Multiple Choice Question Generation

Using NLP Models

Gangaram Arvind Sudewad
20CS30017

01 Introduction

What's the focus?

NLP

Concerns with the interaction of machine and humans in natural languages, leading to applications like chatbots, language translation, sentiment analysis, and text summarization.

MCQG

Involves automatically generating questions and options from a given text or content. The goal is to create relevant questions and options that can help in comprehension. For now only English text.

Multiple-Choice Questions (MCQs):

- Assessment format where test-takers choose the correct answer from a set of options.
- Questions typically consist of a stem (the question itself) and a list of answer choices.
- One correct answer and several incorrect options are provided.

Distractors:

- Incorrect answer options provided alongside the correct answer in MCQs.
- Designed to challenge test-takers by resembling the correct answer but ultimately being incorrect.

Ex: What is the capital of France?

A) Paris (Correct)
B) Berlin
C) London
D) Rome

MOTIVATION

Dataset: Generating a large-scale corpus of Context, question-answer triplets of acceptable quality

Education: Generating quality multiple choice questions with no bias and repetition to evaluate student performance

Customization: Customizing content into questions -multiple choices pairs, for chat boxes and Q/A system

02 Related Works

Works Related to MCQ generation

- Aldabe et al.[1] introduced ArikIturri, tailored for Basque language tests, using linguistically analyzed corpora in XML format.
- Yuni et al.[6] focused on creating automatic factual open cloze questions from informative sentences, based on Part-of-Speech tagging rules.
- Folajimi et al.[3] developed a system generating logical questions from input text, employing a three-step strategy: selecting sentences, identifying subject and context (Gap Selection), and analyzing optimal question formation.
- Chidinma et al.[5] proposed automatic distractor generation for multiple-choice English vocabulary questions using novel sources and semantic similarity.
- Bidyut et al.[2] applied NLP to identify discourse connectives in narrative texts for AQG, extracting text from user materials to generate questions.

Distractor Generation in ArikIturri:

- Utilizes techniques for generating distractors in multiple-choice and error correction questions.
- Techniques vary based on noun inflection forms and verb forms.

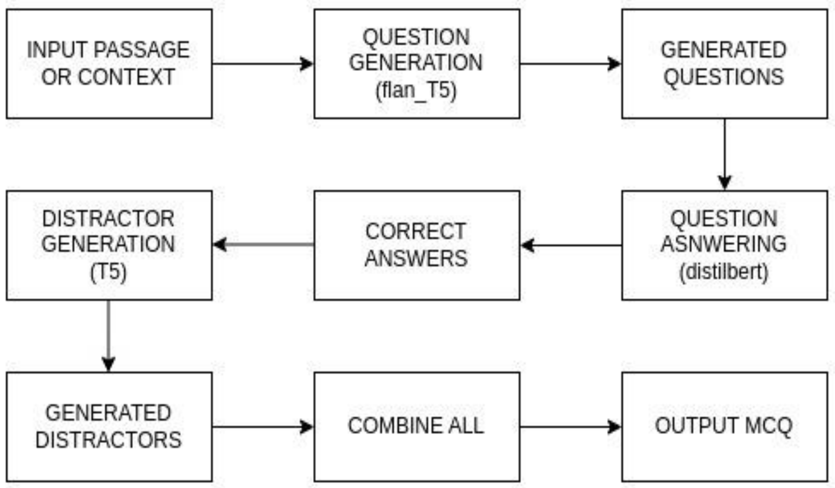
- **For Noun Inflection Forms:**
  This involves changing and copying different forms of words, like how they change for different situations or numbers.

- **For Verb Forms:**
  Involves changes in subject and object persons, verb modes, tenses, aspects, and verb paradigms. This means changing who does what in a sentence, how they do it, and when it happens.

03 Methodology

The multiple-choice question generation process involves preprocessing input text, training a transformer model iteratively, constructing and tokenizing prompt templates, generating questions, answering questions, and generating distractors.

- The question generation process begins with the FlanT5 model, designed for crafting context-specific questions.
- These questions are then fed into the DistilBERT model, which specializes in question answering to pinpoint the correct answers within the provided context.
- Finally, the T5 model generates distractors, carefully crafting incorrect options for multiple-choice questions to accompany the correct answers.



Dataset

SQuAD

Stanford question answering dataset was used. Extracted sentence-question-answer triplets.

RACE

The Reading Assessment for Comprehension and Evaluation dataset is being utilized to train and validate model for the task of distractor generation.

SQuAD

Id, title, context, question, answers, text, answer_start

"id": "573382d24776f41900660c39",
"title": "Warsaw",
"context": "Warsaw is home to many companies and institutions...",
"question": "What was Warsaw ranked the 7th greatest of?",
"answers":
  "text": ["emerging market", "emerging market", "emerging market"],
  "answer_start": [470, 470, 470]

RACE

Race_id, article, questions, options answers

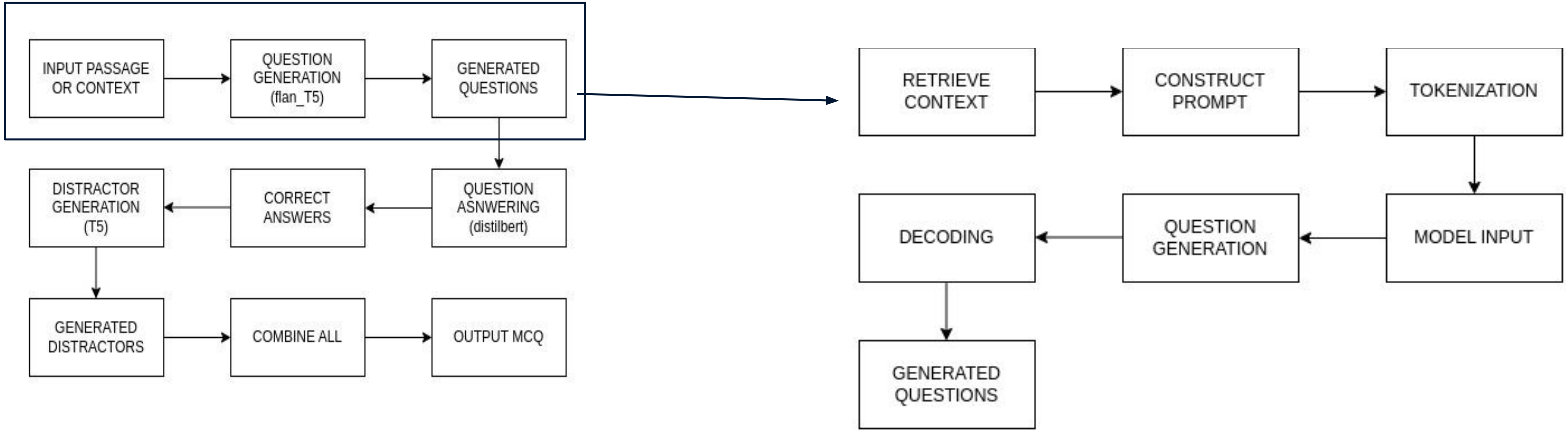
"race_id": "middle1624.txt",
"article": "Working mothers are growing in number...",
"questions":
["What is increasing with working mothers?", "What do children of working moms achieve?", "How do moms interact with kids?"],
"options":
[["Childcare options", "Family income", "School enrollment", "Maternal stress"], ["Higher grades", "Lower self-esteem", "Fewer issues", "Better development"], ["Limited communication", "Decreased attachment", "Increased involvement", "Reduced supervision"]],
"answers": ["Maternal stress", "Higher grades", "Increased involvement"]

Sequential Phases for MCQ Generation

Question Generation

Context Retrieval: Retrieve the context from the dataset using the specified index.

Training Transformer Model: Train the flanT5 model, incorporating parameter analysis, zero-shot inference, dataset tokenization, LORA(Low rank adaption) configuration, and PEFT(Pre-training with Extracted Feature Transfer) fine-tuning.



Prompt Construction: Construct a prompt template by formatting the retrieved context into a template string, including fixed text, context, and a placeholder for the question.
"Generate Question from the following context. + {context} + Question"
Tokenization: Tokenize the prompt template, converting the input text into numerical representations understandable by the model. Convert the tokenized input into PyTorch tensors for further processing.

Model Input: Pass the tokenized input to the flanT5 model for question generation.

Question Generation: Utilize the generate() method of the model to generate questions based on the provided context. The generated output is a sequence of token IDs representing the question.

Decoding: Decode the generated token IDs using the tokenizer to obtain the final question text. Skip special tokens indicating the beginning and end of the sequence during decoding.

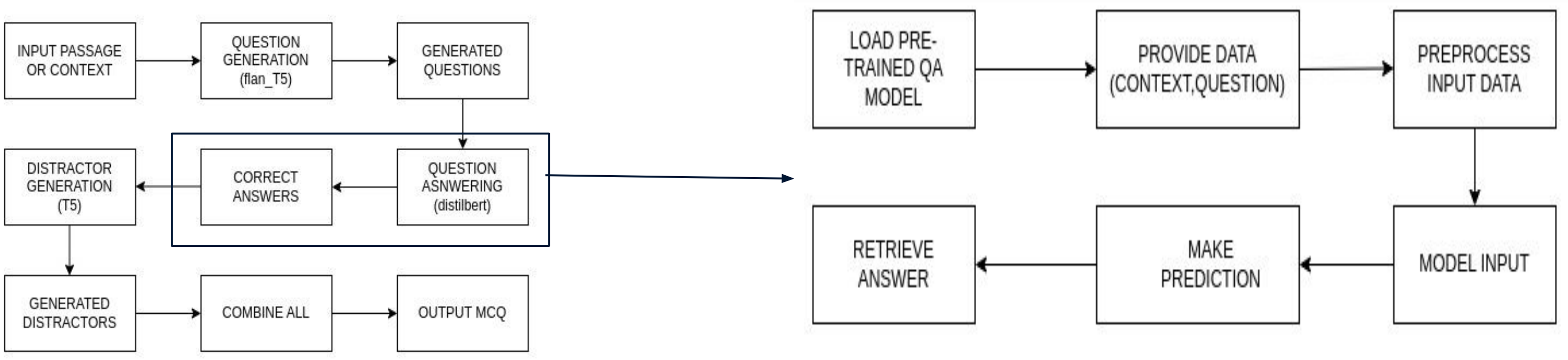
Training Transformer Model:

Training the flan 5 model used for question generation where,
Input: Tokenized context-question pairs for training.
Output: Model predictions (tokenized questions) and loss during training.
Training Data Samples: 87,599 samples then reduced to 17520 samples from the SQuAD training split.
Testing (Validation) Data Samples: 10,570 samples reduced to 2114 from the SQuAD validation split.

Step	Training Loss
200	2.729700
400	0.093100
600	0.082500
800	0.078700
1000	0.076300
1200	0.074600
1400	0.074200
1600	0.070800
1800	0.073300
2000	0.071900

Question Answering

- **Model Training:**
  - Iteratively adjust parameters to minimize prediction errors.
  - Prepare data, fine-tune model, and periodically evaluate performance.
  - Save checkpoints to monitor progress and enable pausing, resuming.
- **Loading Trained QA Model:**
  - Load pre-trained QA model trained on a large dataset.
  - Understand natural language questions and provide answers based on context.



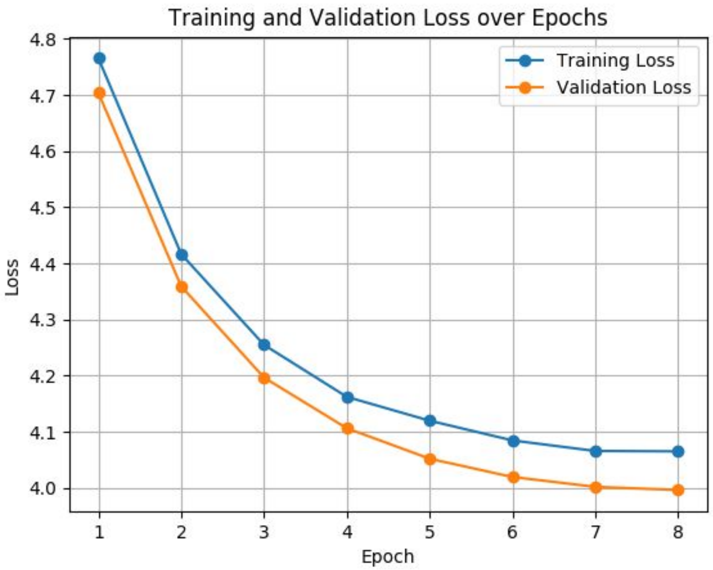
Question Answering

- **Providing Data:**
  - Input context and question for QA model analysis.
- **Preprocessing Data:**
  - Tokenize and pad input data to prepare for model input.
- **Making Prediction:**
  - Feed preprocessed data into trained QA model.
  - Utilize learned parameters and pre-training knowledge to generate answer.
- **Retrieving Answer from Model:**
  - Retrieve generated answer from model's output.
  - Use for further analysis or presentation to the user.

Training Transformer Model:

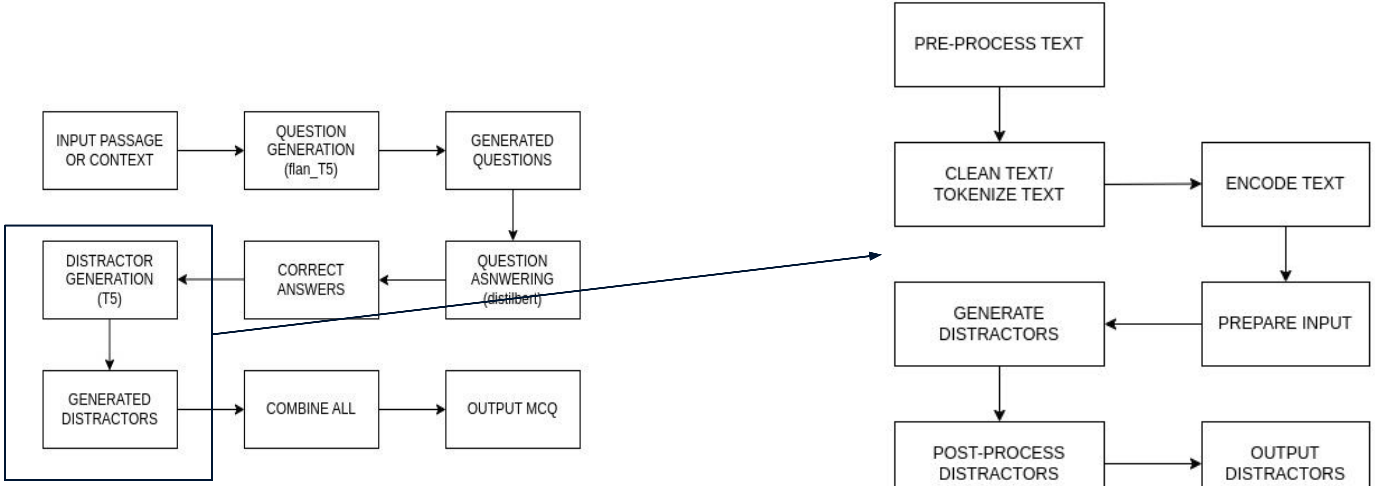
Input during Training: Tokenized context-question pairs with answer positions.
Output during Training: Predicted start and end positions of answers used for computing loss.
Training Data Samples: Consist of approximately 80,000 tokenized context-question pairs from the SQuAD training split.
Testing (Validation) Data Samples: Include around 10,000 tokenized context-question pairs from the SQuAD validation split, utilized for evaluating model performance

Epoch	Training Loss	Validation Loss
1	4.7657	4.7041
2	4.4165	4.3591
3	4.2548	4.1965
4	4.1621	4.1062
5	4.1196	4.0519
6	4.0845	4.0193
7	4.0657	4.0018
8	4.0650	3.9961



Distractor Generation

- **Tokenization and Encoding:**
  - Tokenization breaks text into tokens for model comprehension.
  - Encoding maps tokens to numerical identifiers for processing.
- **Preparing Input:**
  - Combine encoded tokens of question, answer, and context for model input.
  - Organize data into a format suitable for processing.



Distractor Generation

- **Training Model:**
  - Prepare RACE dataset, initialize and optimize T5 model with Adam optimizer.
  - Training the T5 model for distractor generation where, input consists of concatenated question-answer-context triplets for training and generated distractors for testing (validation). There are 87,866 training samples. There are 4,887 testing (validation) samples.
- **Generating Distractors:**
  - Use trained model T5 to generate distractors for multiple-choice questions.
  - Enhance question depth and complexity with alternative options.
- **Post-processing Distractors:**
  - Clean generated distractors to ensure coherence and suitability for use.
  - Remove special tokens or unwanted characters.
- **Outputting Distractors:**
  - Output generated distractors for further use or evaluation.
  - Seamlessly integrate into final question generation pipeline or utilize independently.

04 Results

Results

In the analysis of the model-generated question, it's evident that there is a discrepancy between the intended question and the question generated by the model. This highlights a potential area for improvement in the model's understanding of context and its ability to generate relevant questions.

'title': 'Warsaw', 'context': "Warsaw, especially its city centre (Śródmieście), is home not only to many national institutions and government agencies, but also to many domestic and international companies. In 2006, 304,016 companies were registered in the city. Warsaw's ever-growing business community has been noticed globally, regionally, and nationally. MasterCard Emerging Market Index has noted Warsaw's economic strength and commercial center. Moreover, Warsaw was ranked as the 7th greatest emerging market. Foreign investors' financial participation in the city's development was estimated in 2002 at over 650 million euro. Warsaw produces 12% of Poland's national income, which in 2008 was 305.1% of the Polish average, per capita (or 160% of the European Union average). The GDP per capita in Warsaw amounted to PLN 94 000 in 2008 (c. EUR 23 800, USD 33 000). Total nominal GDP of the city in 2010 amounted to 191.766 billion PLN, 111696 PLN per capita, which was 301,1 % of Polish average. Warsaw leads the region of East-Central Europe in foreign investment and in 2006, GDP growth met expectations with a level of 6.1%. It also has one of the fastest growing economies, with GDP growth at 6.5 percent in 2007 and 6.1 percent in the first quarter of 2008.",
Generated Questions: In 2006, how many companies were registered in Warsaw?
In what year was the GDP per capita in Warsaw estimated at over 650 million euro?

Evaluating the similarity between between generated questions and Stanford Question Answering Dataset (SQuAD 2.0) ground truth questions using BERT embeddings based on the cosine similarity metric.
It calculates key similarity metrics, such as average, maximum, minimum, standard deviation, and range, for quality evaluation.

Generated Qns: What was fought between the colonies of British America and New France?
Most Similar Ground Truth Qns: Who fought in the French and Indian war?
Similarity Score: 0.648 (64.8%)

Metric	value	percentage
Avg similarity	0.568	56.8%
Max similarity	0.676	67.6%
Min similarity	0.144	14.4%
Std deviation	0.155	-
Range	0.504	-

Question Answering

For the question answering part the example outputs are given as follows:

Question: How many programming languages does BLOOM support?
Context: BLOOM has 176 billion parameters and can generate text in 46 natural languages and 13 programming languages.
Ground Truth: 46 natural languages and 13 programming languages
Generated Answer: 176 billion parameters and can generate text in 46 natural languages.
Similarity Score: 0.4999

Question: What is Warsaw's economy characterized by?
Context: " Warsaw's economy, by a wide variety of industries, is characterised by FMCG manufacturing, metal processing, steel and electronic manufacturing and food processing. **"**
Ground Truth: FMCG manufacturing, metal processing
Generated Answer: metal processing
Similarity Score: 0.7071067811865475

Distractor Generation

Some example output of the generated distractors is shown below:

Ex.1: Question: ”What is the best title for the passage?**”**
Q: What is the best title for the passage?
Generated Distractors:
A: New Rules for international visitors **B:** Djokovic's challenge
C: Djokovic's application for special permission to enter the United States
D: World number two Novak Djokovic's dream
Correct: Djokovic's application for special permission to enter the United States

Ex.2: Question: Why is Djokovic hoping for a positive decision?
Generated Distractors:
A: To be unvaccinated **B:** To receive proof of vaccination
C: To be allowed to play at Indian Wells and the Miami Open **D:** To be unvaccinated
Correct: To be allowed to play at Indian Wells and the Miami Open

Ex.3: Question: How many Grand Slam men's titles has Djokovic won?
Generated Distractors:
A: 3 **B:** 16 **C:** 22 **D:** 10
Correct: 22

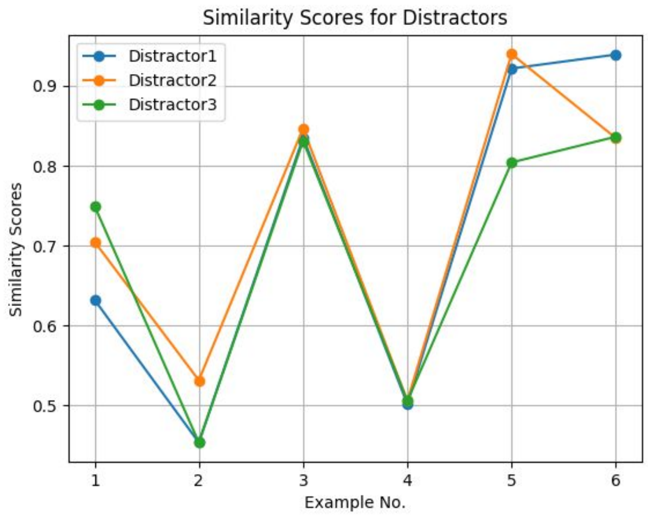
Ex.4: Question: Who had a goal ruled out for offside?
Generated Distractors:
A: Ben Chilwell **B:** Joao Felix **C:** Kai Havertz **D:** Kai Havertz
Correct: Joao Felix

Ex.5: Question: Which of the following best describes the families of the astronauts on the ISS?
Generated Distractors:
A: They are generous and brave
B: They are caring and thoughtful
C: They are kind and brave
D: They are strong and hard-working
Correct: They are caring and thoughtful

Ex.6: Question: Where did the clerk find the necklace at last?
Generated Distractors:
A: In the river
B: On the trees
C: On the tree
D: In the lake
Correct: On the tree

SIMILARITY SCORES

Ex. No.	Distractor1	Distractor2	Distractor3
1	0.63207	0.70482	0.74962
2	0.45337	0.53134	0.45337
3	0.83514	0.84683	0.83109
4	0.50160	0.50563	0.50563
5	0.92177	0.93989	0.80415
6	0.93919	0.83458	0.83636



05 | Future work

Future Works and Upgrades for MCQs Generation Tool

- **Ensemble Models:** Combining outputs of multiple models can enhance question and distractor quality and diversity.
- **Data Augmentation:** Adding more examples and variations to training data boosts model generalization and diversity in question generation.
- **Feedback Mechanisms:** User feedback on generated questions and distractors refines models and enhances performance over time.
- **Multi-Modal Input:** Incorporating images, audio, or video alongside text improves comprehensiveness and relevance of generated content.
- **Evaluation Metrics:** Developing robust metrics aids in quantitative assessment of question and distractor quality for model improvement.
- **Deployment and Integration:** Integrating MCQ generation with educational platforms streamlines real-world usage in educational settings.

References

[1] Itziar Aldabe, Maddalen Lopez De Lacalle, Montse Maritxalar, Edurne Martinez, and Larraitz Uria. Arikiturri: an automatic question generator based on corpora and nlp techniques. In Intelligent Tutoring Systems: 8th International Conference, ITS 2006, Jhongli, Taiwan, June 26-30, 2006. Proceedings 8, pages 584–594. Springer, 2006.
[2] Bidyut Das and Mukta Majumder. Factual open cloze question generation for assessment of learner's knowledge. International Journal of Educational Technology in Higher Education, 14:1–12, 2017.
[3] YO Folajimi and OE Omojola. Natural language processing techniques for automatic test questions generation using discourse connectives. Journal of Computer Science and Its Application, 20(2):60–76, 2013.
[4] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations. arXiv preprint arXiv:1704.04683, 2017.
[5] Chidinma A Nwafor and Ikechukwu E Onyenwe. An automated multiple-choice question generation using natural language processing techniques. arXiv preprint arXiv:2103.14757, 2021.
[6] Yuni Susanti, Takenobu Tokunaga, Hitoshi Nishikawa, and Hiroyuki Obari. Automatic distractor generation for multiple-choice english vocabulary questions. Research and practice in technology enhanced learning, 13(1):15, 2018.

THANK YOU!