Training Question Answering Models From Synthetic Data

Raul Puri[‡*], **Ryan Spring**[§], **Mohammad Shoeybi**[‡],
Mostofa Patwary[‡*], **Bryan Catanzaro**[‡]
[‡]NVIDIA, Santa Clara, California, USA
[§]Rice University, Houston, Texas, USA.

Abstract

Question and answer generation is a data augmentation method that aims to improve question answering (QA) models given the limited amount of human labeled data. However, a considerable gap remains between synthetic and human-generated question-answer pairs. This work aims to narrow this gap by taking advantage of large language models and explores several factors such as model size, quality of pretrained models, scale of data synthesized, and algorithmic choices. On the SQuAD1.1 question answering task, we achieve higher accuracy using solely synthetic questions and answers than when using the SQuAD1.1 training set questions alone. Removing access to real Wikipedia data, we synthesize questions and answers from a synthetic text corpus generated by an 8.3 billion parameter GPT-2 model and achieve 88.4 Exact Match (EM) and 93.9 F1 score on the SQuAD1.1 dev set. We further apply our methodology to SQuAD2.0 and show a 2.8 absolute gain on EM score compared to prior work using synthetic data.

1 Introduction

One of the limitations of developing models for question answering, or any Deep Learning application for that matter, is the availability and cost of labeled training data. A common approach to alleviate this need is semi-supervised learning, wherein one trains a model on existing data and uses it to label more data for training (Zhu, 2005; Chapelle et al., 2009; Zhu and Goldberg, 2009; Kingma et al., 2014). This technique has demonstrated benefits in recent literature for image classification (Xie et al., 2019) and question answering (QA) tasks (Alberti et al., 2019a; Dong et al., 2019). However, the complexities of generating questions and answers in natural language proves challenging for existing methods, with a large gap in quality remaining between synthetic and human-generated data.

Text	Albert Einstein is known for his theories of special relativity and general relativity. He also made important contributions to statistical mechanics, especially his mathematical treatment of Brownian motion, his resolution of the paradox of specific heats, and his connection of fluctuations and dissipation. Despite his reservations about its interpretation, Einstein also made contributions to quantum mechanics and, indirectly, **quantum field theory**, primarily through his theoretical studies of the photon.
117M	Which two concepts made Einstein's post on quantum mechanics relevant?
768M	Albert Einstein also made significant contributions to which field of theory?
8.3B	Because of his work with the photon, what theory did he indirectly contribute to?
Human	What theory did Einstein have reservations about?

Table 1: Questions generated by models of increasing capacity with the ground truth answer highlighted in bold. As model size grows, question quality becomes increasingly coherent, complex, and factually relevant.

In this work, we close this gap using only synthetic questions generated from large models. We also show that naive scaling of Alberti et al. (2019a) alone is insufficient. We demonstrate that answer candidate generation is foundational to synthetic question quality, and there are issues aligning the answer distribution which do not improve with scale. Throughout this paper we focus primarily on question generation as a data augmentation method for existing question answering tasks. Focus on this type of semi-supervised setting is a necessary first step to enable future work applying question generation to low-resource tasks.

Similar to prior work (Alberti et al., 2019a; Dong et al., 2019), we use a 3-step modeling pipeline consisting of unconditional answer extraction from text, question generation, and question filtration. Our approach for training question generators on labeled data uses pretrained GPT-2 decoder models

* Corresponding authors: Raul Puri raulpuric@berkeley.edu, and Mostofa Patwary mpatwary@nvidia.com

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 5811–5826,
November 16–20, 2020.

and a next-token-prediction language modeling objective, trained using a concatenation of context, answer, and question tokens. We demonstrate that pretraining large generative transformer models up to 8.3B parameters improves the quality of generated questions. Additionally, we propose an overgenerate and filter approach to further improve question filtration. The quality of questions produced by this pipeline can be assessed quantitatively by finetuning QA models and evaluating results on the SQuAD dataset.

We demonstrate generated questions to be comparable to supervised training with human-labeled data. We measure this by using the ratio of accuracies between a QA model trained on synthetically generated data and a model trained on human labelled data. For answerable SQuAD1.1 questions we recover 100.8% of fully supervised EM and 100.1% of fully supervised F1 scores, when training on purely synthetic questions and answers generated from unlabeled data. Specifically, we achieve scores of 88.4 and 94.1 compared to supervised training which achieves 87.7 EM and 94.0 F1. Finetuning the resulting model on real SQuAD1.1 data reaches 89.4 EM and 95.1 F1 score, which is higher than any prior BERT-based approach. In Table 1, we show that the generated questions are qualitatively similar to ground truth questions, with the quality improving with the model size.

Going further, we show that QA models can be successfully trained from fully synthetic data, by running question and answer generation on a corpus generated from an unconditional GPT-2 model, and achieve an EM of 88.4 and F1 of 93.9. This approach performs comparably to generating questions from real data recovering 100.3% of fully supervised EM and F1 scores.

In summary, our contributions are as follows:

- We demonstrate that finetuning a model on synthetic questions and answers generated from a synthetic corpus creates a QA model better in SQuAD1.1 EM and F1 scores than one trained from human-labeled data.
- We show that by scaling the model size, using better pretrained models, and leveraging large synthetically generated data, we achieve state of the art results and show 1.7 absolute gain on SQuAD2.0 EM score compared to prior work using synthetic data.
- Through detailed ablation studies we identify that the quality of answer generation is fundamental to high fidelity question generation and properly aligning the answer distribution boosts scores by 19.8 EM points.

2 Method

In this work we seek to generate high quality training data for SQuAD style extractive question answering over a given set of documents D. This requires us to sample (c, q, a) triples for given paragraph contexts $c \in D$ according to probability $p(q, a|c)$, where q is a question resulting in answer a, which exists as a contiguous span of text in c. Leveraging the roundtrip consistency method (Alberti et al., 2019a), we achieve this by using a three step approach consisting of Answer Generation $\hat{a} \sim p(a|c)$, Question Generation $\hat{q} \sim p(q|\hat{a}, c)$, and Roundtrip Filtration $\hat{a} \stackrel{?}{=} a^* \sim p(a|c, \hat{q})$. As illustrated by Algorithm 1 in the Appendix A.1, the synthesized dataset of triples is then used to finetune and train a BERT-based QA model similar to (Devlin et al., 2018).

2.1 Answer Generation: $\hat{a} \sim p(a|c)$

For a model to perform well on a specific dataset, we need to match its answer distribution. Our goal is to learn an answer candidate generator $p(a|c)$, that acts as a prior for the dataset's answer distribution. Earlier work (Dhingra et al., 2018; Lewis et al., 2019) using named entity and noun phrase answer candidates performed best only on those portions of the data distribution.

To achieve this we finetune a BERT-style transformer model with hidden size H for extractive span selection. However, unlike BERT finetuning for question answering we omit the question tokens. This yields an unconditional answer extractor model $p(a|c)$ that predicts the start and end of a token span $(s, e) = a$. Similar to (Alberti et al., 2019a) we used an answer extraction head that models start and end tokens jointly.

$$p(a|c; \theta_A) = \frac{e^{f(a,c;\theta_A)}}{\sum_{a''} e^{f(a'',c;\theta_A)}}$$

$$f(a, c; \theta_A) = \text{MLP}(\text{CONCAT}(\text{BERT}(c)[s], \text{BERT}(c)[e]))$$

Our MLP layer consists of one hidden layer with hidden size $2H$, followed by a ReLU nonlinearity, and a projection from activations to logits.

2.2 Question Generation: $\hat{q} \sim p(q|\hat{a}, c)$

We develop a conditional question generation model, $p(q|a, c)$ using a pretrained GPT-2 model. As input to our model, we concatenate context tokens, answer tokens, and question tokens into a single sequence, separated by end of sequence tokens. We trained the question generation model with a left to right next token prediction loss modeled over the entire concatenated sequence. This method of multi-input controlled text generation draws on inspiration from prior work (Puri and Catanzaro, 2019; Raffel et al., 2019; Dong et al., 2019). More details and visualizations of the input representation and training loss can be found in Figure 3 in the Appendix A.2. To sample from our learned model we concatenate the context tokens with the answer tokens and autoregressively sample output question tokens.

To aid our model with generation we employ start and stop word filtration. We prepend '*question:*' and append '*:question*' tokens to the questions in our training dataset. During inference time, if the model does not sample a sequence containing both the start and stop words we discard the example entirely.

2.3 Roundtrip Filtration: $\hat{a} \stackrel{?}{=} a^* \sim p(a|c, \hat{q})$

In roundtrip filtration (Alberti et al., 2019a) an extractive question answering model $p(a|c, q)$ is trained on the available labeled data. When a new question, answer, and context triple $(c, \hat{q}, \hat{a})$ is generated we apply the QA filtration model $p(a|c, \hat{q})$ to the context and question. The resulting answer a^* from the model is compared to the answer $\hat{a}$ from the triple. If the two are equivalent then the question is considered admissible.

In the original work, however, the authors draw attention to the precision of the method. While it does discard invalid questions, several valid questions are discarded as well. To avoid losing valuable pieces of information to train our question answering models we propose generating two questions, instead of one question, for each candidate answer. Roundtrip filtration is then applied to each question individually. If a triple is decided as acceptable then it is kept regardless of whether the other triple is acceptable, leading to a scenario where both can be kept. This method is similar to prior work in overgeneration and reranking of generated questions (Heilman and Smith, 2010a).

3 Experiment Setup

For all the implementations and training of transformer models we use the Megatron-LM codebase (Shoeybi et al., 2019). For off-the-shelf weights and implementations of BERT-Large we rely on the HuggingFace's transformers codebase (Wolf et al., 2019). For model configurations, hidden size, number of layers, and attention heads, we used the configurations detailed in Megatron-LM. To finetune our GPT-2 models we reused the pretraining hyperparameters detailed in Appendix A.3, except for a batch size of 32, and a learning rate of 2e-5 decaying to zero over six epochs of finetuning data. Finetuning our BERT models for filtration, answer generation, and question answering was all done with a learning rate of 1e-5 and a cosine decay scheduled over 2 epochs of training data. BERT pretraining details are described in Appendix A.4. We refer to our models as BERT-345M (345 million parameters) and BERT-1.2B (1.2 billion parameters) and the original BERT model as BERT-Large.

To train and evaluate the question generation pipeline for our ablation studies in sections 5 and 6 we used a data partitioning scheme as shown in Figure 4 in Appendix A.5. A similar data pipeline has been employed in concurrent work of Klein and Nabi (2019). We split the SQuAD training data into equal portions, partitioning the data randomly into two sets of documents. One half of the documents is used to train the answer generator, question generator, and filtration models while the second half of the documents is used to generate synthetic data to finetune a QA model. The finetuned QA model is then evaluated on SQuAD dev set, where the evaluation results are used as a surrogate measure of synthetic data quality. The partitioning of the dataset is done to avoid leakage and overfitting between the data seen at training time and generation time thereby testing the generalization capabilities of our models. Since shuffling is done randomly we repeat this process 5 times with different seeds for every ablation study and report the mean of our results. Due to the large hyperparmeter space we do not perform any learning rate search and use the static learning schedules as described above. We note that the data partitioning is done only for the ablation studies and hyperparameter selection. For the final models in Figure 1 and Tables 2 & 3, we use the entire SQuAD dataset.

Text Source	Source Data Size	finetune data	# Questions	EM	F1
Wikipedia	638 MB	Synthetic	19,925,130	88.4	94.1
		+SQuAD	20,012,729	**89.4**	**95.2**
8.3B GPT-2	480 MB	Synthetic	17,400,016	88.4	93.9
		+SQuAD	17,487,615	**89.1**	**94.9**
SQuAD1.1	14MB	SQuAD	87,599	87.7	94.0

Table 2: Finetuning BERT-345M on synthetic and human-generated data. Using 1.2B parameter models we synthesize question answer pairs from real Wikipedia corpus and synthetic corpus generated from an 8.3B GPT-2 model. Completely synthetic data does better than training with real data. Finetuning with real SQuAD1.1 data afterwards further boosts performance.

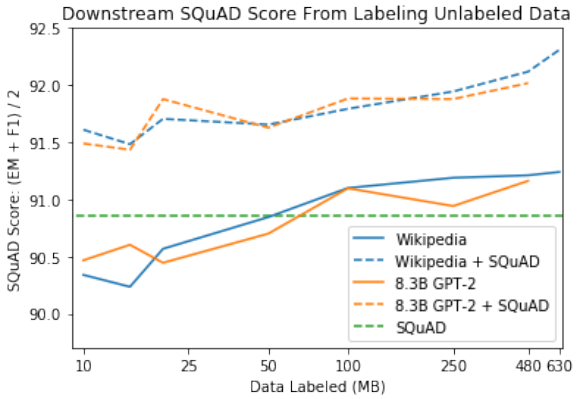

Figure 1: Effect of labeling data size on downstream SQuAD1.1 score. After finetuning BERT-345M models on synthetic data we finetune further on human generated SQuAD1.1 data.

4 Results

In this section we present our results using the best combination of models, algorithms, and parameters. In the following sections, we will perform detailed ablation study and show contributions from each of these choices.

We train a 1.2 billion parameter answer generator, question generator, and question filtering model. In these experiments we use the entire SQuAD1.1 dataset instead of only training on half of the labeled data since we are not doing any model or hyperparameter search. We then use these models to label synthetic data from two sources outside of SQuAD1.1 . We first label data from real Wikipedia documents with the overlapping documents from the SQuAD1.1 training and dev set removed. In parallel we label data from synthetic Wikipedia documents generated by an 8.3B GPT-2 model. This model was first trained with the Megatron-LM codebase for 400k iterations before being finetuned on only Wikipedia documents for 2k iterations. This allows us to generate high quality text from a distribution similar to Wikipedia by using top-p ($p = 0.96$) nucleus sampling.

Implementation	EM	F1
BERT-Large (Alberti et al., 2019a)	78.7	81.9
+ 3M Questions	80.1	82.8
UniLM (Dong et al., 2019)	80.5	83.4
+ 9M Questions	84.7	87.6
BERT-Large	77.4	80.6
+ 3M Questions	81.6	84.5
BERT-345M	84.9	88.2
+ 3M Questions	85.8	88.6
+ 8M Questions	**86.4**	**89.2**

Table 3: Comparison with prior work. Improvements in question generation allow for improved SQuAD2.0 score even without generating unanswerable questions.

Table 2 shows results when the synthetic data is finetuned on a BERT-345M QA model. We showcase that we are able to recover and surpass the performance of human-labeled data by only using synthetic data generated from synthetic corpus. Using questions synthesized on real Wikipedia data we do even better. Finetuning this model afterwards on the actual SQuAD1.1 dataset allows us to achieve a 1.7 and 1.2 point boost to our EM and F1 scores.

In Figure 1 we examine the relationship between SQuAD1.1 score and the amount of text labeled. We find that the performance of training with purely synthetic data observes a log-linear relationship that begins to saturate at approximately 100 MB of text labeled. However, finetuning these models on labeled SQuAD1.1 data demonstrates continued improvement even beyond saturation. The performance of these post finetuned models continues to improve even past 500 MB of data labeled.

In question generation we aim to effectively expand the knowledge and capabilities of our QA models by generating synthetic QA pairs not in the original dataset. The unsupervised LM takes this further to generate these QA pairs from synthetic text. Generating synthetic questions from synthetic text is important because it draws upon knowledge not explicitly stated in the original corpus. While preliminary, it's noteworthy that the fully synthetic experiment works so well, and our results point towards a promising direction for future work. We provide QA examples generated from real and synthetic Wikipedia in Appendix A.7 & A.8.

Comparison with prior work. To quantify the improvements in question generation quality de-

rived from improvements to language models and our generation techniques we compare our results to the original roundtrip consistency work from (Alberti et al., 2019a). We generate 3 million questions from real Wikipedia text and finetune the public BERT-Large model on this data. We then finetune the model on the human-generated SQuAD2.0 dataset and evaluate on the dev set. Unlike prior work we do not generate any unanswerable questions, yet we find in Table 3 that our synthetic data approach outperforms the prior work. This is despite our BERT-Large baseline underperforming the numbers reported in (Alberti et al., 2019a) by a full point. We also compare our methods with the state of the art in synthetically trained SQuAD2.0 (Dong et al., 2019) and find that with a similar number of questions we outperform existing methods, and with even more labeled data this trend persists.

Data Labeling Cost: To label 8 million datapoints we used approximately 5200 GPU hours and 7200 CPU hours. With Azure costs of ∼3$ per GPU hour this comes out to about 16 thousand dollars for the whole dataset or .2¢ per datum. With software optimization we expect further reductions.

5 Model Scale

We show in this section that as we improve pretraining tasks, pretraining scale, and model scale, synthetic data also improves. To show improvements in question generation we track the resulting SQuAD1.1 evaluation score when a BERT-style model is finetuned on the synthetic data. Table 4 summarizes the benefits of using larger models for answer generation, question generation, and question filtration. The following subsections ablate this result to show the contributions from scaling individual components of the synthetic data pipeline.

Model Size				# Questions	EM	F1
Answer	Question	Filter	QA			
345M	345M	345M	345M	116721	85.3	92.0
1.2B	**1.2B**	**1.2B**	**345M**	**184992**	**87.1**	**93.2**
Human Generated Data			345M	42472	86.3	93.2

Table 4: SQuAD1.1 performance using synthetic data. Downstream QA models used in all experiments are 345M parameters.

5.1 Scaling Question Generation

Question generation plays a critical role in our synthetic data pipeline: it must synthesize linguistically and logically coherent text even if the text does not exist within the provided context. In this section we investigate the relationship between question generator scale and downstream SQuAD1.1 performance. We isolate the quality of question generation by using ground truth answers from the SQuAD1.1 dataset to generate questions and finetune a BERT model before evaluating it on the SQuAD1.1 dev set. We perform no question filtration in between generation and finetuning. From our experiments in Table 5 we find that SQuAD1.1 performance increases monotonically. Additionally, the number of valid samples that pass stopword filtration increase with larger models, indicating bigger models maintain coherency during sampling. For comparisons with prior work we train a question answering model with our BERT model (BERT-345M) and the original BERT-Large model. (Klein and Nabi, 2019) use a feedback loop to improve the question generator and BERT-Large question answering model. Compared to our work we find that a similarly parameterized set of models achieve equal if not better performance despite using only a single supervised pass through the data and no feedback loop.

Question Generator	# Questions	EM	F1
117M	42345	76.6	85.0
345M (Klein and Nabi, 2019)	-	75.4	84.4
345M (w/ BERT QA model)	42414	76.6	84.8
345M	42414	80.7	88.6
768M	42465	81.0	89.0
1.2B	42472	83.4	90.9
8.3B	**42478**	**84.9**	**92.0**
Human Generated Data	42472	86.3	93.2

Table 5: Effect of question generator scale on SQuAD1.1 performance. Ground truth answers are used to generate questions without filtration for finetuning.

Answer Generator	#Questions	EM	F1
BERT-Large	227063	77.7	87.6
BERT-345M	229297	79.1	87.9
BERT-1.2B	**229067**	**79.2**	**88.3**
Human Generated Answers	42472	83.7	91.1

Table 6: Comparison of answer generator pretraining and scale on SQuAD1.1 accuracies. Our 1.2 billion parameter question generator is used for generating questions.

5.2 Scaling Answer Generation

Answer generation is equally important in our data generation pipeline. Answer generation is the first

component of the pipeline and must be precise to avoid compounding errors. For answer generation we use an unconditional extractive BERT model that predicts start and end spans jointly over a given sentence. From each probability distribution we sample the entire nucleus ($p = 0.9$) or the top-5 spans, choosing whichever is smaller. We arrive at this implementation based on our ablation studies in section 6.3. In qualitative studies we found that our model consistently generates diverse and valid answers (see samples in Appendix A.7), however, they rarely overlap with the answers from the SQuAD dataset making automatic metrics that rely on n-gram overlap such as BLEU not a suitable metric for analysis. To test the quality of the selected answers, we generate questions from our 1.2 billion parameter question generator and finetune a question answering model on the synthesized questions without any filtration. In Table 6 we compare answer generation quality using our two trained models and the original BERT-Large model from (Devlin et al., 2018). We find that improvements [1] in pretraining dramatically improve answer generation quality by 1.4 EM and 0.3 F1 between BERT-Large and our 345 million parameter answer generation model. We find that increasing model scale further to 1.2 billion parameters improves answer generation quality F1 by 0.4 while EM only improves by 0.1. Although these represent improvements in question quality only achieved by newer models, answer generation seems to be a large bottleneck as we discuss in section 6.1.

5.3 Scaling Question Filtration

We use the 1.2 billion parameter question generator from section 5.1 to generate questions for filtration. As described in more detail in section 6.3 we overgenerate two questions for every answer. We then filter these questions with roundtrip filtration before finetuning a question answering model. In Table 7 we find that our 345 million parameter BERT model modestly outperforms the public BERT-Large model when using synthetic answers to generate questions while our 1.2 billion parameter BERT model further improves on this score by more than a whole point. In the previous section improvements to pretraining scale and tasks made a larger difference on answer generation than increasing model scale. However, here we see the opposite: improvements to pretraining tasks results only in a modest improvement to question filtration, while increasing model size results in much more substantive improvements. We hypothesize that this is due to the larger model's ability to correctly answer more questions, and therefore allow more valid and high quality samples through to the finetuning phase as indicated by the number of questions generated by the technique.

[1] Improvements include using sentence order prediction instead of next sentence prediction heads (Lan et al., 2019), whole word masking, masked ngram prediction instead of random masked prediction (Joshi et al., 2019), rearrangement of residual connection and layer norm layers (Shoeybi et al., 2019), and inclusion of data from RealNews (Zellers et al., 2019), WebText (Gokaslan and Cohen, 2019), and Common Crawl Stories (Trinh and Le, 2018)

Filter Model	# Questions	EM	F1
Synthetic Questions + Real Answers			
BERT-Large	45888	84.5	91.4
BERT-345M	34341	84.2	91.4
BERT-1.2B	**47772**	**85.6**	**92.4**
Synthetic Questions + Synthetic Answers			
BERT-Large	177712	85.5	91.9
BERT-345M	144322	85.9	92.5
BERT-1.2B	**184992**	**87.1**	**93.2**
Human Generated Data	42472	86.3	93.2

Table 7: Effect of pretraining and scale on question filtration. Synthetic questions and answers were both generated with 1.2 billion parameter models. Before finetuning, overgeneration and filtration were performed with the models ablated here.

Question Generator	# Questions	EM	F1
345M	**42414**	**80.7**	**88.6**
345M (no pretraining)	42408	42.7	51.4
345M (no stopwords)	42486	75.5	84.5
Human Generated Questions	42472	86.3	93.2

Table 8: Effect of question generator modeling choices. Questions are generated from ground truth answers without any filtration.

6 Modeling Choices

While developing our synthetic data generation pipeline we explored several modeling and algorithmic choices before scaling up the model size and data quantity used. We pursued three axis of investigation, ablating choices for each model component of our pipeline at a time.

6.1 Question Generation

To study question generation in isolation we used our 345 million parameter model to generate questions from ground truth SQuAD1.1 answers. The results of our analysis can be found in Table 8. We first investigated the use of pretrained models and found that pretraining our GPT-2 model was cru-

Answer Generator	# Questions	EM	F1
NER	132729	59.3	70.5
Independent Spans	83534	77.2	87.1
Joint Spans	**229297**	**79.1**	**87.9**
Paragraph-level Joint Spans	226672	77.3	86.9
Human Generated Answers	42472	83.7	91.1

Table 9: Effect of answer generator modeling choices. Model based answer generation is performed with BERT-345M and questions are generated using a 1.2B parameter model. No filtration is applied to the generated questions.

cial for achieving reasonable question generation. We then examined the effect of stopword filtration in our question generator. Stop word filtration is performed by sampling autoregressively from the model and discarding the sample if the sample does not end with a ": question" token before the end of text. This is similar to forcing a question to end in a question mark. Stop word filtration therefore doesn't filter long or short results, it helps prevent samples that are malformed questions and don't end properly (i.e. randomly trail off with a preposition). We found that this provided a substantial boost to EM and F1 scores of 5.2 and 4.1 respectively. The goal of employing this technique was to catch generations that ramble onwards without stopping, or produce end of text prematurely in the middle of a question. On manual inspection we found qualitatively that this technique helped when generating questions on text that featured heavy use of symbols and foreign language. In these cases the model struggled with out of distribution vocabulary, autoregressive sampling degenerated, and no stopword was produced.

6.2 Answer Generation

In our experiments we found answer generation to be a significant bottleneck in performance. In section 5.2 we found that scaling up model size allows us to close the gap between human and synthetic training performance. However, these scaling analysis were performed with our best model. In Table 9 we show that the choice of model is critical to closing the gap. Starting with a Named Entity Recognition (NER) model we find that it gets a dismal EM and F1 score. This is due to entities comprising only of $\sim 50\%$ of the answer distribution for SQuAD1.1 . It's necessary to use a learned model to model the diverse set of answers present SQuAD1.1 . We then tried to use the most common SQuAD1.1 model, which models

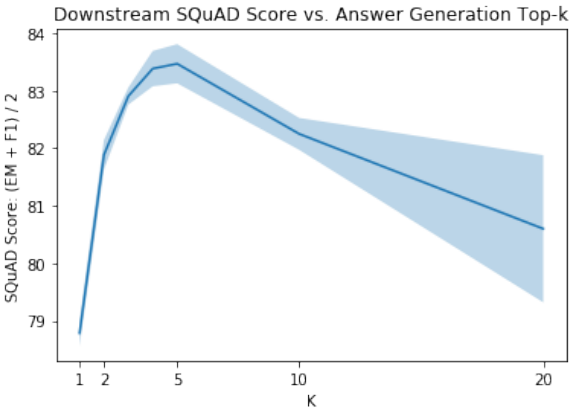

Figure 2: Effect of top-k answer generation on downstream SQuAD1.1 performance. For a particular value of k we sample all top-k candidate answers (within a nucleus of $p = 0.9$) from a sequence according to a 345M parameter answer generator.

the start and end of a span independently, to extract answers from individual sentences. This performed noticeably better, boosting our score to 77.2 EM, despite producing fewer answers than NER extraction. However, upon inspection we found that modeling the span independently resulted in sampling repetitive answers. We then tried using the answer generator from (Alberti et al., 2019a) which models the start and end of a span jointly as a conditional random field. This is the model we ended up choosing as it performed the best with an exact match score of 79.1. Lastly, we also considered jointly modeling answer spans over an entire paragraph instead of a single sentence. However, we found that it performed worse than independent span modeling over sentences.

When sampling our answer candidates we used all top-k answers comprising of the top-p ($p = 0.9$) nucleus of the distribution. We performed an ablation study to select k as we found that this had a noticeable impact on downstream accuracy. In Figure 2 we found that there was an optimal spot of $k = 5$ answers per sentence. When generating answers sampled from an entire paragraph we used $k = 24$ as we found that there were 4.86 sentences per paragraph on average. In general, answer generation proves to be a bottleneck in our question generation pipeline. The difficulty in answer generation is that not only must the answers be useful and well-formed, one must solve a one-to-many modeling problem to sample multiple answers from one passage. We believe that this might also be a contributing factor behind the poorer performance of the paragraph-level answer generation.

Filter Model	# Questions	345M QA		Large QA	
		EM	F1	EM	F1
Synthetic Questions + Real Answers					
None	42472	83.4	90.9	79.0	87.0
Roundtrip (RT)	24310	84.0	91.3	76.5	84.4
Overgenerate & RT	**47772**	**85.6**	**92.4**	**81.7**	**88.7**
Synthetic Questions + Synthetic Answers					
None	229297	79.1	87.9	78.2	86.8
Roundtrip (RT)	93866	86.3	92.7	84.1	90.5
Overgenerate & RT	**184992**	**87.1**	**93.2**	**85.2**	**91.5**
Human Generated Data	42472	86.3	93.2	82.4	89.7

Table 10: Effect of filtration modeling choices on questions generated from ground truth and synthetic answers. 1.2 billion parameter models are used for every stage of the generation pipeline. Questions from no filtration are used in the other experiments with a second set of questions generated in overgeneration experiments.

6.3 Question Filtration

Both question generation and answer generation sometimes produces poor answers. As we show in Table 10 generating synthetic data from synthetic answers without filtering deteriorates significantly, while roundtrip consistency combats this effect to perform 7.2 EM points better. However, we find that even on questions generated from ground truth answers roundtrip filtering throws away questions associated with perfectly good answers. Throwing away data significantly hurts BERT-Large whose pretrained features are not as robust as our BERT-345M model and require more finetuning data. To combat this we take an approach similar to overgeneration and reranking (Heilman and Smith, 2010a) where we generate two questions per answer and feed each into roundtrip filtration independently. We term this overgeneration and filtration. This helps avoid losing important answers in our synthesized training set. To perform overgeneration we sample one question with top-k ($k = 40$) sampling and one with top-p ($p = 0.9$) nucleus sampling. This leads approximately to a whole point of improvement for our model in both the case with and without ground truth answers.

7 Related Work

Early work using rule based question generation (Heilman and Smith, 2010b) proposed the idea of over-generating and re-ranking questions with regression models learned over handcrafted linguistic features. Du et al. (2017) used learned LSTM models on extractive question answering datasets such as SQuAD . These early works focused primarily on generating questions without explicit extracted answers in the text. Subramanian et al. (2017) proposed a two-stage neural model which added a model to estimate candidate answers and using the answers to generate questions. The current state of the art leverages transformer based language modeling including (Alberti et al., 2019a; Dong et al., 2019; Zhu et al., 2019; Klein and Nabi, 2019).

(Alberti et al., 2019a) uses seq2seq models to generate questions, and then enforce answer consistency on synthetic questions to filter out poorly generated questions in a technique called roundtrip consistency. (Dong et al., 2019) uses a unified transformer rather than a seq2seq model to generate QA data in conjunction with roundtrip consistency. (Zhu et al., 2019) learn a model that generates unanswerable questions from an answerable example.

The process of generating answers for answer-aware question generation in recent literature has primarily leveraged cloze fill-in-the-blank passages to highlight an answer in a given context. Some work uses NER or linguistic parsers to select passages for cloze translation as in (Lewis et al., 2019; Dhingra et al., 2018). These methods are only able to generate answers for a subset of questions as SQuAD1.1 is only made up of 52% Named Entity Answers. More recent work such as (Alberti et al., 2019a; Dong et al., 2019) use model based approaches to match the answer distribution of QA datasets and extract more complex answers.

To improve the quality of synthetic data generation and downstream QA models, improving language model quality is crucial. In addition to pretraining task innovation, BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019) and ALBERT (Lan et al., 2019) have showed that increasing the size of available pretraining data directly improves downstream discriminative task performance. T5 (Raffel et al., 2019), GPT-2 (Radford et al., 2019), CTRL (Keskar et al., 2019), Megatron-LM (Shoeybi et al., 2019), and (Puri and Catanzaro, 2019) have shown that increasing language model scale improves the quality, coherency, and correctness of text generation. The models used in (Raffel et al., 2019; Keskar et al., 2019; Radford et al., 2019; Puri and Catanzaro, 2019; Boyd et al., 2020) also demonstrate that larger models allow for better control in conditional language generation.

SQuAD style extractive question answering is not the only form of question answering. There are many other datasets covering a wide range of QA such as multihop (Yang et al., 2018; Welbl et al.,

2018; Talmor and Berant, 2018), Yes-No question (Clark et al., 2019), trivia questions (Joshi et al., 2017; Dunn et al., 2017), analytical questions (Dua et al., 2019), conversational and generative QAs (Reddy et al., 2019), unanswerable questions (Rajpurkar et al., 2018; Alberti et al., 2019b), and large multitask question answering datasets (Talmor and Berant, 2019). While these are outside the scope of the current work, the insights developed improving quality for extractive SQuAD questions will aid question generation in other domains.

8 Conclusions and Future Work

We build upon existing work in large scale language modeling and question generation to push the quality of synthetic question generation. With our best models, we generate large question answering datasets from unlabeled Wikipedia documents and finetune a 345 million parameter BERT-style model achieving 88.4 EM score. Finetuning the resulting model on real SQuAD1.1 data further boosts the EM score to 89.4. This amounts to a 1.7 point improvement over our fully supervised baseline. Finally, we generate synthetic text from a Wikipedia-finetuned GPT-2 model, generate answer candidates and synthetic questions based on those answers, and then train a BERT-Large model and achieve similar question answering accuracy. Doing so required us to scale model size for our answer generators, question generators, and filtration models. We hope that better synthetic questions will enable new breakthroughs in question answering systems and related natural language tasks.

Of particular interest for future work is handling low-resource question answering domains. For such a regime, one needs to analyze the effect of domain transfer and bootstrapping from a very small human labelled dataset. Extension of this work to unanswerable and boolean questions is also a future work direction. More generally application of this work to multi dataset question generation with datasets such as MultiQA (Talmor and Berant, 2019) is a promising avenue for future work.

References

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019a. Synthetic qa corpora generation with roundtrip consistency. *arXiv preprint arXiv:1906.05416*.

Chris Alberti, Kenton Lee, and Michael Collins. 2019b. A bert baseline for the natural questions. *arXiv preprint arXiv:1901.08634*.

Alex Boyd, Raul Puri, Mohammad Shoeybi, Mostofa Patwary, and Bryan Catanzaro. 2020. Large scale multi-actor generative dialog modeling. *arXiv preprint arXiv:2005.06114*.

Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. 2009. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Bhuwan Dhingra, Danish Pruthi, and Dheeraj Rajagopal. 2018. Simple and effective semi-supervised question answering. *arXiv preprint arXiv:1804.00720*.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *arXiv preprint arXiv:1905.03197*.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. *arXiv preprint arXiv:1705.00106*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *arXiv preprint arXiv:1903.00161*.

Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*.

Aaron Gokaslan and Vanya Cohen. 2019. Openwebtext corpus.

Michael Heilman and Noah A Smith. 2010a. Good question! statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 609–617. Association for Computational Linguistics.

Michael Heilman and Noah A Smith. 2010b. Good question! statistical ranking for question generation. In *Human Language Technologies: The 2010*

Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 609–617. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2019. Spanbert: Improving pre-training by representing and predicting spans. *CoRR*, abs/1907.10529.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. 2014. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pages 3581–3589.

Tassilo Klein and Moin Nabi. 2019. Learning to answer by learning to ask: Getting the best of gpt-2 and bert worlds. *arXiv preprint arXiv:1911.02365*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel. 2019. Unsupervised question answering by cloze translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4896–4910, Florence, Italy. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2017. Mixed precision training. *arXiv preprint arXiv:1710.03740*.

Raul Puri and Bryan Catanzaro. 2019. Zero-shot text classification with generative language models. *arXiv preprint arXiv:1912.10165*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using gpu model parallelism. *arXiv preprint arXiv:1909.08053*.

Sandeep Subramanian, Tong Wang, Xingdi Yuan, and Adam Trischler. 2017. Neural models for key phrase detection and question generation. *CoRR*, abs/1706.04560.

Alon Talmor and Jonathan Berant. 2018. Repartitioning of the complexwebquestions dataset. *arXiv preprint arXiv:1807.09623*.

Alon Talmor and Jonathan Berant. 2019. MultiQA: An empirical investigation of generalization and transfer in reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4911–4921, Florence, Italy. Association for Computational Linguistics.

Trieu H Trinh and Quoc V Le. 2018. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing.

Qizhe Xie, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. 2019. Self-training with noisy student improves imagenet classification. *arXiv preprint arXiv:1911.04252*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *Advances in Neural Information Processing Systems*, pages 9051–9062.

Haichao Zhu, Li Dong, Furu Wei, Wenhui Wang, Bing Qin, and Ting Liu. 2019. Learning to ask unanswerable questions for machine reading comprehension. *arXiv preprint arXiv:1906.06045*.

Xiaojin Zhu and Andrew B Goldberg. 2009. Introduction to semi-supervised learning. *Synthesis lectures on artificial intelligence and machine learning*, 3(1):1–130.

Xiaojin Jerry Zhu. 2005. Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences.

A Appendices

A.1 Question Generation Pipeline

Algorithm 1 Pipeline for generating and evaluating synthetic data.

1. Sample answer candidates from paragraphs using a BERT model.
2. Generate questions from answer candidates and paragraphs using a GPT-2 model.
3. Apply a BERT roundtrip consistency model to filter generated question answer pairs.
4. Train a BERT QA model using filtered synthetic questions and evaluate on development set.

A.2 Question Generation Input Representation

We develop a conditional question generation model, $p(q|a, c)$ using a pretrained GPT-2 model. As input to our model, we concatenate context tokens, answer tokens, and question tokens into a single sequence, separated by end of sequence tokens. We use three segment type embeddings to help the GPT-2 decoder model distinguish between different parts of the input. We also use answer segment type embeddings to highlight the presence of the answer span in the provided context tokens.

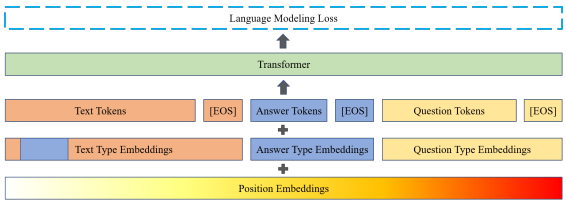

Figure 3: Question Generation input representation and language modeling loss. Answer type embeddings highlight the answer's presence in the text.

A.3 GPT-2 Pretraining Details

The GPT-2 models (Radford et al., 2019) used for question generation were each pretrained on the 174GB corpora used in Megatron-LM: Wikipedia (Devlin et al., 2018), OpenWebText (Gokaslan and Cohen, 2019), RealNews (Zellers et al., 2019), and CC-Stories (Trinh and Le, 2018). Unless otherwise noted, our GPT-2 models were trained at a batch size of 512 for 300k iterations with 3k iterations of warmup, Adamw (Loshchilov and Hutter, 2018) for optimization, a learning rate of 1.5e-4 decaying linearly to 1e-5, weight decay of 0.01, global gradient norm clipping of 1.0, and a normal initialization of $\theta \sim \mathcal{N}(0, 0.02)$.

A.4 BERT Pretraining Details

To train our BERT models we relied on a pretraining regime similar to ALBERT. We used a n-gram masked language modeling task in conjunction with a sentence order prediction task. Unlike ALBERT we did not utilize weight sharing and we used a GPT-2 style ordering of residual connections and layer normalization. We found this greatly improved stability and allowed us to train significantly larger BERT models than prior work (Lan et al., 2019) without encountering training instabilities and overfitting. We trained our BERT models with the same hyperparameters as GPT-2 except using learning rate of 1e-4 and a batch size of 1024 over 2 million iterations with 10k iterations of warmup.

A.5 Training and Evaluation Data Flow

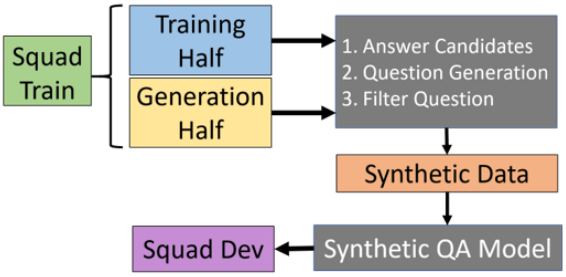

Figure 4: Data flow for training and evaluating question generation pipeline.

A.6 Training Infrastructure

All our models were trained with mixed precision training (Micikevicius et al., 2017) on NVIDIA V100 GPUs. Pretraining took place on anywhere from 4 to 32 DGX-2H servers for our largest models. Finetuning only required one DGX-1V, except in the case of finetuning the 8.3B parameter question generator which required eight DGX-1Vs.

A.7 Samples Generated from Wikipedia Documents

Below are synthetic question and answering pairs synthesized from real Wikipedia documents. Question and answer generation and filtration were performed by 1.2 billion parameter models finetuned over the entire SQuAD1.1 dataset. Generated answer spans are bolded in the text.

Question: What indicates there must be data deletion early on in the visual pathway?

Context: Evidence suggests that our visual processing system engages in bottom-up selection. For example, **inattentional blindness** suggests that there must be data deletion early on in the visual pathway. This bottom-up approach allows us to respond to unexpected and salient events more quickly and is often directed by attentional selection. This also gives our visual system the property of being goal-directed. Many have suggested that the visual system is able to work efficiently by breaking images down into distinct components. Additionally, it has been argued that the visual system takes advantage of redundancies in inputs in order to transmit as much information as possible while using the fewest resources.

Question: What type of antibiotic is cefalotin?

Context: Cefalotin (INN) or cephalothin (USAN) is a **first-generation cephalosporin** antibiotic. It was the first cephalosporin marketed (1964) and continues to be widely used. It is an intravenously administered agent with a similar antimicrobial spectrum to cefazolin and the oral agent cefalexin. Cefalotin sodium is marketed as Keflin (Lilly) and under other trade names.

Question: What did "Wanted Dead or Alive" rank on the Billboard Hot 100?

Context: "Wanted Dead or Alive" is a song by American rock band Bon Jovi. It is from their 1986 album "Slippery When Wet". The song was written by Jon Bon Jovi and Richie Sambora and was released in 1987 as the album's third single. During a February 20, 2008 encore performance in Detroit, Jon Bon Jovi told the crowd about running into Bob Seger at a Pistons game. As he introduced his song "Wanted Dead or Alive", he said it was inspired by Seger's "Turn the Page" hit and called the song the band's anthem. The song peaked at **#7** on the "Billboard" Hot 100 chart and #13 on the Mainstream Rock Tracks chart, making it the third single from the album to reach the Top 10 of the Hot 100. As a result, "Slippery When Wet" was the first hard rock/glam metal album to have 3 top 10 hits on the "Billboard" Hot 100.

Question: Who played the role of Othello in the scene?

Context: The book begins when Kostya and his fellow students are waiting for their first lesson with the Director. They are excited and nervous at the prospect of meeting, and are surprised when he tells them that their first exercise is to put on a few scenes from a play. Kostya and two of his friends perform scenes from "Othello", with **Kostya** taking the leading role. Afterwards the Director tells them their mistakes.

Question: Who broke the Phantom's mind?

Context: In the final episode of the game, it is revealed that Fulbright is in fact deceased, and that the Fulbright seen throughout the game is an international spy known as the Phantom posing as him, as well as the one behind most of the game's major events. Seven years prior to the game's events, the Phantom was the catalyst of the UR-1 Incident, having murdered Metis Cykes, Athena's mother, sabotaged the HAT-1 shuttle, and leaving Simon Blackquill to take the fall for the crime after seemingly incriminating evidence was found to point to Simon as the only suspect. Simon willingly allowed himself to be imprisoned in order to protect Athena and to draw the Phantom out, but Athena suffered severe trauma from the ordeal, having believed for 7 years that she had actually murdered her mother, when in fact she stabbed the Phantom in the hand in self-defense. In the present day, the Phantom attempted to finish their case, murdering Clay Terran and bombing both the HAT-2 shuttle and a courtroom in a desperate attempt to destroy incriminating evidence from the UR-1 incident. The Phantom possesses a unique psychological makeup, showing very little, if any, emotion of any sort, nor any fear. The Phantom also has no sense of self, claiming they do not know what their original gender, face, nationality, or identity even was in the beginning; having taken on so many disguises and identities, the Phantom is an endless void. However, **Phoenix, Apollo, and Athena** eventually managed to break the emotionless Phantom severely in court, causing them to suffer a severe identity crisis, moments before an unseen sniper rifle takes the Phantom's life.

Question: Who was Miss United Kingdom in 1997?

Context: **Vicki-Lee Walberg** (born 11 October 1975) is a model who was Miss United Kingdom in 1997, and made the top 10 at the Miss World 1997 pageant. She was the last title holder to advance to the semifinal of the contest. Walberg later went on to work in television and was a 'Dolly Dealer' in Bruce Forsyth's Play Your Cards Right on ITV during its 2002 revival.

Question: What was the final score for the Tottenham home match against Newcastle United?

Context: He scored his first Premier League hattrick in a 4-0 away win on Boxing Day against Aston Villa. On 5 January 2013, Bale scored in the FA Cup third round fixture against Coventry City as well as assisting Clint Dempsey on both of his goals in a 3-0 win. On 30 January, Bale scored a magnificent solo effort in the 1-1 draw with Norwich City. Bale then scored against West Bromwich Albion in a 1-0 away win on 3 February. Bale then took his goal tally of the season to 15 goals with a brace against Newcastle United in a match which Spurs won **2-1**. This took Spurs into third place, and strengthened their Champions League ambitions.

Question: Who was arrested along with Ernst Sekunna?

Context: The arrests started in March 1917, with **Chandra Kanta Chakraverty** "a thin-faced, falsetto-voiced Hindu, a native of Bengal, and a speaker of many languages", and the German, Ernst Sekunna, being arrested on charges of conspiracy. Most of the others were arrested on April 8, including Franz Bopp, the German Consul General for San Francisco, E. H. von Schack, Deus Dekker and Wilhelm von Brincken. The Indian Nationalists were accused of taking "advantage of American neutrality to plot on American soil against the allies" at "the expense of the laws and hospitality of the United States". The two men had also taken out trade names to do business as "The Oriental Society", "The Oriental Kitchen",and the "Oriental Review", and purchased of land in an isolated part of New York State.

Question: What protected the hulls of the Chiyoda?

Context: "Chiyoda" was a belted cruiser based on a much scaled-down version of the Royal Navy's. The hull was made of 84 watertight compartments, protected with **Harvey armor**. Originally designed to carry 12.6 inch Canet guns, the plan was abandoned due to excessive top weight. Instead, the design was changed so that her main battery consisted of ten QF 4.7 inch /40 naval guns in single mounts, mounted one each in the bow and stern, and four on each side in sponsons. The use of the Elwick quick-firing technology resulted in an increase in the rate of fire by sixfold over previous cruiser designs. Her secondary battery consisted of 14 QF 3 pounder Hotchkiss and three 11-mm, 10-barrel Nordenfelt guns. She was also equipped with three Whitehead torpedo tubes mounted on the main deck. As was standard practice at the time, the prow was reinforced for ramming.

A.8 Samples Generated from GPT-2 Documents

Below are synthetic question and answering pairs synthesized from fake Wikipedia documents sampled unconditionally from an 8.3B GPT-2 model. Question and answer generation and filtration were performed by 1.2 billion parameter models finetuned over the entire SQuAD1.1 dataset. Generated answer spans are bolded in the text.

Question: What is a clique in a DAG?

Context: The main purpose of the conjecture is to quantify the perfect matchings of the vertices of a graph, in a way that can be related to the number of cliques. A perfect match of two vertices means that if the graph is "cut along the line segment connecting these two vertices", then the pair of vertices forms an optimal matching. A clique is **a small subgraph** that contains all but one pair of vertices in the graph and so these perfect matchings form an "array" of cliques with the same size as the original graph, and thus can be described by the same number of cliques.

Question: What property is the difference between Bis(diphenylphosphino)methane and benz(diphenylphosphino)methane?

Context: Bis(diphenylphosphino)methane has been found to be a **sterically hindered** isomer of benz(diphenylphosphino)methane (CHPH) and therefore it has an oxidation number of 1.

Question: When did Henry II invade Normandy?

Context: During the reign of Louis VII of France, Eleanor was awarded by her husband the County of Anjou. In **1157**, Henry II of England invaded Normandy to take possession of that duchy, defeating Louis's troops in the Battle of Brémule. Louis's grandson and heir, William Adelin, left Anjou for his home in the south of France, where he was crowned at Toulouse on 24 April 1158.

Question: What does Dick Grayson use as his name?

Context: Meanwhile, on his return to the fifth dimension, the leader of the Faceless Ones is killed in the ensuing battle and his daughter is captured. She asks the Faceless Ones for an escape plan and is told that she must first find her father's "labyrinth". The Faceless Ones then freeze her in time and her journey begins. Batman, now imprisoned in Arkham Asylum is visited by Dick Grayson in his new identity of **Nightwing**. Nightwing informs him that he has broken his parole and is now hunting him. Batman is shocked to discover that Nightwing has come to Arkham because of a deal he made with the Riddler to help him track down some of Batman's other enemies. Batman is sent by the Joker to assist Nightwing, Deadman, Deathstroke, and Lex Luthor, in tracking down Deadman's apparent killer. Batman eventually learns that the person who really killed Deadman was his fellow Justice League member, Zauriel. Zauriel is revealed to be a deeply troubled angel-like figure who blames the world for the suffering and death that he has witnessed as he has been with Batman since the death of Damian Wayne. The story arc culminated in a battle in the House of Mystery between the Spectre and Zauriel in an attempt to bring the demon back to Hell. In the end, Batman accepts Zauriel's invitation to follow him back to the fifth dimension to spare him any further pain and humiliation.

Question: Who do Jim, Pam, Dwight, Oscar, and Jim's father, Henry attend the wedding reception for?

Context: At the photo shoot, Andy Bernard (Ed Helms) and Erin Hannon (Ellie Kemper) go on a fake zombie honeymoon in the office, having an intimate moment that is interrupted when they encounter a horde of the undead. Michael and Dwight then stop the zombies from approaching Andy and Erin and create a barricade. The horde is scared off, but the building must be sealed off because the zombies have damaged the power generator, resulting in a total loss of power. After the power returns, Jim, Pam, Dwight, Oscar, and Jim's father, Henry (Brock Peters), begin gathering their families and friends to go to **Erin and Andy**'s wedding reception in the Scranton branch's conference room.

Question: What was the title of 50 Cent's first album?

Context: "I Got Mine" is a song by American rapper 50 Cent from his debut studio album "**Get Rich or Die Tryin'**" (2003). The song features a guest appearance from fellow New York City rapper Nas, who was also featured on the previous single from "Get Rich or Die Tryin'", "Hate Me Now".

Question: What happens to a star when it bursts into a thermonuclear reaction?

Context: When the star explodes, the material is compressed to several hundred times its original size, igniting a thermonuclear reaction. This reaction causes the star to **explode outward**. The first stage of the supernova explosion is not yet far enough away to reach this red giant stage, so the star is engulfed in a supernova explosion. As the star is heated up by the supernova explosion, the outer layers of the star collapse. The compression that occurred when the shock wave reached the star's surface begins to occur at the point where the star's surface meets its core. This core-surface compression heats up and accelerates the remaining core material, producing a shock wave that expands out from the core.

Question: What style was used in This Wonderful Life's production?

Context: In 2009, Maine College of Art (main campus) presented "This Wonderful Life" as the kick-off production to their 2009/2010 theater season. Director Todd Ziegler created a minimalist approach to the production, relying mostly on the basic premise and atmosphere of the film to create a world. The Main Stage theater was transformed into an **Art Deco**-esque set with minimal set pieces, provided by Redlich + Feuer Design. This setting was contrasted by the minimalistic approach to lighting, provided by Brian Claypool, that lent the production a somber tone. In keeping with the Art Deco styling, costume design and construction was done entirely by students of the Department of Theater and Dance. The music was provided by the joint choirs of the college and the Maine All State Honor Choir.

Question: Which road through the Texas scrublands is a controlled access road?

Context: The western terminus of US 83 is located on the southeast corner of the Texas-New Mexico border at the Van Horn, Texas-Van Horn, Texas city limit line. From the border the highway follows Texas State Highway 116, which crosses US 87 in Van Horn and overlaps US 70. US 83 then crosses US 87 again near Marfa, intersecting US 87 Business and Texas State Highway 292. US 83 continues west from Marfa along Highway 290, a route now called the Trans-Pecos Highway. While **US 290** is a controlled-access road, it still has a large number of at-grade intersections, due to the rugged terrain. Between Marfa and Valentine, US 83 travels through the Texas scrubland of the Big Bend.

Question: Who was in charge of the SOE during World War II?

Context: By 1939, the Republican cause was being supported by both the Soviet Union and the Third Reich. The SOE, led by **Colonel Hugh Sinclair**, had been active in the country since 1934, delivering weapons and propaganda material to the Republicans via agents such as future French Resistance leader Francois de La Rocque. This work came to an abrupt end in April 1939, when the Germans invaded the country. Sinclair organised a flight to France, but only about a dozen agents and journalists escaped from the country.