QuizUp: An AI-Powered Framework for Automated MCQ Generation from PDFs

Bharateesha LVN
AIML Department
PES University
lvnb2004@gmail.com

Nagathejas M S
AIML Department
PES University
thejasnaga@gmail.com

Himanshu k
AIML Department
PES University
himanshuk7104@gmail.com

Malleshappa D patil
AIML Department
PES University
mdpatil2004@gmail.com

Abstract:

This paper presents QuizUp, a web-based intelligent learning tool that leverages artificial intelligence to transform document-based learning. The application enables users to upload PDF, TXT, and DOCX files and automatically generate multiple-choice questions (MCQs), true/false questions, and fill-in-the-blank assessments. By integrating advanced AI techniques, QuizUp provides a personalized and adaptive learning experience with customizable difficulty levels and question types. The system demonstrates the potential of AI in educational technology by offering an intuitive, flexible platform for self-assessment and knowledge verification.

Keywords:

—question generation, AI-powered learning, educational technology, multiple-choice questions, natural language processing

I. Introduction

Modern educational approaches increasingly rely on technology to enhance learning experiences. Traditional study methods often lack interactive and personalized assessment tools. QuizUp addresses this gap by providing an AI-driven platform that transforms static documents into dynamic learning resources. The objectives include:

A. Motivation

The primary objectives of QuizUp include:

- Automated question generation from various document formats
- Customizable assessment creation with multiple difficulty levels
- Support for different question types to enhance learning engagement
- Leveraging advanced AI models for intelligent content parsing and question generation

B. Unique Contributions

Unlike existing quiz generation tools, QuizUp offers:

- Comprehensive support for PDF, TXT, and DOCX file formats
- Three distinct question types: MCQs, True/False, and Fill-in-the-Blanks
- Adaptive difficulty levels (Easy, Medium, Hard)
- AI-powered content extraction and question generation

II. System Architecture

A. Overall Design

The QuizUp application follows a modular web-based architecture with the following key components:

- Frontend: Flask-based web interface
- Backend: Text extraction and AI-powered question generation
- AI Model: Google's Generative AI (Gemini 1.5 Pro)

B. Technical Components

1. **File Processing**
   - Supports PDF, TXT, and DOCX file formats
   - Utilizes libraries: `pdfplumber`, `docx`, `csv`
   - Secure file handling with `werkzeug` utilities
2. **Text Extraction**
   - Extracts text from different document types
   - Handles encoding and formatting challenges
   - Provides consistent text representation for AI processing
3. **Question Generation**
   - Implements context-aware prompt engineering
   - Generates questions based on:
     - Input text
     - Selected difficulty level
     - Chosen question type
   - Uses Google's Generative AI for intelligent content processing

III. Methodology

A. Question Generation Algorithm

The core question generation process involves:

1. Text preprocessing and normalization
2. Context analysis using AI model
3. Difficulty-specific prompt engineering
4. Intelligent question and answer generation

Difficulty Level Strategies

- **Easy**: Focus on direct facts, simple vocabulary
- **Medium**: Balanced between recall and application
- **Hard**: Complex scenarios, critical analysis

B. Question Type Handling

1. **Multiple Choice Questions (MCQ)**
   - Four distinct answer options
   - Randomized correct answer placement
   - Plausible distractors
2. **True/False Questions**
   - Binary answer format
   - Text-based verification
   - Contextual explanations
3. **Fill-in-the-Blanks**
   - Context-driven blank spaces
   - Semantically relevant options
   - Focuses on key terms and concepts

IV. Implementation Details

A. Technology Stack

- **Backend**: Python, Flask
- **AI Model**: Google Generative AI (Gemini 1.5 Pro)
- **File Processing**: pdfplumber, python-docx
- **PDF Generation**: FPDF

B. User Interface

- Responsive web design
- Drag-and-drop file upload
- Customization options for:
  - Number of questions
  - Question type
  - Difficulty level

V. Experimental Results

A. Key Achievements

- Successful AI-powered question generation
- Support for multiple document formats
- Customizable learning assessment tool

B. Limitations

- Dependency on AI model's interpretation
- Potential variations in question quality
- Performance variations with complex documents

VI. Conclusion and Future Work

QuizUp represents an innovative approach to automated learning assessment. Future enhancements will focus on:

- Expanding supported document formats
- Improving AI model fine-tuning
- Enhancing user interface and experience
- Implementing more advanced question generation techniques

VII. Acknowledgments

We thank the Department of Computer Science and Engineering(AIML) at PES University for their support and guidance.

References

1. J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint arXiv:1810.04805.
2. L. Yang, B. Duan, and L. Wang, "Automatic Question Generation from Text: A Survey and Perspectives," Proceedings of the 2020 AAAI Conference.
3. S. Rao and H. Daumé III, "Neural Question Generation from Text: A Preliminary Study," ACL 2018 Workshop on Question Generation.
4. H. Xu, Z. Wang, and W. Sun, "Toward AI-Powered Education: Self-learning Chatbots Based on Reinforcement Learning," IJCAI Proceedings.
5. M. Heilman and N. Smith, "Question Generation via Overgenerating Transformations and Ranking," EMNLP 2010.