# Introduction

The Quora QA answering dataset consists of a single file with two columns - `question` and `answer` and doesn't have context column. There are no additional documents which can be used for information retrieval. The topics are broad and can range from global affairs, education and healthcare to relationships. This makes it a particularly challenging task of open domain QA in a closed book setup. The model is expected to answer questions using the knowledge stored in its parameters.

Moreover as the data is scraped from the internet, it can make the LLM notoriously prone to hallucinations. GPT2 is trained on web data which contains a lot of unfiltered content from the internet, which is far from neutral. Open-AI has warned that "models like GPT-2 do not distinguish fact from fiction, we don't support use-cases that require the generated text to be true."

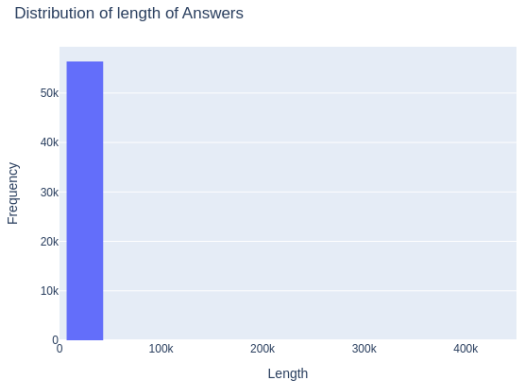
| Question | Answer | |
|---|---|---|
| Where can I find packers and movers in Noida? | http://www.buzznoida.com /scategory/packers-movers/81.aspx | Dead and unsafe links |
| What exercise can lose face fat? | You can attain perfectly shaped cheekbones by losing weight in your face areas with these 7 effective steps. _____ Subscribe To Our YT Channel : [LINKED_TEXT: My Weightloss Project] [URL: https://shorturl.at/jnX16] For Interesting Health, Fitness, Nutrition Tips & Hack | Sponsored content |
| Why are Democrats afraid of Christians? | Well that is a weird way to misspell "disgusted by" Maybe you meant "ashamed of" that is closer. Autocorrect sucks huh. | Non-inclusive, hate speech |
| are roos scared of saltwater crocodiles in Australia? | They [redacted] aught to be . | obscene |

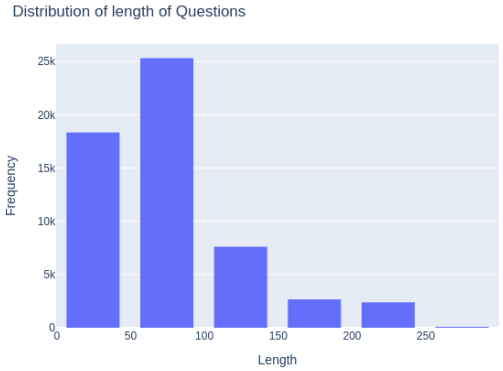
We can observe from this table that the data has a lot of problems.

The dataset contains several urls some even included in the question ( **6** in total). **11k** answers contain  urls. URL in the text is a very good indicator of spam and sponsored content. Even if they are included for reference, these shouldn't be the part of model training data as URLs can be broken and don't directly provide information. We want the model to itself have the knowledge to answer questions instead of linking to URLs.

Quora is very popular forum and it attracts all sorts of audience. It is a major task to moderate the content for obscenity. Therefore, to ensure we don't train the model on these words, we should find and remove them. There is a popular [repo] containing list of bad words. We can use it for substring match and replacement. Word/String match may not be accurate as substring match and may give false negatives. About **200** unique questions contain such words.

We can observe that questions have lengths upto **300** characters. But, the answers can be pretty long upto **450k** characters. As majority of the answers are within **50k** chars we discard the longer ones as they are usually spam.

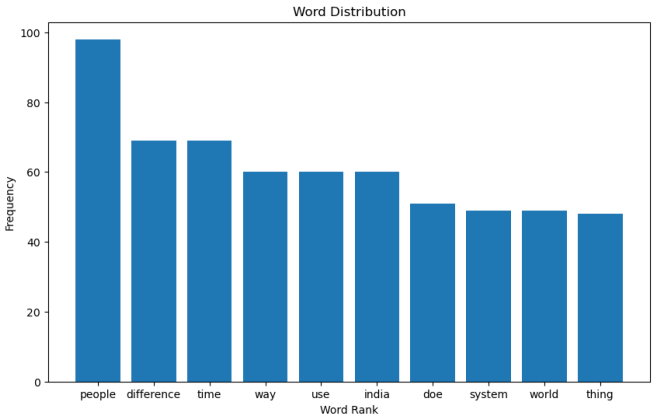



There are several exact duplicate questions and only **~3k** unique questions. Each answer may focus on different aspects of the question. We want to maintain diversity in the dataset because the main objective is to build a conversational and interesting QA agent so we keep them.

## Question text analysis

For analysis, the text is first tokenized into words. Then non-alphanumeric characters are removed. The resulting text is lower cased, removed of stop words, and lemmatized. Lemmatization gives more meaningful words as compared to stemming. This is particularly useful for topic modelling. We have only ~3k questions so the higher computational complexity is still practical. The most common words should be those used in question construction - “What”, “Where” etc and not informative. Therefore, part-of-speech (POS) tagging is used to identify the nouns. This gives a resulting total of **~5k** unique tokens.




People is the most frequent token. As expected, India is among the most frequent words. Difference may because of being frequently used in question of type – *What is the difference between …*

LDA topic modelling

| |
|---|
| Topic 0: 0.008*"country" + 0.006*"number" + 0.005*"travel" + 0.005*"language" + 0.004*"people" + 0.004*"life" + 0.004*"system" + 0.004*"say" + 0.004*"ancient" + 0.004*"kind" |
| Topic 1: 0.069*"i" + 0.018*"get" + 0.008*"dog" + 0.006*"use" + 0.006*"want" + 0.004*"difference" + 0.004*"im" + 0.004*"buy" + 0.004*"india" + 0.004*"cat" |
| Topic 2: 0.010*"i" + 0.009*"good" + 0.008*"use" + 0.008*"people" + 0.008*"think" + 0.007*"one" + 0.006*"know" + 0.005*"way" + 0.005*"make" + 0.005*"system" |
| Topic 3: 0.020*"i" + 0.007*"one" + 0.007*"india" + 0.006*"doe" + 0.005*"make" + 0.004*"learn" + 0.004*"difference" + 0.004*"mean" + 0.004*"show" + 0.003*"war" |

These topics are not complete random and can have meaning and interpretability.

1. High level overall concepts - like country, travel, language, people, life.

2. popular topics like  cat and India

3. verbs - use, think, and, make

4. most common words in questions - like what is the `difference`, how to `make`, how to `learn`, what does it `mean`, how to `show`.

# Literature Survey

## LM as Knowledge Base

Language models can be used as flexible knowledge bases. They require no schema engineering, allow practitioners to query about an open class of relations, are easy to extend to more data, and require no human supervision to train. In this paper they demonstrated using ***BERT*** to achieve SOTA on open-domain question answering.

## T5 for QA

In this paper they demonstrated scaling T5 to perform competitively open-domain question answering datasats. But, the model size around 11B parameters may be prohibitively expensive for fine-tuning under resource constrained settings. They also found cleaning the dataset by removing unanswerable questions from the benchmarks significantly improved their performance. This demonstrates the importance of clean datasets.

## BART

BART is a state-of-the-art model on a range of a tasks like abstractive dialogue, question answering, and summarization tasks. It is based on BERT (due to the bidirectional encoder) with a GPT decoder.

## GPT-2

It is trained on a lot of unsupervised Web data. It has  excellent 0-shot performance but generated text can be prone to hallucinations. GPT2 is trained on web data which contains a lot of unfiltered content from the internet, which is far from neutral. Open-AI has warned that "GPT-2 do not distinguish fact from fiction, we don’t support use-cases that require the generated text to be true." It doesn't make sense to use gpt for finetuning a QA system.

# Methodology

## Dataset cleaning

### Extractive Summarization of answers

We may want to summarize the longer answers (say having more than 500 words) for practical reasons. We can extract the important sentences and phrases from the answer using TextRank. TextRank construct a graph using the sentences as nodes and semantic similarity as edge weights. It iteratively ranks each sentence assigining better rank to sentences connected to important sentences. It is fast and useful for keyword extraction and summarization. This will be better than truncating the long answers.

To further improve the data quality, we may use LLM to generate a coherent abstractive summary from the extracted text. But, for now I am going to directly use TextRank generated summary. It is practical because we need to run it for almost each of the 50k rows. Also, as we are going to use a pretrained LLM for finetuning, we can assume that the model will not forget the grammar and only memorize the facts

### Removing redundant answers by 0 shot classification

There are multiple answers to a single question (upto 106 answers). We may not need all of them. Without relying human evaluators, we can use a capable llm to score these answers. This would be better than using heuristics like text similarity. Judging an answer can be a complex task as we don't have the voting information. Therefore, I am using a popular model based on Microsoft's DeBERTa-v3-base. It was trained on multiple Natural Language Inference (NLI) datasets and is suitable for zero-shot classification. It has ~200M params and inference time is reasonable.

We want classify whether the given answer does actually answers the question or not. We can prompt with question and the given answer and ask whether it is reliable. Using the confidence score from the llm we can sort the answers. We can keep answers with the top **20%** of scores. This will result in **~5x** reduction in dataset size resulting in ~15k samples.

## Data Augmentation

As we have a small training set size ~10k samples, we can add more samples by augmenting the existing ones. It will also make the model more robust against the diverse data - by simulating spelling mistakes and synonym replacements. NLPaug is a standard

open-source library for this purpose. We apply these to questions only as the model is expected to answer any given question. This increases training data to 12k.

# Results

We split the data into train, dev and test with (0.8, 0.1 and 0.1) ratios.

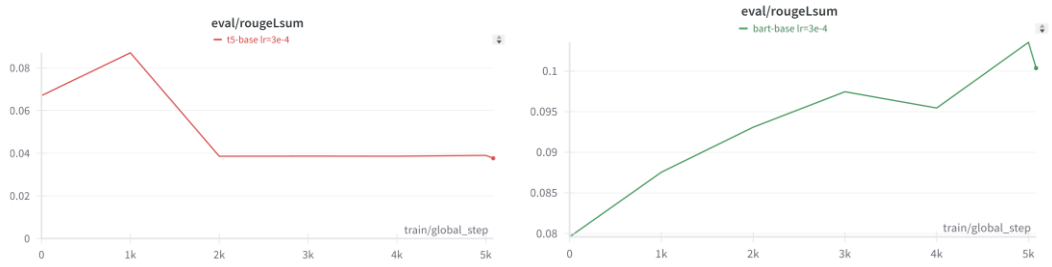
We focus on **T5-base** and **BART-base.**

Initially, both of these models were directly used to evaluate the performance. Then they were finetuned, to see which model improves more.

Test set results

| Model | Loss | Rouge1 | Rouge2 | RougeL | RougeLsum |
|---|---|---|---|---|---|
| T5 (no finetune) | 12.87 | 0.083 | 0.013 | 0.068 | 0.067 |
| T5 | 5.45 | 0.04 | 0.004 | 0.038 | 0.038 |
| BART (no finetune) | 8.69 | 0.095 | 0.018 | 0.074 | 0.08 |
| BART | 4.01 | 0.12 | 0.03 | 0.095 | 0.1 |

Using a learning rate of 3e-4 for T5-base which should work well for QA as reported [here] we found it was overfitting. The validation error didn't improve.




But, for the same learning rate BART was able to learn.

We observe the dev and test performance marginally drops after augmentation. But, training loss is the least sofar, so it is overfitting. We will decrease the lr.

Test

| Model | Loss | Rouge1 | Rouge2 | RougeL | RougeLsum |
|---|---|---|---|---|---|
| BART | 4.01 | 0.12 | 0.03 | 0.095 | 0.1 |
| BART augmented | 4.106 | 0.11 | 0.023 | 0.094 | 0.1 |
| BART aug + mirror + lr=2e-5 | 3.83 | 0.11 | 0.026 | 0.087 | 0.092 |

### Mirror Task

The motivation is that multi-task learning will make the llm more generalizable and improve data efficiency. We can give the llm more complete knowledge of the dataset by also asking it to predict the question for a given answer.

Steps

1. finetune the llm to predict the question for each answer.
2. then, finetune it on the original task of answer generation.

There is reduction in test loss by this method.

## Conclusion

We explored the QA dataset, cleaned and found interesting topics. We fine-tuned multiple models on the reduced dataset and found that BART performs better that T5. This can be attributed to the fact that T5 has more parameters and requires more data. We also perfomed data augmentation but observed that it can lead to overfitting if it is too similar to exisiting training data specially when using high learning rate. We perfomed an interesting experiment involving both answer and question prediction which results in singificant loss in test loss.