing question bank generation systems are difficult to implement in a real-world educational setting where slides are the primary mode of presentation. The current educational situation at the university level is ignored. Therefore, a system is needed that can assist the instructors of postgraduates.

1.5 Proposed Solution

The entire research is carried out in a very systematic manner. The step-by-step research flow is depicted in **Figure 1-2**. The first step is to identify the issue. The optimum remedy for the problem identified in the first phase was then provided. We conducted a thorough literature review to assist us in determining the best solution to the situation. We looked over the study that had been done on our proposed solution, examined it, and compared it. Then we used several tools and strategies to put our framework into action. Experts then validate our proposed framework.

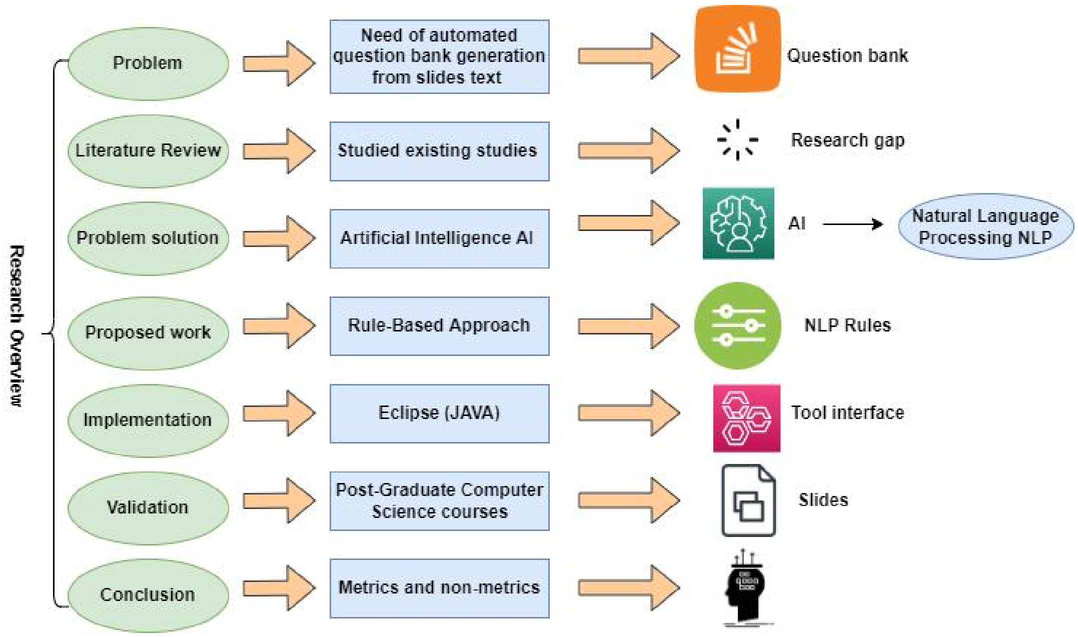

Figure 1- 2-Research Review

Our work presents a novel framework to automatically generate open cloze questions and answers from the lecture slides of software engineering courses at the postgraduate level. The proposed framework employs Artificial Intelligence (AI) techniques like Natural Language Processing (NLP) etc. as shown in **Figure 1-3** to enable the automatic generation of questions bank from PowerPoint slides. Particularly a set of rules are developed to extract the questions and their corresponding answers from the given input lecture slides as a pdf file. Furthermore, a complete algorithm is developed for the execution of rules to generate questions. A sophisticated user interface tool i.e., the QBG tool is also developed. The feasibility of the proposed framework is demonstrated through the lectures taken from the department of computer and software engineering department belonging to the national university of sciences and technology (NUST) Islamabad, Umm ul Qura university Saudia Arabia.

Figure 1- 3-Flow of the proposed methodology

1.6 Thesis Organization

The report is organized as follows:

- **Chapter 1** gives the introduction to the proposed topic, aims, objectives, and motivation.
- **Chapter 2** presents the review of state of art in the context of automatic MCQ question generation techniques.
- **Chapter 3** discusses the proposed Question bank generation tool framework using the NLP technique for Postgraduates levels.
- **Chapter 4** gives the implementation detail.
- **Chapter 5** discussed the experimentation including the setup used for implementation, results obtained, and their discussion.
- **Chapter 6** discussed the discussion and limitations of the proposed work.
- **Chapter 7** concludes the topic by suggesting some future work that is not under the scope of this research but can be implemented in the future.

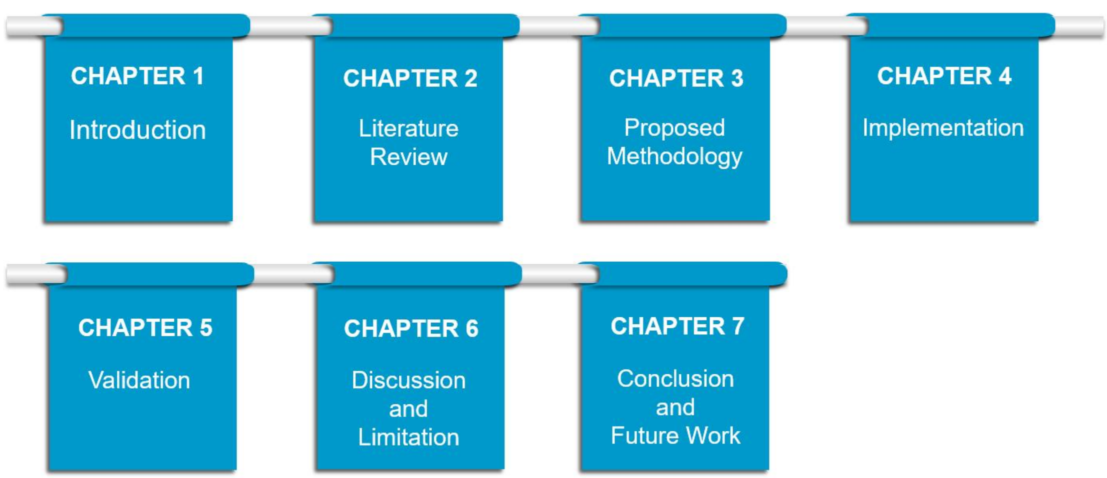

Figure 1- 4-Thesis Outline

Chapter 2

Literature Review

Chapter 2

Literature Review

Automatic question bank generation is an important area of research in the field of natural language processing. In the last decade, researchers have paid high attention to automatic question generation. In this section, a thorough but critical attempt is made to review the existing state of the art and identify its shortcomings. The existing literature that has been discussed below is mostly focused the natural language processing and machine learning techniques. The advantages and drawbacks of existing methodologies using these techniques have also been discussed in detail in this section.

Typically, Question Generation QG can be divided into three distinct categories: syntax-based, semantic-based, and template-based.

1. **Syntax-based Approach**
   The objective of a syntax-based method [11] is to change declarative phrases into interrogative sentences utilizing a variety of transformations. Syntactic elements of the input, such as POS or parse-tree dependency relations, are used to drive question creation via syntax-based approaches. These techniques don't require that you understand the semantics of the input (i.e., entities and their meaning).
2. **Semantic Approach**
   When employing a semantic approach [12], semantic role labelling can be used to determine the semantic parse of a sentence (SRL). Compared to the syntax-based method, this one can offer a more in-depth level of analysis. Additionally, it makes the appropriate adjustments.

3. **Template-Based Approach**: There are no transformation rules in a template-based method [13]. Using pre-made question templates, this technique takes pertinent information out of the text. Using templates, questions are generated. Using predefined text and placeholders that can be filled with values to create questions, templates define the questions' outer structure. Additionally, templates outline the characteristics of the entities (syntactic, semantic, or both) that can take the place of placeholders.
4. **Rule-Based Approach**: The use of rules is used to produce questions. Approaches that use text as input are frequently accompanied by rules. Typically, rule-based systems annotate sentences with syntactic and/or semantic data. They then match the input to a pattern provided in the rules using these annotations. These rules describe how to choose an appropriate question type (for example, picking appropriate wh-words) and how to manipulate the input to create questions (e.g., converting sentences into questions).

5. **Statistical methods**: This is where training data is used to learn how to transform questions. Recently, neural networks [8,13, 48] have been used to automatically create questions from enormous datasets. Question creation, for example, has been treated as a sequence-to-sequence prediction issue in Gao et al. (2018) [50], in which the question generator creates a sequence of text representing a question given a segment of text (typically a sentence) (using the probabilities of co-occurrence that are learned from the training data). Kumar et al. (49) also used training data to estimate which word(s) in the input text will be replaced by a gap (in gap-fill questions).

2.1 Syntax-based Approach

The author in [6] proposed a syntactic-based approach for English grammar question retrieval. They suggested a syntactic-based strategy for retrieving English grammar questions that may effectively retrieve related grammar questions with comparable grammatical focus. They suggest a new syntactic tree, the parse-key tree, to capture the grammatical focus of English grammar problems in the proposed syntactic-based approach. They then presented two kernel functions to compute the similarity between two parse-key trees of the query and grammar questions in the collection, namely relaxed tree kernel and part-of-speech order kernel. Danon et al. [11]

proposed a syntactic-based approach for question generation for a cyber security domain in which the Word2Vec model is used. They proposed a four-component pipeline, which obtains as input a training corpus of domain-specific documents, along with a set of declarative sentences from the same domain and generates as output a set of factoid questions that refer to the source sentences but are slightly different from them, so that a question-answering system or a person can be asked a question that requires a deeper understanding and knowledge than a simple word-matching. BASUKI et al [33] used a syntactical approach to make a question generation system by using textbooks as a dataset. This study used a syntactical approach to offer an Automatic Question Generation (AQG) method for generating Open Domain Indonesian questions. This research incorporates four stages, namely: the identification of declarative sentences for 8 coarse-class and 19 fine-class sentences, the classification of features for coarse-class sentences and the classification rules for fine-class sentences, the identification of question patterns, and the extraction of sentence's components as well as the rule generation of questions.

Table 2- 1-Summary of studies using Syntactical Approaches

Sr.	Author	Techniques Applied	Domain	Result
1.	Danon [11]	Syntactical Approach Word2Vec model	Cyber Security	MCQs
2.	BASUKI [33]	Syntactical Approach	Open Domain Indonesian questions	MCQs
3.	LantingFan [6]	Syntactical Approach	English grammar	MCQs

2.2 Semantic Approach

Many researchers use a semantic-based approach and introduce different types of methodologies according to the intended corpus. Tianlin Zhang et al. [27] use a semantic-based approach in which some rules and NLP technologies (parsing, named entity recognition, etc.) have been used for finding the relation between subject and object. The proposed algorithm is searching for an appropriate word and then deciding question types like what when etc. This technique is specially designed for generating questions in the medical field.

Lekshmi R Pillai [26] combined the approach of Word Sense Disambiguation (WSD) and Semantic Role Labelling (SRL) for generating questions. Text subjected into word sense disambiguation, then into Wordnet Database and then checking roles i.e., actors and participants to make factoid like questions i.e., Who did what? In short Factoid based questions (who, what, when, etc).

Susmita et al. [2] focused on generating diverse questions having the same answers from a single text. The author here also elaborates that literature offers many methods in which answers, and passage needs to be given in proposed algorithms and then they will generate questions corresponding to the answers. But their work overcomes this limitation and without manual intervention, automatic answers and questions are generated. In the proposed system, the sentence is fed into the algorithm (NES) and split into three parts in which each part contains some focused words (keywords), then these three parts are fed into another algorithm, and questions are created based on these focused contents.

General Deepak et al. [29] used the TD-IDF measure first to obtain relevant topics. Then summarized paragraphs by proposed algorithms (find a set of proper nouns, adjectives, adverbs) and then frequent words are filtered out. Sentences are extracted containing these keywords and used for questions. In this technique, pivotal nouns, adjectives, and adverbs are considered blanks.

In [1], Araki et al have presented two different methods that automatically generate questions from multiple sentences for the medical field. The first method requires learners to take specific inference steps such as event and entity coreferences over multiple sentences. The second method generates questions using patterns extracted from the relations between entities and events in a corpus. This technique is applied in the medical field and from the literature it has been found that this is a good technique for biology concepts.

Afzal et al [30] introduced an unsupervised dependency-based semantic relations approach consisting of three main components. The first component used IE methodologies to extract semantic relations and in the second component, automatic question generation occurs using these semantics. Ainuddin Faizan et al. [20] proposed a technique that is choosing named entities from the given input and considering them as blanks. They demonstrated a method for automatically creating multiple-choice questions (MCQs) from slide content. It extracts named entities from slides and searches a knowledge library to generate a variety of MCQs with acceptable answer and alternatives. The author used three decks of slides with the following topics: 1) World War I (33 slides), 2) The Atom (14 slides) and 3) Geography/General knowledge (37 slides).

J. Leo [7] and Stasaski [42] proposed an ontology-based approach for generating questions for the medical field only. The ontology enables the generation of descriptive questions (e.g., "Describe X," which is to list its attributes), discriminative questions (e.g., "What is the difference between X and Y," which is to identify which of their attributes are not shared by them), closed questions (e.g., "Is it true that X has the property Y," and even inverse questions (e.g., "Which properties X does not have") in addition to factual.

Thinh Leet al. [8] proposed a technique in which key concepts are identified from a sentence by using WordNet as a semantic source to generate questions. The author used wiki pages as a dataset. Teo, N. H. I et al. [10] proposed an ontological-based approach for automatic MCQ question generation for the domain of undergraduate operating system courses.

S. S. R. Adithyaet al. [14] used regular expression and POS tagging to generate questions from Wikipedia pages by searching the corresponding articles. Atchariyachanvanich et al. [16] used a database course to generate questions as queries by using a reverse SQL question generation algorithm.

Table 2- 2-Summary of studies using Semantic Approaches

Sr.	Author	Techniques Applied	Domain	Result
1.	Teo, N. H. I et al. [10]	Ontology	Undergraduate operating system course. three textbooks	MCQs
2.	A.Agarwal [41]	Dynamic self-evolving Concept Network technique	8^{th} physics	Questions
3.	Leo [7], Stasaski [42], Susmita et al. [2], Araki [1]	Ontology	Medical education scenario i.e., biology book, gastroenterology and hepatology, cardiology, internal medicine, and Orthopedics.	MCQs
4.	Zhang [27]	Semantic approach (Named entity recognition)	Medical field	Factual question
5.	General Deepak et al. [29]	TD-IDF measures to find nouns, adjectives, and adverbs	Open-domain	Fill in the blanks

2.3 Statistical Methods

Lelkes et al [5] introduced NewsQuizQA and applied the transformer encoder-decoder model for quiz-style question generation tasks. The author generates questions for news stories only. Killawala et al [15] used Neural Network models for the general domain to generate factoid questions, Fill in the blanks, true-false, and MCQs. This technique is computationally overhead due to the pretraining of the model.

Yibo Sun et al [28], and Angelica Willis et al [22] applied a neural network encoder-decoder model to generate multiple questions against pre-identified answers. They used Quora and MARCO datasets in their technique and deals with simple questions. Bang Liu [23] introduced a neural network model technique for generating multiple factoid questions. SQUAD dataset is used for training purposes in this technique.

Wei Yuan [24] used a neural network and proposed a New linguistic technique QAF is invented instead of POS and NER to generate questions. Bok Lee[46] proposed a hierarchical conditional variational autoencoder (HCVAE) for generating QA pairs given unstructured texts as contexts while maximizing the mutual information between generated QA pairs to ensure their consistency. Rajpurkar [47] used a neural network model by using SQUAD datasets. Duan [48] used neural networks and convolutional neural networks to improve existing question answering systems. Zhou et al.[51] proposed a unified model to predict the question type and to generate questions simultaneously by using SQUAD and MARCO datasets.

Table 2- 3-Summary of studies using Statistical Approaches

Sr.	Author	Techniques Applied	Domain	Result
1.	Killawala [15]	Neural Network	General domain	Factoid questions, Fill in the blanks, true-false, MCQs
2.	Yibo Sun [28] Angelica Willis [22]	Neural Network (Encoder decoder model)	Quora MARCO dataset	Generate multiple questions against pre-identified answers
3.	Bang Liu [23]	Neural Network	SQUAD dataset	Generating multiple Factoid questions
4.	Adam D. Lelkes [5]	Neural Network	News Stories	Domain-specific simple questions
5.	Wei Yuan [17]	Neural Network and The proposed new linguistic technique QAF is invented instead of POS and NER	SQUAD and MARCO dataset	Domain-specific simple questions

2.4 Rule-based Approach

In [3] Shitu et al. used a pattern matching technique with a regular expression for automatic question generation purposes in a Bangladesh railway domain. Deena Gnanasekaran [36] introduced rule-based approaches and Natural Language Processing (NLP) techniques for generating questions and input paragraphs selected from the computer science domain.

Das et al [34] generated factual open cloze questions by extracting informative sentences from the input corpus by applying rules over them. This algorithm takes simple sentences only from the corpus which means a sentence having no more than one independent clause and no dependent clause. In short, complex sentences are skipped by the algorithm. News reports on Cricket matches were taken by the system as input and produced factual questions as output.

Manish Agarwal [18] presented an automatic open cloze question generation (OCQG) system. This approach consisted of two steps. In the first step, relevant and informative sentences were selected, and keywords were identified in the selected sentences in the second step. News reports on Cricket matches were taken by the system as input and produced factual OCQs as output.

Pedro Azevedo [32] introduced a technique for the generation of factoid questions. POS and NER techniques are used to select a sentence and then match some rules to generate factoid questions from them. Their system uses Named Entity Recognition to extract entities from the text (NER). POS tagging can also be used to find patterns in sentences and extract information that could be questioned. Regular expressions are used to match specific patterns and generate questions that follow a set of rules.

Table-2-4 shows the comparison of the techniques that are using rule-based approaches for the automatic generation of questions.

Table 2- 4-Summary of studies using Rule-based Approaches

Sr.	Author	Techniques Applied	Input	Output
1.	Tanzim Tamanna Shitu [7]	Pattern Regular Expressions	Domain-specific i.e., Bangladesh Railway	Factoid Question
2.	Manish Agarwal and Prashanth Mannem [18]	Sentence selection by regular expression	Domain-specific i.e., Biology book	Fill in the blanks
3.	Pedro Azevedo [32]	Application of POS and NER to select a sentence and then applied some rules	Open-domain i.e., Wiki documents Controlled documents, Entity questions w/o ORG Entity questions Dependency questions	Factoid Question
4.	Bidyut Das [34]	Rules for key generation and then generate fill in the blanks	Eight Wikipedia pages namely, Amitabh Bachchan, Ramayana, Mahabharat, Yoga, Internet, India, Sachin Tendulkar, Bengal Tiger	They search (dependency parsing) only simple sentences and make questions from them.
5.	Deena Gnanasekaran [36]	Rule-based approach	Computer Science domain	Procedural questions

Tanzim [7] used pattern regular expressions to make factoid questions for the railway information dataset. This approach is not beneficial for education. Manish selected the sentences for questions by using regular expressions for creating questions for the medical field. Azevedo is generating factoid questions for the non-educational domain. In [20], sentences with one clause are fed into the proposed algorithm to make sentences. On the other hand, Deena [36] is generating automatic questions for the computer science domain. It is seen that the rule-based approach is used for non-educational domains by the majority of the researchers. In our work, we employ this approach for slides contents that will benefit educational institutions at the post postgraduate level.

2.4 Research Gaps

Although there exist several state-of-the-art studies for the automated generation of questions bank, these are mostly content-based techniques where rich contents (e.g., paragraphs, book chapters, etc.) are considered for question bank generation. Books are used mostly in lower education levels i.e., secondary, and intermediate levels of education. The mode of education at a higher level is lecture slides and existing literature do not use educational dataset at a postgraduate level where the mode of delivering the lecture slides. Therefore, the real-time educational scenario at a higher level has not catered yet. Datasets used in existing research are either from different books of intermediate level, some Wikipedia pages, and some other standard datasets i.e., SQUAD is highly used by researchers available on different research websites. SQUAD is used for statistical approaches. Existing methodologies and datasets used in them have some drawbacks given below:

- The datasets taken from the books that are being used by researchers in different techniques are beneficial for education but still not beneficial for real-time educational scenarios where instructors deliver lectures by using PowerPoint slides and students also use these slides for self-learning or preparing for the exam.
- There isn't a single paper except [20] in the literature that discusses question generating using PowerPoint slides. Researchers and practitioners did not include slides as input to their proposed question-making algorithms.

Table 2-5 shows the approach that is using PowerPoint slides for generating questions. This approach is only finding named entities from the slides and asking a question. Named entities here include nouns i.e.., the name of a person, place or a thing. Therefore, it can be concluded that methodologies designed to extract questions from such types of datasets don’t give beneficial results for education.

Table 2- 5-Approaches using PowerPoint slides as input

Sr.	Author	Techniques Applied	Domain	Result
1.	Faizan [20]	Used semantic annotation tool for selecting named entities	World War I, The Atom, Geography/General knowledge	MCQs

2.5 Contributions:

Based on the extensive review of the literature and to overcome all the above-mentioned shortcomings and drawbacks, this research aims to use a dataset containing lectures of the software engineering courses that consist of PowerPoint slides. The lectures are taken from the department of computer and software engineering department belonging to the national university of sciences and technology (NUST) Islamabad, Saudi electronics university. Text is extracted from the lectures taken as a dataset for this research. Then proposed rules base methodology is applied to the extracted text. When rules are applied to the given input, a file of automated questions bank generation is produced as a result containing suitable assessment questions in it. The rule-based approach is best suited for the text extracting from the intended lectures as compared to other benchmark artificial intelligence and machine learning techniques e.g., syntax-based approach and template-based approach in which neural network models are highly used. Unlike other deep learning models, it does not need a large dataset for training purposes. That is the reason, this approach consumed less complexity of the machine. Expert opinion is used for the validation of the generated results.

Chapter 3

Methodology

Chapter 3

Methodology

A novel and efficient methodology is proposed by utilizing machine learning techniques to aid in the daily basis assessment procedure at the postgraduate level. The proposed methodology used natural language processing NLP for the generation of an automated question bank. A high-level view of the proposed methodology is shown in **Figure-3-1**. In this methodology, NLP techniques and some rules have been made to generate the questions automatically. Rules are composed by using regular expressions.

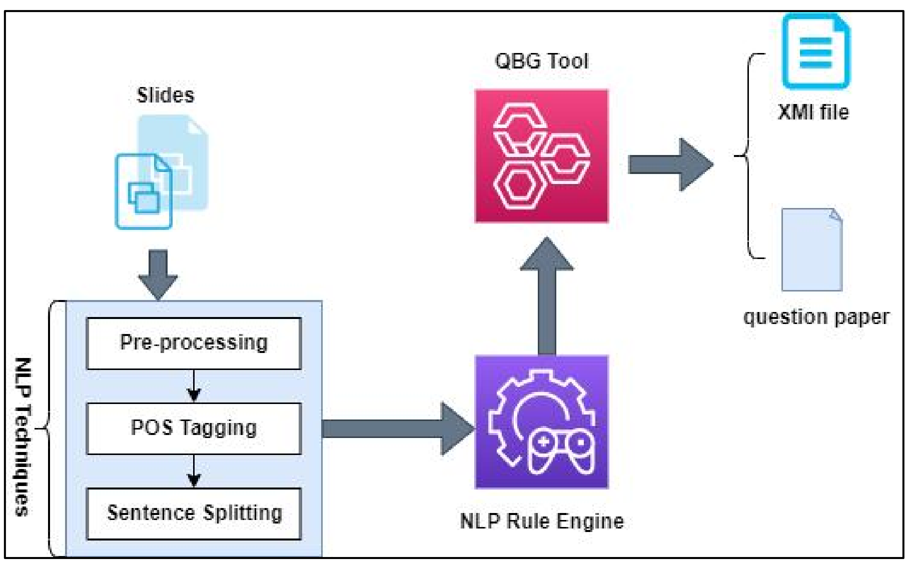

Figure 3- 1-High Level view of Proposed Framework

In this methodology, the raw slides are first pre-processed to straighten out semantic aberrations in the raw data. This is followed by Parts-of-speech tagging to segregate and tag specific structural elements. After sentence splitting, we use a rule-based approach to generate the questions automatically. As part of our proposed methodology, we have created a comprehensive set of rules based on regular expressions for effective question generation. Furthermore, A complete framework NLQBG is designed for the application of proposed rules in the proposed approach. Question bank generation tool (QBG) is generated as a graphical user interface GUI for the demonstration of the proposed approach.

3.1 Proposed Algorithm

The proposed algorithm consists of several steps to automate the generation of questions from an input text. The steps of the algorithm are shown in **Figure-3-2.**

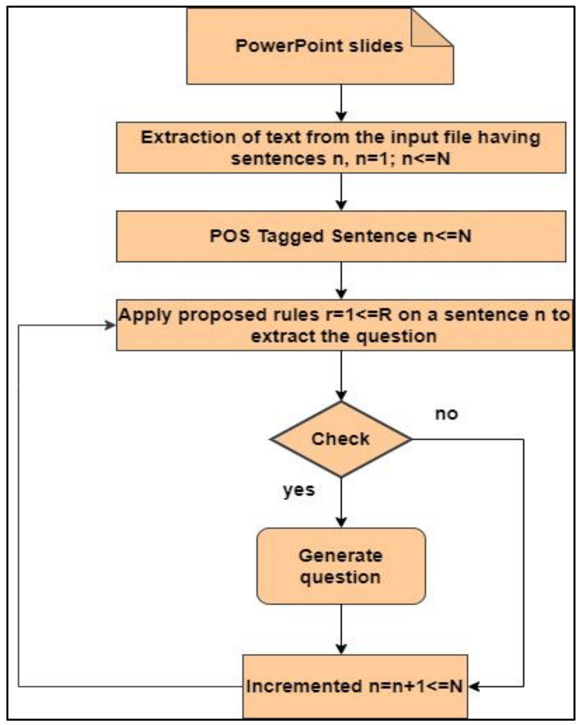

Figure 3-2 Steps of Proposed Framework

Before the implementation of the proposed rules to the text. The input text first needs to be passed on some steps. Therefore, the whole process is carried out in the following steps.

1. Preprocessing
2. Parts of speech tagging (POS)
3. Sentence Spitting
4. Proposed Rules

3.1.1 Preprocessing

Text is extracted from the input pdf file of a lecture slide. The initial state of the extracted text is rough and not capable to pass into the rules. Therefore, the cleaning process is done over it in which punctuation of the text string is carefully catered. Extra punctuation marks are removed to clean the text, and some are placed to get the text in a format. As in PowerPoint slides full stop "." mark is often not present by the completion of every sentence. But in the proposed technique, every sentence is separately needed so that rules are applied to it efficiently.

3.1.2 Parts of speech tagging (POS)

The important part of the proposed technique is POS tagging which is done by a POS tagger library. A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads the text in some language and assigns parts of speech to each word (and another token), such as noun, verb, adjective, etc. (shown in table 3-1), although generally computational applications use more fine-grained POS tags like 'noun-plural'. A list of POS tagged words abbreviation is given in **Table 3-1.** Maxent dictionary of Stanford POS tagger is used in the proposed technique to place tags on every single word of the input string.

Table 3- 1-List of POS tags

Sr.#	Tag	Description	Examples
1	NN	Noun in singular form	diagram
2	NNP	Proper noun in singular form	UML
3	NNS	Noun in plural form	classes
4	VBZ	Verb, 3rd person singular, present	Simplifies, shows
5	VBN	Verb, Past Participle	Related, applied, tagged
6	VBP	Verb, non 3rd person, singular present form	Are, allow
7	CC	Coordinating conjunction	And, but, or
8	MD	modal	Must, will, may, should, can, cannot
9	IN	Preposition/Subordinating conjunction	With, for, during
10	DT	Determiner	The, a, an, some
11	CD	cardinal number	One, two, three
12	HYPH	Hyphen, special character	"-"
13	LRB/ RRB	special character	"(" , ")"
14	JJ	Adjective	appropriate
15	WRB	Wh-adverb	when
16	RB	adverb	together
17	VB	Verb, base form	Need, be
18	WDT	Wh-adverb	that
19	PRP	Pocessive pronoun	we

3.1.3 Sentence Splitting

After POS tagging on the whole text string, it splits into single sentences. The string is converted into a string array by using the split function. The string array is ready to be fed into the proposed rules. Sentence splitting is needed before the text is fed into the proposed algorithm because a rule-based approach is used, and every sentence is separately needed. Every single sentence is one by one fed into the algorithm to check the synchronization with the proposed rule.

3.1.4 Proposed NLP Rules

This section comprises a set of NLP rules that cover mapping of the sentences in natural language to question generation. To extract questions, these rules are to be realized over the text extracted by the lectures of the postgraduate courses in natural language with the help of NLP techniques.

Rules are comprised of regular expressions. These regular expressions are then applied to the text using the string-matching technique supported by the Regular Expression Library. The basic purpose of rules is to match each input sentence with them to see the pattern of the sentences and select the sentences from which MCQs can be generated. These rules will be applied one by one to every sentence of the text string to check the synchronization of the rules with the pattern of the sentences. The pattern is the combination of some (Parts of Speech) POS tags. If the pattern of a sentence would match the pattern of any one of the rules, then that sentence is selected for generating the question. When a sentence is selected, then a question is generated according to the format.

There are ten rules proposed for the extraction of questions and answers from a given input pdf file. The result from the study shows that as the complexity of a sentence increases the number of syntactical rules to be considered, therefore, for achieving maximum accuracy more rules to be included [54]. The proposed rules with their description, regular expression, examples are as follows:

1) **Rule using the pattern of proper noun and verb combination**

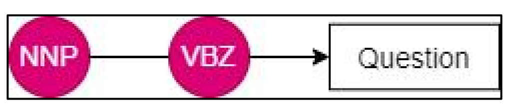

Figure 3- 3-Graphical representation of Rule1

Regular Expression: “\\w+NNP\\b.\\w+VBZ\\b”

Description: A noun with its POS value NNP **and** VBZ refers to the verb.

This regular expression matches the pattern of string-like NNP proceeded by verb VBZ. If in a sentence combination of NNP and VBZ come, then that sentence is selected for MCQ.

POS Tagging:

UML_NNP is_VBZ a_DT general_JJ purpose_NN notation_NN may_MD limit_VB its_PRP$ suitability_NN for_IN modelling_VBG some_DT particular_JJ domains_NNS.
Applying_VBG Constraint_NN in_IN Profile_NNP OCL_NNP Editor_NNP opens_VBZ within_IN Constraint
The above sentence has been selected for generating questions by rule1.

Extraction of right answer:"\\w+NNP\\b.*?(?=\\w+VBZ\\b)"
Regular expression extracts the key option for the answer and separates the question statement. The further format will place the blank and make an MCQ question.

Extraction-Example:

Question: ________________is a general-purpose notation may limit its suitability for modelling some particular domains?

Key option: **UML**

Question: Applying Constraint in ....opens within Constraint?

Key option: Profile OCL Editor

---

2) Rule using the Pattern of consecutive two singular nouns and verb combination

Regular Expression: \\w+NN\\b.\\w+NN\\b.(?=\\w+VBZ\\b|\\w+VBP\\b)

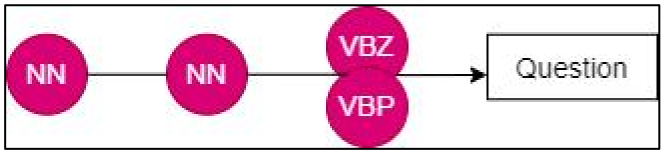

Figure 3- 4-Graphical representation of Rule2

Description: Two consecutive nouns with POS value NN proceeded by the verb with POS value VBZ or VBP.

The regular expression matches the pattern of a string with two NN proceeded by verb VBZ. If in a sentence combination of these words come, then that sentence is selected for MCQ.

POS Tagging:

When_WRB using_VBG a_DT façade_NN controller_NN leads_VBZ to_IN low_JJ cohesion_NN and_CC high_JJ coupling_NN**.**

Extraction of right answer:"\\w+NN\\b.*?(?=\\w+VBZ\\b)"

Regular expression extracts the key option for the answer and separates the question statement. The further format will place the blank and make an MCQ question.

Extraction-Example:

When does use a .... lead to low cohesion and high coupling?

Key option: façade controller

---

3) Rule using the pattern of singular noun and verb combination

Regular Expression: "\\w+NN\\b.(?=\\w+VBZ\\b|\\w+VBP\\b)"

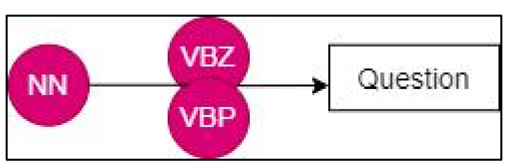

Figure 3- 5-Graphical representation of Rule3

Description: Noun with its POS value NN **and** VBZ| VBP refers to the verb.
This regular expression matches the pattern of a string with noun NN proceeded by verb VBZ and VBP. If in a sentence combination of NN and VBZ come, then that sentence is selected for MCQ.

POS Tagging:

When_WRB a_DT profile_NN is_VBZ applied_VBN,_, instances_NNS of_IN the_DT appropriate_JJ stereotypes_NNS should_MD be_VB created_VBN for_IN those_DT elements_NNS that_WDT are_VBP instances_NNS of_IN metaclasses_NNS with_IN required_VBN extensions_NNS
This sentence has been selected for generating questions referred to by rule 3.

Extraction of right answer:"\\w+NN\\b.(?=\\w+VBZ\\b|\\w+VBP\\b|\\w+VBN\\b)"

This regular expression extracts the key option for the answer and separates the question statement. The further format will place the blank and make an MCQ question.

Extraction-Example:

Question 1: When a ....is applied, instances of the appropriate stereotypes should be created for those elements that are instances of meta classes with required extensions?

4) Rule using the pattern of a proper noun and round bracket combination

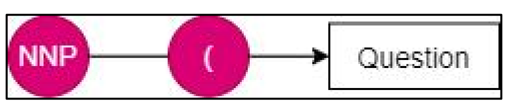

Figure 3- 6-Graphical representation of Rule4

Regular Expression:"\\w+NNP\\b.\\ ("

Description: Noun with its POS value NNP **and** (refers to the punctuation mark "round open bracket".

This regular expression matches the pattern of string-like NNP proceeded by verb (. If in a sentence round bracket comes after a noun with POS value NNP, then that sentence is selected for MCQ. Brackets here include the abbreviation of NNP that proceed by bracket. So, the question can be asked here about the abbreviation or full form of that noun NNP.

POS Tagging:

Building_NNP UML_NNP Models_NNPS with_IN OCL_NNP (_-Object_NN Constraint_NNP Language_NNP)_-RRB-

This sentence has been selected for generating questions.

Extraction of right answer:"(? <=\\().*?(?=\\))"

This regular expression extracts the key option for the answer and separates the question statement. The further format will place the blank and make an MCQ question.

Extraction-Example:

Question: what is the abbreviation of OCL?

Key option: Object Constraint Language

---

5) Rule using the pattern of plural noun and verb combination

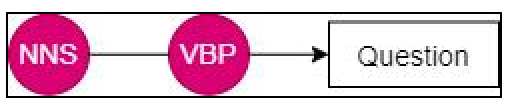

Figure 3- 7-Graphical representation of Rule5

Regular Expression:

" \\b(\\w+NNS\\b|\\w+VBP\\b) (?:\\W+\\w+){0,0}?\\W+(\\w+NNS\\b|\\w+VBP\\b)\\b"

Description: Noun with its POS value NNS **and** VBP refers to the verb.
This regular expression matches the pattern of string in which consecutive nouns with POS value NNS proceeded by verb VBP is present. If a sentence combination of NNS and VBP comes consecutively, then that sentence is selected for MCQ.

POS Tagging:

Because_IN **modeling_NN languages_NNS** do_VBP not_RB have_VB to_TO be_VB text_NN based_VBN,_, and_CC often_RB are_VBP n't_RB (_-LRB- they_PRP can_MD,_, for_IN example_NN,_, have_VBP a_DT graphical_JJ syntax_NN,_, like_IN UML_NNP)_-RRB-,_, we_PRP will_MD need_VB a_DT different_JJ mechanism_NN for_IN defining_VBG languages_NNS in_IN the_DT MDA_NNP context_NN ._.
The above sentence has been selected for generating the questions.

Extraction of right answer:

"\\w+NN\\b.?\\w+NNS\\b.(?=\\w+VBZ\\b|\\w+VBP\\b|\\w+VBN\\b)"
"\\w+NN\\b.?\\ here means if any noun with POS value NN may or not present before NNS can be considered. This regular expression extracts the key option for the answer and separates the question statement. The key option here is NNS. The further format will place the blank and make an MCQ question.

Extraction-Example:

Question1: Because ....do not have to be text-based, and often are not (they can, for example, have a graphical syntax, like UML), we will need a different mechanism for defining languages in the MDA context?
Key option: Modelling Languages
Question2: In the pattern scope hierarchy, ...are the medium scale patterns?
Key option: Design Pattern

6) **Rule using the pattern of a proper noun, singular noun, and verb combination**

Figure 3- 8-Graphical representation of Rule6

Regular Expression:"\\w+NNP\\b.\\w+NN\\b.\\w+VBZ\\b"

Description: Noun with its POS value NNP, noun with its POS value NN **and** VBZ refers to the verb.

This regular expression matches the pattern of string-like NNP proceeded by NN which is proceeded by the verb VBZ. If in a sentence combination of NNP NN and VBZ come, then that sentence is selected for MCQ.

POS Tagging:

View_NNP architecture_NN is_VBZ popular_JJ choice_NN for_IN documenting_VBG the_DT architecture_NN

The above sentence has been selected for generating the questions.

Extraction of right answer:"\\w+NNP\\b.*?(?=\\w+VBZ\\b)"

This regular expression extracts the key option for the answer and separates the question statement. The key option here is NNP and NN. Further formatting steps will place the blank and make an MCQ question.

Extraction-Example:

Question: ....is a popular choice for documenting the architecture?

Key option: View architecture

---

7) Rule using the pattern of a plural noun, conjunction and plural noun combination

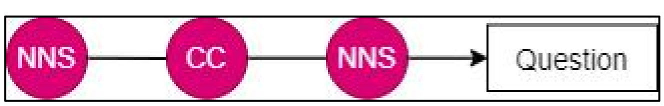

Figure 3- 9-Graphical representation of Rule7

Regular Expression:"\\w+NNS\\b.\\w+CC\\b.\\w+NNS\\b"

Description: Noun with its POS value NNS, and conjunction with POS value CC and VBZ refers to the verb.

This regular expression matches the pattern of a sentence in a string in which two nouns with POS value NNS are joined by a conjunction CC. If in a sentence combination of the above pattern is found, then that sentence is selected for MCQ.

POS Tagging:

As_IN with_IN parts_NNS,_, properties_NNS can_MD be_VB connected_VBN to_IN other_JJ **properties_NNS** or_CC parts_NNS using_VBG connectors_NNS

The above sentence has been selected for generating the question.

Extraction of right answer:"\\w+NNS\\b.?(?=\\w+CC\\b)"

This regular expression extracts the key option for the answer and separates the question statement. The further format will place the blank and make an MCQ question.

Extraction-Example:

Question: As with parts, properties can be connected to other ....or parts using connectors?

Key option: properties

8) **Rule using the pattern of plural noun, modal, verb base form and verb combination**

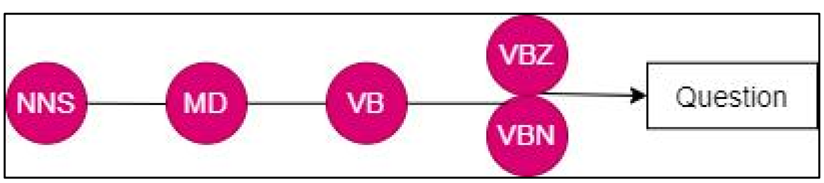

Figure 3- 10-Graphical representation of Rule 8

Regular Expression:"\\w+NNS\\b.\\w+MD\\b.\\w+VB\\b.\\w+VBN\\b|VBZ\\b"

Description: Noun with its POS value NNS, MD refers to prepositions **and** VBZ|VBN refers to the verb.

The regular expression matches the pattern of string-like NNS proceeded by verb VBZ with the help of MD and VB prepositions. If a sentence combination of NNS and VBZ comes in the pattern mentioned above, then that sentence is selected for MCQ.

POS Tagging:

Requirements_NNS should_MD be_VB ranked_VBN according_VBG to_IN importance_NN and_CC change_NN

The above sentence has been selected for generating questions.

Extraction of right answer:"\\w+NNS\\b.?(?=\\w+MD\\b)"

This regular expression extracts the key option for the answer and separates the question statement. The further format will place the blank and make an MCQ question.

Extraction-Example:

Question: ....should be ranked according to importance and change?

Key option: Requirements

---

9) **Rule using the pattern of consecutive two singular noun, plural noun, and verb combination**

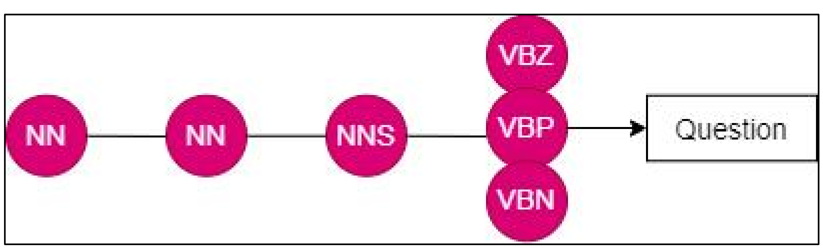

Figure 3- 11-Graphical representation of Rule 9

Regular Expression:

“\\w+NN\\b.\\w+NN\\b.\\w+NNS\\b.(?=\\w+VBZ\\b|\\w+VBP\\b|\\w+VBN\\b)”

Description: Noun with its POS value NN **and** VBZ|VBP|VBN refers to the verb.

The regular expression matches the pattern of string-like two consecutive nouns NN proceeded by verb VBZ|VBP|VBN. If in a sentence combination of NN, NN and VBZ come, then that sentence is selected for MCQ.

Input Example:

When several target pattern elements are specified, they must be separated by commas (", ").

POS Tagging:

When_WRB several_JJ **target_NN pattern_NN elements_NNS** are_VBP specified_VBN,_, they_PRP must_MD be_VB separated_VBN by_IN comas_NNS.

The above sentence has been selected for generating the questions.

Extraction of right answer:"\\w+NN\\b.*?(?=\\w+VBZ\\b|\\w+VBP\\b|\\w+VBN\\b)"

This regular expression extracts the key option for the answer and separates the question statement. The further format will place the blank and make an MCQ question.

Extraction-Example:

Question: When several ....are specified, they must be separated by commas (", ")?

Key option: target pattern elements

---

10) Rule using the pattern of noun and hyphen combination

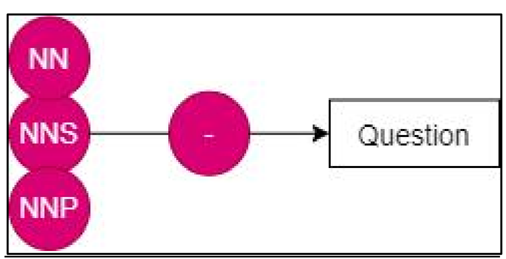

Figure 3- 12-Graphical representation of Rule10

Regular Expression:"(?=\\w+NN\\b|\\w+NNS\\b|\\w+NNP\\b).\\–"

Description: Noun with its POS value NN, NNS **and** NNP. “|” sign represents “or”. It means presence of either NN or NNS or NNP.

This regular expression finds the noun with POS value NN or NNS or NNP proceeded by the hyphen sign “-“. It is because nouns proceeded by hyphen sign are defining themselves in the text so here term name can be asked from that sentence that is referring proceeded line. If such a pattern is found in a sentence, then that is selected for the MCQ.

POS Tagging:

Weak_JJ conflict_NN –_HYPH statements_NNS not_RB satisfiable_JJ together_RB under_IN some_DT boundary_NN condition_NN

The above sentence has been selected for generating the questions.

Extraction of Right Answer:".*?(?=\\–)"

This regular expression extracts the key option for the answer and separates the question statement. The further format will place the blank and make an MCQ question.

Extraction-Example:

Question: ....refers to– statements not satisfiable together under some boundary condition.

Key option: Weak Conflict

QBG Engine includes all of these rules implemented on an individual sentence of the text in order to extract the questions.

3.1.5 Application Scenarios

Rules will be applied to the sentences of the input in two scenarios which are explained below.

Steps of the algorithm: Steps of the algorithm are as follows which include the application of proposed rules to the sentences. Rules will be applied in two scenarios.

Scenario I- Single question generation from a single sentence

Unlike scenario I where all of the rules from r=1 to R are applied to a single sentence, only one rule is checked on a single sentence shown in **Figure 3-13 (a).** For example, if a pattern of any one rule matches with a pattern of a sentence, then a question is generated from that sentence and here iteration of the loop is terminated, and n is incremented by 1 here. then next iteration executes to check the pattern of the next sentence with rules and so on.

The first sentence of the text is set to n=1 to generate the question from a given sentence. In an iteration, all rules are applied sequentially to a sentence of the text. If the pattern of any one of the rules matches with a pattern of sentence n=1, then a question is generated from that sentence.

When the rules are checked on a single sentence in an iteration, the following are the states that occur.

1. A question will be generated if any one of the rules applies to a given sentence n=1.
2. No question will be generated.

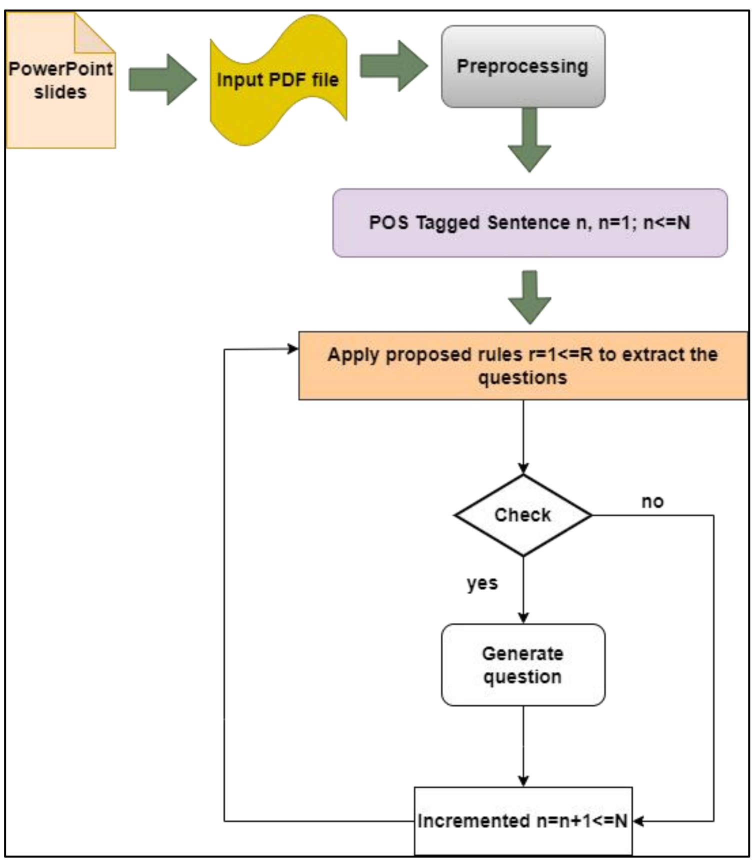

Figure 3-13 (b)- Scenario I- Single question generation from a single sentence

Then, the value of n is incremented by 1 i.e., n=1+1, and the next sentence will be checked on the rules and the whole algorithm is applied to all of the sentences up to N.

Scenario II-Multiple question generation from a single sentence

The purpose of this scenario is to check either more than one rule can be applied to a single sentence and to extract more than one question from a single sentence shown in **Figure 3-13 (a).** Therefore, every proposed rule is applied to every sentence in different iterations. Following is the whole process:

The first sentence of the text is set to n=1 to generate the question from a given sentence. Similarly, the rule is assigned a number r and the total number of proposed rules is R. In an iteration, all rules are applied sequentially to a sentence of the text. If a pattern of a rule matches with a pattern of a sentence, then a question is generated from that sentence and then the next rule is checked on that sentence.

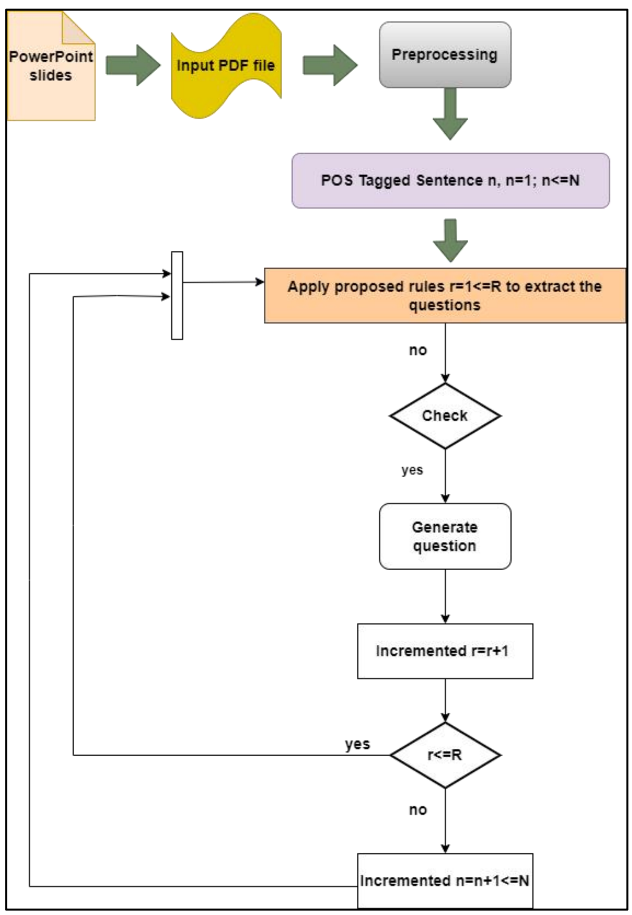

Figure 3- 13 (b)- Scenario I-Multiple question generation from a single sentence

For example, all the proposed rules from r=1 to R will be applied to the sentence n=1 in a single iteration and goes up to the sentence N going through multiple iterations.

When the rules are checked on a single sentence in an iteration, the following are the states that occur.

10) Single question will be generated if the single rule applies to a given sentence n=1.
11) More than one question will be generated if more than one rule applies to a given sentence n=1.
12) No question will be generated.

In an iteration value of n will be the same. So, when all rules are checked in a single iteration on a sentence and the aforementioned states have been found then n will be incremented to 1 i.e., n=1+1=2. It means the second sentence n=2 is now passed from the steps of the whole algorithm. It will iterate the steps of the algorithm up to sentence N.

The **Figure 3-3** demonstrates the complete flow of the proposed algorithm. The results are stored in the XML file and a text file. The text files can be downloaded into the device to conduct the assessment.

Chapter 4

Implementation

Chapter 4

IMPLEMENTATION

This chapter provides the implementation details of our proposed framework. Section 4.1 gives the detail of the tools and languages used. The proposed QBG tool interface along with a description is presented in Section 4.2. Section 4.3 discusses the question generation in detail.

4.1 Tools and Languages

Based on the proposed algorithm, a complete (Question Bank Generation) QBG Tool is implemented in Eclipse IDE [37] using JAVA language. The input of QBG is a PDF file of the lecture slide. The output of QBG is an XML file of generated questions.
The process is summarized as follows:

- QBG receives lecture slide documents as input, and text is retrieved from the papers.
- Text is given a light pre-processing to put it in the right format.
- The OpenNLP Maxent tagger library is used to identify noun-verb conjugations, prepositions, and adjectives.
- Regex (Regular expression library) made it easier to match each word's pertinent tags, and then to match the resulting string with the suggested rules for extracting MCQ questions.
- By using the suggested rules, the questions are located and saved in the XMI file and a downloadable text file.

There is no need for manual intervention because all QBG procedures are totally automated.
Figure 4-1 depicts the Eclipse platform's user interface. The proposed methodology is depicted in **Figure 3-1** as a high-level view. The processes of the QBG tool are summarized in this diagram. Some preprocessing techniques, including a few state-of-the-art NLP techniques, were adapted from previous publications. **Figure 3-1** depicts the proposed methodology by representing the system's inputs and outputs. In the following sections, we'll go through each phase of the procedure in greater depth.

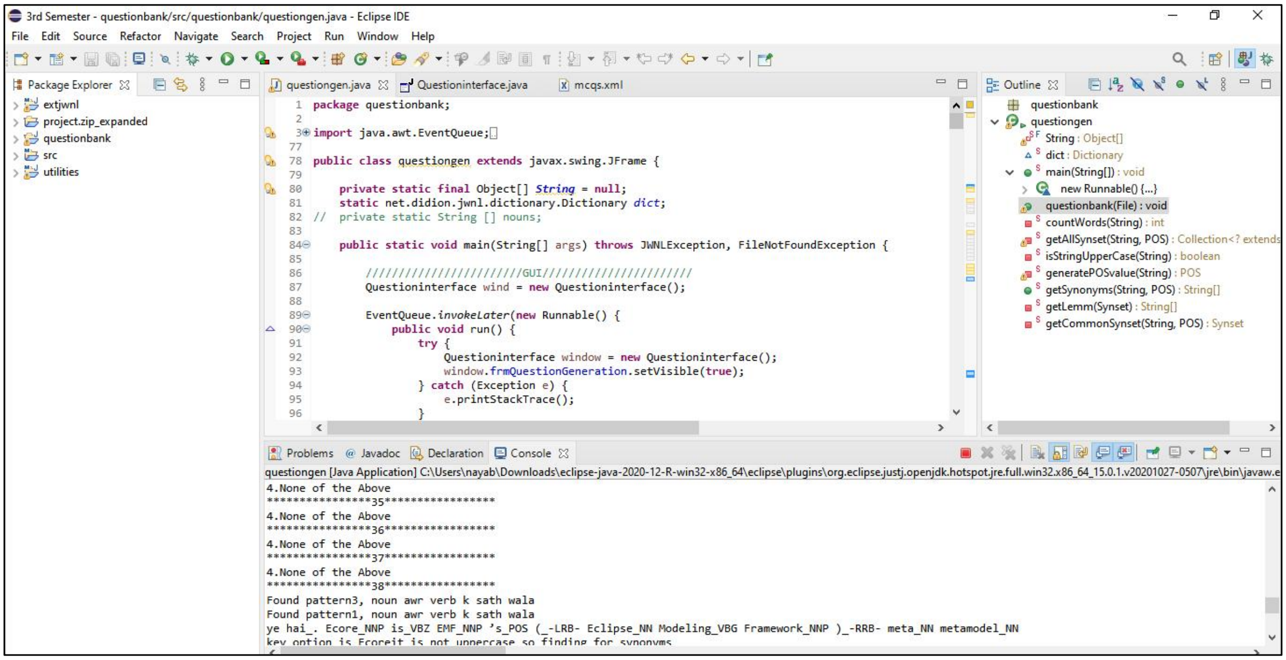

Figure 4- 1-Interface of the Eclipse platform

4.2 Tool Interface

Figure 4-2 depicts the main interface of the QBG tool. As input, the tool uses a pdf file containing the intended text to produce questions. Then, as demonstrated in the tool via menu buttons, it has many functionalities. The output of menu buttons is displayed in the text field labeled "output" below. By just pressing the option in the menu, the output can be downloaded to the user's device.

The functions of buttons are briefly described below:

Extracted text

Text that is extracted from the input pdf file can be obtained by clicking this button.

POS Tagging

POS tags applied by the maxent tagger library on the extracted text can be seen by clicking this button.

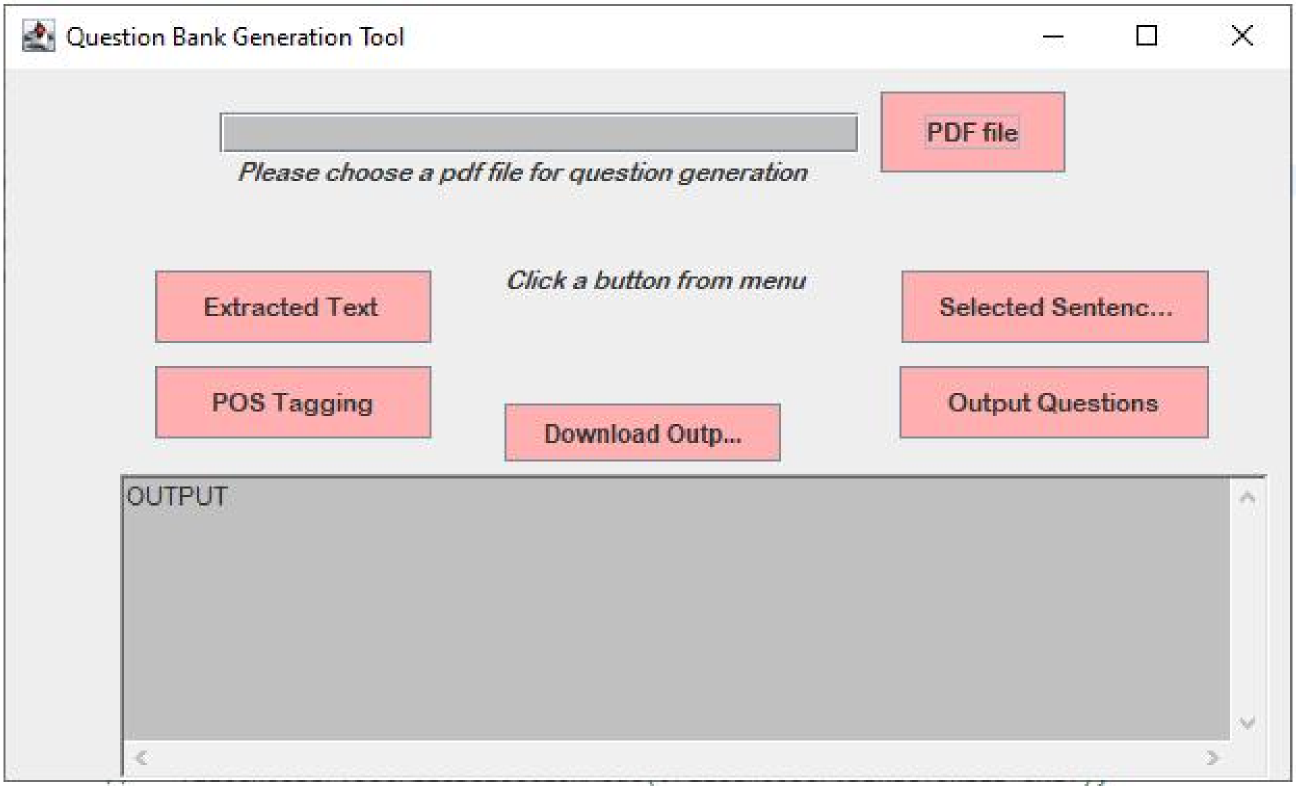

Figure 4- 2-Interface of QBG TOOL

Selected Sentences

The text or sentences selected by applying proposed NLP rules for the extraction of questions can be seen by clicking this button.

Output Questions

This is the main button through which extracted questions from a given input file can be obtained and the result is shown on the output area given on the tool.

Download Output

When this button is clicked, the output present on the output area is downloaded to the location of your choice in your device i.e., laptop, or computer.

4.3 Questions Generation Detail

The basic steps required to operate the QBG tool have already been discussed at the start of section 4. So, these steps must be understood to generate the questions and answers from the lectures. QBG tool has to input a pdf file to extract the questions from that file. By clicking the pdf file button, it will ask the file you need to upload from the device. When a file is successfully uploaded into the QBG tool, the output can be taken from the menu buttons according to the choice. But the sequence of buttons tells the functionality provided by the QBG tool. Although the main desired output of the QBG tool is a file containing questions. But other intermediate outputs can be seen by using menu buttons from the tool to understand the whole functionality.

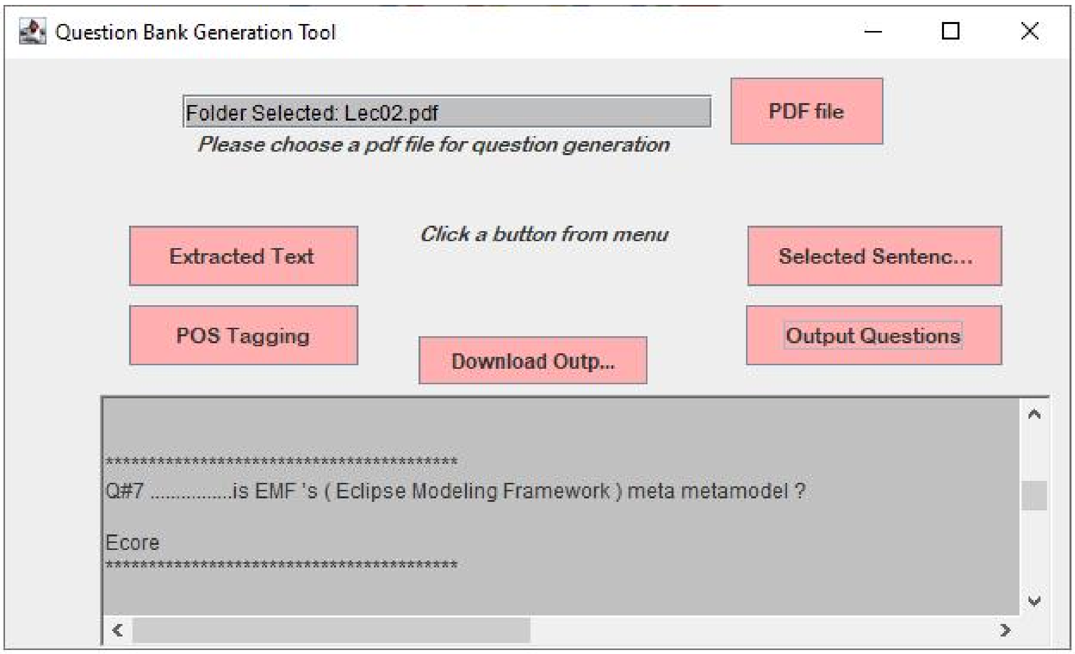

Figure 4- 3-Output of QBG tool comprising questions

Each button's functionality has already been discussed. Because the name of the button suggested its functionality, the output can be seen on the tool's output area by clicking the button sequentially according to the desired output. Furthermore, by clicking the "download output" button, the output can be downloaded into the device. Questions that have been generated by the tool are obtained by clicking the button “Output questions”. **Figure 4-3** shows the generated questions and answers in the output area of the tool.

Now, these are the questions that are generated by the tool from the uploaded pdf file. This is the desired output of the NLP rules. Questions in the output area of the tool shown in **Figure 4-3** can be downloaded into the device for future use. In the menu buttons of the QBG tool, there is a button “Download output” is used for downloading the file comprising the questions and their corresponding answers.

Figure 4-4 shows the method how downloading the file. It clearly illustrates that by clicking the button "Download output", it will ask the desired location of the device at which you want to save the output questions paper file. The extension of the file is .txt. You can keep the file name according to your choice. **Figure 4-4** shows the downloaded file comprising questions and answers. Instructors can use these files or question papers as assessment items while taking assessments of the students.

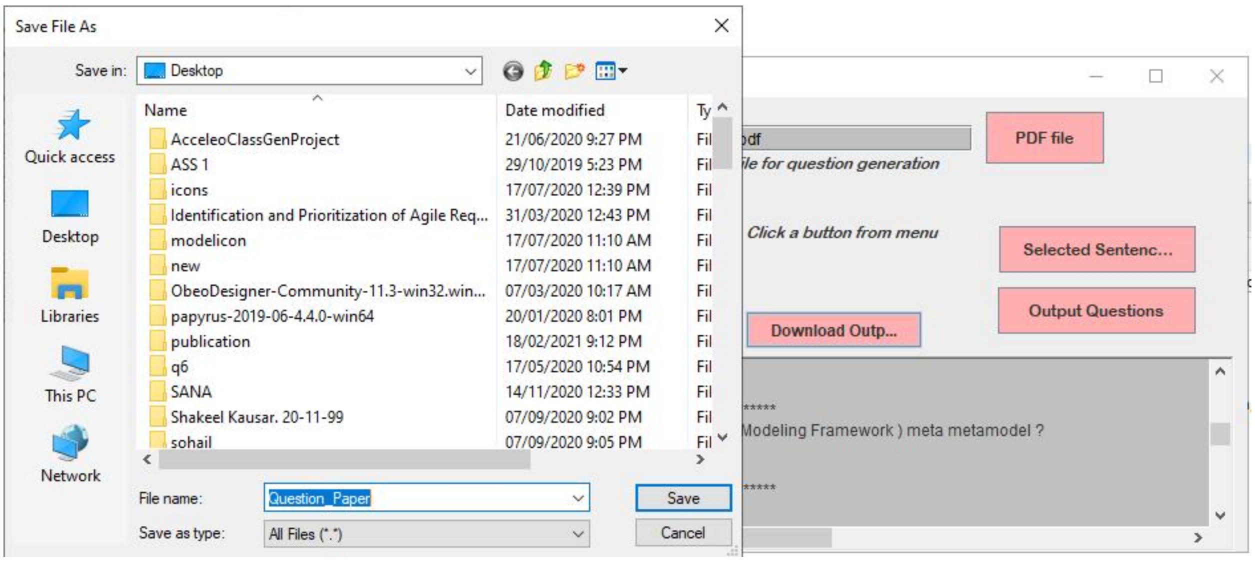

Figure 4- 4-Downloading the generated questions as a file

Chapter 5

Validation

Chapter 5

Validation

The applicability of the proposed framework is presented in this section with the help of courses from computer and software engineering departments as case studies. Section **5.1** discusses the datasets used as case studies. Section **5.2** discusses the results obtained by the QBG tool for different courses. Results of scenario 1 and scenario II are in detail discussed in sections **5.2.1** and **5.2.2**.

5.1 Datasets

The dataset for a question bank generation is taken from the computer and software engineering departments of different universities. The dataset includes the lecture slides of different courses in the software engineering field. Lectures of Postgraduates software engineering courses i.e.,

1. Model-Driven Software Engineering MDSE (NUST National university of sciences and technology, Pakistan)
2. Software System Design and Architecture SSDA (NUST National university of sciences and technology, Pakistan)
3. Software Requirement Engineering SRE (NUST National university of sciences and technology, Pakistan)
4. Databases (umm ul Qura university of Saudia Arabia)

PowerPoint slides from lectures from the aforementioned courses are included in the datasets. Lectures from these courses are used to test the efficiency of the proposed question bank creation mechanism in the proposed system. The results are then evaluated by specialists from these courses to verify and validate the proposed system's functionality.

It is to be noted that these lectures include PowerPoint slides, and they are in the form of pdf text files. The proposed algorithm first extracts the text from the slides and then further steps are applied to the text. The text in the lectures is not initially capable of being fed directly into the algorithm to produce questions. It needs to be preprocessed very carefully to get it ready to generate questions. Preprocessing is done before being fed into the proposed algorithm. This has been discussed in detail in section 3.

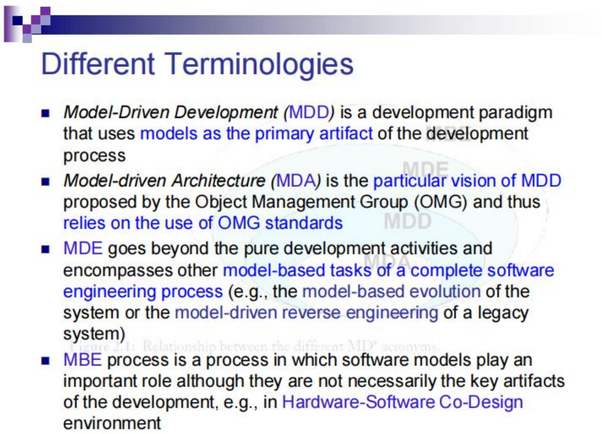

Figure 5- 1-MDSE lecture slide

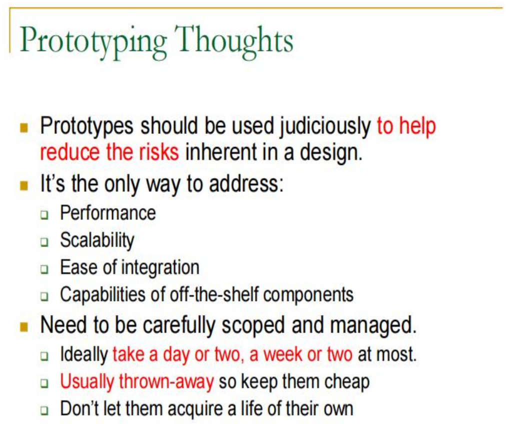

Figure 5- 2-SDA lecture slide

Figures 5-1 and 5-2 depict representative slides from MDSE and SDA presentations. As the figures clearly illustrate, the language in the slides does not adhere to a suitable punctuation standard. Each phrase does not end with a full stop. From the standpoint of slides, there is no problem. However, before feeding this text format into the algorithm, it must be cleansed to generate questions automatically.

5.2 Results

NLP rules have been applied to the first five lectures of all the courses used in the datasets. Results have been taken from two scenarios separately to compare the results.

```
output - Notepad
File Edit Format View Help
4 By default , it is split between two plugins an " edit " ............includes adapters that provide a structured view and perform command based editing of the model
objects ;_?
ANSWER:plugin

5 an " editor " ............provides the UI for the editor and wizard ?
ANSWER:plugin

6............unifies Java , XML , and UML ?
ANSWER:EMF

7 An ...............is essentially the Class Diagram subset of UML with ( * ?
ANSWER:EMF model

8 ...............forms the vocabulary of ecore metamodel ( M ) Nov?
ANSWER:Ecore metametamodel

9............is the root of every model object ?
ANSWER:EObject

10...............is used to create instances of the classes and values of the data types that belong to the package An EPackage 's associated EFactory is accessed via
the eINSTANCE reference ?
ANSWER:A factory

11 ...............is an Eclipse project which allows you to easily create your own graphical modeling workbench by leveraging the Eclipse Modeling technologies ?
ANSWER:Sirius

12 It is composed of a set of Eclipse editors ( diagrams , ...............and trees , workflows ) which allow the users to create , edit and visualize EMF models ?
ANSWER:tables

13 A ...............is a diagram used to visually organize information ?
ANSWER:mind map

14 A ...............is often created around a single concept , drawn as an image in the center of a blank landscape page , to which associated representations of
ideas such as images , words and parts of words are added ?
ANSWER:mind map
```

Figure 5- 3-A text file of Extracted Questions

This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<QUESTIONS>
  ▼<QUESTION0>
    ▼<question>
       ...............refers to- MDA in Practice?
       <option1/>
       <option2>Lecture </option2>
       <option3/>
       <option4/>
     </question>
   </QUESTION0>
  ▼<QUESTION1>
    ▼<question>
       This ............is suitable and heavily used for textbased languages , like programming languages ?
       <option1/>
       <option2>method </option2>
       <option3/>
       <option4/>
     </question>
   </QUESTION1>
  ▼<QUESTION2>
    ▼<question>
       Because ...............do not have to be text based , and often are n't ( they can , for example , have a graphical syntax , like UML ) , we will need a different mechanism for
       defining languages in the MDA context ?
       <option1/>
       <option2>modeling languages </option2>
       <option3/>
       <option4/>
     </question>
   </QUESTION2>
  ▼<QUESTION3>
    ▼<question>
       Because a ............is also a model , a metamodel itself must be written in a welldefined language ?
       <option1/>
       <option2>metamodel </option2>
       <option3/>
       <option4/>
     </question>
   </QUESTION3>
  ▼<QUESTION4>
```


Figure 5- 4-XML file of questions

Figure 5-4 shows the XML file of the output of the QBG tool. XML file comprises the questions and their corresponding answers. XML file is basically to store the output.

5.2.1 Results of Scenario I

The rules have been implemented in this situation in such a way that IF-ELSE conditions are employed in a loop. So, if a rule pattern synchronizes with a sentence pattern, a question is formed from that sentence, and the iteration of the loop ends here, and the next iteration begins checking the pattern of the next sentence with rules, and so on. In this fashion, the QBG tool has generated questions from the courses of the aforementioned courses. The detailed results of the courses are shown in **Tables 5-1, 5-2**, and **5-3**. The accuracy of the QBG tool was calculated with the use of tables. **Table 5-1** summarizes the results of the five lectures of the "SSDA" course. **Table 5-2** summarizes the results of the five lectures of the "MDSE" course. **Table 5-3** summarizes the results of the five lectures of the "SRE" course. **Table 5-4** summarizes the results of the five lectures in the "Database" course.

Table 5- 1-Summary of SSDA LIST

Sr.	Lecture#	Questions by QBG tool		
		Correct questions	Wrong questions	Total
1.	1	28	3	31
2.	2	7	4	11
3.	3	10	4	14
4.	4	22	3	25
5.	5	21	5	26
total		**88**	**19**	**107**

For each generated question through the QBG tool, we have calculated and compared the number of Correct, Incorrect questions with the opinion of a human expert. This leads to measuring the accuracy of the QBG tool for case studies.

Particularly, **correct** questions represent those questions that are correctly generated by the QBG tool from PowerPoint slides. **Incorrect** are those questions that are incorrectly generated by the QBG tool. Incorrect questions here mean the questions that are not meaningful. Based on the aforementioned parameters, the accuracy is calculated as given below. The formula of accuracy is taken from [14].

From the table 5-1, it has shown that the total questions generated by the QBG tool from the lecture slides under observation are=107 (Placeholder1)

Wrong questions generated by the QBG tool are = 19

Correct questions generated by QBG tool are =88

bolds= $\frac{\textbf{True Questions}}{\textbf{True Questions+Wrong Questions}}$

Accuracy= $\frac{88}{88+19} = 82.2\%$

Table 5- 2-Summary of MDSE LIST

Sr.	Lecture#	Questions by QBG tool		
		Correct questions	**Wrong questions**	**Total**
1.	1	28	8	36
2.	2	7	4	11
3.	3	11	1	12
4.	4	20	6	26
5.	5	12	0	12
6.	6	32	5	37
total	--	**110**	**24**	**134**

From the table 5-2, it has shown that the total questions generated by the QBG tool from the lecture slides under observation are= 134

Wrong questions generated by QBG tool are = 24

Correct questions generated by QBG tool are =110

Accuracy= $\frac{110}{110+24} = 82.0\%$

Table 5- 3-Summary of SRE LIST

Sr.	Lecture#	Questions by QBG tool		
		Correct questions	**Wrong questions**	**Total**
1.	1	19	11	30
2.	2	22	2	24
3.	3	8	2	10
4.	4	15	2	17
5.	5	20	8	28
6.	6	12	4	16
total	--	**96**	**29**	**125**

From the table 5-3, it has shown that total questions generated by QBG tool =125

Wrong questions generated by QBG tool are = 29

Correct questions generated by QBG tool are =96

Accuracy= $\frac{True\ Questions}{True\ Questions+Wrong\ Questions}$

Accuracy= $\frac{96}{96+29} = 77\%$

Table 5- 4-Summary of Database LIST

Sr.	Lecture#	Questions by QBG tool		
		Correct questions	**Wrong questions**	**Total**
1.	17	48	6	54
2.	18	25	4	29
3.	19	70	10	80
4.	20	25	2	27
5.	5	71	8	79
total		**239**	**30**	**269**

From the Table 5-4, it has shown that total questions generated by QBG tool =269

Wrong questions generated by QBG tool are = 30

Correct questions generated by QBG tool are =239

Accuracy= $\frac{True\ Questions}{True\ Questions+Wrong\ Questions}$

Accuracy= $\frac{239}{239+30} = 89\%$

Now accumulative accuracy of all the courses is calculated. The above tables, it has shown that the total questions generated by the QBG tool from the lectures under observation are=107+134+125+269=635

Wrong questions generated by QBG tool are = 19+24+29+30=90

Correct questions generated by QBG tool are =88+110+96+239=533

Accuracy= $\mathbf{\frac{True\ Questions}{True\ Questions+Wrong\ Questions}}$

Accuracy= $\frac{533}{533+90} = 83.9\%$

Accuracy=84%

5.2.2 Results of Scenario II

Every rule was applied sequentially to each sentence of the input text in this scenario. The goal of this scenario is to see if a statement can be used to extract more than just a question. As a result of applying rules in this manner, it has been discovered that some sentences are synchronized with multiple rule patterns. As a result, a single line can create multiple questions. Aside from that, some of the issues have been brought up such as the discovery of duplicate questions. For example, some rules, such as rule #3 and rule #4, are closely related. So, if two successive nouns NN are preceded by VBZ in a sentence, both rules pose the same question with minor differences. A fruitless attempt was made to consolidate these closely related regulations. Despite their apparent resemblance, their patterns are distinct, and they must be used independently to match the patterns of other texts.

Table 5- 5-Summary of MDSE LIST

Sr.	**Lecture#**	**Questions by QBG tool**		
		Correct questions	**Wrong questions**	**Total**
1.	1	37	3	40
2.	2	10	3	13
3.	3	13	1	14
4.	4	25	4	29
5.	5	14	4	18
6.	6	38	5	43
total	**--**	**137**	**20**	**157**

For each generated question through the QBG tool, we have calculated and compared the number of Correct, Incorrect questions with the opinion of a human expert. This leads to measuring the accuracy of the QBG tool for case studies.

Particularly, **correct** questions represent those questions that are correctly generated by the QBG tool from PowerPoint slides. **Incorrect** are those questions that are incorrectly generated by the QBG tool. Based on the aforementioned parameters, the accuracy is calculated as given below. The formula of accuracy is taken from [14].

From the Table 5-5, it has shown that the total questions generated by the QBG tool from the lecture slides under observation are=

Total questions generated by QBG tool =157

Wrong questions generated by QBG tool are = 20

Correct questions generated by QBG tool are =137

Accuracy= $\frac{137}{137+20} = 87.2\%$

Table 5- 6-Summary of SSDA LIST

Sr.	Lecture#	Questions by QBG tool		
		Correct questions	**Wrong questions**	**Total**
1.	1	29	3	32
2.	2	9	4	13
3.	3	15	2	17
4.	4	32	6	38
5.	5	30	7	37
total		**115**	**22**	**137**

From the table 5-6, it has shown that total questions generated by QBG tool =137

Wrong questions generated by QBG tool are = 22

Correct questions generated by QBG tool are =115

Accuracy= $\frac{\boldsymbol{True\ Questions}}{\boldsymbol{True\ Questions+Wrong\ Questions}}$

Accuracy= $\frac{115}{115+22} = 84\%$

Table 5- 7-Summary of SRE LIST

Sr.	Lecture#	Questions by QBG tool		
		Correct questions	**Wrong questions**	**Total**
1.	1	38	4	42
2.	2	25	3	28
3.	3	10	2	12
4.	4	17	3	20
5.	5	27	5	32
6.	6	18	1	19
total	**--**	**135**	**18**	**153**

From the table 5-7, it has shown that Total questions generated by QBG tool =153

Wrong questions generated by QBG tool are = 18

Correct questions generated by QBG tool are =135

Accuracy= $\frac{135}{135+18} = 88.2\%$

Table 5- 8-Summary of Database LIST

Sr.	Lecture#	Questions by QBG tool		
		Correct questions	Wrong questions	Total
1.	17	66	10	77
2.	18	32	4	36
3.	19	80	30	110
4.	20	34	2	36
5.	5	77	38	115
total		**289**	**84**	**374**

From the table 5-8, it has shown that Total questions generated by QBG tool =125

Wrong questions generated by QBG tool are = 29

Correct questions generated by QBG tool are =96

Accuracy= $\frac{\boldsymbol{True\ Questions}}{\boldsymbol{True\ Questions+Wrong\ Questions}}$

Accuracy= $\frac{96}{96+29} = 77\%$

Now accumulative accuracy of all the courses is calculated. The above tables, it has shown that the total questions generated by the QBG tool from the lectures under observation are =157+137+153+374=821

Wrong questions generated by QBG tool are = 18+22+20+84=144

Correct questions generated by QBG tool are =135+115+137+298=685

Accuracy= $\frac{685}{685+144} = 83.4\%$

Accuracy=83.4%

It has already been discussed that duplication of questions is found in Scenario II. E.g., some rules like rule#3 and rule#4 are closely related to each other. So, if in a sentence two consecutive nouns NN come proceeded by VBZ, then both rules make the same question with a little difference. An attempt has done to merge these closely related rules but failed. Although they are closely related but their patterns are different, and they both are needed separately to match the pattern of different sentences.

Scenario I provide more accuracy than **Scenario II**, as evidenced by the aforementioned accuracies of both scenarios. Duplication of questions is evident in **Scenario II** as well, as previously discussed. Even though **Scenario I** give more accuracy than **Scenario II**. **Scenario II**, on the other hand, gives a diversity of questions that **Scenario I** is unable to provide due to the application of a single rule to a statement. Therefore, **Scenario II** is preferable to **Scenario I** since it offers us a bigger number of questions.

5.2.3 Comparison with a Previous Study [20] that used Slides as Input

As discussed earlier the author in [20] has used slides as input into the proposed algorithm in order to generate questions. In literature, there is only a study [20] available that is using slide content. Therefore, in this section, a comparison is done with our proposed approach with [20]. **Table 5-9** shows the summarized result of the questions generated by the proposed QBG tool by taking the slides used as input by [20]. The evaluation criteria used by the author is the rating of the questions taken by 50 users. While the accuracy of the proposed QBG tool is calculated by validating the correct and incorrect questions. Correct and incorrect questions are distinguished by the experts of particular subjects.

Table 5- 9-Comparison between proposed QBG tool and [20]

Sr.	Author	Techniques Applied	Domain	Tool	Output	Accuracy
1.	Faizan [20] 2018	Used semantic annotation tool for selecting only named entities i.e., nouns	World War I, The Atom, Geography/General knowledge	No tool	Factoid nature MCQs	User evaluation Maximum question quality rating =43.33%
2.	Nayab	Rule-based Approach 10 proposed rules to cover maximum text	Computer Science i.e., Software Engineering courses	QBG tool	Fill in the blanks	84%

This approach [20] is only finding named entities from the slides and asking a question. Named entities here include nouns i.e.., the name of a person, place or a thing. The author did not provide the material used in his research so that we can use that dataset in the QBG tool to see the result.

5.2.4 Comparison with a Previous Study [10] that used Operating system book as Input

Now we take the study using the same computer science domain as input but using rich content i.e., book. Chapter 11 of this book containing 20 pages has been used as input in the QBG tool in order to check the results. Table **5-10 (a)** shows the results generated by the QBG tool.

Table 5- 10 a)-Results generated of input used in [10]

Sr.	Input	**Questions by QBG tool**		
		Correct questions	**Wrong questions**	**Total**
1.	Operating system	207	0	207

Table 5- 10 b)-Comparison between proposed QBG tool and [0]

Sr.	**Author**	**Techniques Applied**	**Domain**	**Tool**	**Output**	**Accuracy**
1.	Faizan [20] 2018	Used semantic annotation tool for selecting only named entities i.e., nouns	Computer Science i.e., Operating system	MCQG tool	MCQs	Whether or not the generated questions have a meaningful or useful question, this will be explored in future studies

2.	Nayab	Rule-based Approach 10 proposed rules to cover maximum text	Computer Science i.e., Model-Driven Software Engineering, Database, Software System Design and Architecture, Software Requirement Engineering	QBG tool	Fill in the blanks	100%

Table 5-10 (b) shows the comparison results with a previous study using an operating system book as the input of the tool. The author of this research is not giving clear accuracy and mentioned that questions either are meaningful or not will be discussed in future studies. Our proposed tool gave a good result on the operating system course containing rich content. Therefore, we can say that the proposed methodology is good enough to make questions.

Chapter 6

Discussion and Limitations

Chapter 6
Discussion and Limitations

Sub-section 6.1 contains a detailed discussion of the proposed research work and **sub-section 6.2** deals with the limitation of the research.

6.1 Discussion

This article introduces a novel framework to automatically generate assessment questions from the contents of lecture slides. The feasibility of the proposed framework is evaluated through the lecture slides of courses university-level as case studies. In our approach, we used the post-graduate-level courses for question generation where the mode of delivering the lectures is PowerPoint slides by the instructors. The results, as given in **Tables 5-6, 5-7**, and **5-8**, prove that the proposed framework can generate a question bank from a lecture with high accuracy. We target different courses of software engineering courses to generate questions. Every course has around 15-20 lectures taken from the relevant expert and every lecture has been used to generate the questions from the QBG tool to see the results. But in this work, the results of the initial few lectures are included due to space limitations.

The goal is to demonstrate the efficacy of the proposed framework, which is accomplished fairly through given case studies. In this regard, the QBG tool is freely available [56] for further evaluation using the various case studies of choice. The biggest advantage of the proposed framework is to deal with the inappropriate text of slides content without any particular template, where reduced text content and structure in slides results in poorer annotation quality when compared to long, well-articulated texts. Therefore, it is quite hard to manage the text of slides as compared to texts that are well written. In this case, we used a few pre-processing steps to get the text in a good format before applying the proposed algorithm. These steps are derived from a standard procedure that is widely used in natural language processing. The given preprocessing steps for the text of slides contents certainly improve the accuracy of the QBG tool. The QBG tool, on the other hand, can generate assessment questions from any text. It has been observed that the QBG tool gives more accuracy to those lectures that are written with good text structure when calculating results. Depending on the style of the given text, the accuracy of the QBG tool may be compromised in this case.

The proposed framework is the first and significant step toward the automation of assessment questions from the content i.e., PowerPoint slides. The proposed framework is highly beneficial for both industry and academia. Particularly, the proposed framework helps in building the right product at the right time at a comparatively lower cost. It yields a positive result, and validation demonstrates that it will greatly assist university instructors. Questions are generated for various courses, and it has been discovered that using the QBG tool to create questions is effective. An accuracy of roughly 80% was obtained while preparing the questions for each course. The key benefit of the proposed framework is that it significantly reduces the development time of quizzes and papers as the automatically generated assessment questions.

The precision of the QBG tool can be improved by paying close attention to the preprocessing procedures. Even though our method for text preparation is sound. However, as we all know, the content on the lecture slides is not plain text. Natural language processing (NLP) deals with text that should be simple and well-formatted. Furthermore, the punctuation marks in the sentences on the lecture slides are incorrect. And any recommended technique for generating questions will look at the sentences independently to generate the question from a specific sentence. As a result, to adapt the text of PowerPoint slides, adequate pre-processing processes are required.

The proposed framework is highly extendable and supports further enhancements as per requirements. For example, the proposed tool generates open closes i.e., fill-in-the-blank questions. The same approach can be used to generate multiple-choice MCQ questions and subjective type questions. Therefore, it is straightforward to incorporate more rules in the proposed framework for the generation of other types of questions like subjective type questions, generation of distractors to make questions like MCQs, etc. Subsequently, the rules can be implemented in the QBG tool with simplicity.

6.2 Limitation

We only provide the foundation of the proposed framework in this article by focusing on the fill-in-the-blank questions with the correct option from PowerPoint slides. In this regard, the proposed framework currently lacks questions with distractors. We believe that by using the methodology (Section 3) and implementation approach provided, such missing distractors concepts can be easily incorporated into the proposed framework (Section 4).

Chapter 7

Conclusion and Future Work

Chapter 7
Conclusion and Future Work

This work presented a novel framework to automatically generate open cloze questions and answers from the lecture slides of software engineering courses at the postgraduate level. The proposed framework employed Artificial Intelligence (AI) techniques like Natural Language Processing (NLP) etc. to enable the automatic generation of questions banked from PowerPoint slides. Particularly a set of rules are developed to extract the questions and their corresponding answers from the given input lecture slides as a pdf file. Furthermore, a complete algorithm is developed for the execution of rules to generate questions. A sophisticated user interface tool i.e., the QBG tool is also developed. The feasibility of the proposed framework is demonstrated through the lectures taken from the department of computer and software engineering department belonging to the national university of sciences and technology (NUST) Islamabad and umm ul Qura university Saudia Arabia.

The proposed algorithm used a rule-based approach which is best suited for the text extracted from the intended lectures as compared to other benchmarks i.e., artificial intelligence and machine learning techniques in which neural network models are highly used. Unlike other deep learning models, it does not need a large dataset for training purposes. That is the reason, this approach consumed less complexity of the machine.

Our results showed reasonable performance giving a mean accuracy rate of **84%.** For validation, comparison results are made by taking input from previous studies and good results with high accuracy are obtained. Based on the obtained results, it can be concluded that:

- Rule-based approach providing high accuracy gave promising results to the real-time educational environment of the courses of postgraduate’s level leading all state-of-the-art results.
- Proposed approach provides the fast and easiest way to generate questions automatically that benefit the real-time education environment. Instructors can easily generate the

questions by taking only a few seconds and making an assessment activity for the students on regular basis.

- Students can also get benefit by using this tool in their self-learning for the exams and especially during a covid-19 pandemic crisis.

The evaluation results prove that the proposed framework is capable of generating the open cloze questions from the lecture slides of Software engineering courses with high accuracy. Consequently, the proposed framework is the first and significant step towards the automation of generating questions at the postgraduate level from the PowerPoint lecture slides. This leads to several benefits for the instructors like taking students' assessments regularly without consumption of time and effort.

The proposed framework is highly extendable for further enhancements. Although the rule-based approach is efficiently used to extract the cloze questions and gave very graceful results with high accuracy there is still room to enhance the NLP rules to generate questions to increase the accuracy. The rule-based approach can be further used to extract the distractors of the answers and can convert cloze questions to open cloze questions. Furthermore, subjective type questions can be generated using this approach. QBG tool can be upgraded by inserting more functionalities. This will be more beneficial to the instructors to make full fledge question papers for exam points of view.

REFERENCE

1. Jun Araki, Dheeraj Rajagopal, Sreecharan Sankaranarayanan, Susan Holm, Yukari Yamakawa, Teruko Mitamura: “Generating Questions and Multiple-Choice A”, In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 1125–1136, Osaka, Japan. The COLING 2016 Organizing Committee.
2. Susmita Gangopadhyay; S.M Ravikiran: “Focused Questions and Answer Generation by Key Content Selection”, 2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM 2020), New Delhi, India.
3. Tanzim Tamanna Shitu, Nazia Zaman, K. M. Azharul Hasan: “ Domain-Specific Factoid Question Answering by Regular Expression Generation”: 2020 IEEE Region 10 Symposium (TENSYMP), Dhaka Bangladesh.
4. [4] Warren P. du Plessis, “ Automated Generation of Test Questions and Solutions”: 2020 IFEES World Engineering Education Forum - Global Engineering Deans Council (WEEF-GEDC), Cape Town, South Africa.
5. [5] Adam D. Lelkes, Vinh Q. Tran, Cong Yu: “Quiz-Style Question Generation for News Stories”: Proceedings of the Web Conference 2021April 2021.
6. [6] LantingFang, “Syntactic based approach for grammar question retrieval”: Volume 54, Issue 2, March 2018.
7. [7] J. Leo ·G. Kurdi ·N. Matentzoglu · B. Parsia1 ·U. Sattler · S. Forge · G. Donato · W. Dowling: “Ontology-Based Generation of Medical, Multi-term MCQs”: International Journal of Artificial Intelligence in Education (2019).
8. Nguyen-Thinh Le & Niels Pinkwart: “Evaluation of a question generation approach using the semantic web for supporting argumentation”, *Research and Practice in Technology Enhanced Learning* volume 10, Article number: 3 (2015).
9. Dhawaleswar Rao CH and Sujan Kumar Saha. 2020. Automatic multiple-choice question generation from the text: A survey. IEEE Transactions on Learning Technologies, 13(1):14–25.

10. Teo, N. H. I.; Nor, N. D. M.; Zain, N. H. M.; Mokhtar, N. A.: "The development of MCQ generating system based on ontology concepts", International Journal of Advanced Trends in Computer Science and Engineering; 9(1.4 Special Issue):583-591, 2020.
11. Danon, G., Last, M.: A syntactic approach to domain-specific automatic question generation. CoRR (2017).
12. Flor, M., Riordan, B.: A semantic rule-based approach to open-domain automatic question generation. In: Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications, pp. 254–263. Association for Computational Linguistics, New Orleans, June 2018.
13. Le N-T, Pinkwart N. Evaluation of a question generation approach using the semantic web for supporting argumentation. *Res. Pract. Technol. Enhance. Learn.* 2015;10(1):1–19. [PMC free article] [PubMed] [Google Scholar].
14. S. S. R. Adithya; Pramod Kumar Singh: "Web authorizer tool to build assessments using Wikipedia articles", TENCON 2017 - 2017 IEEE Region 10 Conference, Penang, Malaysia.
15. Akhil Killawala; Igor Khokhlov; Leon Reznik: Computational Intelligence Framework for Automatic Quiz Question Generation, 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Rio de Janeiro, Brazil.
16. KANOKWAN ATCHARIYACHANVANICH, SRINUAL NALINTIPPAYAWONG, THANAKRIT JULAVANICH: Reverse SQL Question Generation Algorithm in the DBLearn Adaptive E-Learning System, IEEE Access (Volume: 7).
17. Johanna Melly, Gabriel Luthier, Andrei Popescu-Belis: A Consolidated Dataset for Knowledge-based Question Generation using Predicate Mapping of Linked Data, 16th Joint ACL - ISO Workshop on Interoperable Semantic Annotation PROCEEDINGS.
18. Manish Agarwal, Prashanth Mannem, Automatic gap-fill question generation from textbooks, Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications.
19. Guanliang Chen,1 Jie Yang,2 Claudia Hauff,1 Geert-Jan Houben1: LearningQ: A Large-scale Dataset for Educational Question Generation, International AAAI Conference on Web and Social MediaTwelfth International AAAI Conference on Web and Social Media.

20. Ainuddin Faizan, Steffen Lohmann: Automatic Generation of Multiple Choice Questions from Slide Content using Linked Data, Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics June 2018 Article No.: 32Pages 1–8https://doi.org/10.1145/3227609.3227656.
21. Xinyuan Lu, Yuhong Guo: Learning to Generate Questions with Adaptive Copying Neural Networks, https://arxiv.org/abs/1909.08187.
22. Angelica Willis, Glenn Davis, Sherry Ruan, Lakshmi Manoharan, James Landay, Emma Brunskill: "Key Phrase Extraction for Generating Educational Question-Answer Pairs", Proceedings of the Sixth (2019) ACM Conference on Learning @ ScaleJune 2019 Article No.: 20Pages 10, https://doi.org/10.1145/3330430.3333636.
23. Bang Liu, Haojie Wei, Di Niu, Harlan Chen, Yancheng He: "Asking Questions the Human Way: Scalable Question-Answer Generation from Text Corpus".
24. Wei Yuan, Tieke He, Xinyu Dai: " Improving Neural Question Generation using Deep Linguistic Representation", Proceedings of the Web Conference 2021April 2021 Pages 3489–3500, https://doi.org/10.1145/3442381.3449975.
25. Ming Liu, Li Liu, " Automatic Chinese Multiple Choice Question Generation Using Mixed Similarity Strategy", IEEE Transactions on Learning Technologies (Volume: 11, Issue: 2, April-June 1 2018).
26. Lekshmi R Pillai; Veena G.; Deepa Gupta: "A Combined Approach Using Semantic Role Labelling and Word Sense Disambiguation for Question Generation and Answer Extraction", 2018 Second International Conference on Advances in Electronics, Computers, and Communications (ICAC), Bangalore, India.
27. Tianlin Zhang, Liu, Pei Quan: " Domain-specific automatic Chinese multiple-type question generation", 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), DOI: 10.1109/BIBM.2018.8621162.
28. Yibo Sun; Duyu Tang; Nan Duan; Tao Qin; Shujie Liu; Zhao Yan; Ming Zhou: "Joint Learning of Question Answering and Question Generation", IEEE Transactions on Knowledge and Data Engineering (Volume: 32, Issue: 5, May 1, 2020), DOI: 10.1109/TKDE.2019.2897773.
29. Gerard Deepak; Naresh Kumar; G VSN Sai Yashaswea Bharadwaj; A Santhanavijayan: "OntoQuest: An Ontological Strategy for Automatic Question Generation for e-assessment

using Static and Dynamic Knowledge", 2019 Fifteenth International Conference on Information Processing (CIPRO), Bengaluru, India.

30. Naveed Afzal · Ruslan Mitkov: "Automatic generation of multiple-choice questions using dependency-based semantic relations", *Soft Comput* 18, 1269–1281 (2014). https://doi.org/10.1007/s00500-013-1141-4.
31. Shivani G. Aithal, Abishek B. Rao & Sanjay Singh: "Automatic question-answer pairs generation and question similarity mechanism in question answering system", *Appl Intell* 51, 8484–8497 (2021). https://doi.org/10.1007/s10489-021-02348-9.
32. Pedro Azevedo, Bernardo Leite, Henrique Lopes Cardoso, Daniel Castro Silva, and Luís Paulo Reis, "Exploring NLP and Information Extraction to Jointly Address Question Generation and Answering", Artificial Intelligence Applications and Innovations. 2020; 584: 396–407.Published online 2020 May 6. DOI: 10.1007/978-3-030-49186-4_33.
33. S. Basuki, S.F. Kusuma: " Automatic question generation for 5W-1H open domain of Indonesian questions by using syntactical template-based features from academic textbooks", Journal of Theoretical and Applied Information Technology 96(12):3908-3923, June 2018.
34. Bidyut Das & Mukta Majumder: "Factual open cloze question generation for assessment of learner's knowledge", Int J Educ Technol High Educ 14, 24 (2017). https://doi.org/10.1186/s41239-017-0060-3.
35. Junghyuk Park, Hyunsoo Cho, Sang-Goo Lee: "Automatic Generation of Multiple-Choice Fill-in-the-Blank Question Using Document Embedding", June 2018DOI: 10.1007/978-3-319-93846-2_48.
36. Deena Gnanasekaran, Raja Kothandaraman, and Kannan Kaliyan: "An Automatic Question Generation System Using Rule-Based Approach in Bloom's Taxonomy", Volume 14, Issue 5, 2021, DOI: 10.2174/2213275912666191113143335.
37. https://www.eclipse.org/downloads/packages/release/juno/sr2/eclipse-ide-java-developers
38. P. Shanthi Bala and G. Aghila: "Q-Genesis: Question Generation System Based on Semantic Relationships".
39. Arjun Singh Bhatia, Manas Kirti, and Sujan Kumar Saha: "Automatic Generation of Multiple Choice Questions Using Wikipedia", In Maji P., Ghosh A., Murty M.N., Ghosh K., Pal S.K. (eds) Pattern Recognition and Machine Intelligence. Premi 2013. Lecture Notes in

Computer Science, vol 8251. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-45062-4_104.

40. Shivani G. Aithal1 · Abishek B. Rao1 · Sanjay Singh, "Automatic question-answer pairs generation and question similarity mechanism in question answering system", *Appl Intell* **51,** 8484–8497 (2021). https://doi.org/10.1007/s10489-021-02348-9.
41. A.Agarwal; N. Sachdeva; R. K. Yadav; V. Udandarao; V. Mittal; A. Gupta; A. Mathur, "EDUQA: Educational Domain Question Answering System Using Conceptual Network Mapping", ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Brighton, UK.
42. Katherine Stasaski, Marti A. Hearst: "Multiple Choice Question Generation Utilizing An Ontology", Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications.
43. Girish Kumar; Rafael E. Banchs; Luis Fernando D'Haro, " Automatic fill-the-blank question generator for student self-assessment", IEEE Frontiers in Education Conference (FIE 2015), El Paso, TX, USA.
44. Silvia Ferdiana Kusuma; Daniel O Siahaan; Chastine Fatichah, " Automatic Question Generation In Education Domain Based On Ontology", 2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM), Surabaya, Indonesia.
45. Muhammad Wasim; Waqar Mahmood; Muhammad Nabeel Asim; Muhammad Usman khan: "Multi-Label Question Classification for Factoid and List Type Questions in Biomedical Question Answering", IEEE Access (Volume: 7).
46. Dong Bok Lee, Seanie Lee, Woo Tae Jeong, Donghwan Kim, Sung Ju Hwang: "Generating Diverse and Consistent QA pairs from Contexts with Information-Maximizing Hierarchical Conditional VAEs".
47. Pranav Rajpurkar, Robin Jia, Percy Liang: " Know What You Don't Know: Unanswerable Questions for SQuAD".
48. Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, Shi Wang: "Answer-focused and Position-aware Neural Question Generation", Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, Association for Computational Linguistics.

49. Nan Duan, Duyu Tang, Peng Chen, Ming Zhou: Question Generation for Question Answering, Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark.
50. Kumar, P.S., "A novel approach to generate MCQs from domain ontology Considering DL semantics and open-world assumption". Web Semantics: Science, Serv.
51. Gao, Y., Wang, J., Bing, L., King, I., Lyu. MR (2018): "Difficulty controllable question generation for reading comprehension". Tech. rep.
52. Wenjie Zhou, Minghua Zhang, Yunfang Wu: "Question-type Driven Question Generation", Computation and Language (cs.CL) 2019.
53. Jinnie Shin, Qi Guo, and Mark J. Girl: "Multiple-Choice Item Distractor Development Using Topic Modeling Approaches", Centre for Research in Applied Measurement and Evaluation, Department of Educational Psychology, University of Alberta, Edmonton, AB, Canada.
54. https://www.techtarget.com/searchenterpriseai/definition/natural-language-processing-NLP
55. MARYAM HAMDANI, WASI HAIDER BUTT, MUHAMMAD WASEEM ANWAR, IMRAN AHSAN, FAROOQUE AZAM, AND MUDASSAR ADEEL AHMED, " A Novel Framework to Automatically Generate IFML Models From Plain Text Requirements": *Digital Object Identifier 10.1109/ACCESS.2019.2959813,* IEEE Access (Volume: 7).
56. https://github.com/Nayab-Gull/thesis_ms_se