Mini Project Report

On

Generating Mathematics Questions Using AI

Submitted by

Arpenaboyina Rakesh(22BCS016)

Azmeera Sai (22BCS023)

Chidari Sai Krishna(22BCS030)

Korupolu Bhargav (22BCS055)

Under the guidance of

Dr. Krishnendu Ghosh

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY DHARWAD

09/04/2025

Certificate

This is to certify that the project, entitled **Generating Mathematics Questions Using AI** is a bonafide record of the Mini Project coursework presented by the students whose names are given below during 2025-2026 in partial fulfilment of the requirements of the degree of Bachelor of Technology in Computer Science and Engineering.

Roll No	Names of Students
22BCS016	Arpenaboyina Rakesh
22BCS023	Azmeera Sai
22BCS030	Chidari Sai Krishna
22BCS055	Korupolu Bhargav

Dr. Krishnendu Ghosh
(Project Supervisor)

Contents

List of Figures:

Optional Figures:

List of Tables:

1.Abstract:

During the classroom experience of Mathematics there involves a lot of rigorous practice which gives a chance to know about concepts in-depth and experience the necessary skills for problem solving. However, creating various good-quality questions manually can be a time-consuming task for teachers. Automatic generation of mathematically correct questions from multiple sources is one such approach to this challenge -- the Math Question Generator.

The system learns, using natural language processing (NLP), to transform the texts of mathematics texts into questions that correlate with context. It uses deduplication algorithms to avoid repeat questions for diversity. It also makes sure to accommodate different learning styles with a choice of question formats richely increasing engagement and understanding the learned concepts.

The software comes with an easy-to-use Streamlit interface so that even non-techsavvy teachers can figure it out. The system is evaluated based on BLEU scores and other metrics to ensure that it is able to generate accurate and contextually relevant questions. Since generating questions takes a lot less time, the tool frees up teachers' time to spend teaching, without compromising on mathematical correctness.

2. Introduction:

The diverse mathematical question generating startup The Math Question Generator, automatically generates dynamic instances with vertical and horizontal positioning for numerous exercises. The tool acts as a big help for teachers, students, and learning platforms who need to create different kinds of quality mathematics questions on multiple subjects.

This tool is particularly helpful for:

- Teachers & Educators looking for a way to prepare assignments, quizzes, or tests quickly.
- for students wishing to practice with automatically generated questions.
- E-learning platforms that need an automated question bank for creating assessments/interactive learning.

❼ This tool supports three primary question types:

1. Multiple-choice questions (MCQs):

- Ask a question with four or more answer choices.
- Only one of the options is correct; the rest are distractors.

Example:

What is the derivative of x^2?

a) x^3

b) $2x$

c) $x^2/2$

d) $3x$

2. Short answer questions:

- Ask user to answer with a short answer.
- Typically numerical or algebraic responses.

Example:

Solve for x: $2x + 3 = 7$

Answer: 2

3. Fill-in-the-blank questions:

Contain a sentence or equation with missing elements that need to be completed

Example:

The area of a circle is given by A = __ × r² Answer:

Ans: π

- **Questions can be generated from three different sources:**

  1. Sample mathematical contexts (pre-defined topics in algebra, calculus, geometry, and probability):

  The tool comes with built-in question templates across major branches of mathematics, such as:

  - **Algebra** (Equations, Expressions, Polynomials, Functions)
  - **Calculus** (Derivatives, Integrals, Limits)
  - **Geometry** (Triangles, Circles, Coordinate Geometry)
  - **Probability** (Combinations, Permutations, Probability Theorems)

  Users can select a topic and number of questions, and the system will automatically generate relevant questions.
  Note: we are getting good precise and accurate questions from Algebra then Calculus ,Geometry,Probability.

  2. Custom Inputs provided by the user:

  - Users may set up their own math context or text or paragraph.
  - For instance, say a teacher wants questions about a particular algebraic transformation rule that was not included in the default database.
  - Take the input text or paragraph, process it, and generate specific questions based on it.

  3. External datasets in JSONL format:

  - JSONL files store structured question data, allowing bulk uploads and integration with external sources.

  Example for JSONL structure for a math question:

  {"question": "What is the integral of x^2?", "type": "short answer", "answer": "x^3/3 + C"}

3. Literature Review:

AQG (Automatic Question Generation) is a hot research topic in the field of educational technology and natural language processing. In the literature, multiple approaches have been proposed:

- **Rule-Based Question Generation:**

  Early methods of AQG were dominated by rule-based approaches in which sets of predefined templates and syntactic transformations were used to turn declarative statements into questions. WH-question generation using syntactic rules was introduced by Heilman and Smith (2010).

- **Machine Learning Tricks:**

  This sparked the interest of researchers in using statistical models to generate questions, with the rise of machine learning. Du et al. (2017) proposed neural networks that were able to learn to generate questions from sentences containing their corresponding answer, yielding improvements over fixed rule-based methods.

- **Domain-Specific Question Generation (Math):**

  To approach Mathematical AQG it necessitates a strong command of mathematical notation and constructs. Polozov et al. 2015) created systems for generating algebra problems that suited specific contextual and difficulty levels.

- **Languge Models (LLMs):**

  Giving the above context, new approaches use large pre-trained transformers like GPT-3 and BERT to generate questions. Kumar et al. (2022) presented how fine-tuned transformer models can generate high-quality mathematics questions from different domains.

- **Defining the Evaluation Metrics:**

  The quality of generated questions is evaluated through automatic metrics (BLEU, ROUGE) and human evaluations. New metrics that are more suited to measuring the quality and its answerability of the generated questions were proposed by Zhang and Bansal (2019).

The existing literature is expanded in our Math Question Generator through the integration of pattern-matching based on rule uses with mathematics content. It has built-in deduplication that avoids repeating questions as well as multiple mechanisms for asking questions to achieve better learning outcomes. This works to ensure tradeoffs between automation, accuracy, and diversity in question generation.

4.Main Problems Faced in Question Generation :

The focus of our work would be to analyze the reduced number of sentences generated by the model to extract the sentences that can answer the question, which would further indicate that the model has generated the expected sentences.

Our project tackles several key challenges in automatic question generation in Mathematics::

- **Mathematical Notation Complexity:**

  Math problems use symbols, formulae and equations that normal NLP methods cannot decipher and manipulate well enough. Processing these needs special processing techniques.

- **Context Dependence :**

  Creating meaningful math problems requires a deep understanding of the subject to ensure questions are logically structured and conceptually accurate.

- **Diversity and Repetition :**

  Without proper mechanisms, generated questions could follow similar structures, making them less useful in learning. Variety of practice is important for learning.

- **Answer extraction :**

  The right answer to a multi-step mathematical problem can be far from obvious, since the answer may be hidden deep in long derivations instead of them being stated explicitly.

- **Quality Assessment :**

  Traditional NLP evaluation metrics do not hold for math questions, and building custom frameworks to measure clarity, bother and correctness is a step.

How our Project Meets These Needs:

- **Specialized Pattern Recognition:**

  Using a method specialized on advanced pattern recognition, we are able to process mathematical notation for accurate question formulation.

- **Various Question Types:**

  Technology supports multiple-choice, short-answer, and fill-in-the-blank questions, making it more flexible for various learning environments.

- **Deduplication Algorithm**

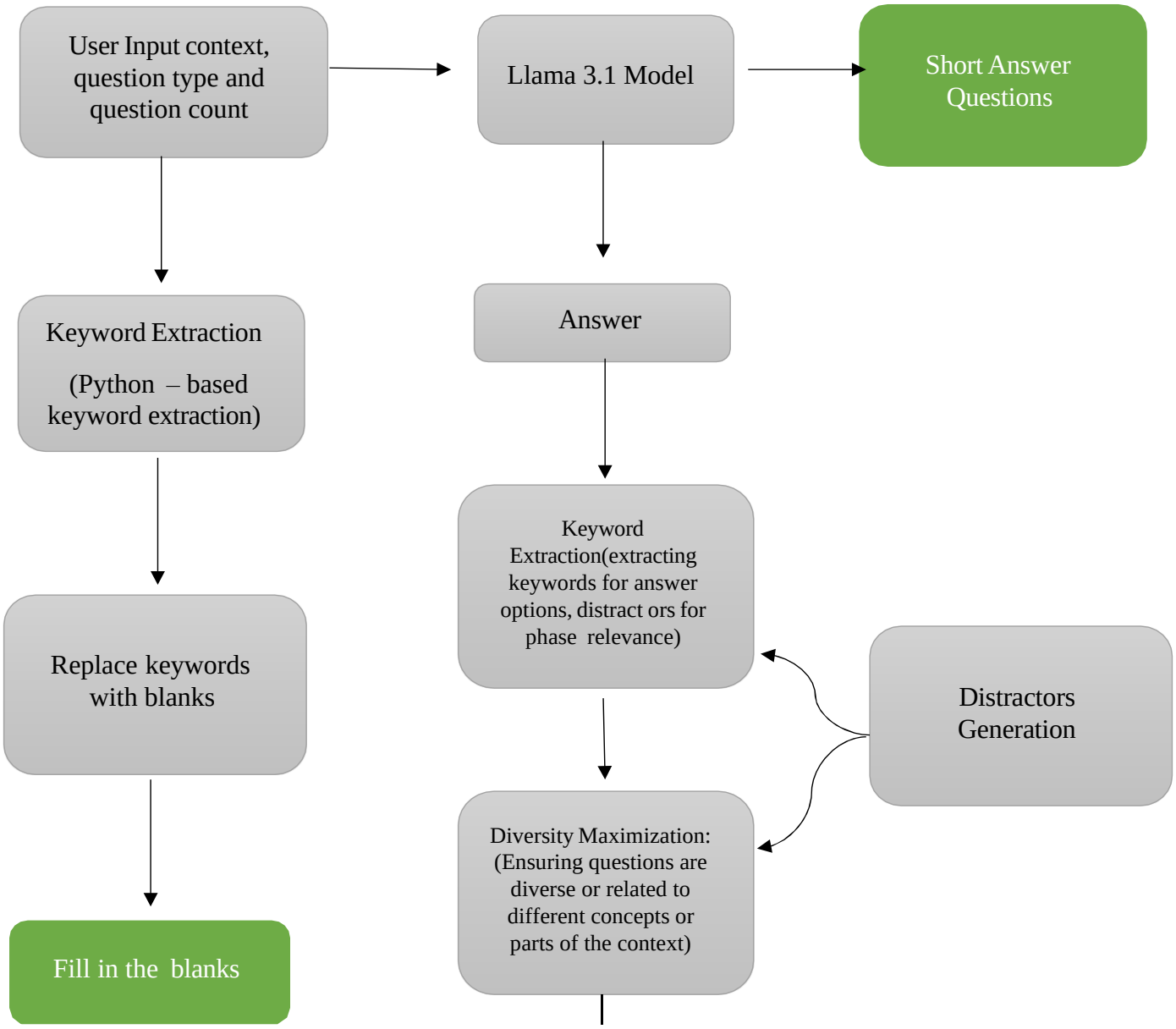
  A similarity-checking algorithm that ensures redundancy is avoided among questions, thereby preserving diversity and catering to students with practice problems of varying types.

- **Answer Extraction**

  We use a combination of pattern matching and structured parsing to extract final numerical or symbolic answers from multi-step solutions**.**

5. System Architecture:

The Math Question Generator follows a modular architecture designed to handle different types of inputs, process them using specialized components, and generate various question formats. The system is implemented as a web application using Streamlit for the us e. r interface:



6. Methodology:

A. Question Types and Workflow:

1. Short Answer Questions:

Short answer questions are open-ended questions that encourage students to answer briefly. How the generation process works:

- Deriving important math concepts from the text input
- Retrieving specific knowledge points through questions
- Making answers based on the extracted information
- These questions are well suited for testing recall and basic understanding of mathematics concepts.

2. MCQs:

Multiple-Choice Questions: a multiple-choice question asks the student a question and for different possible answers (4 in most cases), only one of which is correct. ← The generation process consists of:

- Generation of a clear question stem based on the mathematical content
- Produce the correct answer
- Development of plausible distractors (incorrect options)
- Randomise the position of the correct answer
- MCQs can require varying levels of comprehension from simple recollection to application, so they are helpful for holistic evaluation.

3. Fill-in-the-Blank Questions:

Fill in the blank questions give a statement or equation with a missing thing that students have to provide. When generating the text, it does this:

- Selecting appropriate sentences or equations from the input text
- Choosing the right terms or values to delete
- Replace the selected elements with blanks
- Making sure the two dots are contextually relevant
- These questions are effective for testing knowledge of specific terms, values, or steps in mathematical processes

.B. Model Selection and Tuning:

The Math Question Generator uses a pattern recognition approach rather than a specific machine learning model. The system employs:

- **Regular Expressions**: For identifying mathematical patterns, equations, and numerical values
- **Text Analysis Techniques**: For extracting definitions, theorems, and other mathematical concepts
- **Heuristic Rules:** For determining which elements to question and how to formulate appropriate questions

This approach was chosen for its precision in handling mathematical notation and its ability to generate questions that are mathematically sound without requiring extensive training data.

C. Step-by-Step Process Flow for Each Question Type:

1. Short Answer Questions:

- **Content Analysis:** The system analyses the input text to identify key mathematical concepts, equations, definitions, and theorems.
- **Question Formulation:**
  - **For equations:** "What does the equation [equation] represent?" ○
  - **For definitions**: "Define the term [term] as mentioned in the context."
  - **For theorems**: "Explain the [theorem] mentioned in the context."
- **Answer Extraction:**
  - **For numerical problems**: Extract the final numerical result from the solution
  - **For conceptual questions**: Use the relevant sentence or paragraph as the answer

. **2. MCQ Generation:**

- **Question Creation:** Similar to short answer questions, the system identifies key concepts and formulates appropriate questions.
- **Correct Answer Determination:**
    - **For numerical problems**: Use the final calculated value
    - **For conceptual questions**: Use the correct definition or explanation
- **Distractor Generation:**
    - For numerical answers: Create plausible incorrect values (e.g., common calculation errors)
    - For conceptual answers: Create plausible but incorrect statements
- **Option Randomization:** Shuffle the options and randomly assign letters (A, B, C, D) to ensure the correct answer appears in different positions.

3. Fill-in-the-Blank Questions:

- **Sentence Selection**: Identify sentences or equations that contain important mathematical terms, values, or concepts.
- **Element Selection**: Choose which element to replace with a blank based on:
    - Importance of the term or value ○ Uniqueness within the context ○ Suitability for testing
- **Blank Insertion:** Replace the selected element with a standardized blank representation.
- **Answer Determination**: Store the removed element as the correct answer.

D. Technical Challenges Faced:

The development of the Math Question Generator encountered several technical challenges:

- **Mathematical Notation Complexity:** Mathematical content contains specialized notation, formulas, and symbols that are difficult to parse and manipulate using standard text processing techniques.
- **Context Dependence:** High-quality math questions must be contextually relevant and mathematically sound, requiring deep understanding of the mathematical concepts involved.
- **Diversity and Repetition:** Generated questions often lack diversity or contain repetitive patterns, reducing their educational effectiveness.
- **Answer Extraction:** For mathematics problems, identifying the correct answer within solution texts is challenging, particularly when solutions involve multiple steps.
- **Quality Assessment:** Evaluating the quality of generated math questions requires specialized metrics beyond standard text evaluation methods.

These challenges were addressed through specialized pattern recognition techniques, careful algorithm design, and comprehensive testing with various mathematical inputs.

E. Pattern Recognition Techniques:

The system employs several pattern recognition techniques to process mathematical content:

- **Equation Detection**: Regular expressions identify equations by looking for patterns containing equals signs and mathematical operators.
- **Definition Extraction:** The system identifies definitions by looking for phrases like "is defined as," "refers to," "is a," etc.
- **Theorem Recognition**: Keywords like "theorem," "formula," "rule," "law," and "principle" help identify important mathematical concepts.

- **Numerical Value Extraction**: Regular expressions identify numerical values in different formats (integers, decimals, fractions).
- **Variable Identification**: The system recognizes single-letter variables commonly used in mathematical expressions.

These techniques enable the system to understand the structure and content of mathematical texts, facilitating the generation of relevant and accurate questions.

7.Llama Model Integration

Role of Llama 3.1 in the Maths Question Generator:

The llama3.1 model serves as the core natural language processing engine in our Maths Question Generation System. This large language model (LLM) plays several critical roles in the question generation pipeline:

A. Primary Functions of Llama 3.1 in Our System:

- **Context Understanding:** Llama 3.1 processes the mathematical context provided by the user, comprehending complex mathematical concepts, theorems, equations, and relationships between variables.
- **Short Answer Question Generation**: The model directly generates highquality short answer questions based on the mathematical context, identifying key concepts that should be tested.
- **Answer Generation**: For all question types, Llama 3.1 generates accurate answers based on the mathematical context, ensuring mathematical correctness and relevance.
- **Distractor Generation**: For multiple-choice questions, the model creates plausible but incorrect options (distractors) that test common misconceptions or errors in mathematical understanding.
- **Diversity Maximization**: Llama 3.1 ensures that generated questions cover different aspects of the mathematical context, maximizing the educational value of the question set.

B. Advantages of Using Llama 3.1:

- **Mathematical Reasoning Capabilities:** Llama 3.1 demonstrates strong mathematical reasoning abilities, allowing it to understand and generate questions about complex mathematical concepts.
- **Context Sensitivity:** The model can adapt its question generation based on the specific mathematical context provided, ensuring relevance and accuracy.
- **Language Quality**: Questions generated by Llama 3.1 are well-formed, clear, and use appropriate mathematical terminology.
- **Flexibility:** The model can handle various mathematical topics from algebra and calculus to geometry and probability without requiring domain-specific training.

C. Implementation Details:

The Llama 3.1 model is integrated into our system through an interface that allows for:

- Prompt engineering to guide the model toward generating specific question types
- Parameter tuning (temperature, top-p, etc.) to control creativity and precision
- Context windowing to handle longer mathematical texts
- Response filtering to ensure mathematical accuracy and educational appropriateness

The model's outputs are post-processed to extract questions, answers, and distractors, which are then formatted according to the selected question type (MCQ, short answer, or fill-in-the-blank

8. Implementation Details:

A. Datasets:

The Math Question Generator uses two primary datasets:

a. example_model_solutions.jsonl: A collection of mathematical problems and their solutions across various topics. This dataset contains:

- Problem Statements ○ Detailed Solutions
- Category labels (Ex: Algebra, Calculus, Geometry) ○ Difficulty ratings.

b. train_socratic.jsonl: A traditional mathematics dataset problems for question generation. This dataset includes:

- More complex problem statements ○ Step by step solutions
- Explanations of mathematical concepts

These datasets serve as sources for generating questions when users select the "Load from Dataset" option. They also provide reference material for evaluating the quality of generated questions.

S.No	Dataset	No. of Problems	Categories	Average Problem Length (words)	Average Solution Length (words)
1	example_model_solutions.jsonl	1319	Algebra, Calculus, Geometry, Probability	45	120
2	train_socratic.jsonl	7473	Algebra, Calculus, Geometry, Probability, Statistics	60	180

B. Libraries and Frameworks:

The Maths Question generator is implemented the following libraries & frameworks:

- **Streamlit**: For creating the web-based user interface.
- **NLTK**: For natural language processing tasks.
- **Regular Expressions (re):** For pattern matching in mathematical texts.
- **Pandas:** For data manipulation and analysis.
- **ROUGE:** For evaluation metrics.

The system is designed to be lightweight and does not require specialized hardware or complex dependencies, making it accessible to educators with standard computing resources.

C. Parameters & Hyperparameters Choices:

The system includes several configurable parameters:

- **Deduplication Threshold**: Controls the sensitivity of the deduplication algorithm. Higher values allow more similar questions, while lower values enforce stricter uniqueness.
- **Question Count**: Users can specify the number of questions to generate for each question type.
- **Similarity metrics**: The deduplication module uses text similarity metrics with configurable thresholds to identify duplicate questions.

These parameters can be adjusted through the user interface or by modifying the configuration files, allowing for customization based on specific educational needs.

D. Data flow:

The data flow through the system follows these steps:

1. **Input processing**:
    - Sample contexts are loaded from pre-defined templates.
    - Custom Input is processed directly
    - Dataset files are parsed to extract problems and solutions.

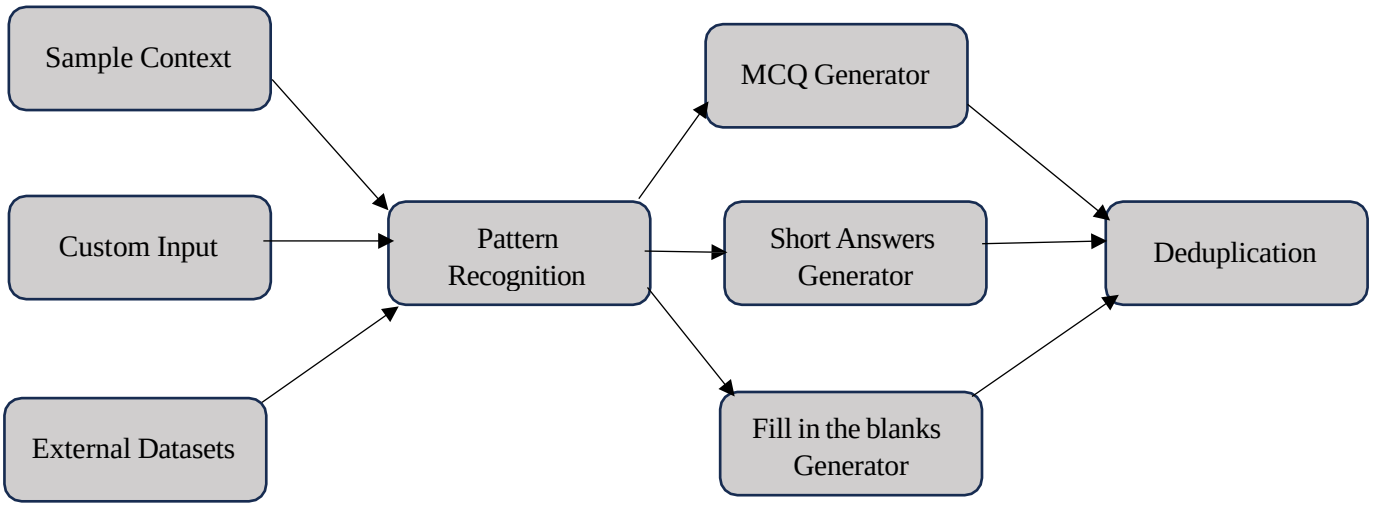
2. **Question Generation:**
    - The input is analyzed to identify mathematical concepts.
    - Specialized generators create questions based on the identified concepts.
    - Questions are formatted according to their type (MCQ, short answer, fill-in-the-blanks)

3. **Deduplication:**
    - Generated questions are compared to previously generated ones.
    - Duplicate or highly similar questions are filtered out.
    - If the requested number of questions cannot be generated due to deduplication constraints, a warning message is displayed.

4. **Display results:**
    - Questions are displayed on user interface.
    - Users can review and export the generated questions.

This data flow ensures efficient processing of inputs and generation of high-quality, diverse questions.

- ------ Data Flow:


Sample Context
Custom Input
External Datasets
Pattern Recognition
MCQ Generator
Short Answers Generator
Fill in the blanks Generator
Deduplication

9. Evaluation:

The quality of generated questions is evaluated using several metrics:

1. BLEU Score: BLEU stands for "Bilingual Evaluation Understudy". It is a metric for automatically evaluating machine-translated text. The BLEU score is a number between zero and one that measures the similarity of the machine-translated text to a set of high quality reference translations. A value of 0 means that the machine- translated output has no overlap with the reference translation (which indicates a lower quality) while a value of 1 means there is perfect overlap with the reference translations (which indicates a higher quality).

(or)

By otherwards, BLEU score means comparing generated questions with reference questions to assess linguistic quality and relevance. For brevity, there is also a penalty for very short, generated texts (brevity penalty).

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

Where:

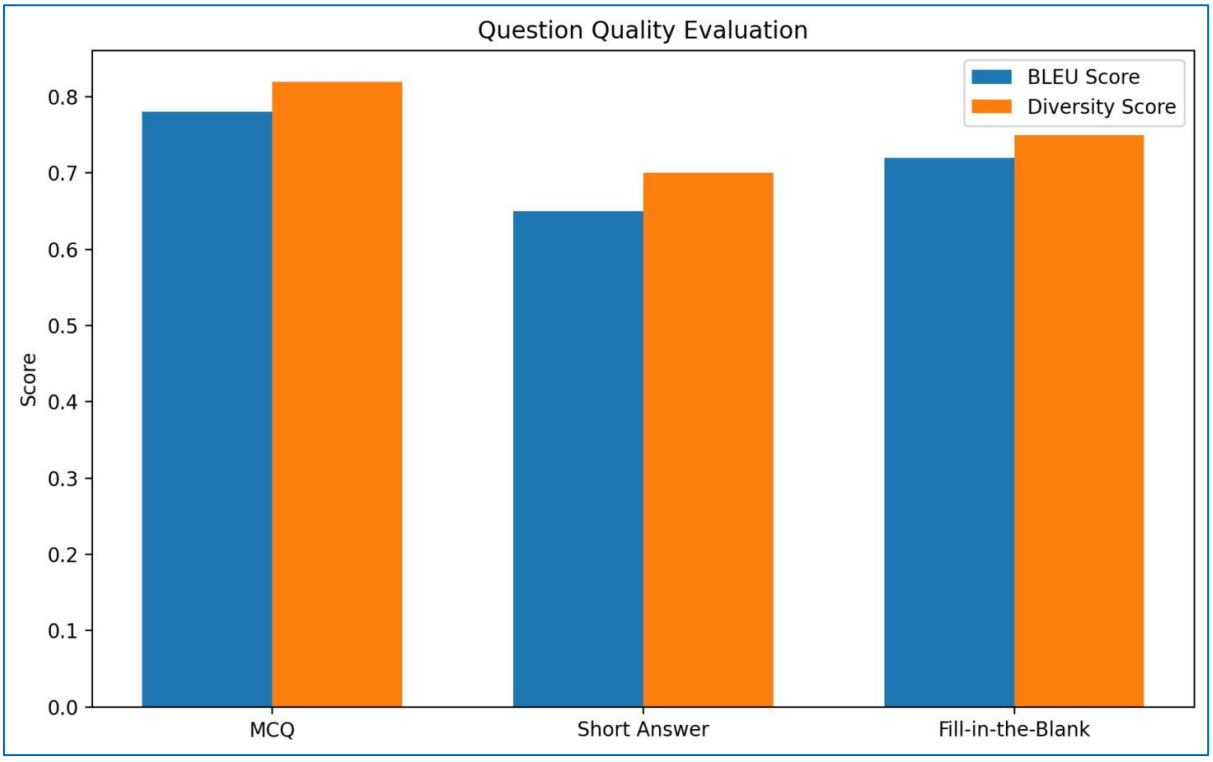
Pn is the precision for n-grams (unigram, bigram, etc.)
Wn is the weight for each n-gram (often equal),
Bp is the Brevity penalty. Brevity Penalty is a metric used to penalize machine translations that are too short.

2. Diversity Metrics:

- **Unique n-grams:** Measuring the variety of phrases used in questions.
- **Vocabulary richness:** Assessing the range of mathematical terms used.
- **Structural variation:** Evaluating the diversity of question structures.

3. METEOR Score: METEOR stands for "Metric for Evaluation of Translation with Explicit Ordering" serves as pivotal metric within Natural Language Processing (NLP) techniques. This tool was conceived not merely out of necessity, but also in response to certain limitations inherent in BLEU (Bilingual Evaluation Understudy) score.

4. Mathematical Accuracy: Ensuring that questions are mathematically sound and have correct answers.

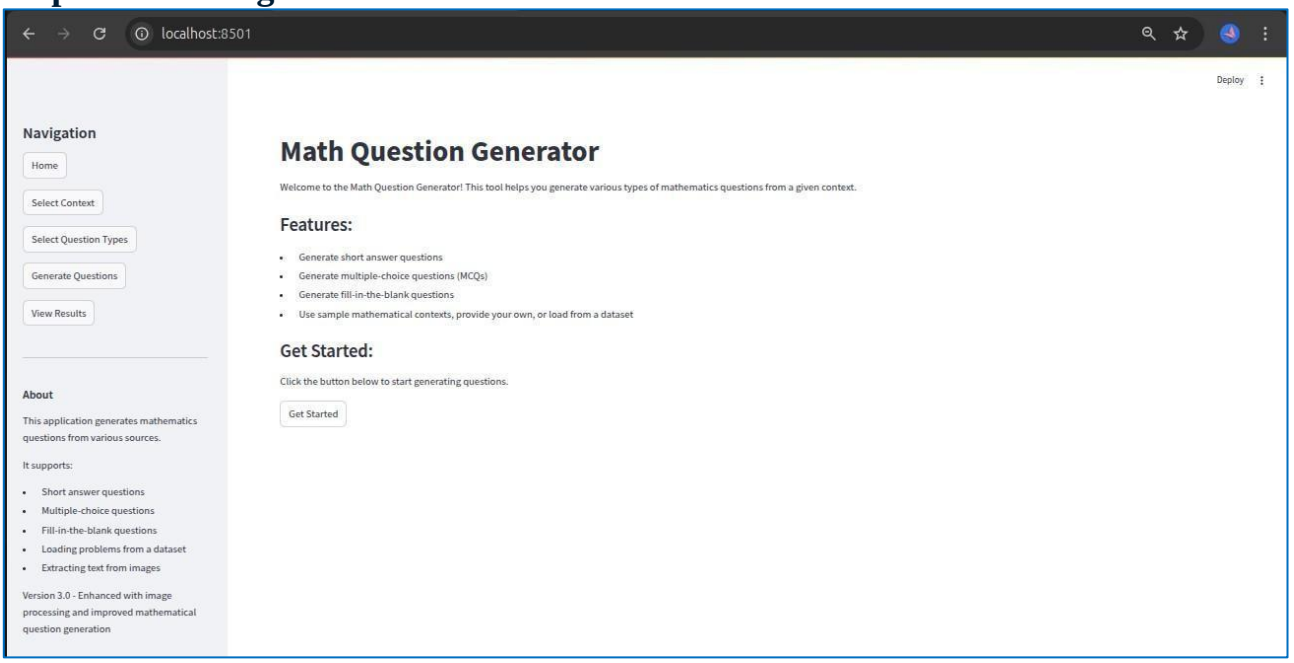


- ----Evaluation metrices for Different Question Types:

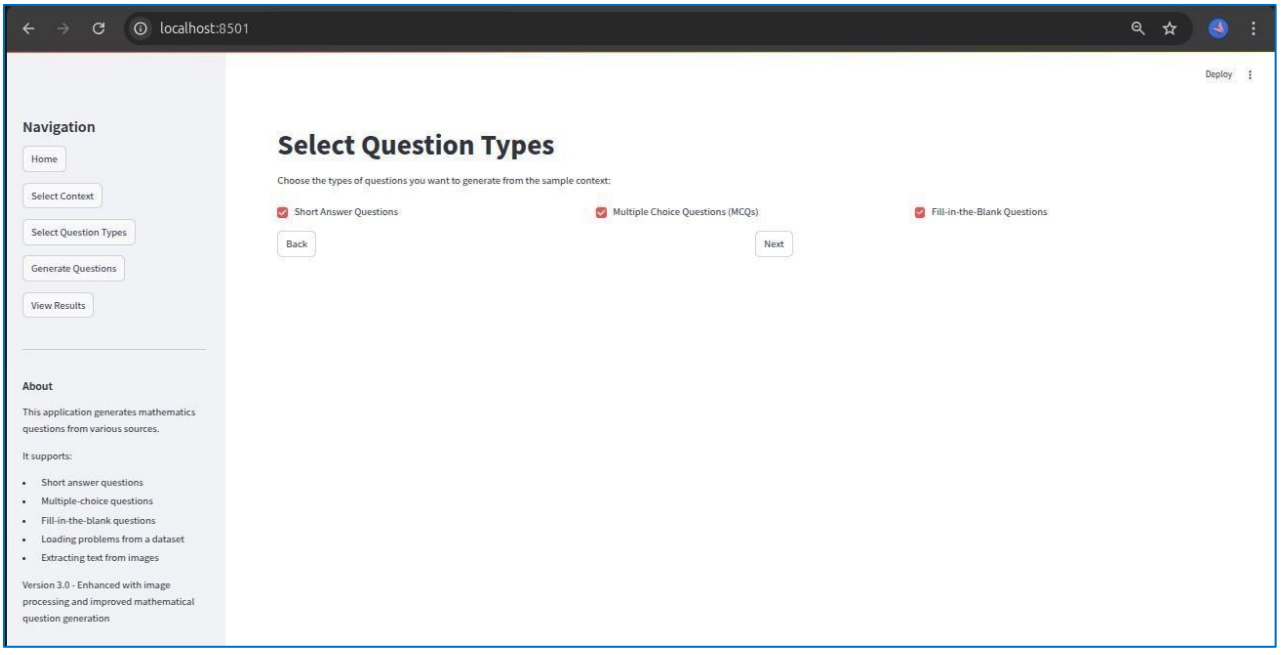
Question Type	BLEU score	Diversity Metrics	METEOR score	Mathematical Accuracy
MCQs	0.7800	0.8200	0.8500	High
Short Answer	0.6500	0.7000	0.7200	Medium
Fill in the blank	0.7200	0.7500	0.7000	High

10. Screenshots and User Interface:

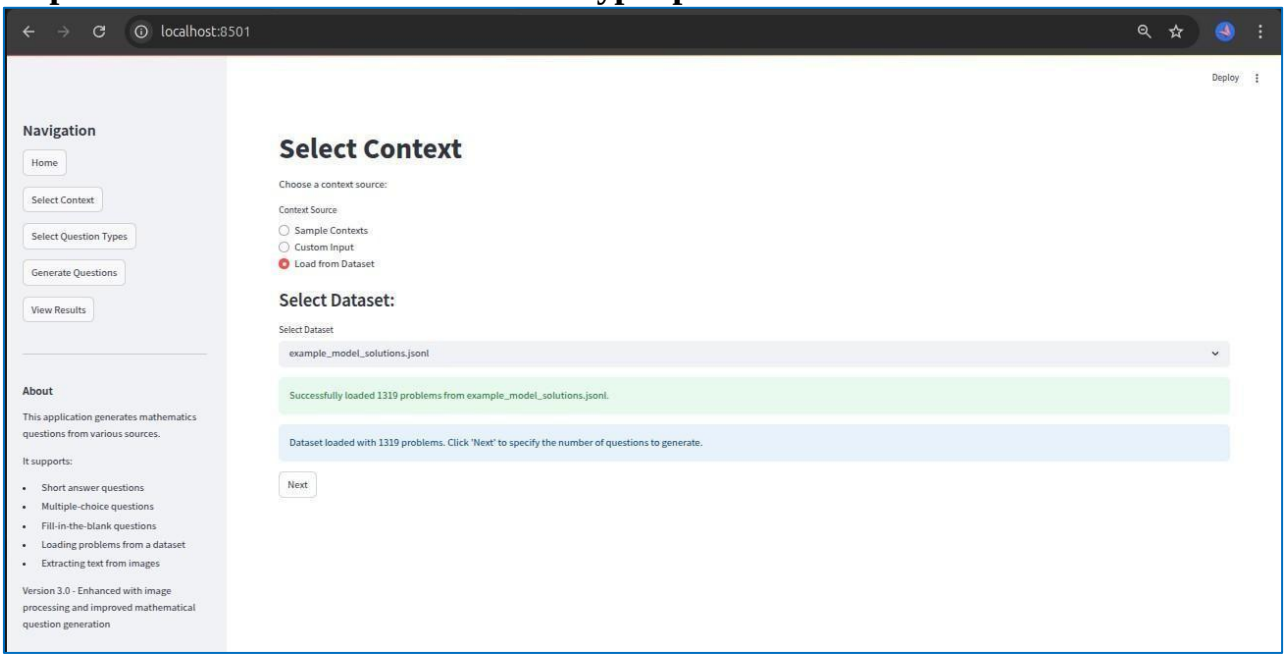
Step-1. Home Page:



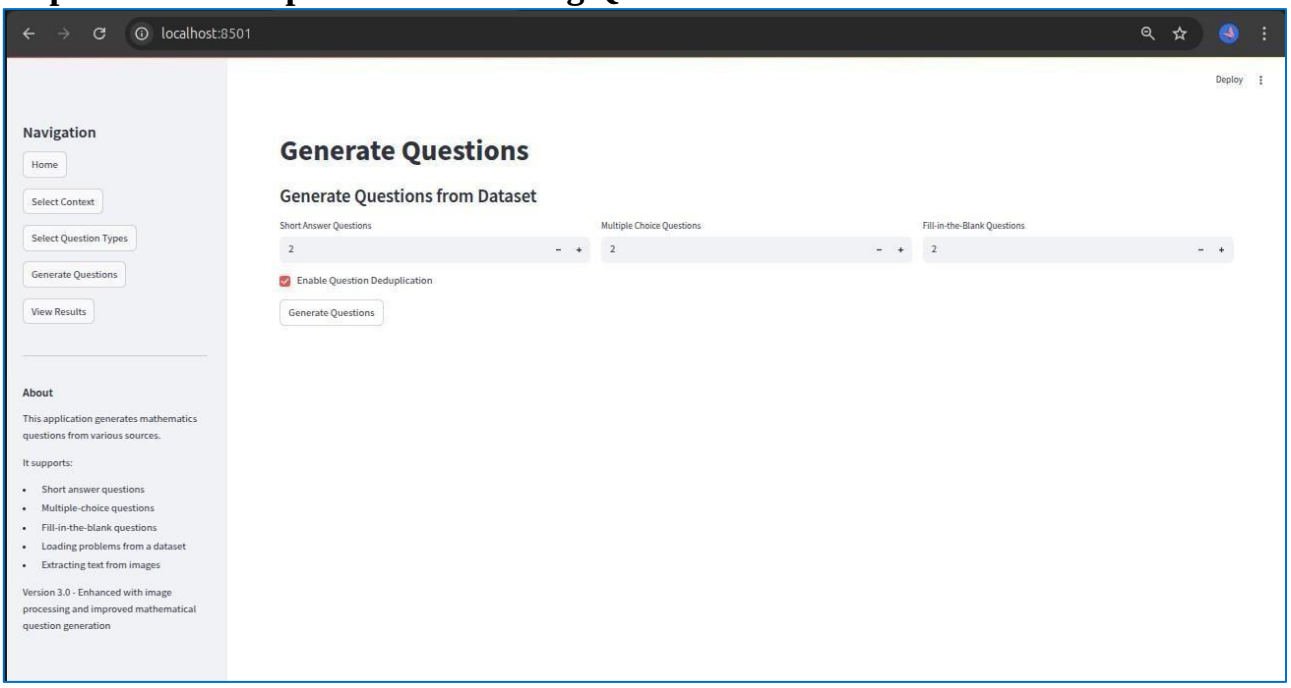
Step-2: User Selects the Types of Questions:



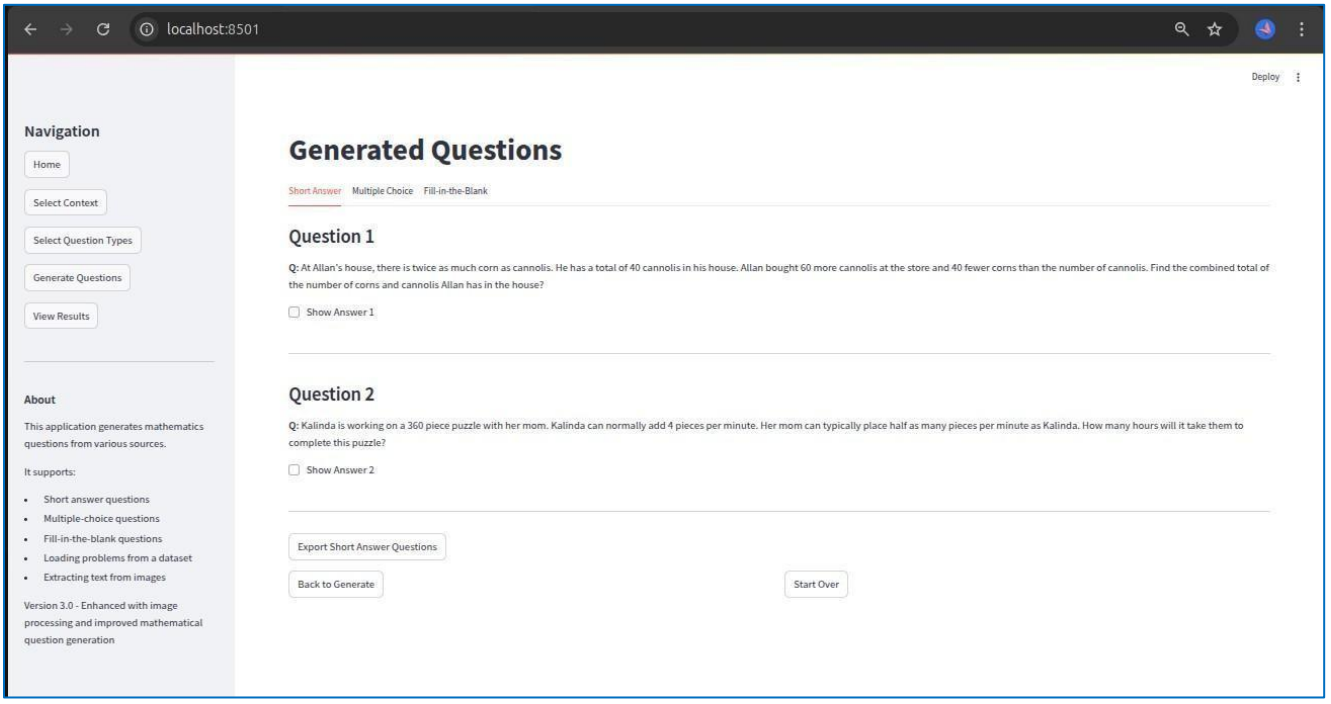
Step-3: User Selects one of the context types provided:



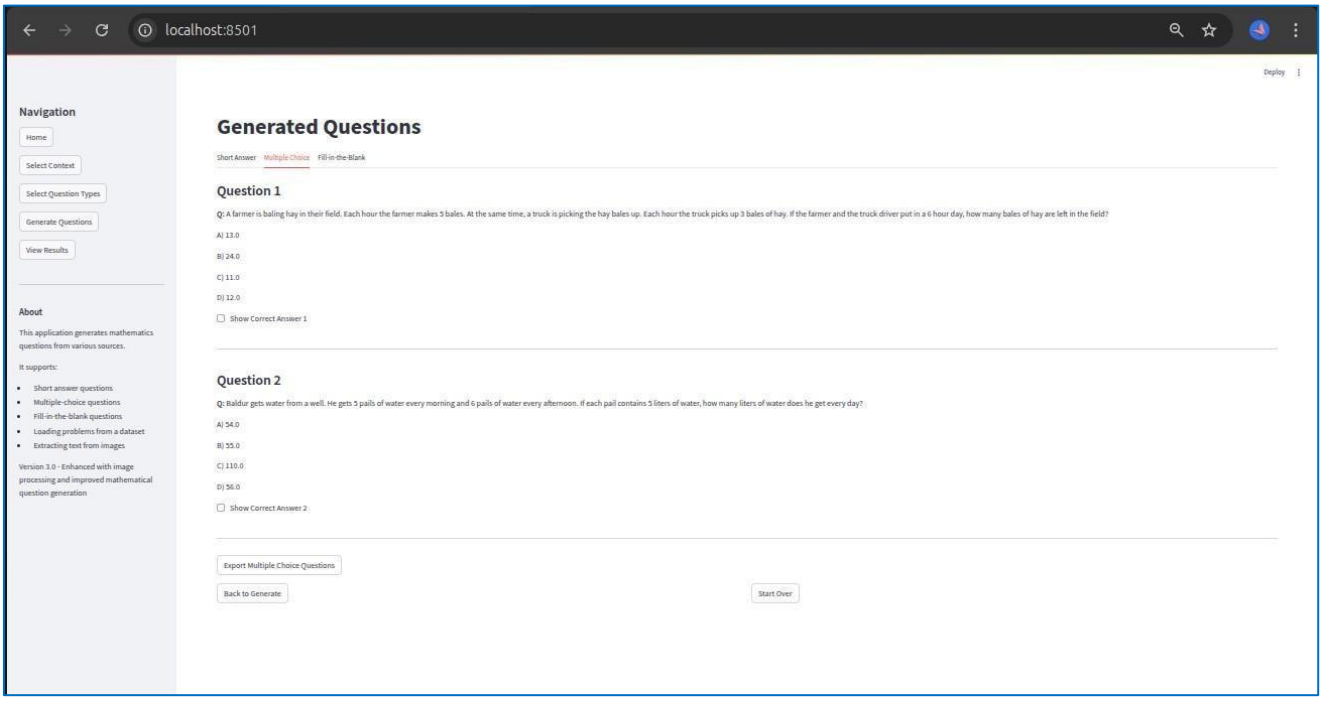
Step-4: Previous option of Generating Questions:



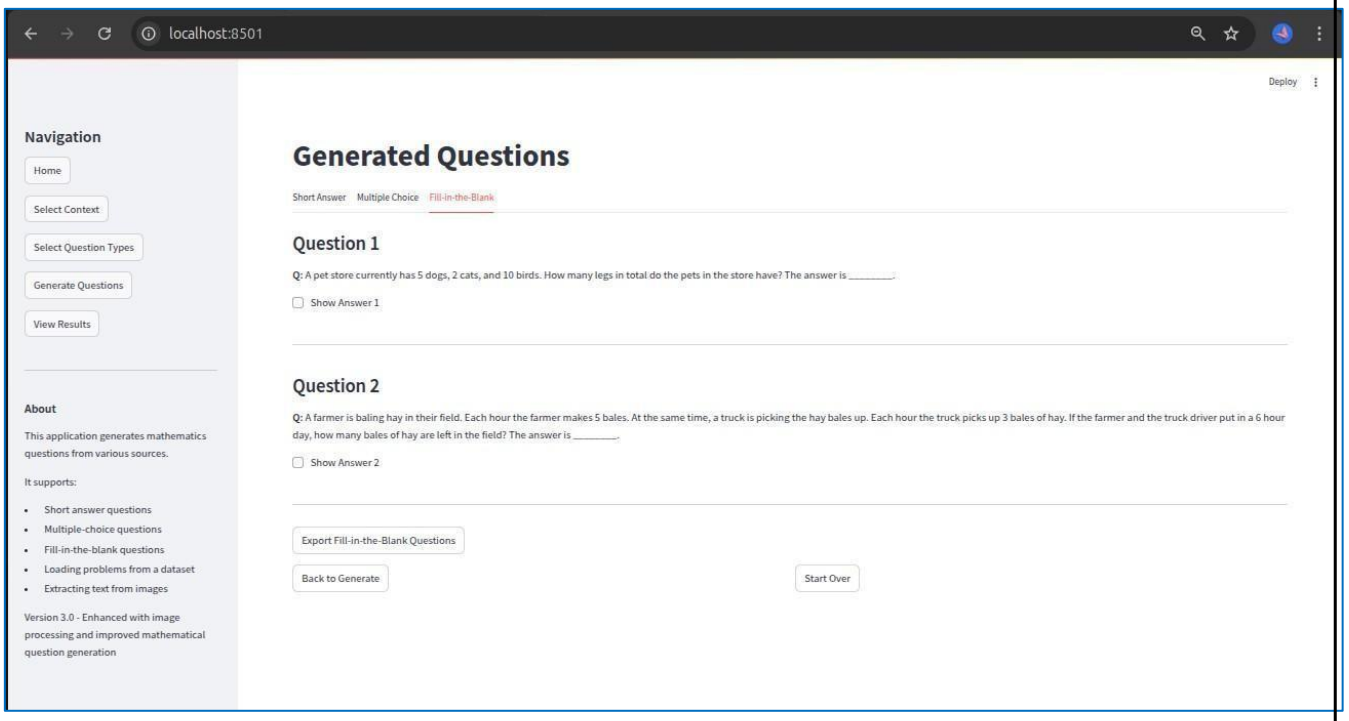
Step-5: Short Answer Type Questions Generated:



Step-6: MCQ Type Questions Generated:

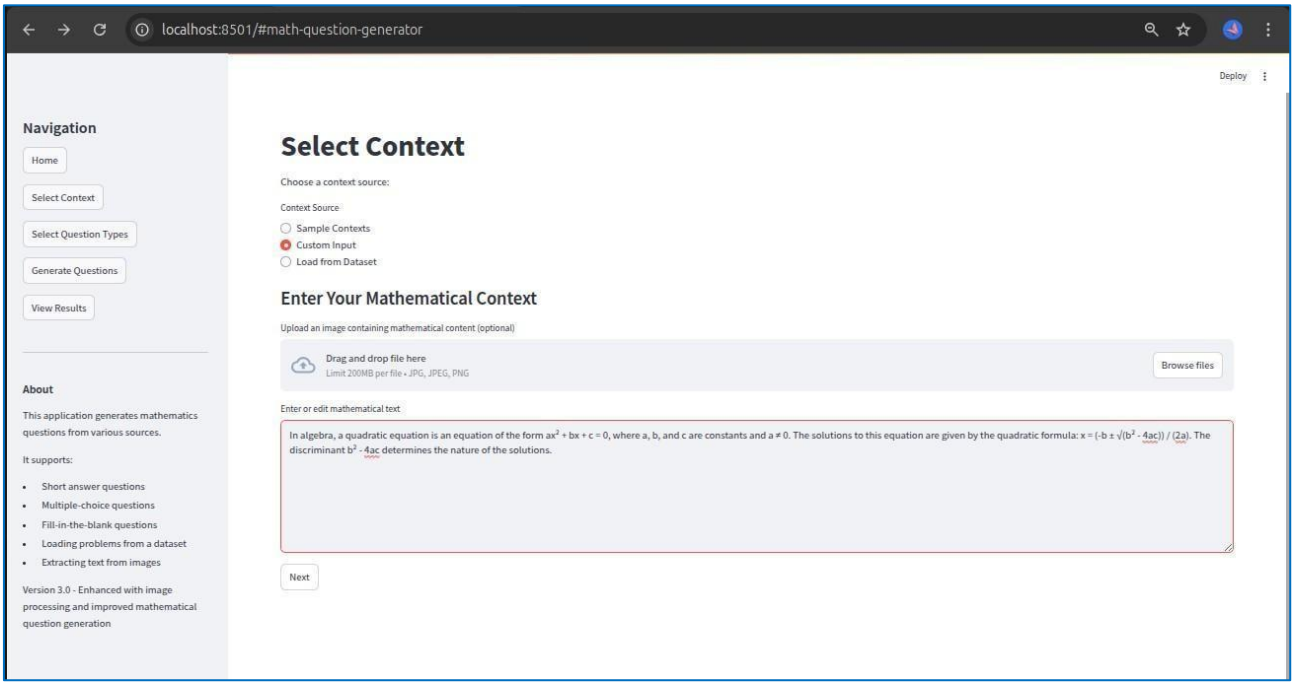


Step-7: Fill in the blanks Questions Generated:

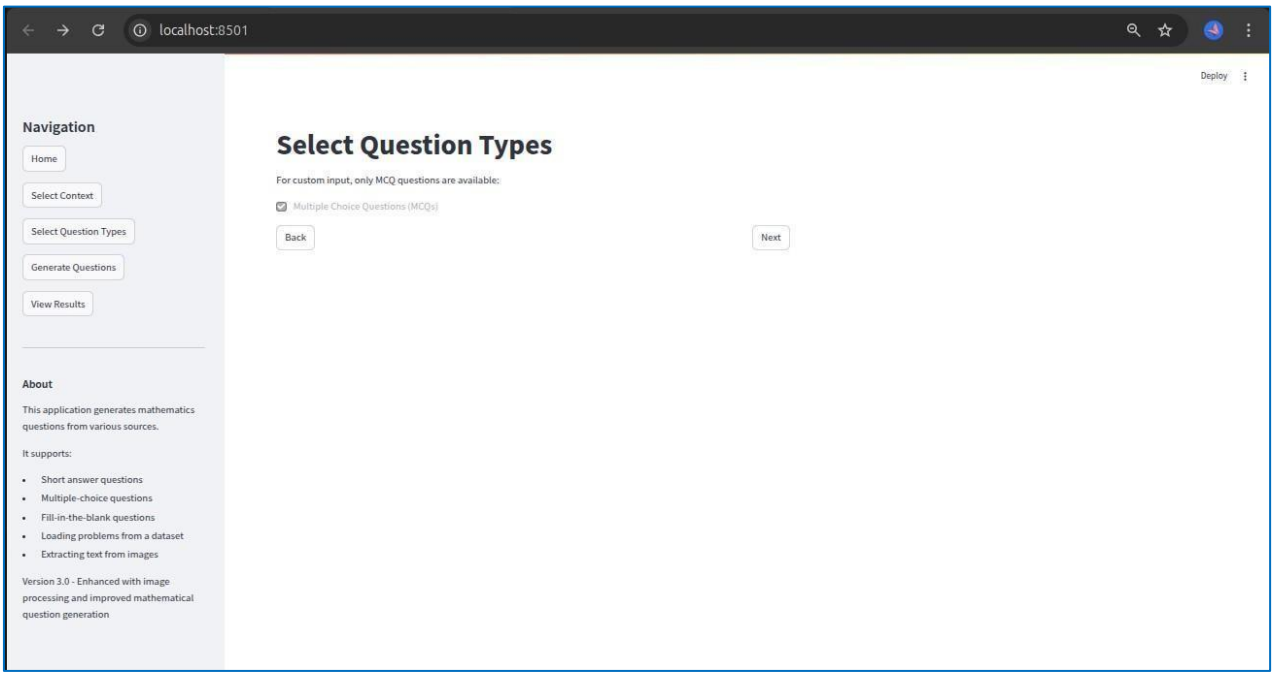


Optional Step:

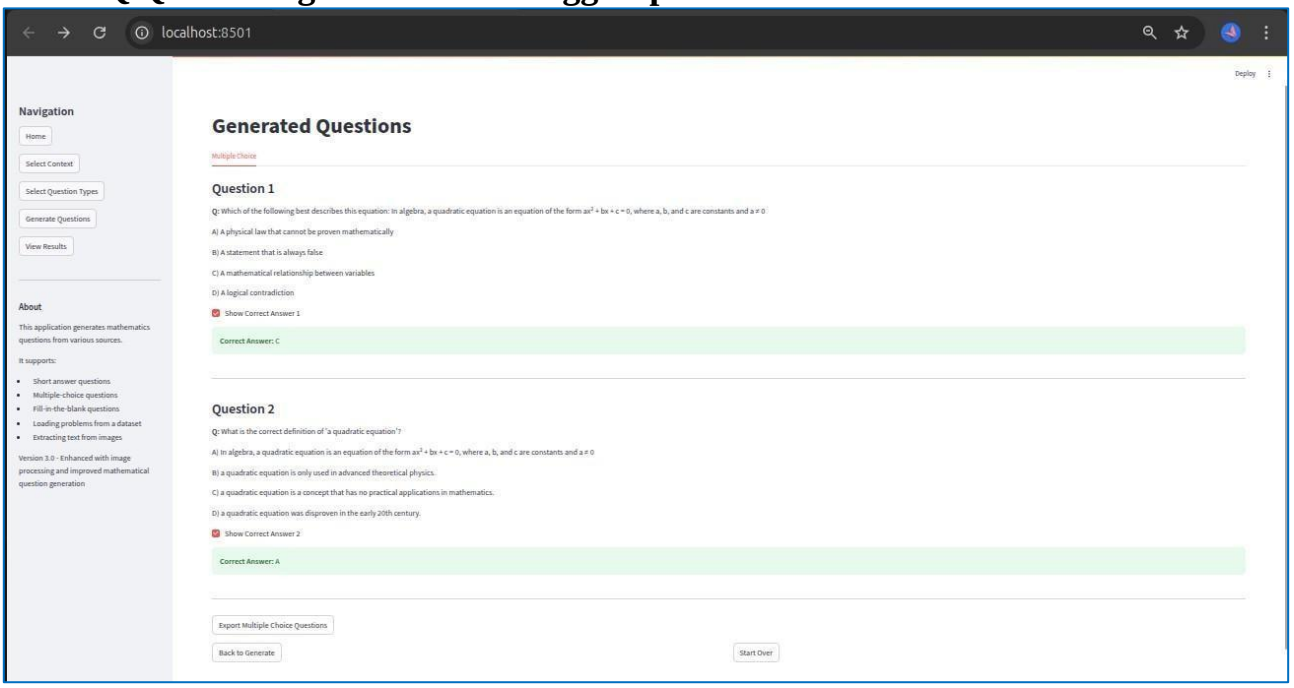
a. User can also enter own paragraph or Upload Files:



b. For Custom Input only MCQ Questions:



C. MCQ Questions generated with toggle option for Answer:



11. Conclusion and Future Works:

Significant improvements to Math Question Generator, now capable of handling innumerable nuances of mathematical notation, equations, and so on. Now a system correctly offers text with mathematical content in custom input, creates nicely formatted questions with different question formats (MCQ, fill-in-blank, and short answer) and randomizes correct answers in position selecting randomly from available options. This enhanced pattern recognition allows the generator to extract equations, formulas, numbers, and variables from mathematical contexts and generate more accurate and mathematically- inclined questions. There is still need for future work despite all this improvement. It could be improved with support for more complex mathematical expressions, such as those involving integrals, limits, or matrices.

Some dependencies by default on fresh installs would be; LaTeX renderer for better visual rendering of mathematical expressions. The addition of a math expression parser would enable the system to compute solutions on the fly, rather than relying solely on pre-computed solutions. According to the initial analysis, the data is somewhat biased towards the variables being treated, particularly for higher-level topics in calculus, linear algebra, and statistics, and could improve with additional tuning of the question generation algorithms to more accumulated and difficult questions. Finally, using a machine learning technique to get sense of context and develop context aware questions would greatly augments the relevance and quality of output questions.

12. References:

- Heilman, M., & Smith, N. A. (2010). Good question! Statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 609-617).

- Du, X., Shao, J., & Cardie, C. (2017). Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers) (pp. 1342-1352).

- Polozov, O., O'Rourke, E., Smith, A. M., Zettlemoyer, L., Gulwani, S., & Popović, Z. (2015). Personalized mathematical word problem generation. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

- Kumar, V., Balasubramanian, S., Nenkova, A., & Joshi, A. (2022). Automatic generation of high-quality multiple-choice math word problems. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.

- Zhang, S., & Bansal, M. (2019). Addressing semantic drift in question generation for semi-supervised question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing* (pp. 2495-2509).

- Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics* (pp. 311318).