Addressing Question Imbalance in SQuAD

Gagan Gopinath*, Sai Raghava Mukund Bhamidipati*, Tarun Anand*
University Of Wisconsin-Madison
{ggopinath, bhamidipati3, tanand3}@wisc.edu

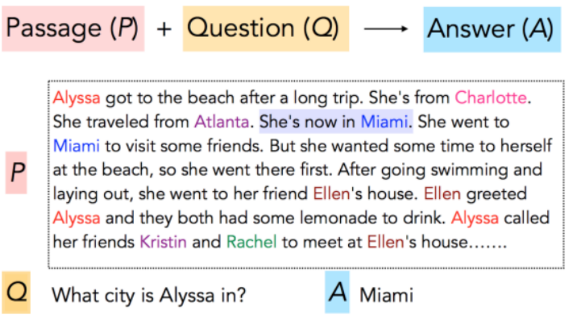
1 Introduction

The ability to answer questions is an important task to establish further progress in machine comprehension of human language. Question answering is a multi disciplinary domain that involves information retrieval and natural language processing. This requires models capable of converting human text into an internal representation, that can generate answers to user questions. Question Answering has a long history dating from the 1970s, but the initial focus was on the development of domain specific models, that utilized a pre-existing database to answer user generated questions. However, research in the past decade has yielded improvements in the field, where general purpose question answering systems can easily be built on given text, using different model architectures.

Question answering can be broken down into the following three categories-

- IR-Based Factoid Question Answering - This involves answering a question by using context obtained from the internet, or from relevant documents.
- Knowledge-Based Question Answering - This involves using context derived from structured databases, which are queried to obtain the relevant, valid answer.
- Using multiple sources - These systems utilize multiple sources to obtain the relevant answer, and a prominent example is IBM's Watson.

Among IR-Based Question Answering category, there are different kinds on questions that can be asked like, Multiple Choice Questions, True/False Questions, Fill in the blanks, etc. But, we specifically focus on Reading Comprehension based Factoid Question Answering task. In NLP, it is used as an important metric to measure natural language understanding. This is analogous to how a professor teaches in class, gives an exam and analyzes the student's understanding of the class based on his/her answers to the question.



Alyssa got to the beach after a long trip. She's from Charlotte. She traveled from Atlanta. She's now in Miami. She went to Miami to visit some friends. But she wanted some time to herself at the beach, so she went there first. After going swimming and laying out, she went to her friend Ellen's house. Ellen greeted Alyssa and they both had some lemonade to drink. Alyssa called her friends Kristin and Rachel to meet at Ellen's house.......

What city is Alyssa in?

Miami

Figure 1: Example of Reading Comprehension based Question Answering

Reading comprehension based question answering has 3 components namely Passage (P), Question (Q) and Answer (A) as seen in Figue 1. Each Passage (P) is used as context for answering the respective questions. From the example above we can observe that the passage speaks about Alyssa's trip to Miami. The Question (Q) could have different answers or different interpretations, but given the context, we know that the answer is Miami. It is important to note that the answer (A) is a continuous span of text from the passage. So, the model is trained to predict the start and end indices representing the answer from a given passage.

During our analysis of the task using BERT and DistilBERT models, we observed some very interesting trends and patterns. Numeric answers were performing better than non-numeric answers, performance on long passages were better than on short passages, etc. The analysis that particularly caught our attention was our discovery of certain existing imbalances in the dataset that can fundamentally affect language modelling. We notice that the questions can be divided into interrogative categories (what,why,where, when etc.) with a high disparity in the numbers for each category. The dataset largely comprises of what questions, a moderate amount of who and how questions, and a very few where and why questions. Due to this imbalance, most existing models on SQuAD report much higher F1 scores on the question categories that are over represented, and perform poorly on the under represented question categories as seen in Figure 2. Additionally, this imbalance is not apparent since the overall model accuracy is influenced largely by the over represented categories. This results in less than ideal models that perform poorly on questions that are routinely encountered in natural language conversations.

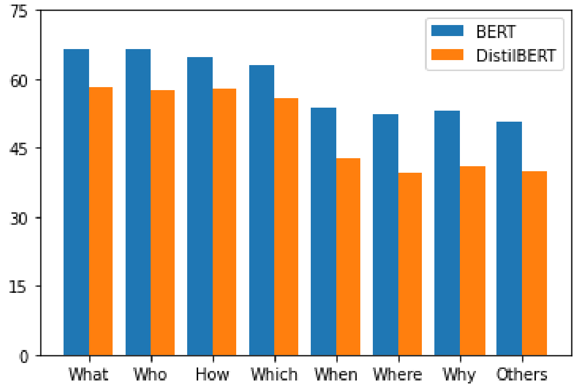


Figure 2: BERT and DistilBERT F1 scores for each question category

Although different architectures and techniques have been used to improve the F1 scores on Question Answering task in general, there isn't any specific research focusing on the problem we are targeting to solve. We believe that it is extremely important to target this issue since at some point in time, these models will be used in a wide range of real-world applications where they are expected to perform equally well on all types of questions and showcase their generality. This stems from the fact that it is equally likely for someone to ask a 'why' question in comparison to a 'what' question.

In order to improve the real world performance of question answering models trained on SQuAD, we explore three approaches -

- Statistical methods such as undersampling
- Question Generation techniques to augment the dataset
- Augmenting the dataset with additional questions from pre existing datasets

We offer an analysis on the different approaches taken, as well as the benefits and demerits of each approach. The Dataset augmentation approaches performed the best at addressing the existing imbalances. The rest of the paper is structured as follows. Section 2 provides a brief description of related works, while Section 3 provides more information on our approach and the reasoning behind it. Section 4 presents the experimental setup we used and Section 5 provides analyses on the results we obtain. We conclude in Section 6 along with providing a framework for future work.

2 Literature Survey

Prior work implementing different model architectures on SQuAD and SQuAD 2.0 exists. An important implementation is the original BERT paper (1) which utilizes SQuAD as a benchmark. In this extremely influential paper, the BERT model is introduced as a powerful language representation model, that drastically improves upon the performance of its contemporaries. BERT incorporates a novel pre-training and fine-tuning approach which reduces the time required for learning, as well as reduces the need for heavily-engineered, task specific architectures. Despite its non specific nature, it is capable of outperforming many task-specific architectures on various sentence-level and token-level tasks. BERT involves pre-training on unlabelled data over different tasks, and uses a pre-training corpus consisting of the BooksCorpus (4) and English text from wikipedia. Fine-training on SQuAD 2.0 is performed to measure its performance. BERT finetuning is relatively straightforward, since it is possible to model downstream tasks through the self-attention heads. This is done by representing the input question and passage as a single packed sequence, and using just a start and end vector. Questions that do not have an answer are treated as having an answer span with start and end at the [CLS] token. Sum of log-likelihoods of the correct positions is taken as the training objective. With fine-tuning for 2 epochs, BERT was capable of obtaining results comparable to those at the top of the leaderboards, and achieves a 5.1 F1 improvement over other models that do not use BERT as one of their components.

Another important model architecture is that of DistilBERT (5), which uses a smaller model, that is pre-trained with knowledge distillation. This results in lighter models, that require smaller computational power, but still obtain performances similar to larger models. Knowledge distillation is a compression technique where a student model which is compact is trained to obtain performance similar to a larger teacher model. DistilBERT utilizes the same general architecture as BERT, with a 2x reduction in number of layers. It also takes advantage of the common dimensionality between teacher and student networks, to initialize the student by taking one layer out of two. Finetuning DistilBERT on SQuAD results in performances withing 3 points of the original BERT model, despite a 40% reduction in model size.

Using conventional machine learning techniques can improve the effectiveness of BERT on SQuAD 2.0 as shown by Hulbard et. al (7). Using a context aware convolutional network, initialized with filter generators has a positive effect on BERT's performance. Replacing traditional CNN filters with meta-filters which are applied over the input sequences results in the generation of a filter-feature map, which can recognize the important and relevant tokens from the given context. Using this approach results in BERT predicting the answerable and non answerable SQuAD 2.0 questions reasonably well. Another approach followed by Hulbard et. al involves using recurrent neural networks with LSTM units to maintain an internal state representation of the question's significance. However, this approach results in significant overfitting of the non answerable questions, with a poor performance on answerable questions.

During the course of our analyses, we discovered that the SQuAD dataset faced certain limitations due to the presence of a skewed dataset. One approach we looked at to correct this imbalance was to generate further questions to address the skew. We aimed to automatically generate questions from the SQuAD dataset based on the work of Zhou et. al (6). Zhou et. al introduce a novel framework called the Neural Question Generation (NQG) framework, that can generate natural language questions from a given text, without the need to specify rules for the same. It utilises a sequence-to-sequence model with an enriched encoder that uses lexical features to generate answer focused questions. The word vector is concatenated, along with the lexical feature embedding vectors and an answer position indicator, which serves as the input to a bidirectional Gate Recurrent Unit (BiGRU) encoder. It also deploys an attention-based GRU decoder that decodes sentences and answer information in order to generate new questions. They use the SQuAD dataset as a test set to determine the effectiveness and quality of the questions generated via a mixture of human ratings and ablation tests. When compared to the baseline score, NGQ outperforms it by a factor of 0.76, indicating that the generated questions are quite related to the given sentences and answer spans.

Another Question Generation technique we utilize is based on the work of Grover et. al (14) which uses the T5 transformer (12) . The T5 consists of an Encoder-Decoder Transformer (13) with a pre-normalization layer and a model configuration similar to BERT. The T5 transformer leverages the effectiveness of transfer learning by utilizing a unified framework that considers every language problem as a text-to-text task. This includes tasks such as machine translation, document summarization, and even classification and regressions. It is pre-trained on the Colossal Clean Crawled Corpus dataset (12) which includes a large amount of text extracted from web crawlers.

3 Proposed Methods

3.1 Imbalance in the Dataset

Q-Type	Train Data	Dev Data
what	77847	7500
who	13904	1081
how	13528	1281
which	8056	490
when	8009	736
where	5320	488
why	1898	191
others	1757	106

Table 1: Breakdown of the type of questions in the dataset

Table 1 displays the type of questions present in the SQuAD dataset. It is apparent that there is a huge disparity in the types of questions present.

3.2 Correcting the Imbalance

Models trained on the SQuAD dataset tend to perform better on the over represented questions, and poorly on the where and why questions as seen in Figure 2. This imbalance results in models performing poorly in real world scenarios, despite obtaining favorable results on the SQuAD dataset. In order to ensure robustness of any model trained on SQuAD, it is imperative that this innate imbalance is accounted for and corrected. The approaches we take to accomplish this task is explained in the remainder of this section.

3.3 Statistical Sampling

In order to obtain a fairly balanced dataset, we look at sampling techniques such as undersampling and oversampling. By randomly undersampling data from the majority category, its overrepresentation in the model's training. This can be observed from Figure 3 wherein the faded parts represent data that is discarded. This increases's the models competency when it comes to the smaller question categories. Similarly, random oversampling of the minority category can increase the models performance when it comes to answering such questions.

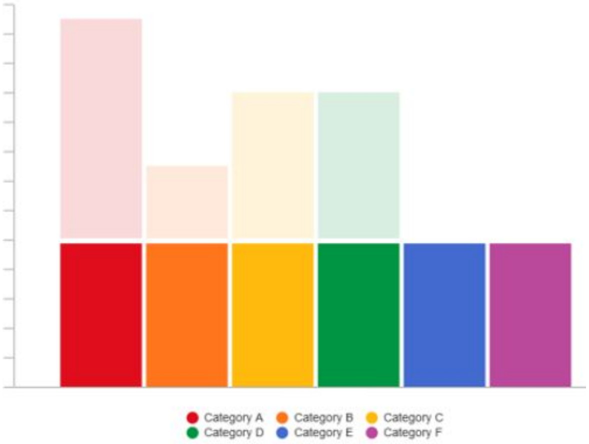


Figure 3: Undersampling technique

3.4 Question Generation

In order to balance the dataset, machine learning based question generation techniques can be leveraged in order to generate more examples from the minority categories. Given the original SQuAD dataset, we use question generation techniques to generate new questions from the same passages. We can observe from Figure 4 that we pass the passage to generate questions-answer pairs from the model which is then filtered. Underrepresented questions from the newly generated questions are sampled and added onto the existing dataset. This task is accomplished by using the T5 transformer previously trained for question generation on the Amazon Customer Reviews Dataset (15).

3.5 Dataset Augmentation

We enhance the dataset by augmenting it with other question answering datasets, in order to ensure a more evenly balanced dataset. To accomplish this task we make use of the Conversational Question Answering Dataset (CoQA) (11). The CoQA dataset consists of conversational questions, with free-form text answers from the relevant passsages. Augmenting SQuAD with CoQA helps us balance the different categories of questions. This can be visualized in Figure 5.

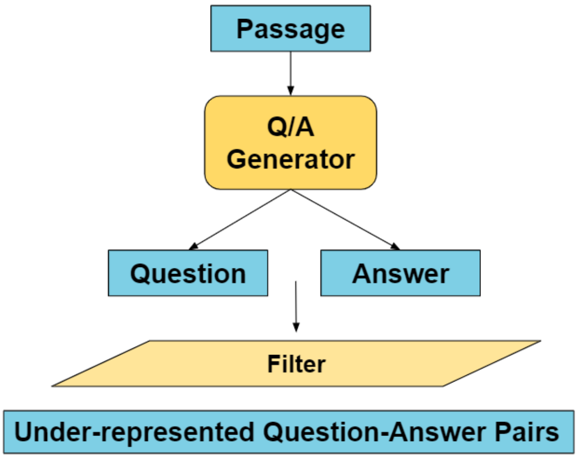


Figure 4: Question Generation flow diagram

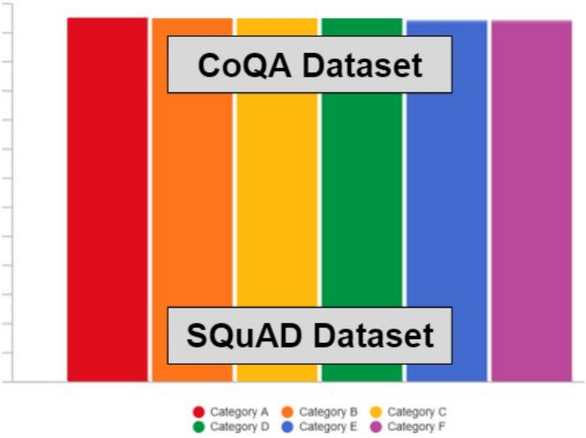


Figure 5: Augmentation technique

4 Experimental Setup

4.1 Dataset

The SQuAD dataset is a large, high quality dataset that serves as an important benchmark in question answering tasks. However, SQuAD consists of questions which are guaranteed to have an answer in the provided context, which decreases its robustness to distracting sentences. This results in models that end up selecting the span that appears to be most related to the question, rather than checking whether the answer is contained within the text. To address these shortcomings the SQuAD 2.0 dataset (3) was constructed, compiling the existing answerable questions from SQuAD, along with unanswerable questions on the same paragraphs. The newly added questions are constructed such that they are relevant to the context, and contain a plausible but incorrect answer.

4.2 Experiments

The primary models we consider are BERT(2) and DistilBERT(5). The reasons for choosing these models are -

- They are the primary models that form the basic backbone architecture of most NLP models.
- A large amount of prior work already utilize these models, thereby making it worthwhile to consider the benefits of balancing the dataset.
- DistilBERT is considered since BERT can take up a large amount of computational resources for training and evaluation.

Since it is computationally inefficient to train these models from scratch, we built them on top of pre-trained models. These pre-trained models were obtained from the Hugging Face open source repository (8). We then followed the following pipeline for cleaning and preprocessing the data -

- Since the dataset contains just the start index for each answer, we tokenize the inputs and format it to include the end index as well
- In typical NLP tasks, long documents are dealt with by truncation. However, that does not appear to be ideal for question answering since removing part of the context could result in losing the answer we are looking for. We deal with large text examples by allowing them to provide several input features, each having a length shorter than the maximum length of the model. Overlap between features is allowed to account for cases where the answer lies at the point of splitting
- We merge the tokens in such a way that the [CLS] token is followed by the question, followed by [SEP], and finally the context. The final prediction will be the first and last token index with reference to the context, which in turn will be the span of our answer

Parameters	Value
Learning Rate	2e-5
Weight Decay	0.01
Epochs	3
Max. Feature Length	384
Dropout	0.1
Activation	GeLU
No. of Layers	12
No. of heads	12

Table 2: BERT configuration

Parameters	Value
Learning Rate	2e-5
Weight Decay	0.01
Epochs	3
Max. Feature Length	384
Dropout	0.1
Activation	GeLU
No. of Layers	6
No. of heads	12

Table 3: DistilBERT configuration

Post the preprocessing and cleaning phase we finetune both the models under the configurations shown in Table 1 and Table 2.

We measure the final model's performance based on the two popular metrics for question answering i.e F1 score and Exact Match (EM). F1 score can be defined as the average overlap between the prediction and ground truth answer. Exact match as the name suggests gives a score of 1 if the predicted answer is exactly the same as ground truth answer, 0 otherwise. It is important to note that DistilBERT makes a tradeoff between architecture complexity and performance as evident in the scores obtained, but it drastically reduces the resources involved in training.

5 Results

Figure 6 and Figure 7 displays the F1 scores obtained for each category of question. The remainder of this Section discusses the trends and inferences that can be gauged.

5.1 Statistical Sampling

The results of our sampling approaches are visible as orange bars in figures 6 and 7. For both BERT and DistilBERT it is evident that the performance on the over represented categories (What,

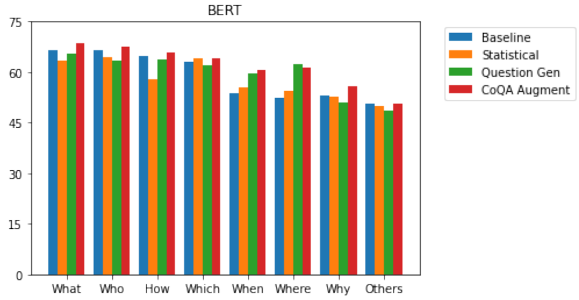


Figure 6: Category wise F1 scores for different approaches - BERT

Who, How) is below that of the baseline implementation. This is because these categories are over represented in the initial dataset, and the baseline models have largely learnt to answer these questions. Our undersampling results in a reduction in the number of questions from these categories, thus resulting in a slight performance reduction. However, the Which,When, and Where categories have a slightly better F1 score when compared to the baseline. This occurs due to a more balanced dataset, ensuring each category has adequate training examples. Our sampling approach results in a reduction in the performance on Why questions. This occurs because the number of questions of this category is extremely less, and required significant oversampling to obtain more equitable numbers. We hypothesize that this results in an overfitting on the training dataset, thereby affecting the performance on the test dataset.

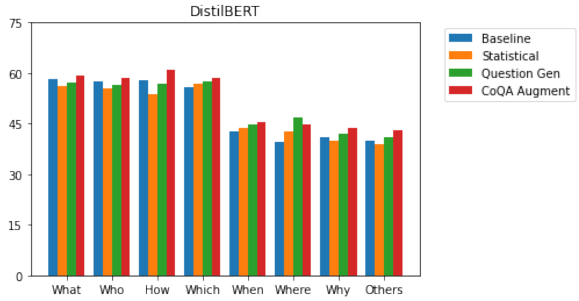


Figure 7: Category wise F1 scores for different approaches - DistilBERT

5.2 Question Generation

After using a T5 transformer to generate new questions from the SQuAD passages, the results obtained from training are visible as green bars in figures 6 and 7. Similar to the Sampling approach, the performances on the over represented categories (What,Who,How) sees a slight decrease for both BERT and DistilBERT. This occurs because no new questions from these categories are included in the augmented dataset. However, the newly generated questions from the Which, When, and Where categories help increase the training samples for the models, thereby leading to an increase in its performance. However, the question generation technique did not produce an even amount of questions from the different categories, and provided very few why questions. Hence, there is no significant improvement in this category.

5.3 Dataset Augmentation

The final approach involved augmenting the CoQA dataset with SQuAD, in order to obtain a more balanced dataset. The results obtained from training are visible as red bars in figures 6 and 7. This approach proved to be the best since it helped increase the number of examples for all categories uniformly with real questions unlike our previous attempt to generate synthetic questions. Another point to note is that the CoQA dataset samples were very similar to SQuAD dataset samples ensuring an even distribution. Due to the presence of more training samples, we can see that both BERT and DistilBERT showed a significant improvement when compared to the baseline, for all categories.

6 Conclusion

Our work looked at enhancing the existing work that looks at SQuAD and making it more practical in real world scenarios. We looked at the BERT and DistilBERT models which form an important backbone of modern question answering systems. We present and explore ideas that can help balance the SQuAD dataset's inherent imbalance. The undersampling approach does help in balancing the dataset, but it affects the accuracy as a whole. Dataset augmentation techniques such as question generation and merging datasets prove to be the best bet at rectifying the imbalance. However, we did run into certain limitations such as the generated questions being lopsided affecting the total performance. Furthermore, the CoQA dataset being used is a conversational dataset, and may not have been the most optimal choice of dataset chosen for merging. Given the limited time and lack of prior experience, we believe our work is a good launch pad for future attempts. Using better question generation techniques, question paraphrasing techniques, and the usage of more relevant datasets can help improve the SQuAD dataset. While we looked at one direction of making these models work better or generalize for a real-world situation, interesting future work could involve other directions like,

- Continuous learning models i.e., models evolving with time as new information comes in and old information becomes outdated. For example, question answering on rapidly changing COVID-19 information.

- Open Domain Question Answering i.e., without having to train the model on very specific datasets, yet making them perform well on multiple tasks.

7 Contributions

Team : We would have virtual Zoom calls atleast twice a week to discuss findings, evaluate new ideas or methodologies, check our progress and split the work equally. On weekends, we would meet in person at the library and work together the whole day. We collaborated over Google Colab where we ran all our code and experiments. The final code would then be pushed to GitHub by one of the team members. We contributed equally to the project as a whole. Following are our respective highlights,

Gagan : Came up with the idea to work on Question Answering after going through latest research work in NLP. Worked on base implementation for Question Answering using DistilBERT on SQuAD dataset including pre-processing, training and evaluation scripts. Wrote scripts to explore and analyze interesting patterns in the performance of our models on different question types which forms the basis of our final project. Worked on analyzing and evaluating the final result. Solely responsible for working on the presentation slides, flow, content and animations. Also contributed 35% to both the reports.

Mukund : Formulated parts of the project hypothesis and research statement. Worked on majority of the code base including dataset generation pipelines for data augmentation technique, under sampling technique and parts of the question generation technique. Trained, validated and fine-tuned both the BERT and DistilBERT models. Also responsible for end to end metric analysis, result analysis and model evaluation scripts. Ran inference and compiled evaluation results for the framework holistically and recorded the obtained metrics and scores.

Tarun : Worked on the dataset category analysis script and formulated sections of the dataset generation pipeline. Based on existing research looked into different question generation approaches and decided to use the T5 model. Wrote the pipeline for question generation from the SQuAD dataset. Researched existing literature to obtain a comprehensive literature survey. Major contributor to the report for both Assignment 3 and Assignment 4.

References

[1] Rajpurkar, Pranav, et al. "Squad: 100,000+ questions for machine comprehension of text." arXiv preprint arXiv:1606.05250 (2016).

[2] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018)

[3] Rajpurkar, Pranav, Robin Jia, and Percy Liang. "Know what you don't know: Unanswerable questions for SQuAD." arXiv preprint arXiv:1806.03822 (2018)

[4] Y. Zhu et al., "Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books," 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 19-27, doi: 10.1109/ICCV.2015.11.

[5] Sanh, Victor, et al. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter." arXiv preprint arXiv:1910.01108 (2019).

[6] Zhou, Qingyu, et al. "Neural question generation from text: A preliminary study." National CCF Conference on Natural Language Processing and Chinese Computing. Springer, Cham, 2017.

[7] Gupta, Suhas. "Exploring Neural Net Augmentation to BERT for Question Answering on SQUAD 2.0." arXiv preprint arXiv:1908.01767 (2019).

[8] Hugging Face - Home of Machine Learning, https://huggingface.co/, accessed on 4/8/2022

[9] Kwiatkowski, Tom, et al. "Natural questions: a benchmark for question answering research." Transactions of the Association for Computational Linguistics 7 (2019): 453-466.

[10] Choi, Eunsol, et al. "QuAC: Question answering in context." arXiv preprint arXiv:1808.07036 (2018).

[11] Reddy, Siva, Danqi Chen, and Christopher D. Manning. "Coqa: A conversational question answering challenge." Transactions of the Association for Computational Linguistics 7 (2019): 249-266.

[12] Raffel, Colin, et al. "Exploring the limits of transfer learning with a unified text-to-text transformer." arXiv preprint arXiv:1910.10683 (2019).

[13] Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

[14] Grover, Khushnuma, et al. "Deep Learning Based Question Generation Using T5 Transformer." International Advanced Computing Conference. Springer, Singapore, 2020.

[15] Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering, R. He, J. McAuley ,WWW, 2016