Generating Mathematics Questions Using AI

Introduction to AI-Generated Math Questions

What is AI-Driven Math Question Generation?

- AI creates mathematical questions with help of data.
- Used in education and adaptive learning.
- Ensures efficient and scalable question creation.

Importance of AI in Mathematics

Why Use AI for Math Questions?

- Saves time for educators.
- Provides a wide range of questions.
- Personalized learning for students.
- Enhances adaptive assessments.

Types of Maths Questions AI Can Generate.

Categories of Questions:

1. **Basic Arithmetic** - Addition, subtraction, multiplication, division.
2. **Algebraic Equations** - Solving for unknown variables.
3. **Geometry** - area, perimeter.
4. **Probability & Statistics** - Mean, median, probability.
5. **Word Problems** - Real-world mathematical applications.
6. **Higher Mathematics** - Calculus, linear algebra.

Types of Maths Questions:

→ MCQ(Multi Choice Questions)

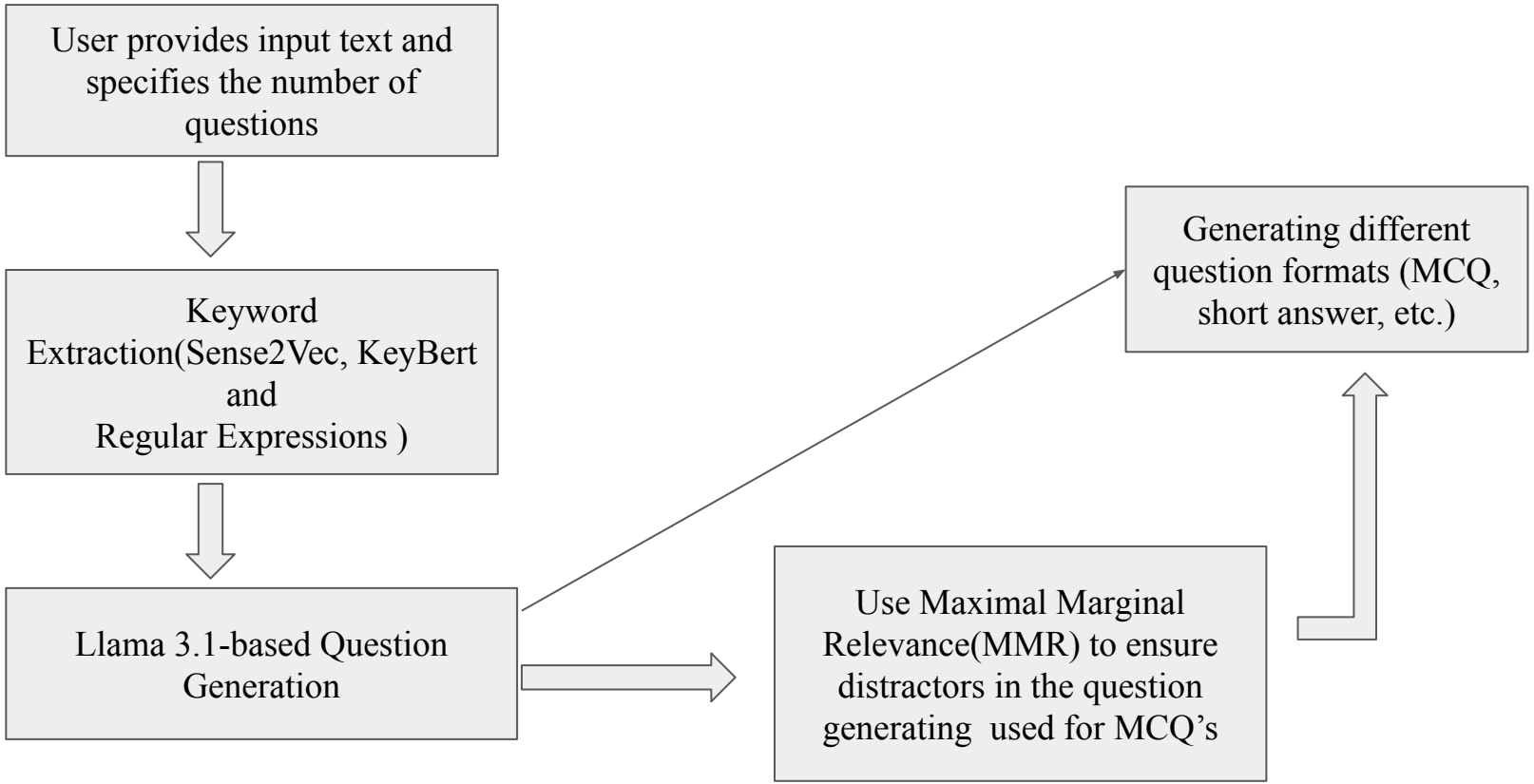
→ Short Answers

Methods AI Uses to Generate Math Questions.

Techniques Involved:

- **Rule-Based Models** - Uses predefined templates and logic.
- **Machine Learning Models** - Learns from data to create new questions.

Project Flow:



Step 1:User Input:

Data:“Generate a Quadratic equation question and find its roots.”

Number of Question:2

Step2: KeyWord Extraction(Sense2Vec & KeyBERT):

AI extracts key terms related to topic.

Extracted Keywords:

→ Quadratic, equations,root.

Sense2Vec :

Sense2Vec is an improved version of Word2Vec that considers the context and meaning of words. Unlike Word2Vec, which treats all occurrences of a word the same, Sense2Vec differentiates between different meanings (senses) of a word based on its part of speech (noun, verb, etc.) or entity type (person, location, etc.).

- Word2Vec represents words based on their surrounding context but treats all occurrences of a word the same.

In Simple Terms:

- **Word2Vec:** "Apple" (company) and "Apple" (fruit) are treated the same.
- **Sense2Vec:** Recognizes "Apple|NOUN" (fruit) vs. "Apple|ORG" (company) as different meanings.

KeyBERT (Keyword BERT): is a keyword extraction technique that uses **BERT embeddings** to find the most relevant words or phrases from a given text. Instead of relying on frequency-based methods, it understands the **context** of words and extracts keywords that truly represent the main topics.

How It Works:

1. Converts the input text into BERT embeddings (vector representations).
2. Compares each word's embedding with the entire text's embedding.
3. Ranks the words/phrases based on semantic similarity and selects the most relevant ones as keywords.

Example:"Artificial Intelligence is transforming industries by automating complex tasks and improving efficiency."

Extracted Keywords (Using KeyBERT):

- **Artificial Intelligence**
- **automating tasks**
- **improving efficiency**
- **transforming industries**

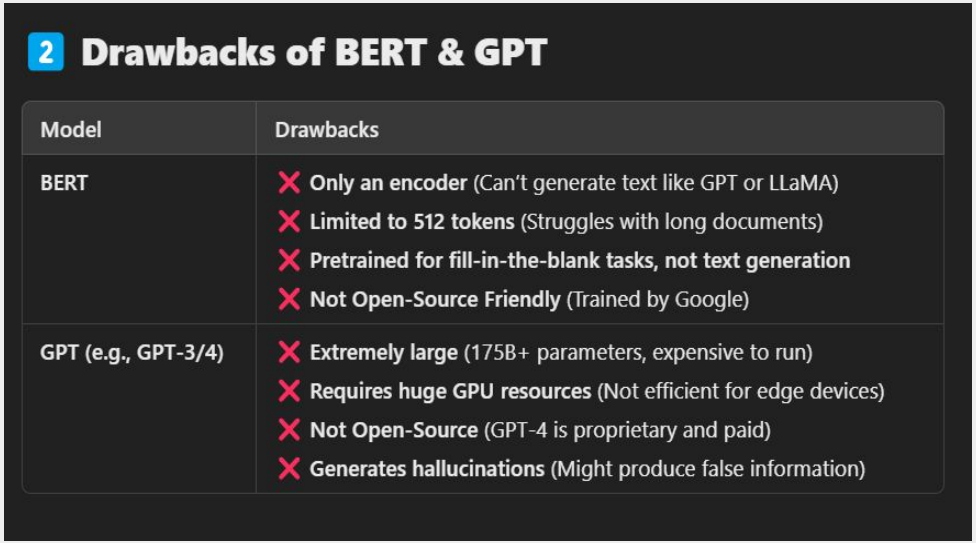
Step 3:LLaMA 3.1 Model Generate a Question:

Once the keywords are extracted, the **LLaMA 3.1 model** generates a relevant question based on the identified key concepts.

Example:Extracted Keywords: Quadratic Equation, Roots, Quadratic Formula

Generated Question:"Find the roots of the quadratic equation 2x2−3x+1=02x^2 - 3x + 1 = 02x2−3x+1=0 using the quadratic formula."

LLaMA 3.1 is a cost-effective, high-performance, and context-aware model, making it ideal for automated question generation compared to older models like GPT, T5, or BERT.



Step 4: MMR Ensures Distractors for MCQ Generation

What is Maximal Marginal Relevance (MMR)?

Maximal Marginal Relevance (MMR) is a technique used in **information retrieval and text summarization** to ensure diversity in selected items while maintaining relevance. In the context of **question generation**, MMR helps in selecting **distractors (wrong answer choices)** for Multiple Choice Questions (MCQs) that are:

1. **Relevant** to the question.
2. **Diverse**, meaning they are not too similar to each other or to the correct answer.

Example: Which planet is known as the **Red Planet**?

A) Mars ✅ (Correct)

B) Apple ❌ (Unrelated) ⟹ Bad Distractors

C) Earth ❌ (related) ⟹ Good Distractors

Step 5: Generating MCQs and Short Questions, the system creates different types of questions based on the input data.

1. **MCQs (Multiple Choice Questions)**
   - These are questions where a user selects the correct answer from multiple options.
   - **Example:**

     What is the capital of France?

     a) Berlin

     b) Madrid

     c) Paris ✅

     d) Rome
2. **Short Answer Questions**
   - These require a brief, open-ended response.
   - Example:

     What is the capital of France?

     Answer: Paris

GSM8K Dataset: Overview and Usage

1. Introduction to GSM8K:

GSM8K (Grade School Math 8K) is a high-quality dataset of **grade school-level math word problems**. It was designed to evaluate and improve the problem-solving capabilities of language models in arithmetic reasoning.

2. Key Features of GSM8K:

- Contains **8,500** math word problems.
- Problems are written in **natural language**.
- The dataset is **manually curated** to ensure high quality.
- Useful for training and evaluating **mathematical reasoning models**.

3. Types of Problems in GSM8K:

The dataset includes problems related to:

- Basic arithmetic (addition, subtraction, multiplication, division)
- Fractions and decimals
- Ratios and proportions
- Time, distance, and speed
- Word-based algebraic problems

4.Example Problem from GSM8K:

{"question": "Janet\u2019s ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day at the farmers' market?",
"answer": "Janet sells 16 - 3 - 4 = <<16-3-4=9>>9 duck eggs a day.\nShe makes 9 * 2 = $<<9*2=18>>18 every day at the farmer\u2019s market.\n#### 18"}

5.Test and Train Files in GSM8K:

- **Train Set (train.jsonl)**: Contains **7,500** problems for model training.
- **Test Set (test.jsonl)**: Contains **1,000** problems to evaluate model performance.

6.Why so we have two types of files:

The dataset has both **train** and **test socratic** files because they serve different purposes:

1. **Test Set &Train (test.jsonl & train.jsonl)**:
   - This file contains **standard** math word problems and solutions.
   - It is used to **evaluate** an AI model's ability to solve problems **independently** without additional guidance.
2. **Test Socratic & Train Socratic Files (test_socratic.jsonl & train_socratic.jsonl)**:
   - This version **breaks down problems into guided steps** using **Socratic questioning** (asking smaller, leading questions).
   - Helps measure how well AI models **reason through a problem** rather than just predicting the answer.
   - Useful for **prompt engineering** and improving the interpretability of AI responses.

7.How to Download GSM8K:

GitHub Repository: https://github.com/openai/grade-school-math

SVAMP Dataset: Overview and Usage

- **SVAMP (Simple Variations of Arithmetic Math Problems)** is a **benchmark dataset** for evaluating the generalization ability of machine learning models in solving math word problems.
- It is designed to test whether an AI model can **solve math problems in different linguistic variations** while maintaining the correct mathematical reasoning.
- **Key Focus:** Evaluates robustness of AI models in solving math word problems by introducing **linguistic variations** to existing questions.
- **Example:** **"**Emma went to the market and bought 3 apples. She already had 5 apples at home. How many apples does she have now?"
- **GitHub Repository**: https://github.com/arkilpatel/SVAMP.git

Dataset Structure & Columns

The dataset consists of **980 rows** and **9 columns**. Below is a breakdown:

Column Name	Description	Example
Question	Unique Id or reference for the Math Problem	“number0”
Numbers	The numerical values used in the Math Problem	“7 2”,”6 9”
Equation	The mathematical equation needed to solve the problem	“7 + 2”
Answer	The final correct numerical answer	“9”
Group Number	Categorizes similar types of questions together	“[1,2,3,4]”
Grade	The intended grade level for the problem	“1” (Grade 1)
Type	The mathematical operation type.	“Addition”
Body	The structured problem statement before generating the full question	“Ellen has n”
Ques_Statement	The final, complete math word problem.	“How many apples are in the basket”

AIMO External Dataset:

It is an external Dataset to be used in the AI Mathematical Olympiad Kaggle competition.
This dataset is a compiled version of two benchmark math dataframes for solving math problems such as:
MATH: "MATH is a new dataset of 12,500 challenging competition mathematics problems.
GSM8K: "a dataset of 8.5K high quality linguistically diverse (variety of languages spoken by people around the world) grade school math word problems created by human problem writers.

Download Dataset by the following code:

```
import kagglehub
path = kagglehub.dataset_download("alejopaullier/aimo-external-dataset")
print("Path to dataset files:", path)
```

Dataset contains the following column names:

Problem	Level	Type	Solution	Stage	Source

This dataset contains total 21,293 rows and 6 columns.

Detail Explanation of the dataset:

Column Name	Description	Example
Problem	Contains detailed math problems. Usually written in descriptive form (e.g., algebraic word problems or geometric challenges).	"Solve for x: 3x + 2 = 11"
Level	Indicates the difficulty of the problem.Ranges from Level1(easy) to Level5(Hard)	"Level 1"
Type	Specifies the specific category(Eg:Algebra, Geometry,Calculus)	"Algebra"
Solution	Outlines the solution or strategy to solve the problem.Sometimes include the steps to approach the problem.	"First subtract 2 from both sides."
Stage	Indicates the current phase of the AI model’s learning process.	"train"
Source	The origin of the problem.Provides insight into where the problems were sourced.	"MATH"

1. BLEU Score (Bilingual Evaluation Understudy):

A BLEU (BiLingual Evaluation Understudy) score is a metric that measures how similar a machine-translated text is to a reference translation.

It measures the precision of n-grams (usually unigram, bigram, trigram, and 4-gram) between the generated question and the reference question.

BLEU is calculated by finding the ratio of matched n-grams between the generated and reference question. For brevity, there is also a penalty for very short generated texts (brevity penalty).

where,

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

- P_n is the precision for n-grams (unigram, bigram, etc.),
- w_n 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.

Steps to Calculate BLEU Score:

Example:

We have a **generated question** and a **reference question**:

Reference question: "What is the sum of 3 and 5?"

Generated question: "What is 3 plus 5?"

1. Preprocessing:

We tokenize both the reference and generated questions into unigrams:

Reference: ["What", "is", "the", "sum", "of", "3", "and", "5"]

Generated: ["What", "is", "3", "plus", "5"]

2. Count n-grams:

For **unigrams (n=1)**, we count how many words in the generated question match those in the reference.

Reference unigrams: "What", "is", "the", "sum", "of", "3", "and", "5"

Generated unigrams: "What", "is", "3", "plus", "5"

So, there are **4 matches** out of 5 words in the generated question.

3. Precision:

For **unigrams**:

$$p_1 = \frac{\text{Number of matched unigrams}}{\text{Total number of unigrams in the generated question}}$$

$p_1 = \frac{4}{5} = 0.8$

4. **Brevity Penalty (BP)**:

BP=exp(1- length of generated/ length of reference)

= exp(1- $\frac{5}{8}$)

=exp(0.375)≈1.454.

5. **Calculate the BLEU score**:

$BLEU = BP \times \exp(\Sigma w_n \log(p_n))$

$BLEU = BP \times \exp(w_1 \log(p_1))$ (since unigrams, N=1)

$BLEU = 1.454 \times \exp(1 \times \log(0.8))$ (since $w_1=1$ for unigram & $p_1=0.8$)

$BLEU = 1.454 \times 0.907 \approx 1.318$.

METEOR Score: A Step-by-Step Process

1: Introduction

Title: Understanding METEOR: A Better Evaluation Metric

Key Points:

- METEOR (**Metric for Evaluation of Translation with Explicit ORdering**) is an NLP evaluation metric.
- Used for **machine translation, text summarization, and question generation**.
- Improves over BLEU by incorporating **semantic similarity, stemming, synonyms, and word order**.

2: What Does METEOR Take as Input?

Inputs:

- **Generated Text** → AI or machine-generated text.
- **Reference Text** → Human-written text used as a benchmark.

Example:

- **Reference:** "Find the sum of 2 and 3."
- **Generated:** "Calculate the addition of 2 and 3."

3: Step-by-Step Process in METEOR

Step 1: Tokenization & Preprocessing

- Splits text into **tokens (words, punctuation, etc.)**.
- Converts text into **lowercase**.
- **Types of Tokenization Used:**
  - **Word Tokenization** → Splits into words.
  - **Whitespace Tokenization** → Uses spaces as delimiters.
  - **Punctuation Handling** → Removes unnecessary punctuation.
  - **Stemming & Lemmatization** → Converts words to root form.

Step 2: Exact Word Matching

- Identifies **direct word matches** between generated and reference text.
- **Finds identical words** → Words that are present in both the generated and reference texts in the same form.
- **Assigns a match score** → The more exact matches, the higher the initial similarity score.

Step 3: Stemming & Synonym Matching

- Uses **WordNet** to find word stems and synonyms.
- Example:
  - Reference: *"Find the sum of 2 and 3."*
  - Generated: *"Calculate the addition of 2 and 3."*
  - Recognizes **"sum" and "addition"** as similar.

Stemming Match:

- **Partial credit** (e.g., 0.8–0.9 weight) for words that match after stemming.

Synonym Match:

- **Partial credit** (e.g., 0.6–0.8 weight) for words that are synonyms according to **WordNet**.

Step 4: Precision, Recall & F-Mean Calculation

Precision (P):

- **Precision (P) = (Number of matched words) / (Total words in generated text)**

Recall (R):

- **Recall (R) = (Number of matched words) / (Total words in reference text)**

F-Mean Score:

- Balances **precision** and **recall** with higher weight on recall.
- Formula:
- **F-Mean = (10 × P × R) / (R + (9 × P))**
- METEOR places **more emphasis on recall** than precision because recall is more important for capturing the meaning of the reference sentence.
- **Why High Recall is Needed**
  Reference: "Find the sum of 2 and 3."
  Generated 1 (High Precision, Low Recall): "Find sum." → ✅ Matches "Find" and "sum" but **misses key details** ("of 2 and 3").
  Generated 2 (High Recall, Balanced Precision): "Calculate the addition of 2 and 3." → ✅ **More words covered**, even though "Calculate" and "addition" differ from "Find" and "sum".

Step 5: Word Order Penalty Computation

- Identifies incorrect **word ordering**.
- Computes a **penalty score** based on the number of chunks (continuous matched sequences).
- **Penalty = 0.5 × (Chunks / Matched words)**

Example of Chunks
Reference:
"The quick brown fox jumps over the lazy dog."
Generated (Correct Order, One Chunk):
✅ "The quick brown fox jumps over the lazy dog."

- Matched words: **9**
- Chunks: **1**
- **Lower penalty (better score)**

Generated (Incorrect Order, Multiple Chunks):
❌ "The fox jumps over the lazy dog quick brown."

- Matched words: **9**
- Chunks: **3** (["The", "fox", "jumps", "over", "the", "lazy", "dog"], ["quick"], ["brown"])
- **Higher penalty (worse score)**

Step 6: Final METEOR Score Calculation

- Uses formula:
  METEOR Score = F-mean × (1 - Penalty)
- Generates a **final similarity score (0 to 1).**

4: METEOR's Output

Outputs:

- **METEOR Score (0-1):** Higher score = better similarity.
- **Precision & Recall Values:** Intermediate calculations.
- **Penalty Score:** Deduction for incorrect word order.

5: Advantages of METEOR Over BLEU

✅ **Considers synonyms & stemming** (unlike BLEU) ✅ **Better recall-based scoring** ✅ **More aligned with human judgment**


Thank you!