# Generating Tricky Multiple-Choice QA Pairs from Contexts using Hierarchical Conditional VAEs

Stanford CS224N Custom Project

**Davyn Sudirdjo**
Department of Computer Science
Stanford University
davyn@stanford.edu

## Abstract

Globalization has created an influx of international students wanting to study at universities in English-speaking countries. One of the basic requirements is taking the English proficiency test, such as the SAT, ACT, TOEFL, TOEIC, and IELTS. However, there is a lack of accessibility to practice questions. Question-Answer Pair Generation (QAG) is a potential solution to overcome this data scarcity challenge. This project proposes a QAG+ model that generates QA pairs using an infoormation-maximizing hierarchical conditional variational autoencoder (Info-HCVAE) and generates tricky and subtly wrong multiple choice answers using Hugging Face Transformers instead of the traditional QAG one-to-one mapping. The baseline evaluation is using the correct answers from other questions. Varying thresholds of a similarity metric, evaluated against the BLEU score, is then used to generate tricky wrong answers. Four different QA models were tested on the different difficulty levels to get a bearing of how a human would perform in these tests. A comparison was also made between the four models' performances on real TOEFL tests versus generated multiple choice questions from the same TOEFL texts. Our findings show that our QAG+ model does well enough in mimicking a real test based on how the four models performed, but an exploration on the semantic and contextual similarity across answers is key to improving the model further.

## 1 Key Information to include

- External collaborators (if you have any): N/A
- External mentor (if you have any): N/A
- Sharing project: No

## 2 Introduction

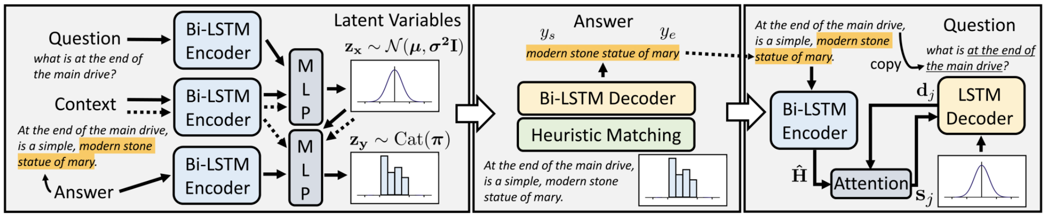
In the last five years, much forward progress has been made in the field of Question Answering (QA). The use of deep neural networks and the capitalization of a wide array of pretrained language model has resulted in human-like performance on various QA benchmarks, such as Google's BERT and ELECTRA, as well as Meta's (formerly Facebook's) RoBERTa, which achieved state-of-the-art performance on the Stanford Question Answering Dataset (SQuAD). Similar to but still lagging slightly behind QA, significant progress has been made in the field of Question Generation (QG) and Question-Answer Generation (QAG). We need more QA data to improve existing QA models, but there is a huge overhead to create and tag these QA pairs. These generative (QG and QAG) models are very useful to overcome the data scarcity obstacle.

The application of QAG models is not limited to improving QA models, however. It could also be a very powerful tool in the education sector, particularly in the test prep sub-sector. Presently, test prep material is limited in quantity and accessibility. Students would have to either go to tutoring centers or purchase multiple study guides from various publishers to gain access to these practice questions. For this project, we chose to create a generative multiple choice model for the English language proficiency tests because of its market size and how generalizable it is across the different tests (SAT, ACT, TOEFL, TOEIC, IELTS, etc.).

In this project, we propose a QAG+ model, the "+" symbol denoting an extension of existing QAG models to incorporate the generation of wrong multiple choice answers. There are currently several well-performing QAG models such as HarvestingQG, MaxoutQG, and SemanticQG. However, we will be using the information-maximizing hierarchical conditional variational autoencoder (Info-HCVAE) model, a novel QAG model published in 2020 [1]. We believe the Info-HCVAE model is suited better for this task because of its method of generating the QA pairs, which is by scanning a context and generating answers first, before generating questions based on that answer. Therefore, it will be easier to generate the incorrect multiple choice answers. Additionally, it is information-maximizing, which means that it will encapsulate as much of the context as possible since generating QA pairs from a context is a one-to-many problem (endless pairs of questions can be generated from one answer) and the Info-HCVAE model avoids this problem.

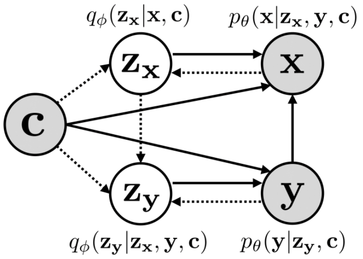
## 3 Related Work

The prior work that we will be referencing the research that initially proposed the novel Info-HCVAE QAG model from the Korea Advanced Institute of Science & Technology (KAIST) [1]. For context, Info-HCVAE is not a new model, but using it for QAG is a new application. The Info-HCVAE has two separate latent spaces for question and answer conditioned on the context, where the answer latent space is additionally conditioned on the question latent space. During generation, this Info-HCVAE first generates an answer given a context, and then generates a question given both the answer and the context, by sampling from both latent spaces. This probabilistic approach allows the model to generate diverse QA pairs focusing on different parts of a context at each time.


Question
what is at the end of the main drive?
Context
At the end of the main drive, is a simple, modern stone statue of mary.
Answer
Bi-LSTM Encoder
Bi-LSTM Encoder
Bi-LSTM Encoder
MLP
MLP
Latent Variables
$\mathbf{z_x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}^2\mathbf{I})$
$\mathbf{z_y} \sim \mathrm{Cat}(\boldsymbol{\pi})$
Answer
$y_s$
$y_e$
modern stone statue of mary
Bi-LSTM Decoder
Heuristic Matching
At the end of the main drive, is a simple, modern stone statue of mary.
At the end of the main drive, is a simple, modern stone statue of mary.
Question
what is at the end of the main drive?
copy
Bi-LSTM Encoder
$\mathbf{d}_j$
LSTM Decoder
$\hat{\mathbf{H}}$
Attention
$\mathbf{s}_j$

(a) Conditional Posterior / Prior Networks (b) Answer Generation Network (c) Question Generation Network

Formally, given a context $\mathbf{c}$ which contains $M$ tokens $\mathbf{c} = (c_1, \ldots, c_M)$, the model generates QA pairs $(\mathbf{x}, \mathbf{y})$ where $\mathbf{x} = (x_1, \ldots, x_N)$ is the question containing $N$ tokens and $\mathbf{y} = (y_1, \ldots, y_L)$ is its corresponding answer containing $L$ tokens. The paper aims to accomplish the QAG task by learning the conditional joint distribution of the question and answer given the context, $p(\mathbf{x}, \mathbf{y} \mid \mathbf{c})$, from which the QA pairs can be generated, and thus $(\mathbf{x}, \mathbf{y}) \sim p(\mathbf{x}, \mathbf{y} \mid \mathbf{c})$. The conditional joint distribution is approximated using a Info-HCVAE framework with separate latent spaces $\mathbf{z_x}$ and $\mathbf{z_x}$ for each $\mathbf{x}$ and $\mathbf{y}$) respectively, with the reason being that the answer is always a finite span of the $\mathbf{c}$ which can be modeled using a categorical distribution, while the question needs a continuous latent space because there could be unlimited valid questions from $\mathbf{c}$.


$q_\phi(\mathbf{z_x}|\mathbf{x}, \mathbf{c})$
$p_\theta(\mathbf{x}|\mathbf{z_x}, \mathbf{y}, \mathbf{c})$
$\mathbf{z_x}$
$\mathbf{x}$
$\mathbf{c}$
$\mathbf{z_y}$
$\mathbf{y}$
$q_\phi(\mathbf{z_y}|\mathbf{z_x}, \mathbf{y}, \mathbf{c})$
$p_\theta(\mathbf{y}|\mathbf{z_y}, \mathbf{c})$

The figure on the left is the directed graphical model for Info-HCVAE. The gray and white nodes denote observed and latent variables. The model comprises of two conditional prior networks $p_\psi(\mathbf{z_x} \mid \mathbf{c})$ and $p_\psi(\mathbf{z_y} \mid \mathbf{z_x}, \mathbf{c})$ to model context-dependent priors, two conditional posterior networks $q_\phi(\mathbf{z_x} \mid \mathbf{x}, \mathbf{c})$ and $q_\phi(\mathbf{z_y} \mid \mathbf{z_x}, \mathbf{y}, \mathbf{c})$ to approximate the true posterior distributions of latent variables for each QA pair, answer generation networks, as well as question generation networks. The posterior and prior networks use the pre-trained word embedding network from BERT, and the generative networks use the whole BERT as a contextualized word embedding network.

# 4 Approach

## 4.1 Data Preprocessing

**Preprocessing 1:** To pretrain the Info-HCVAE model, the datasets will first be preprocessed such that they are all formatted in the same format as SQuAD. The final formal is as follows:

```
data:  List[item] → List of items.
item:  Dict[context, qas] → Dict of context and qas key-value pairs.
context:  str → String of paragraph content.
qas:  Dict[question, answer] → Dict of question and answer key-value pairs.
question:  str → String of a question.
answer:  List[a] → List of correct answers.
a:  str → String of an answer.
```

After preprocessing the data, we pretrained the Info-HCVAE model from the KAIST study on the SQuAD dataset. This pretrained model is used to generate the initial QA pairs.

**Preprocessing 2:** We preprocess each dataset to extract all the contexts in a list format. We feed the contexts into the Info-HCVAE model to generate the QA pairs first, and then feed the QA pairs into the second layer to generate the QA+ pairs.

## 4.2 Finetuning Info-HCVAE

The Info-HCVAE QAG model [1] is trained on the three datasets which can be found in section 5. After generating the QA pairs, three incorrect answers for each QA pair are generated to get the QA+ pairs. For the baseline model, answers to three other distinct questions will be used. For the rest of the model, the three incorrect answers will be generated using Hugging Face and the GPT-2 model will be used to ensure grammatical correctness. The model will also take three levels of difficulty, with the difficulties defined by Word2Vec similarity thresholds sing the pre-trained Word2Vec model from `gensim`. The higher the minimum and maximum similarity thresholds, the more difficult the QA+ pair will be to solve. The incorrect answers will be content-related, with aspirations for further research into incorrect answers relating to grammar, vocab, and context. We modified the code on the model's `generate_qa.py` file such that it follows the structure of the project.

## 4.3 Addressing the Correct "Incorrect" Answers Issue

Word2Vec similarity does not equal definition similarity, meaning that the incorrect answers generated may actually be correct answers after all. To address this issue, we propose using the Bilingual Evaluation Understudy (BLEU) metric. We set the correct answer as the reference text, and compute the BLEU score for each of the three generated answers. If the BLEU score is above a certain threshold, the generated answer is deemed too similar and will be regenerated.

## 4.4 Finetuning QA Models

The evaluation metrics revolve around having QA models answering the multiple choice questions. We finetune each QA model such that they can answer from a choice of answers rather than finding an answer on their own. The training loss is computed using the following cross entropy loss function:

$$H(y, \hat{y}) = -\sum_{i=1}^{n} y_i \log(\hat{y}_i) \tag{1}$$

# 5 Experiments

## 5.1 Data

In this project, the model will be trained and tested on Stanford's (SQuAD), Google's (Natural Questions) dataset, and Microsoft's (NewsQA) dataset. With SQuAD and Natural Questions, we used the training set given and randomly divided the dev set by two to create the validation and test sets. With NewsQA, we did a random 60/20/20 split to create the training, validation, and test sets.

### 5.2 Evaluation methods

**Quantitative Evaluation 1:** We propose a new metric called Accept BLEU (A-BLEU) to evaluate how good the each generated incorrect answer is. Recall that in section 4.4, we introduced the BLEU metric to ensure that the generated incorrect answers are not correct. Although the model will end up only outputting the acceptable incorrect answers, we wanted to investigate how good the model is in generating incorrect answers in general. The equation for A-BLEU is defined as the following:

$$\texttt{A-BLEU} = \frac{a}{a+r} \cdot 100\% \tag{2}$$

where $a$ represents the number of acceptable incorrect answers and $r$ represents the number of rejected incorrect answers due to a higher BLEU score than the threshold.

**Quantitative Evaluation 2:** We run several QA models that can answer multiple-choice questions, namely BERT, RoBERTa, ALBERT, and ELECTRA on the generated QA+ pairs. For each of the four models, we run a base model and a finetuned model. The input contexts to generate the QA+ pairs are the test sets of all three datasets. There will be four sets of QA+ pairs: baseline, easy, medium, and difficult. We will evaluate each model's accuracy on each set of QA+ pairs to develop an understanding of whether the generated wrong answers are difficult enough for a human to make a mistake in. The accuracy of these QA models in answering the QA+ pairs will be used to determine the effectiveness of the project's QAG+ model. We use the Word2Vec similarity as thresholds to compute the wrong answers, and they are defined as follows:

$$\texttt{Easy} \geq 0.5\texttt{, Medium} \geq 0.75\texttt{, Hard} \geq 0.9 \tag{3}$$

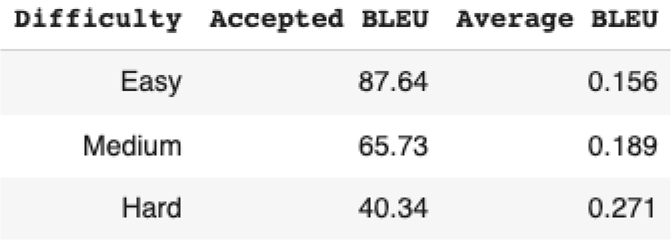
**Quantitative Evaluation 3:** We use 50 real past TOEFL reading comprehension texts as the contexts, and generated the same number of QA+ pairs per text as the real tests, with the difficulty being preset to a Word2Vec threshold of $\geq 0.75$. We run the four finetuned QA models on the original QA+ pairs, as well as the generated QA+ pairs and compared the model's performances in each set of QA+ pairs. Due to the time constraints of this project, we were not able to run a qualitative evaluation with human subjects. Therefore, we feel that this is presently the best way to find out how well our QAG+ model does in simulating an actual English language test.

### 5.3 Experimental details

To pretrain the Info-HCVAE model, we used the default hyperparameters from the original study. To finetune the four QA models, we trained each model on all three datasets. To do so, we set the learning rate `lr` to $5 \times 10^{-5}$, the number of epochs `epoch` to 10, the batch size `batch_size` to 4. The learning rate is optimized using the Adam optimizer and we used the cross entropy loss function for the training loss.

### 5.4 Results

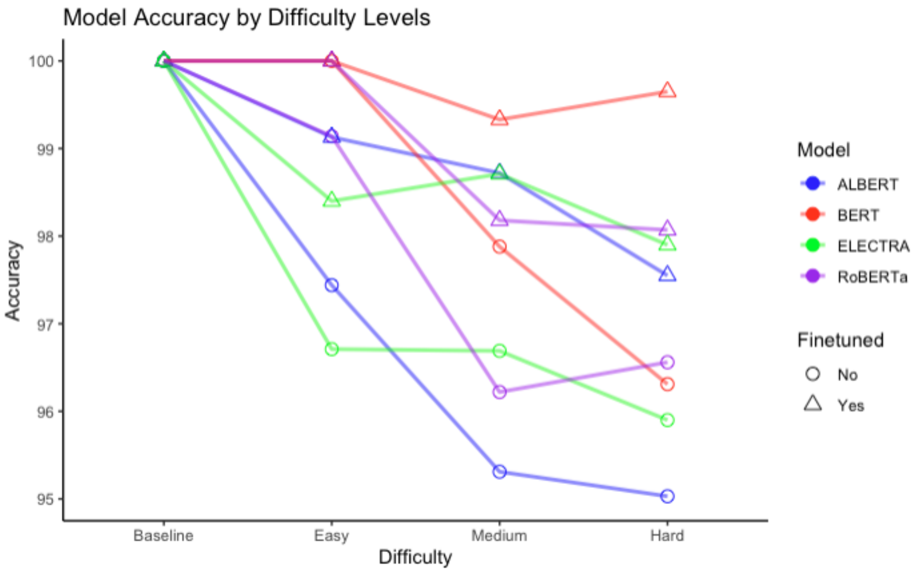
**Quantitative Evaluation 1:** The following table is the A-BLEU score and the average BLEU score of all generated answers (both accepted and rejected) for each respective difficulty level.



| Difficulty | Accepted BLEU | Average BLEU |
|---|---|---|
| Easy | 87.64 | 0.156 |
| Medium | 65.73 | 0.189 |
| Hard | 40.34 | 0.271 |

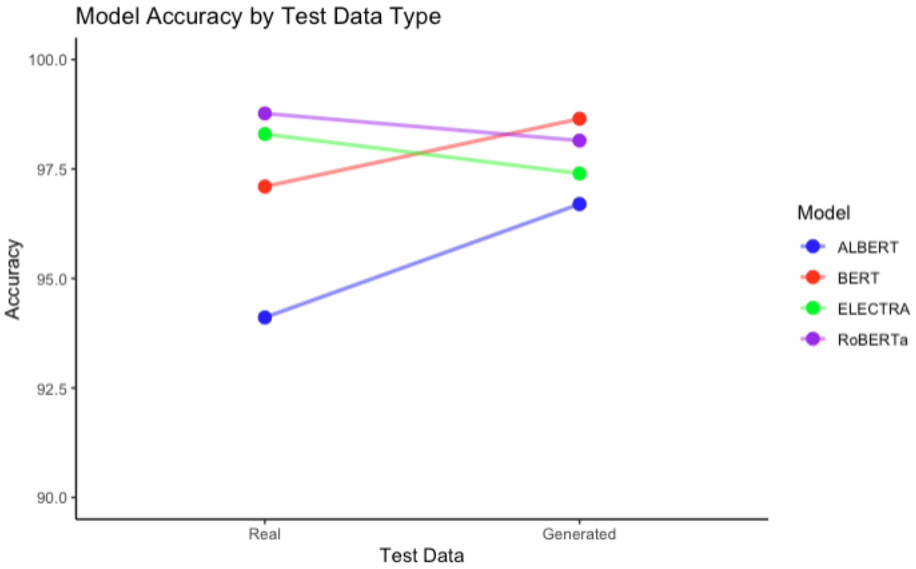
We observe that as the difficulty gets higher, the number of rejected answers increases, with the A-BLEU decreasing. Furthermore, the average BLEU score increases as the difficulty level increases, since the Word2Vec similarity increases by difficulty. This is consistent with our assumption.

**Quantitative Evaluation 2:** The following plot is the test accuracy of each of the four QA models for the varying levels of difficulty, and further broken down by whether the model is finetuned.



We observe that as the difficulty level increases, there is a general decrease in model performance, except for a few outliers such as the finetuned BERT's performance and the base RoBERTa's performance on the hard difficulty level. We also observe that all the finetuned models performed better than their respective base versions. This means that the QA models performed better when they are finetuned to all three datasets.

**Quantitative Evaluation 3:** The following plot is the test accuracy of each of the four QA models given 50 real TOEFL contexts with about 12-14 questions per context.



We observe that BERT and ALBERT performed significantly better on the generated QA+ pairs compared to the real QA+ pairs, while ELECTRA and RoBERTa performed slightly worse on the generated QA+ pairs. From these results, we can deduce that our model may be slightly too easy as the average performance of the four QA models is better on the generated data than in the real data.

## 6 Analysis

Overall, the results show that our QAG+ model generated tricky enough QA+ pairs such that the finetuned QA models only performed slightly better than on the real QA+ pairs from the TOEFL past tests bank. This indicates that we are on the right track to be able to simulate actual tests. However, we also acknowledge that the BLEU score is not perfect in determining the semantic and contextual similarity between the correct answer and the generated incorrect answers.

One of the most significant evaluation metrics that we missed due to time constraints is qualitative evaluation, that is, having several groups of people evaluate the generated answers relative to the context and the question. This would be helpful in determining how well our model can mimic real tests and provide a direction on how we can improve the model such that the generated QA+ pairs is as close as possible to the QA+ pairs a human test-setter would create.

## 7 Conclusion

QAG can be a powerful tool for the test prep industry, as it provides prospective test-takers with more accessibility to practice tests. Our analysis shows that some of the best-in-class QA models still score very well in the different tests with varying difficulties, but only slightly better than the real TOEFL tests. If we were to amend the difficulty level further, we can confidently say that the methods developed in this project can be utilized to generate robust multiple choice questions for reading comprehension texts.

## 8 Further Research

We can do several things to improve the model. Firstly, we could try to find a better way to set the difficulty than Word2Vec. Secondly, we could finetune the BLEU metric specifically for this task, such that the model does not accidentally generate correct "incorrect" answers. Thirdly, we could alter the InfoMax to maximize the how much of the context is captured by the set of QA+ pairs. Last but most importantly, our aspiration is to extend this into more test types and subjects to improve the access to education through technology.

## References

[1] Dong Bok Lee, Seanie Lee, Woo Tae Jeong, Dong Hwan Kim, and Sung Ju Hwang. Generating diverse and consistent question-answer pairs from contexts with information-maximizing hierarchical conditional variational autoencoders. Korea Advanced Institute of Science  Technology, 2020.