Report On

Automatic MCQ generator from PDF

Submitted in partial fulfillment of the requirements of the Course project in Semester VII of Fourth Year Computer Engineering

By
Vipul Bhoir(Roll No.07)
Mrudul Chaudhari(Roll No. 12)
Abhinav Desai(Roll No. 14)

Supervisor
Dr. Anil Hingmire

University of Mumbai

Vidyavardhini's College of Engineering & Technology

Department of Computer Engineering

(2024-25)

Vidyavardhini's College of Engineering & Technology
Department of Computer Engineering

CERTIFICATE

This is to certify that the project entitled **"Automatic MCQ generator from PDF"** is a bonafidework of **"Vipul Bhoir(Roll No.07), Mrudul Chaudhari(Roll No. 12), Abhinav Desai(Roll No. 14)"** submitted to the University of Mumbai in partial fulfillment of the requirementfor the Course project in semester VII of Fourth Year Computer Engineering.

.

Supervisor Dr. Megha Trivedi

Mr. Anil Hingmire Head of Department

Abstract

Quizzes provide a quick way of assessing students' knowledge and understanding on specific topics. The manual creation of quizzes however is a very demanding task. In this project, we present an Automatic Quiz Generation System (AQGS) that generates quizzes from a given knowledge source (text) without human intervention. We use natural language processing techniques to generate high quality quiz questionsin an efficient manner. Our system generates quizzes consisting of multiple choice questions (MCQs). The results of the evaluation shows that the system generates questions that are syntactically correct, semantically correct, contextually relevant and complete.

1 Introduction

1.1 Introduction

Automatic mcq generation focuses on generating questions from knowledge sources with no or minimal human intervention. The questions are in the form of mcqs, which provide a quick way of assessing students' knowledge and understanding. The project aims to develop a system which can generate mcqs consisting of multiple choice questions. Using quizzes as an assessment tool enables instructors to measure the effectiveness of their teaching methods, as well as evaluate the learning progress of students Assessment is concerned with the variety of methods that educators use to evaluate, measure, and document the learning progress, skill acquisition, or educational need of. It often defines the systematic basis for making inferences about the learning and development of students.

1.2 Problem Statement

Despite the numerous benefits of using frequent quizzing as a method for improving students' understanding and long-term retention, the manual creation of test materials can be a challenging, time consuming, and a cognitively demanding task. Only few educators have received formal training in assessment development and as a result a lot of questions generated by instructors are of poor quality
.Giving poor quality quizzes to students inevitably impacts the standards of their evaluations. Also, with the growing popularity of e-learning technologies and adaptive learning platforms, come a huge demand for assessment questions. Quizzes are a powerful, quick, and a time-tested way to support and assess students.

1.3 Project Scope

The project aims to develop an automated system that can extract content from a PDF and generate multiple-choice questions (MCQs) based on the text. The system will analyze the input PDF document, identify key concepts, and formulate MCQs with appropriate options and correct answers. This tool can be especially useful for educators, online learning platforms, and examination authorities to create question banks quickly and efficiently.

2. Requirement Analysis

2.1 Software Requirements:

The system will require the following software:

- python
- visual studio code
- Tidyverse libraries
- Other relevant libraries (e.g., ggplot2, caret, etc.)

2.2 Hardware Requirements

The system will require the following hardware:

A computer with at least 4GB of RAM and 100GB of free disk spaceAn internet connection

Recommended:

- 16 GB RAM

Minimum:

- 8 GB RAM

2.3 Functional Requirements

The system must be able to perform the following functions:

- The system should allow users to upload a PDF file.
- The system should identify key concepts, terms, and important sentences in the text.
- The system should automatically generate multiple-choice questions (MCQs) basedon the extracted content.
- The system should correctly identify and mark the correct answer for each generated MCQ.

2.4 Nonfunctional Requirements

Performance: The system should be able to process and extract text from a 10MB PDF in under 2 minutes. It should generate MCQs for an average-sized academic PDF (30 pages) within 3-5 minutes.

Scalability: The system should be scalable to handle large files (up to 100 MB PDFs) and multiple users accessing the service concurrently.The system should efficiently process multiple PDFs for batch MCQ generation without significant degradation in performance.

Maintainability: The system should be modular and easy to maintain, allowing future updates for additional features (e.g., support for new languages). The codebase should be well-documented for future enhancements or bug fixes

Portability: The system should be deployable across different platforms (web-based, standalone desktop application). It should support multiple operating systems, including Windows, macOS, and Linux.

Extensibility: The system should be extensible, allowing for new features and functionality to be added easily.

3. System Design

3.1 System Design:

The system will be designed to automatically generate Multiple Choice Questions (MCQs) from a PDF document. This involves uploading PDFs, extracting content, generating questions and answers, and exporting results in various formats.

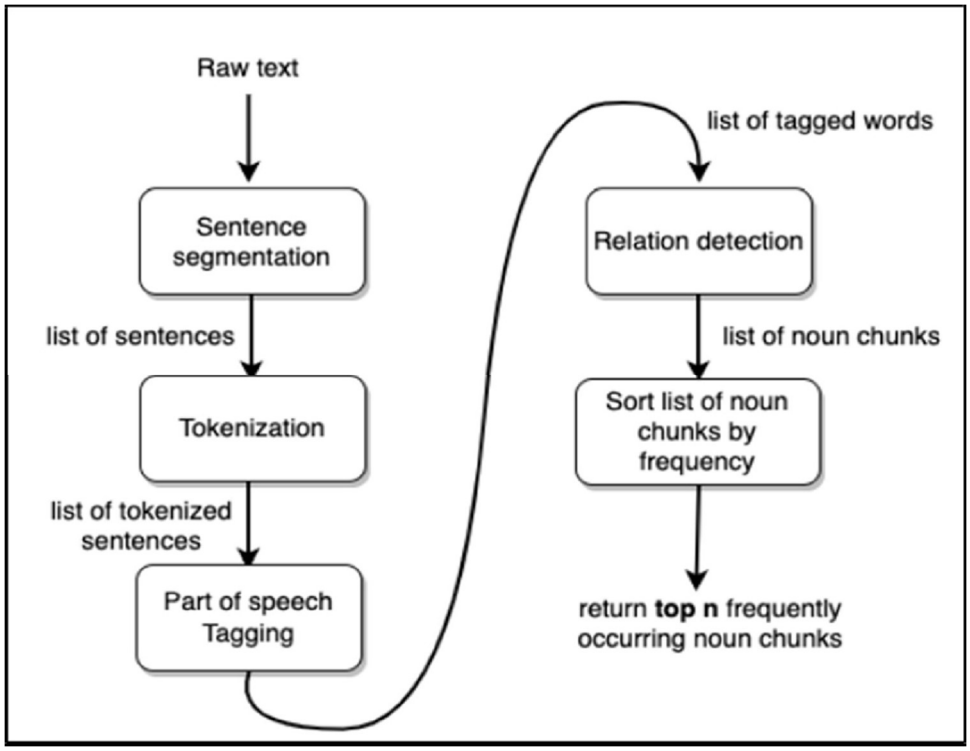
The following are the main modules of the system:
Text Extraction Module: Extract text from text-based PDFs using libraries like PyMuPDF or PDFMiner

Keyword Identification & Contextual Understanding: Identify key concepts or important sentences from the text**.**

MCQ Generation Module: Generate questions based on the extracted content.

Answer Identification: Identify the correct answer for each MCQ from the extracted text.

3.2 Diagram



3.2.1 Project Workflow

3.3 Module Description:

PDF upload module: This module allows users to upload the PDF file containing the content from which the MCQs will be generated.

Keyword identification and concept extraction module: This module identifies key concepts and terms in the text from which questions will be generated. It uses NLP techniques and Identify the most important terms andconcepts in the text.

MCQ generation module: This core module is responsible for generating the actual MCQs. It forms questions based on key sentences and automatically generates options.

Answer identification and validation module: This module identifies and validates the correct answers for each question. It also ensures that the distractors are semantically close but incorrect.

3.2.1 Implementation

4.1 Methodology

1. Pdf data extraction

The first step is to extract the textual content from the PDF files. If the PDF is text-based, we use a PDF parsing library like PyMuPDF or PDFMiner to extract the content. Plain text content that serves as the input for further processing.

2. Text preprocessing

After the raw text is extracted, it needs to be preprocessed for the MCQ generation process. The text is split into sentences or words. Common irrelevant words (e.g., "and," "the") are removed to focus on key concepts. The words are reduced to their base form (e.g., "running"to "run") to standardize the text. This step identifies proper nouns, dates, numbers, etc., that are relevant for MCQ generation.

3. Keyword identification and concept extraction

In this step, important keywords and concepts are identified to form the basis of the questions. The identified terms will be used as "blanks" in the generated questions, or the key concepts will form the basis for question creation. Output will be a listof keywords, concepts, and phrases.

4. Question and Answer generation

This is the core step where multiple-choice questions (MCQs) are generated based on the text. Sentences containing the identified keywords are transformed into questions by replacing the keywords with blanks. The correct answer is extracted from the original sentence.

4.2 Sample Modules

1. Pdf Data Extraction collection

This module handles the extraction of text from the uploaded PDF documents. It uses libraries like PyMuPDF or PDFMiner for extracting text from text-based PDFs and. The extracted text is the foundation for subsequent steps in the MCQ generation process.

2. Text Processing

In this module, the extracted text undergoes preprocessing to prepare it for MCQ generation. This includes tokenization, stop-word removal, lemmatization, and Named Entity Recognition (NER) to identify relevant terms and clean up unnecessary text. This ensures the quality of the text before keyword extraction.

3. Keyword Extraction

This module identifies key concepts or important terms from the preprocessed text. Using techniques such as TF-IDF, BERT embeddings, or Named Entity Recognition (NER), it pinpoints words and phrases that are likely candidates for generating questions.

4. Mcq generation

Based on the keywords and concepts extracted, this module formulates multiple-choice questions. It replaces the keywords in sentences with blanks and generates the correct answer along with plausible distractors using synonym libraries or contextually similar words

4.3 Code :

```
from typing import Annotated

from fastapi import FastAPI, File, UploadFile, Form

import uvicorn

from PyPDF2 import PdfReader

import io

from PDF_Text import get_text_from_pdf

from Response import get_response

from text_pdf import text_to_pdf

app = FastAPI()

@app.post("/upload-pdf/")

async def read_pdf( filename: Annotated[str, Form()], file: UploadFile = File(...)):

    try:

        # Read the PDF file content as bytes

        contents = await file.read()

        pdf_file = io.BytesIO(contents)
```

```
        clean_text = get_text_from_pdf(pdf_file)

        System = "You are the expert system of the world. You know everything."

        start = "Generate the MCQ for the given Text below\n"

        end="mcq generated"

        prompted = start + clean_text + end

        answer = get_response(System , prompted)

        required_text = answer.replace('*', '')

        required_text = required_text.replace("*", "")

        required_text = required_text.replace("#", "")

        result = text_to_pdf(required_text, filename)

        print(result

        return {"Output: ": result}

    except Exception as e:

        return {"Error Internet Connection": e}
```

```
if__name___== '__main__':

    uvicorn.run(app, host = 'localhost', port = 8000)
```

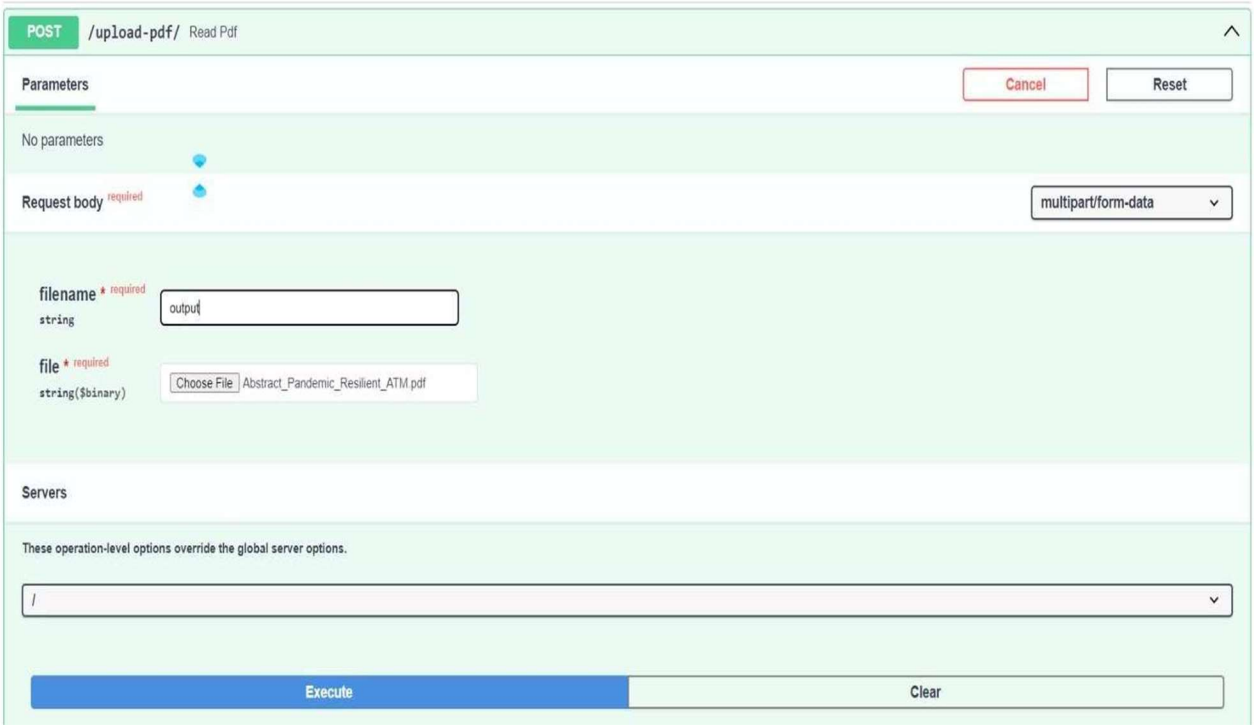
4.3 Output:

Pandemic-Resilient ATM: Enabling Contactless and Card less Transactions using Computer Vision

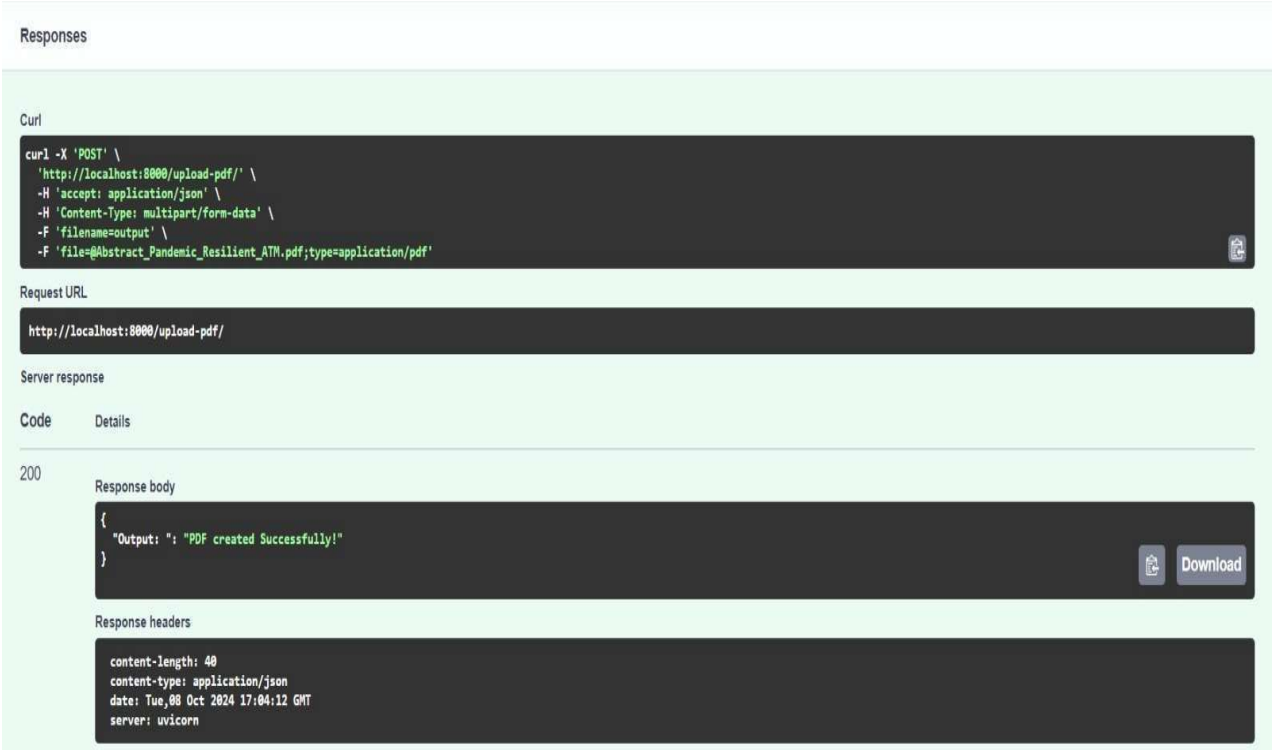
In recent years, Significant advancements in Science and Technology have paved the way for the emergence of hand gesture recognition technology and its diverse applications. This innovative technology has rapidly established itself as a powerful asset in the ongoing battle against the transmission of infectious diseases, with the potential to mitigate the impact of future pandemics that could affect various industries, including banking.

Within the framework of our research, we present a groundbreaking model designed to harness the capabilities of Touchless and Cardless ATMs, thereby providing a secure and seamless financial services experience by avoiding practices like Card Skimming. Our approach introduces a novel use of ATM cameras for hand tracing and detection, employing a sophisticated computer vision model to precisely analyses the intricate motions of an individual's hand. This process serves the dual purpose of identity validation in real-time and the reduction of physical contact, thus not only revolutionizing banking practices but also making substantial contributions to public health measures amid pandemics and beyond.

4.3.1 Input Text



4.3.2 Interface to Upload File in PDF format



4.3.3 Interface showing Pdf uploaded successfully

MCQs for "Pandemic-Resilient ATM: Enabling Contactless and Cardless Transact
using Computer Vision"

1. The main innovation described in the text is:

a) A new type of ATM card with advanced security features.
b) A contactless and cardless ATM system powered by hand gesture recognit
c) A computer vision model to analyze ATM transactions for fraud detectio
d) A hand tracing and detection system for improving security camera foot

2. The technology discussed in the text is beneficial in combating pandemics because:

a) It reduces the need for physical contact during transactions.
b) It eliminates the possibility of spreading viruses through ATM cards.
c) It can track and trace individuals who have used ATMs.
d) It provides a contactless alternative to cash transactions.

3. The key benefit of this technology for banking is:

a) Increased customer convenience and satisfaction.
b) Reduced risk of card skimming and fraud.
c) Improved efficiency and speed of transactions.
d) All of the above.

4. Which statement best describes the functioning of the new ATM system?

a) The system relies on fingerprint scanning to authenticate user identit
b) The system utilizes cameras to track hand movements for user identific
c) The system is equipped with advanced AI to identify and prevent fraudu transactions.
d) The system allows users to access their accounts via a mobile app link the ATM.

5. The text highlights that this technology is:

a) Still in its experimental phase and not yet widely implemented.
b) Ready for widespread implementation in the banking industry.
c) Primarily aimed at developing countries facing limited access to banki services.
d) Part of a broader trend toward contactless and cashless payment system

Answers:

1. b)
2. a)
3. d)
4. b)
5. d)

4.3.4 MCQ Qustions with options and Answer given below

5. Results:

5.1 Results:

The system was tested with multiple types of PDFs,. The PDF Data Extraction Module demonstrated a high level of accuracy in extracting text from text-based PDFs using libraries like PyMuPDF and PDFMiner. After the extraction process, the Text Preprocessing Module successfully cleaned and preprocessed the extracted text. Through tokenization, stop-word removal, and lemmatization using NLTK or spaCy, the module reduced noise and prepared the text for further analysis. This step was crucial for accurate In some cases, the system struggled with long, complex sentences or very dense academic texts, resulting in MCQs that were either too vague or too specific. Over 85% of the generated MCQs were deemed relevant and contextually appropriate by the educators. By utilizing **word embeddings** and contextual understanding models, the system could generate multiple-choice questions with distractors that were semantically close to the correct answer, making the questions more challenging and relevant.

5.2 Conclusion:

The development of the **Automatic MCQ Generator from PDF** represents a significant advancement in the field of educational technology, leveraging automated processes to enhance learning and assessment. The project aims to streamline the traditionally manual and time- consuming task of creating multiple-choice questions (MCQs) from educational materials, ultimately benefiting educators, students, and content creators alike. The system successfully automates the extraction of relevant information from PDF documents and transforms this information into high-quality MCQs. This capability reduces the time educators spend on question creation, allowing them to focus more on teaching and interacting with students. The ability to generate MCQs from educational content automatically is particularly valuable in today's fast-paced academic environment. It not only aids in assessment preparation but also enhances the learning experience by providing students with immediate access to practice questions relevant to their coursework.

References:

[1] Santhanavijayan, A., Balasundaram, S.R., Hari Narayanan, S., Vinod Kumar, S., and Vignesh Prasad, V. (2017) 'Automatic generation of multiple-choice questions for e-assessment', Int. J. Signal and Imaging Systems Engineering, Vol. 10, Nos. 1/2, pp.54–62.

[2] Ayako Hoshino and Hiroshi Nakagawa (2005) " A realtime multiple-choice question generation for language testing: A preliminary study", EdAppsNLP 05: Proceedings of the second workshop on Building Educational Applications Using NLP.

[3] D. R. CH and S. K. Saha, "Automatic Multiple Choice Question Generation From Text: A Survey," in IEEE Transactions on Learning Technologies, vol. 13, no. 1, pp. 14-25, 1 Jan.-March 2020, doi: 10.1109/TLT.2018.2889100.

[4] Deepshree S. Vibhandik, Rucha C. Samant "Automatic / Smart Question Generation System for Academic Purpose", International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 4, Issue 4, July - August 2015.

[5] Susanti, Y., Tokunaga, T., Nishikawa, H. et al. "Automatic distractor generation for multiple- choice English vocabulary questions", RPTEL 13, 15 (2018). https://doi.org/10.1186/s41039-018- 0082-z.