EDU-QUEST

AUTOMATIC QUESTION GENERATION SYSTEM

Name: Talha Khuram, Ahmad Aqeel

Roll no: 22i-0790, 22i-1134

Section: D, B

Final Project Report

1. INTRODUCTION:

EduQuest is an intelligent automatic question generation system designed to generate educational questions directly from PDF documents using advanced Retrieval-Augmented Generation techniques. The system is developed to address the increasing demand for automated educational content creation in modern digital learning environments. With the rapid growth of online learning platforms, educators face significant challenges in continuously generating high-quality, diverse, and relevant assessment material. EduQuest provides an efficient solution that significantly reduces the manual effort required to create questions while maintaining educational relevance and quality.

The system allows users to upload PDF documents such as lecture notes, research papers, textbooks, and technical documents. These documents are processed using a semantic-based retrieval pipeline, after which state-of-the-art large language models generate multiple types of questions. The three supported question types are Multiple Choice Questions, Short Answer Questions, and Long Answer or Descriptive Questions. Each generated question is directly based on the most relevant context extracted from the uploaded document.

By combining semantic search with powerful generative AI models, EduQuest ensures that the generated questions remain contextually accurate and educationally meaningful. The system also includes an evaluation framework that validates the structure, completeness, and quality of the generated questions. This helps in maintaining a reliable standard of output across different models.

Key objectives of the system include automating the question generation process from PDF documents, supporting multiple question types for diverse academic assessments, enabling comparative evaluation of different AI models, assisting educators in quickly generating high-quality assessment content, and demonstrating the real-world effectiveness of Retrieval-Augmented Generation in education.

2. MOTIVATION:

The motivation behind EduQuest arises from several important challenges currently faced in educational institutions and digital learning platforms.

One major challenge is the burden of educational content creation. Teachers and instructors spend a significant amount of time creating quizzes, tests, assignments, and examinations. This manual effort reduces the time available for teaching, mentoring, and student interaction. EduQuest aims to automate this repetitive task and allow educators to focus on more impactful learning activities.

Another key challenge is scalability. With the increasing adoption of digital learning systems, there is a growing need for large and diverse question banks. Creating such large volumes of questions manually is neither practical nor efficient. An automated system like EduQuest provides a scalable solution that can generate numerous questions in a short time.

Consistency and quality also serve as strong motivational factors. Human-generated questions often vary in difficulty, structure, and quality. An automated system ensures uniform formatting, consistent difficulty levels, and standardized quality across different question sets.

Accessibility is another important motivation. Not all educators possess expertise in assessment design. EduQuest democratizes the process of quality assessment creation by enabling any instructor to generate well-structured questions regardless of prior experience.

From a research perspective, this project explores the effectiveness of several AI models including rule-based systems, local large language models, and API-based cloud models. The project contributes to understanding how Retrieval-Augmented Generation can be practically used in education.

Finally, the project demonstrates the practical application of cutting-edge technologies such as vector databases, prompt engineering, large language models, and document processing frameworks in a real-world scenario.

3. SYSTEM ARCHITECTURE:

- Overall System Architecture

EduQuest is built on a modular Retrieval-Augmented Generation architecture. The system begins by taking a PDF document as input. The document is first loaded using a document loader module. The extracted text is then divided into smaller overlapping chunks using a text splitter. Each chunk is converted into numerical embeddings using a sentence embedding model. These embeddings are stored in a vector database.

When a user requests question generation, the query is converted into an embedding and compared with the stored document embeddings. The most relevant chunks are retrieved and passed into a prompt template along with instructions for the selected question type. This prompt is then sent to a Large Language Model, which generates the final questions. The output is passed through a post-processing and evaluation module before being displayed to the user.

- Implementation Methods

The document processing pipeline begins with PDF loading using the PyMuPDF library. This library extracts raw text and metadata from PDF files. After this, the text is divided into overlapping chunks using Recursive Character Text Splitter. The default chunk size is set to 1000 characters, and the overlap between consecutive chunks is 200 characters to preserve contextual continuity.

Next, embeddings are generated using the BAAI bge-small-en-v1.5 embedding model, which produces 384-dimensional vector representations of text. These embeddings are stored in a ChromaDB vector database, which enables fast and efficient similarity search.

The question generation process begins with context retrieval through semantic search. The top four most relevant document chunks are retrieved by default. These chunks are then formatted into structured prompt templates that define the desired output format for each question type. The formatted prompt is passed to the selected language model for question generation. The system then validates the generated output and formats it for display.

The evaluation framework checks whether the output meets predefined structural requirements. Quality metrics analyze completeness, relevance, and answer correctness. The evaluation system also enables model-wise comparison for research analysis.

4. MODELS USED:

- Phi-3-mini

Phi-3-mini is a local transformer-based Large Language Model developed by Microsoft. It contains approximately 3.8 billion parameters and runs using float-16 precision with a memory footprint of around 3.8 GB. The model is privacy-preserving and does not require internet access. However, when running on a CPU, it requires 8 to 10 GB of RAM and takes more than 15 minutes to generate a single question. Due to this limitation, it is disabled by default in the system.

- Gemini 2.0 Flash

Gemini 2.0 Flash is an API-based Large Language Model provided by Google Generative AI. It is the primary recommended model for EduQuest. It offers fast and reliable question generation with an average response time of 10 to 30 seconds. It provides high-quality outputs with a free-tier API option. The main disadvantages are dependency on internet connectivity and potential API rate limits.

- Baseline Rule-Based Model

The baseline model is a template-based rule system that uses pattern matching and predefined question structures. It generates questions instantly and does not depend on any external services. While it is fast and deterministic, it produces lower-quality output and lacks creativity. It is used mainly for performance comparison.

- Additional Models

Phi-2 and TinyLlama models are also supported by the system but are disabled by default. Phi-2 contains 2.7 billion parameters, while TinyLlama contains 1.1 billion parameters. These models are faster than Phi-3 but provide lower question quality.

5. API's USED:

The Google Generative AI API is used to access the Gemini 2.0 Flash model. Authentication is handled through API keys, and retry mechanisms are implemented to manage rate limits.

The Hugging Face Hub API is used for accessing pretrained models and embedding models. Most resources are publicly available.

The system is built using LangChain for orchestration of the Retrieval-Augmented Generation pipeline. ChromaDB is used for vector storage. PyTorch is used for local model inference. The Transformers library is used for loading local language models.

PyMuPDF is used for extracting text from PDFs. Gradio is used to build the web-based graphical user interface. Matplotlib, Pandas, and NumPy are used for dataset analysis and visualization.

Docker and Docker Compose are used for container-based deployment. Python version 3.10 or above is used as the main programming language.

6. IMPORTANT FEATURES AND ASPECTS

EduQuest supports multiple AI models with seamless switching and automatic fallback in case of failure. It supports MCQs, Short Answer Questions, and Long Answer Questions. The RAG-based retrieval pipeline ensures that each question is generated using relevant document context.

The evaluation system checks for proper question structure, presence of correct answers, and option completeness. BLEU and ROUGE metrics are used for similarity evaluation. The system also provides dataset analysis including document length distribution, word counts, and statistical visualizations.

Advanced features include an ablation study framework for chunk size, overlap, retrieval size, and temperature analysis. The system also includes retry logic, memory management, and optimized token usage.

7. EVALUATION METRICS AND MATRIX

The system evaluates generated questions based on several metrics including format validation, presence of question mark, number of options, correct answer detection, length appropriateness, and structural completeness.

The overall quality score is calculated using weighted contributions from these metrics. Performance is measured using generation time and compliance rate. BLEU and ROUGE scores are used when reference data is available.

Gemini achieved an average quality score between 0.75 and 0.85 with a generation time of 10 to 30 seconds. The baseline model achieved a quality score between 0.60 and 0.70 with near-instant generation. Phi-3 achieved high quality but suffered from extreme latency.

8. RESULTS

The system achieved a question generation success rate of over 90 percent across all models. Format compliance ranged between 85 to 95 percent for Gemini and 70 to 80 percent for the baseline model. Error rates remained below 5 percent, mainly due to API timeouts or memory overflow.

The ablation study showed that an optimal chunk size of 1000 characters, overlap of 200 characters, and retrieval size of 4 resulted in the best quality outputs. The dataset consisted of more than 12 PDF documents including research papers, lecture notes, and technical material.

9. LIMITATIONS

Local models are extremely slow on CPU and require large memory. The Gemini API is dependent on internet access and subject to rate limits. The system may miss important information in very long documents due to limited context window size. Images, diagrams, equations, and complex layouts are not handled effectively.

The quality evaluation is fully automated and does not include expert human validation. The difficulty level of questions is not automatically adjusted. The system does not currently support multi-document question generation, question editing, or export to learning management systems.

10. CONCLUSION

EduQuest successfully demonstrates the practical application of Retrieval-Augmented Generation for automatic question generation from educational documents. The system significantly reduces manual effort for educators while ensuring quality and relevance of generated questions.

Gemini 2.0 Flash proved to be the most effective model in terms of speed and quality. Local models remain impractical for CPU-based deployment. Retrieval-based generation significantly improved accuracy and relevance compared to direct generation without context.

Future work includes GPU acceleration, difficulty calibration, support for export formats, multi-document generation, human expert evaluation, and domain-specific model fine-tuning.

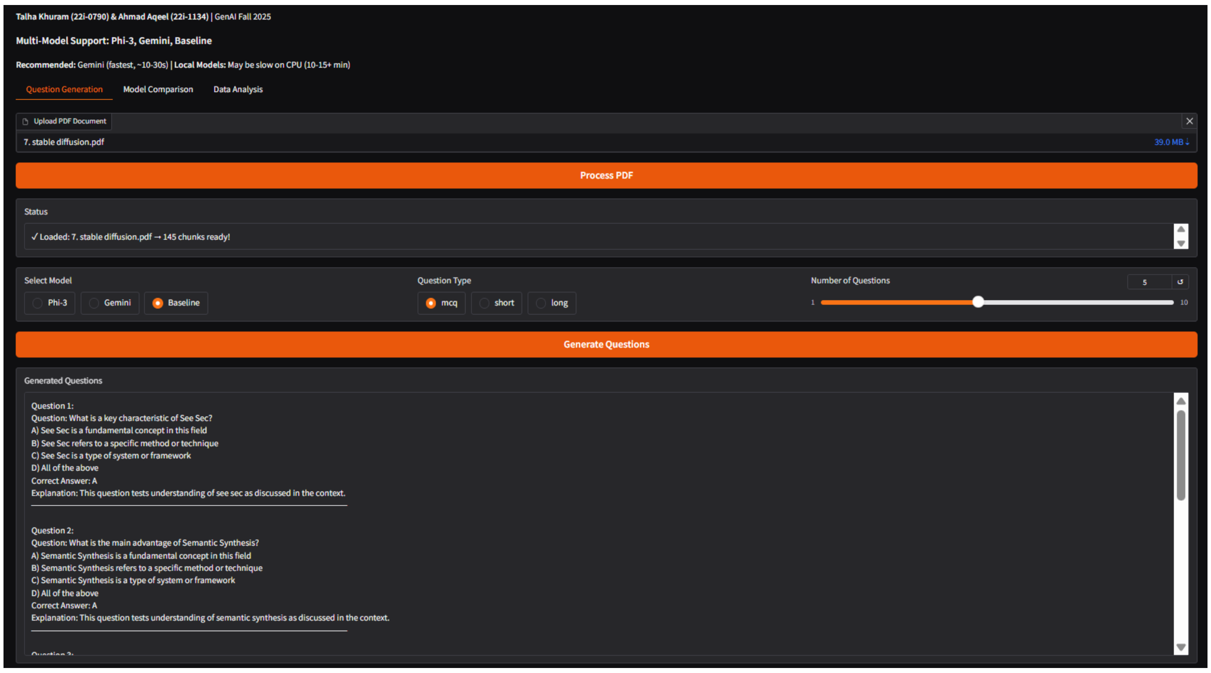
11. REFERENCES:

- Lewis, M. et al. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation. ACL. https://aclanthology.org/2020.acl-main.703
- Raffel, C. et al. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. JMLR. https://arxiv.org/abs/1910.10683
- Zhang, J. et al. (2020). PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. ICML. https://arxiv.org/abs/1912.08777
- Rajpurkar, P. et al. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. EMNLP. https://rajpurkar.github.io/SQuAD-explorer/

- Liu, Y. et al. (2019). RoBERTa: A Robustly Optimized BERT PretraininG Approach. arXiv.
  https://arxiv.org/abs/1907.11692
- Devlin, J. et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL.
  https://aclanthology.org/N19-1423
- Sanh, V. et al. (2019). DistilBERT: Smaller, Faster, Cheaper and Lighter. arXiv.
  https://arxiv.org/abs/1910.01108
- Schick, T., & Schütze, H. (2021). It's Not Just Size That Matters: SmalL Language Models Are Also Few-Shot Learners. NAACL.
  https://aclanthology.org/2021.naacl-main.20
- Gao, T. et al. (2021). SimCSE: Simple Contrastive Learning of Sentence Embeddings. EMNLP.
- https://aclanthology.org/2021.emnlp-main.552
- Xu, Y. et al. (2021). Beyond Question Generation: Automatic Question Answer Pair
  Generation for Educational Applications. ACL.
  https://aclanthology.org/2021.acl-long.82

12. SCREENSHOTS:

Baseline mcq’s:



Talha Khuram (22i-0790) & Ahmad Aqeel (22i-1134) | GenAI Fall 2025

Multi-Model Support: Phi-3, Gemini, Baseline

Recommended: Gemini (fastest, ~10-30s) | Local Models: May be slow on CPU (10-15+ min)

7. stable diffusion.pdf 39.0 MB

Status

✓ Loaded: 7. stable diffusion.pdf → 145 chunks ready!

Select Model: Phi-3 Gemini [✓ Baseline]

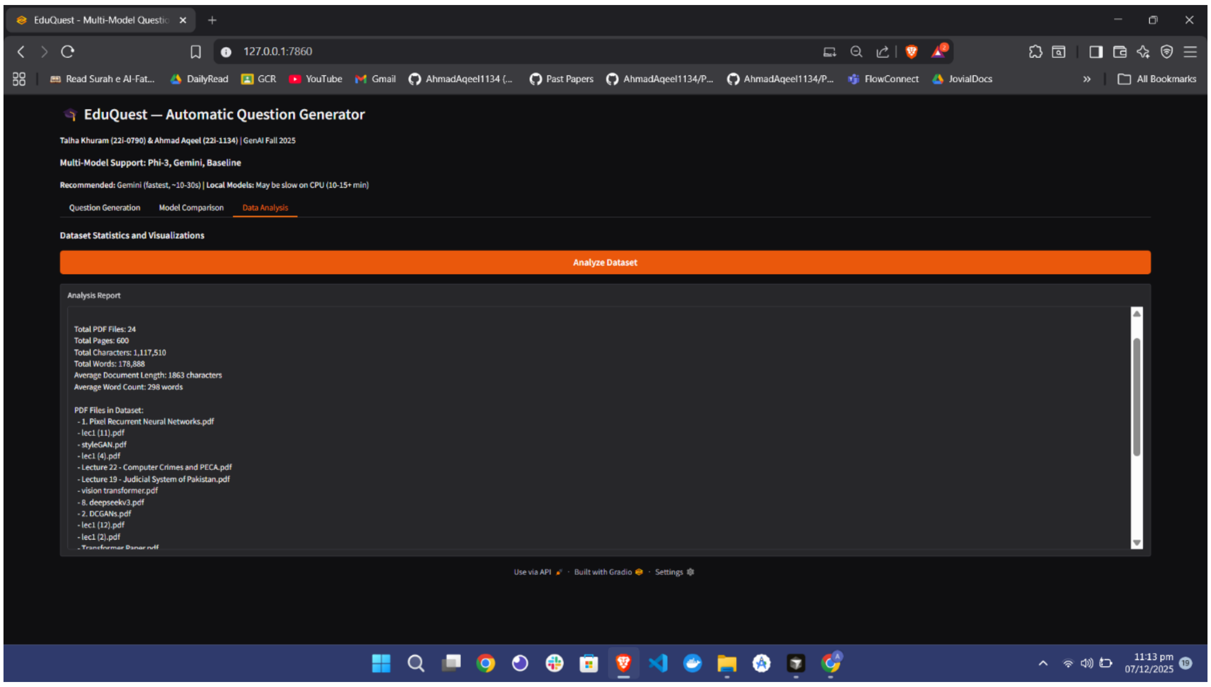
Question Type: [✓ mcq] short long

Number of Questions: 5 (1–10)

Generated Questions

```
Question 1:
Question: What is a key characteristic of See Sec?
A) See Sec is a fundamental concept in this field
B) See Sec refers to a specific method or technique
C) See Sec is a type of system or framework
D) All of the above
Correct Answer: A
Explanation: This question tests understanding of see sec as discussed in the context.
________________________________________

Question 2:
Question: What is the main advantage of Semantic Synthesis?
A) Semantic Synthesis is a fundamental concept in this field
B) Semantic Synthesis refers to a specific method or technique
C) Semantic Synthesis is a type of system or framework
D) All of the above
Correct Answer: A
Explanation: This question tests understanding of semantic synthesis as discussed in the context.
________________________________________

Question 3:
```

Data analysis:


EduQuest — Automatic Question Generator

Talha Khuram (22i-0790) & Ahmad Aqeel (22i-1134) | GenAI Fall 2025

Multi-Model Support: Phi-3, Gemini, Baseline

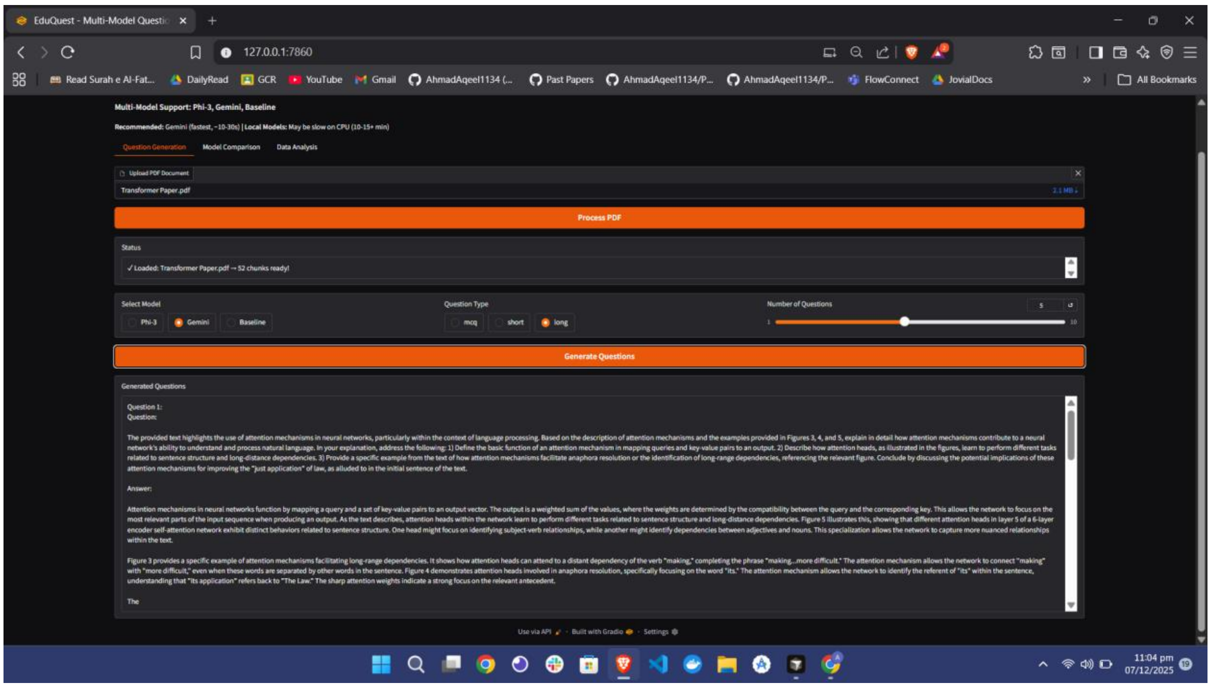
Recommended: Gemini (fastest, ~10-30s) | **Local Models:** May be slow on CPU (10-15+ min)

Dataset Statistics and Visualizations

Analysis Report

```
Total PDF Files: 24
Total Pages: 600
Total Characters: 1,117,510
Total Words: 178,888
Average Document Length: 1863 characters
Average Word Count: 298 words

PDF Files in Dataset:
 - 1. Pixel Recurrent Neural Networks.pdf
 - lec1 (11).pdf
 - styleGAN.pdf
 - lec1 (4).pdf
 - Lecture 22 - Computer Crimes and PECA.pdf
 - Lecture 19 - Judicial System of Pakistan.pdf
 - vision transformer.pdf
 - 8. deepseekv3.pdf
 - 2. DCGANs.pdf
 - lec1 (12).pdf
 - lec1 (2).pdf
```

Gemini long:


Multi-Model Support: Phi-3, Gemini, Baseline

Recommended: Gemini (fastest, ~10-30s) | **Local Models:** May be slow on CPU (10-15+ min)

Upload PDF Document: Transformer Paper.pdf

Status: ✓ Loaded: Transformer Paper.pdf → 52 chunks ready!

Select Model: Phi-3 [✓ Gemini] Baseline

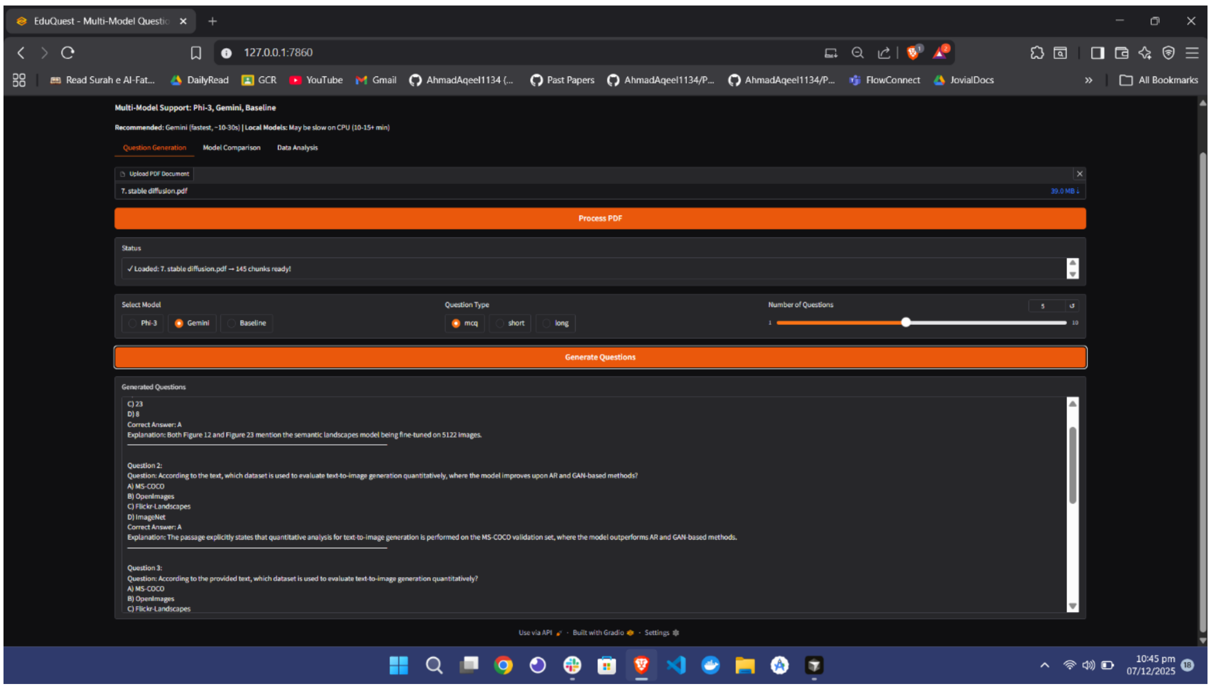
Question Type: mcq short [✓ long]

Number of Questions: 5

Generated Questions

```
Question 1:
Question:

The provided text highlights the use of attention mechanisms in neural networks, particularly within the context of language processing. Based on the description of attention mechanisms and the examples provided in Figures 3, 4, and 5, explain in detail how attention mechanisms contribute to a neural network's ability to understand and process natural language. In your explanation, address the following: 1) Define the basic function of an attention mechanism in mapping queries and key-value pairs to an output. 2) Describe how attention heads, as illustrated in the figures, learn to perform different tasks related to sentence structure and long-distance dependencies. 3) Provide a specific example from the text of how attention mechanisms facilitate anaphora resolution or the identification of long-range dependencies, referencing the relevant figure. Conclude by discussing the potential implications of these attention mechanisms for improving the "just application" of law, as alluded to in the initial sentence of the text.

Answer:

Attention mechanisms in neural networks function by mapping a query and a set of key-value pairs to an output vector. The output is a weighted sum of the values, where the weights are determined by the compatibility between the query and the corresponding key. This allows the network to focus on the most relevant parts of the input sequence when producing an output. As the text describes, attention heads within the network learn to perform different tasks related to sentence structure and long-distance dependencies. Figure 5 illustrates this, showing that different attention heads in layer 5 of a 6-layer encoder self-attention network exhibit distinct behaviors related to sentence structure. One head might focus on identifying subject-verb relationships, while another might identify dependencies between adjectives and nouns. This specialization allows the network to capture more nuanced relationships within the text.

Figure 3 provides a specific example of attention mechanisms facilitating long-range dependencies. It shows how attention heads can attend to a distant dependency of the verb "making," completing the phrase "making...more difficult." The attention mechanism allows the network to connect "making" with "more difficult," even when these words are separated by other words in the sentence. Figure 4 demonstrates attention heads involved in anaphora resolution, specifically focusing on the word "its." The attention mechanism allows the network to identify the referent of "its" within the sentence, understanding that "its application" refers back to "The Law." The sharp attention weights indicate a strong focus on the relevant antecedent.

The
```

Gemini Mcq's:



Multi-Model Support: Phi-3, Gemini, Baseline

Recommended: Gemini (fastest, ~10-30s) | **Local Models:** May be slow on CPU (10-15+ min)

7. stable diffusion.pdf 39.0 MB

Status

✓ Loaded: 7. stable diffusion.pdf → 145 chunks ready!

Select Model: Phi-3 [✓ Gemini] Baseline

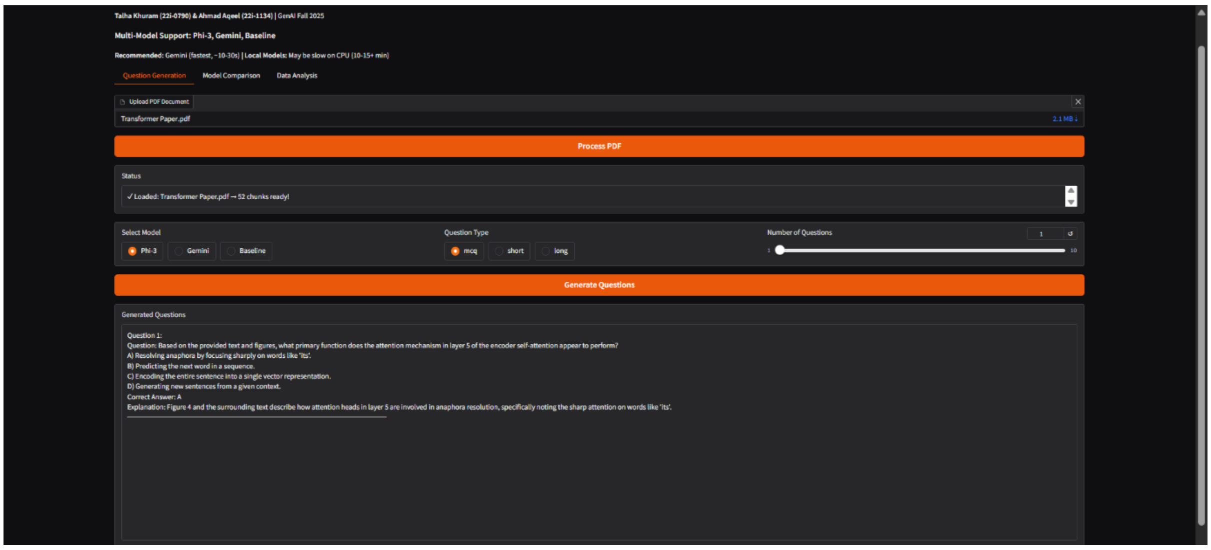
Question Type: [✓ mcq] short long

Number of Questions: 5

Generated Questions

```
C) 23
D) 8
Correct Answer: A
Explanation: Both Figure 12 and Figure 23 mention the semantic landscapes model being fine-tuned on 5122 images.
________________________________________

Question 2:
Question: According to the text, which dataset is used to evaluate text-to-image generation quantitatively, where the model improves upon AR and GAN-based methods?
A) MS-COCO
B) OpenImages
C) Flickr-Landscapes
D) ImageNet
Correct Answer: A
Explanation: The passage explicitly states that quantitative analysis for text-to-image generation is performed on the MS-COCO validation set, where the model outperforms AR and GAN-based methods.
________________________________________

Question 3:
Question: According to the provided text, which dataset is used to evaluate text-to-image generation quantitatively?
A) MS-COCO
B) OpenImages
C) Flickr-Landscapes
```

Phi-3 Mcq's:



Talha Khuram (22i-0790) & Ahmad Aqeel (22i-1134) | GenAI Fall 2025

Multi-Model Support: Phi-3, Gemini, Baseline

Recommended: Gemini (fastest, ~10-30s) | **Local Models:** May be slow on CPU (10-15+ min)

Transformer Paper.pdf 2.1 MB

Status

✓ Loaded: Transformer Paper.pdf → 52 chunks ready!

Select Model: [✓ Phi-3] Gemini Baseline

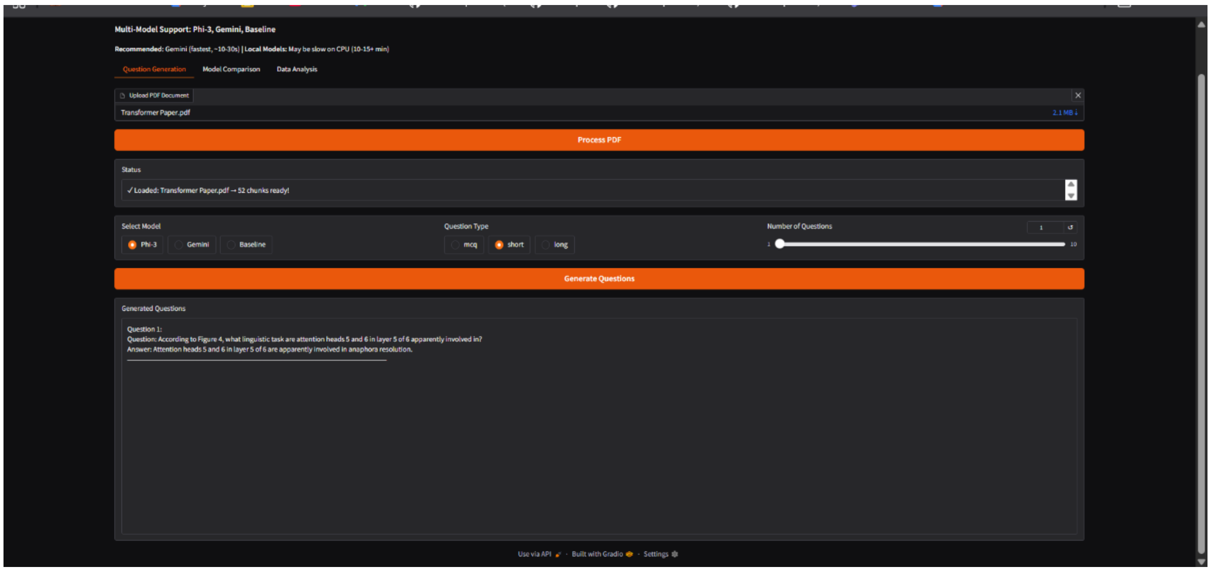
Question Type: [✓ mcq] short long

Number of Questions: 1

Generated Questions

```
Question 1:
Question: Based on the provided text and figures, what primary function does the attention mechanism in layer 5 of the encoder self-attention appear to perform?
A) Resolving anaphora by focusing sharply on words like 'its'.
B) Predicting the next word in a sequence.
C) Encoding the entire sentence into a single vector representation.
D) Generating new sentences from a given context.
Correct Answer: A
Explanation: Figure 4 and the surrounding text describe how attention heads in layer 5 are involved in anaphora resolution, specifically noting the sharp attention on words like 'its'.
________________________________________
```

Phi-3 Short:


Multi-Model Support: Phi-3, Gemini, Baseline

Recommended: Gemini (fastest, ~10-30s) | **Local Models:** May be slow on CPU (10-15+ min)

Question Generation | Model Comparison | Data Analysis

Upload PDF Document: Transformer Paper.pdf (2.1 MB)

Status: ✓ Loaded: Transformer Paper.pdf → 52 chunks ready!

Select Model: [● Phi-3] Gemini Baseline

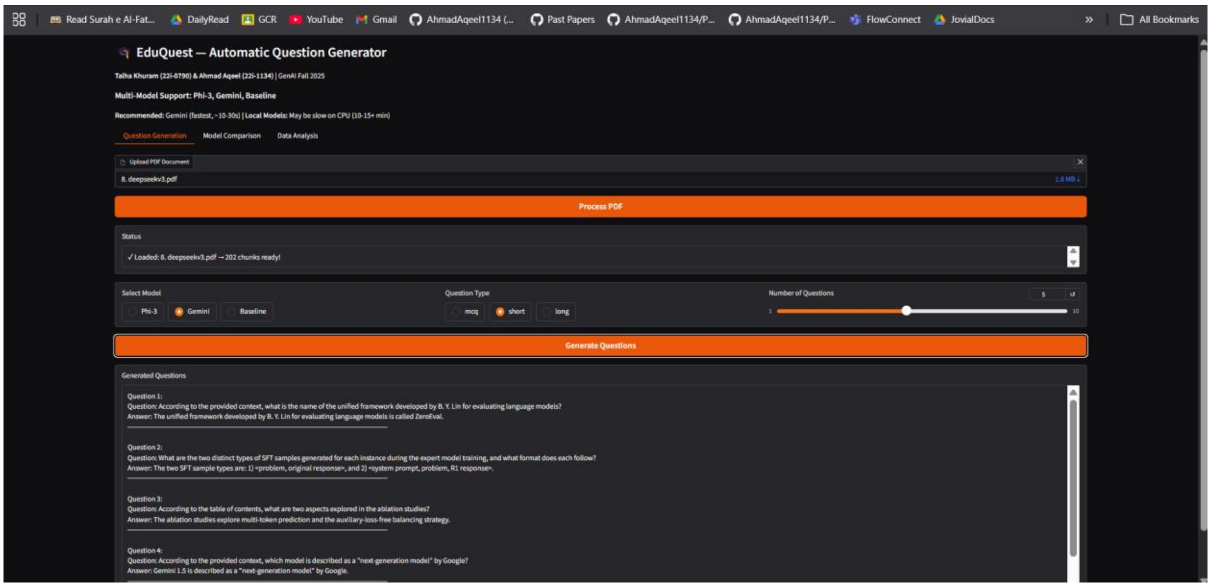
Question Type: mcq [● short] long

Number of Questions: 1

Generated Questions

Question 1:
Question: According to Figure 4, what linguistic task are attention heads 5 and 6 in layer 5 of 6 apparently involved in?
Answer: Attention heads 5 and 6 in layer 5 of 6 are apparently involved in anaphora resolution.

Gemini short:


EduQuest — Automatic Question Generator

Talha Khuram (22i-0790) & Ahmad Aqeel (22i-1134) | GenAI Fall 2025

Multi-Model Support: Phi-3, Gemini, Baseline

Recommended: Gemini (fastest, ~10-30s) | **Local Models:** May be slow on CPU (10-15+ min)

Question Generation | Model Comparison | Data Analysis

Upload PDF Document: 8. deepseekv3.pdf (1.8 MB)

Status: ✓ Loaded: 8. deepseekv3.pdf → 202 chunks ready!

Select Model: Phi-3 [● Gemini] Baseline

Question Type: mcq [● short] long

Number of Questions: 5

Generated Questions

Question 1:
Question: According to the provided context, what is the name of the unified framework developed by B. Y. Lin for evaluating language models?
Answer: The unified framework developed by B. Y. Lin for evaluating language models is called ZeroEval.

Question 2:
Question: What are the two distinct types of SFT samples generated for each instance during the expert model training, and what format does each follow?
Answer: The two SFT sample types are: 1) <problem, original response>, and 2) <system prompt, problem, R1 response>.

Question 3:
Question: According to the table of contents, what are two aspects explored in the ablation studies?
Answer: The ablation studies explore multi-token prediction and the auxiliary-loss-free balancing strategy.

Question 4:
Question: According to the provided context, which model is described as a "next-generation model" by Google?
Answer: Gemini 1.5 is described as a "next-generation model" by Google.