Towards building a Robust *Industry-scale* Question Answering System

Rishav Chakravarti[*], **Anthony Ferritto, Bhavani Iyer, Lin Pan**
Radu Florian, Salim Roukos, Avirup Sil[†]
IBM Research AI
Yorktown Heights, NY
{panl, roukos, raduf, murdockj, bsiyer, avi}@us.ibm.com
aferritto@ibm.com
chakrris@amazon.com

Abstract

Industry-scale NLP systems necessitate two features. 1. Robustness: "zero-shot transfer learning" (ZSTL) performance has to be commendable and 2. Efficiency: systems have to train efficiently and respond instantaneously. In this paper, we introduce the development of a production model called GAAMA (Go Ahead Ask Me Anything) which possess the above two characteristics. For robustness, it trains on the recently introduced Natural Questions (NQ) dataset. NQ poses additional challenges over older datasets like SQuAD: (a) QA systems need to read and comprehend an entire Wikipedia article rather than a small passage, and (b) NQ does not suffer from observation bias during construction, resulting in less lexical overlap between the question and the article. GAAMA consists of Attention-over-Attention, diversity among attention heads, hierarchical transfer learning, and synthetic data augmentation while being computationally inexpensive. Building on top of the powerful BERT$_{QA}$ model, GAAMA provides a $\sim$2.0% absolute boost in F_1 over the industry-scale state-of-the-art (SOTA) system on NQ. Further, we show that GAAMA transfers zero-shot to unseen real life and important domains as it yields respectable performance on two benchmarks: the BioASQ and the newly introduced CovidQA datasets.

1 Introduction

A relatively new task in open domain question answering (QA) is machine reading comprehension (MRC), which aims to read and comprehend a given text and then answer questions based on it. Recent work on transfer learning, from large pre-trained language models like BERT (Devlin et al., 2019) and XLNet (Yang et al., 2019) has practically solved SQuAD (Rajpurkar et al., 2016; Rajpurkar et al., 2018), the most widely used MRC benchmark. This necessitates harder QA benchmarks for the field to advance. Additionally, SQuAD and other existing datasets like NarrativeQA (Kočiskỳ et al., 2018) and HotpotQA (Yang et al., 2018) suffer from observation bias: annotators had read the passages before creating their questions.

In industry research, there is an urgent demand to build a usable MRC QA system that not only provides very good performance on academic benchmarks but also real life industry applications (Tang et al., 2020) in a ZSTL environment. In this paper, to build such a system, we first focus on Natural Questions (NQ) (Kwiatkowski et al., 2019): a MRC benchmark dataset over Wikipedia articles where questions (see Figure 1) were sampled from Google search logs. This key difference from past datasets eliminates annotator observation bias. Also, NQ requires systems to extract both a short (SA, one or more entities) and a long answer (LA, typically a paragraph that contains the short answer when both exist). The dataset shows human upper bounds of 76% and 87% on the short and long answer selection tasks respectively (for a "super-annotator" composed of 5 human annotators). The authors show that systems designed for past datasets perform poorly on NQ.

[*]Work completed while at IBM, author is currently at Amazon
[†]Corresponding author.

Proceedings of the 28th International Conference on Computational Linguistics: Industry Track, pages 90–101
Barcelona, Spain (Online), December 12, 2020

Example 1
Question: when did marley die in a christmas carol?
Wikipedia Page: Jacob Marley
Gold Long Answer: In A Christmas Carol, Marley is the first character mentioned in the first line of the story. Jacob Marley is said to have died seven years earlier on Christmas Eve (as the setting is Christmas Eve 1843, this would have made the date of his passing **December 24, 1836**)...
Gold Short Answer: December 24, 1836
BERT$_{QA}$: seven years earlier
Example 2
Question: who developed the concept of total quality management
Wikipedia Page: Total Quality Management
Gold Long Answer: The exact origin of the term "total quality management" is uncertain. It is almost certainly inspired by Armand V. Feigenbaum's multi-edition book Total Quality Control...
Gold Short Answer: *NULL*
BERT$_{QA}$: Armand V. Feigenbaum

Figure 1: Examples of questions in the NQ dataset. Example 1 contains the short answer in the long answer whereas Example 2 has none.

We propose GAAMA that possesses several MRC technologies that are necessary to perform well on NQ and achieve significant boosts over another industry setting competitor system (Alberti et al., 2019a) pre-trained on a large language model (LM) and then over millions of synthetic examples. Specifically, GAAMA builds on top of a large pre-trained LM and focusses on two broad dimensions:

1. **Improved Attention:** With the reduction of observation bias in NQ, we find a distinct lack of lexical and grammatical alignment between answer contexts and the questions. For example, here is a question to identify the date of an event from the SQuAD 2.0 dataset: *According to business journalist Kimberly Amadeo, when did the first signs of decline in real estate occur?* This question can be aligned almost perfectly with the text in the answering Wikipedia paragraph in order to extract the year *2006*: *Business journalist Kimberly Amadeo reports: "The first signs of decline in residential real estate occurred in 2006."* In contrast, as shown in Example 1 from Figure 1, a question from NQ to identify the date of *marley's death* requires parsing through a number of related sub clauses to extract the answer *December 24, 1836* from the context.

   This need for improved alignment leads us to explore two additional attention mechanisms.

   - **Attention-over-Attention (AoA)** (Cui et al., 2017): on top of BERT's existing layer stack, we introduce a *two-headed* AoA layer which combines *query-to-document* and *document-to-query* attention.
   - **Attention Diversity (AD)** Motivated by (Li et al., 2018), we explore a mechanism that maximizes diversity among BERT attention heads. Intuitively, we want different attention heads to capture information from different semantic subspaces, which BERT currently does not enforce. Finally, we experiment combining the two strategies, yielding a gain of ~1.5% for both short and long answers.

2. **Data Augmentation:** Given the data hungry nature of BERT-based models, we explore three strategies for data augmentation (DA). **Crowd-sourced DA** introduces human annotated Q&A pairs from prior MRC datasets. **Synthetic DA** introduces large amounts of machine generated QA pairs, inspired by the prior successes of (Alberti et al., 2019a; Dong et al., 2019). Unlike previous work, which predominantly relied on computationally expensive beam search decoding, we apply fast and diversity-promoting *nucleus* sampling (Holtzman et al., 2019) to generate 4M questions from a transformer-based question generator (Sultan et al., 2020). **Adversarial DA** performs a novel sentence-order-shuffling to perturb the native NQ data so as to tackle the inherent positional bias in Wikipedia-based MRC as shown by (Min et al., 2019; Kwiatkowski et al., 2019).

We find that, contrary to previous industry research SOTA (Alberti et al., 2019a) on NQ, it is not necessary to perform large scale synthetic DA. Instead we achieve better results with a well aligned Pre-Training (PT, a gain of 1.3–1.6%).

Most QA applications in an industry involve multiples domains e.g. Amazon Kendra[1] for Enterprise Search, Google Search, and IBM Watson Assistant[2] for Customer Service. Hence, there exists a need to develop *one* robust QA system that would work with ZSTL on a plethora of domains. Of course, one could futher fine-tune the system on the new domain to achieve better performance. However, the process is rather expensive as it demands manual human annotation which in real world applications is very scarce (Castelli et al., 2020). Hence, we explore GAAMA's ZSTL effectiveness on two publicly available benchmark bio-medical datasets: BioASQ (Tsatsaronis et al., 2015) and the newly introduced CoVIDQA (Tang et al., 2020). The former is an annual shared task for QA over biomedical documents involving factoid questions. The latter is built on top of the CORD-19 corpus (Wang et al., 2020) consisting of questions asked by humans about the Covid-19 disease. The COVID-19 pandemic has caused an abundance of research to be published on a daily basis. Providing the capability to ask questions on research is vital for ensuring that important and recent information is not overlooked and available to everyone. GAAMA consistently delivers competitive performance when compared to baselines either trained on the target domain or zero-shot transferred to the target.

Overall, our contributions can be summarized as follows: **1.** We propose a novel system that investigates several improved attention and enhanced data augmentation strategies, **2.** Outperforms the previous industry-scale QA system on NQ, **3.** Provides ZSTL capabilities on two unseen domains and **4.** Achieves competitive performance compared to the respective corresponding baselines.

2 Related Work

Most recent MRC systems either achieve SOTA by adding additional components on top of BERT (Devlin et al., 2019) such as syntax (Zhang et al., 2019) or perform attention fusion (Wang et al., 2018) without using BERT. However, we argue that additional attention mechanisms should be explored on top of BERT such as computing additional cross-attention between the question and the passage and maximizing the diversity among different attention heads in BERT. Our work is also generic enough to be applied on recently introduced transformer based language models such as ALBERT (Lan et al., 2019) and REFORMER (Kitaev et al., 2020).

Another common technique is DA (Zhang and Bansal, 2019) by artificially generating more questions to enhance the training data or in a MTL setup (Yatskar, 2018; Dhingra et al., 2018; Zhou et al., 2019). (Alberti et al., 2019a; Alberti et al., 2019b) combine models of question generation with answer extraction and filter results to ensure round-trip consistency to get the SOTA on NQ. Contrary to this, we explore several strategies for DA that either involve *diverse* question generation from a dynamic nucleus (Holtzman et al., 2019) of the probability distribution over question tokens or shuffling the existing dataset to produce adversarial examples.

Recently (Lee et al., 2019; Min et al., 2019) focus on "open" NQ, a modified version of the full NQ dataset for document retrieval QA that discards unanswerable questions. Contrary to that, we specifically focus on the full NQ dataset and believe there is room for improvement from a MRC research standpoint.

3 Model Architecture

In this section, we first describe BERT_{QA}, GAAMA's underlying QA model, and two additional attention layers on top of it. Figure 2 shows our overall model architecture with details explained below.

3.1 Underlying QA model: BERT_{QA}

Given a token sequence $\mathbf{X} = [x_1, x_2, \ldots, x_T]$: BERT, a deep Transformer (Vaswani et al., 2017) network, outputs a sequence of contextualized token representations $\mathbf{H}^L = [\mathbf{h}_1^L, \mathbf{h}_2^L, \ldots, \mathbf{h}_T^L]$.

$$\mathbf{h}_1^L, \ldots, \mathbf{h}_T^L = BERT(x_1, \ldots, x_T)$$

BERT_{QA} adds three dense layers followed by a *softmax* on top of BERT for answer extraction: $\boldsymbol{\ell}_b = softmax(\mathbf{W}_1\mathbf{H}^L)$, $\boldsymbol{\ell}_e = softmax(\mathbf{W}_2\mathbf{H}^L)$, and $\boldsymbol{\ell}_a = softmax(\mathbf{W}_3\mathbf{h}_{[CLS]}^L)$ — where $\mathbf{W}_1$, $\mathbf{W}_2 \in$

[1] https://www.onixnet.com/amazon-kendra

[2] www.ibm.com/watson/assistant

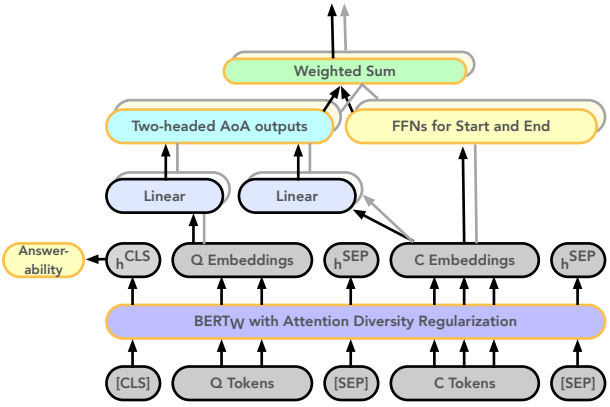

Figure 2: Overall Model Architecture

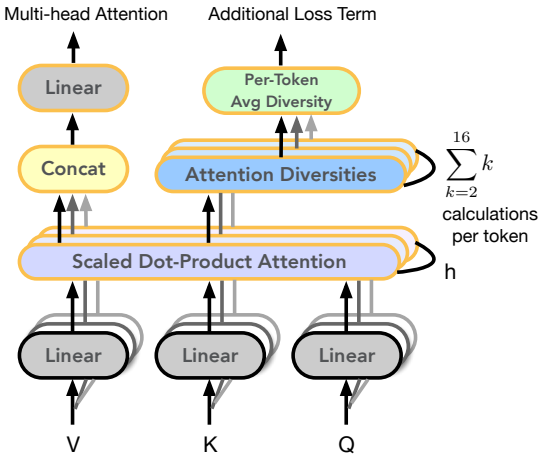

Figure 3: Our Attention Diversity Mechanism

$\mathbb{R}^{1\times1024}$, $\mathbf{W}_3 \in \mathbb{R}^{5\times1024}$, $\mathbf{H}^L \in \mathbb{R}^{N\times1024}$, and $\mathbf{h}^L_{[CLS]} \in \mathbb{R}^{1024}$. $\boldsymbol{\ell}^t_b$ and $\boldsymbol{\ell}^t_e$ denote the probability of the t^{th} token in the sequence being the answer beginning and end, respectively. These three layers are trained during the finetuning stage. The NQ task requires not only a prediction for short answer beginning/end offsets, but also a (containing) longer span of text that provides the necessary context for that short answer. Inspired by prior work from (Alberti et al., 2019b), we only optimize for short answer spans and then identify the bounds of the containing HTML span as the long answer prediction[3]. We use the hidden state of the `[CLS]` token (Devlin et al., 2019) to classify the answer type $\in [short, long, yes, no, null]$, so $\boldsymbol{\ell}^y_a$ denotes the probability of the y^{th} answer type being correct. Our loss function is the averaged cross entropy on the two answer pointers and the answer type classifier:

$$\mathcal{L}_{NQ} = -\frac{1}{3}\left(\sum_{t=1}^{T}(\mathbb{1}(\mathbf{b}_t)\log\boldsymbol{\ell}^t_b + \mathbb{1}(\mathbf{e}_t)\log\boldsymbol{\ell}^t_e) + \sum_{y=1}^{Y}\mathbb{1}(\mathbf{a}_y)\log\boldsymbol{\ell}^y_a\right)$$

where $\mathbb{1}(\mathbf{b})$ and $\mathbb{1}(\mathbf{e})$ are one-hot vectors for the ground-truth beginning and end positions, and $\mathbb{1}(\mathbf{a})$ for the ground-truth answer type. During decoding, the span over *argmax* of $\boldsymbol{\ell}_b$ and *argmax* of $\boldsymbol{\ell}_e$ is picked as the predicted short answer.

3.2 Attention Strategies

In this section, we outline our investigation of the attention mechanisms on top of the above BERT_{QA} model. Our main question: BERT already computes self-attention over the question and the passage in several layers—can we improve on top that?

3.2.1 Attention-over-Attention (AoA)

Our first approach is AoA: originally designed (Cui et al., 2017) for cloze-style question answering, where a phrase in a short passage of text is removed in forming a question. We seek to explore whether AoA helps in a more traditional MRC setting.

Let $\mathbf{Q}$ be a sequence of question tokens $[\mathbf{q}_1, \ldots, \mathbf{q}_m]$, and $\mathbf{C}$ a sequence of context tokens $[\mathbf{c}_1, \ldots, \mathbf{c}_n]$. AoA first computes an attention matrix:

$$\mathbf{M} = \mathbf{C}\mathbf{Q}^T, \tag{1}$$

where $\mathbf{C} \in \mathbb{R}^{n\times h}$, $\mathbf{Q} \in \mathbb{R}^{m\times h}$, and $\mathbf{M} \in \mathbb{R}^{n\times m}$. In our case, the hidden dimension is $h = 1024$. Next, it separately performs on $\mathbf{M}$ a column-wise *softmax* $\alpha = softmax(\mathbf{M}^T)$ and a row-wise *softmax* $\beta = softmax(\mathbf{M})$. Each row i of matrix α represents the document-level attention regarding $\mathbf{q}_i$ (*query-to-document* attention), and each row j of matrix β represents the query-level attention regarding $\mathbf{c}_j$

[3]The candidate long answer HTML spans are provided as part of the preprocessed data for NQ.

(*document-to-query* attention). To combine the two attentions, β is first row-wise averaged:

$$\beta = \frac{1}{n}\sum_{j=1}^{n}\beta_j \tag{2}$$

The resulting vector can be viewed as the average importance of each $\mathbf{q}_i$ with respect to $\mathbf{C}$. This token-to-sequence attention encoded in AoA is a key difference from BERT attention. β is then used to weigh the document-level attention α.

$$\mathbf{s} = \alpha^T\beta^T \tag{3}$$

The final attention vector $\mathbf{s} \in \mathbb{R}^N$ represents document-level attention weighted by the importance of query words.

Since the output of AoA is a vector of document length, to use it for answer start and end prediction we add a *two-headed* AoA layer into the BERT_{QA} model and this layer is trained together with the answer extraction layer during the finetuning stage. Concretely, the combined question and context hidden representation $\mathbf{H}^L$ from BERT is first separated to $\mathbf{H}^Q$ and $\mathbf{H}^C$ [4], followed by *two* linear projections of $\mathbf{H}^Q$ and $\mathbf{H}^C$ respectively to $\mathbf{H}_i^Q$ and $\mathbf{H}_i^C$, $i \in \{1, 2\}$:

$$\mathbf{H}_i^Q = \mathbf{H}^Q\mathbf{W}_i^Q, \tag{4}$$

$$\mathbf{H}_i^C = \mathbf{H}^C\mathbf{W}_i^C, \tag{5}$$

where $\mathbf{H}^Q$, $\mathbf{H}_i^Q \in \mathbb{R}^{M\times 1024}$; $\mathbf{H}^C$, $\mathbf{H}_i^C \in \mathbb{R}^{N\times 1024}$; and $\mathbf{W}_i^Q, \mathbf{W}_i^C \in \mathbb{R}^{1024\times 1024}$. Therefore, the AoA layer adds about 2.1 million parameters on top of BERT which already has 340 million. Next, we feed $\mathbf{H}_1^C$ and $\mathbf{H}_1^Q$ into the AoA calculation specified in Equations (1) - (3) to get the attention vector $\mathbf{s}_1$ for head 1. The same procedure is applied to $\mathbf{H}_2^Q$ and $\mathbf{H}_2^C$ to get $\mathbf{s}_2$ for head 2. Lastly, $\mathbf{s}_1$ and $\mathbf{s}_2$ are combined with $\boldsymbol{\ell}_b$ and $\boldsymbol{\ell}_e$ respectively via two weighted sum operations for answer extraction.

3.2.2 Attention Diversity (AD) layer

It has been shown through ablation studies (Kovaleva et al., 2019; Michel et al., 2019) that removing BERT attention heads can achieve comparable or better performance on some tasks. Our objective is to find out if we can diversify the information captured and train a better BERT model by enforcing diversity among the attention heads.

In a Transformer model, (Li et al., 2018) examine a few methods to enforce such diversity and see an improvement on machine translation tasks. Contrary to that we start with a pre-trained BERT model, take the attention output from scaled dot-product attention and compute the cosine similarity between all pairs of heads:

$$D = \sum_{i=1}^{Head}\sum_{j=1}^{Head}\frac{O^i \cdot O^j}{||O^i||||O^j||}.$$

We then average D for the per-token similarity and add it as an additional loss term. For each token, there are $16 + 15 + ... + 2$ total similarity calculations, 16 being the number of heads in BERT_{QA}. Figure 3 shows the modified structure of Multi-head Attention in the Transformer architecture. We apply this technique during finetuning on NQ and to the last layer of BERT only. It will be interesting to see how this additional training objective affects BERT pretraining, which we leave as future work.

4 Model Training

Our models follow the now common approach of starting with the pre-trained BERT language model and then finetune over the NQ dataset with an additional QA sequence prediction layer as described in section 3.1. Note that unless we specify otherwise, we are referring to the pre-trained "large" version of BERT with Whole Word Masking (BERT_W). BERT_W has the same model structure as the original BERT model, but masks whole words instead of word pieces for the Masked Language Model pre-training task and we empirically find this to be a better starting point for the NQ task.

[4] Superscript L is dropped here for notation convenience; we use the last layer $L = 24$ from the BERT output.

4.1 Data Augmentation (DA)

Model performance in MRC has benefited from training with labeled examples from human annotated or synthetic data augmentation from similar tasks. This includes the prior SOTA on NQ by (Alberti et al., 2019a) where 4 million synthetically generated QA pairs are introduced. In this paper, we similarly adapt and evaluate three different approaches for data augmentation: Crowd-sourced, Synthetic, and Adversarial.

Crowd-sourced DA: We leverage the previously released SQuAD 2.0 MRC dataset that obtained ~130k crowd-sourced question, answer training pairs over Wikipedia paragraphs. Note that we present results using a "pre-training" (PT) strategy where we first train on the augmentation data and, finally, perform fine-tuning exclusively on the NQ domain. We also experimented with a multi-task-learning setup as in (Ruder et al., 2019; Xu et al., 2018), but omit those experimental results for brevity since PT consistently proved to be a better augmentation strategy.

Synthetic DA: We also pre-train a model on 4M automatically generated QA examples. The generation works as follows: similar to (Dong et al., 2019), we first fine-tune a masked LM for question generation using SQuAD1.1 training examples—we choose RoBERTa (Liu et al., 2019) for its extended LM pre-training. Then a SQuAD MRC model trained on ten predefined question types—e.g. *what*, *how*, *when*, and *how many*, as opposed to full-length questions—is used to identify potential answer phrases in NQ training passages. Finally, we use diversity-promoting nucleus sampling (Holtzman et al., 2019) with a nucleus mass of .95 to sample questions from these passage-answer pairs, which has been shown to yield better QA training examples than standard beam search (Sultan et al., 2020).

Adversarial DA: Sentence Order Shuffling (SOS) The SOS strategy shuffles the ordering of sentences within paragraphs from the NQ training set. The strategy is based on an observation in the preliminary BERT_{QA} model that predictions favored earlier rather than later text spans. As noted by (Kwiatkowski et al., 2019), this appears to reflect a natural bias in Wikipedia that earlier texts tend to be more informative for general questions (a default long answer classifier predicting the first paragraph gets a LA F1 of 27.8%). Hence, our perturbation of the sentence ordering is similar in spirit to the types of perturbations introduced by (Zhou et al., 2019) for SQuAD 2.0 based on observed biases in the dataset.

5 Experiments

5.1 Datasets

Source Domain We choose NQ as our source dataset. It provides 307,373 training queries, 7,830 development queries, and 7,842 test queries (with the test set only being accessible through a public leaderboard submission). For each question, crowd sourced annotators also provide start and end offsets for short answer spans[5] within the Wikipedia article, if available, as well as long answer spans (which is generally the most immediate HTML paragraph, table, or list span containing the short answer), if available. The dataset also forces models to make an attempt at "knowing what they don't know" (Rajpurkar et al., 2018) by requiring a confidence score with each prediction. For evaluation, we report the offset-based F1 overlap score. For additional details on the data and evaluation see (Kwiatkowski et al., 2019).

Target Domain To test GAAMA's ZSTL transfer capability, we choose two academic[6] benchmark datasets on a related domain: Bio-medical. The first one uses a subset of the questions and annotations from task 8b of the BioASQ competition (Tsatsaronis et al., 2015). Specifically, we extract 1,266 factoid biomedical questions for which exact answers can be extracted from one of the PubMED abstracts marked as relevant by the annotators. We report the Factoid Mean Reciprocal Rank (MRR) as the evaluation metric. Secondly, we choose the very recent CovidQA (Tang et al., 2020) benchmark to illustrate GAAMA's performance on a globally important transfer learning dataset. This is a QA dataset specifically designed for COVID-19 and manually annotated from knowledge gathered from Kaggle's

[5] ~1% of the questions are annotated with boolean Yes/No instead of short answers. We leave it as future work to detect and generate answers for these types of queries.

[6] Note, we have tested GAAMA's ZSTL successful transfer on several in-house datasets which we cannot publish due to license restrictions.

COVID-19 Open Research Dataset Challenge. It is the first publicly available QA resource on the pandemic intended as a stopgap measure for guiding research until more substantial evaluation resources become available. It consists of 124 question–article pairs (v0.1) and hence does not have sufficient examples for supervised machine learning. CovidQA evaluates the zero-shot transfer capabilities of existing models on topics specifically related to COVID-19. One difference of CovidQA from the other QA datasets we evaluate is that it requires systems to predict the correct *sentence* that answers the question. Hence we intuitively report the P@1, R@3, and MRR based on the official evaluation metric.

5.2 Competitors

We compare GAAMA against three strong competitors from the industry research: **1)** A hybrid of a decomposable attention model for Natural Language Inference (Parikh et al., 2016) and DrQA (Chen et al., 2017), a retrieve and rank QA model, which obtains commendable results on SQuAD. **2)** The NQ baseline system (Alberti et al., 2019b) and **3)** The current industry SOTA on NQ (Alberti et al., 2019a) which utilizes 4 million synthetic examples as pre-training. Architecturally, the latter is similar to us but we propose more technical novelty in terms of both improved attention and data augmentation. We note there is very recent academic work (Zheng et al., 2020)which we omit as GAAMA outperforms them on short answers and more importantly we compare against large scale industry SOTA for the scope of this paper. Since their work is more academic, their model enjoys being computationally more expensive for accuracy than GAAMA as they involve computing graph attentions that are typically more difficult to be run in parallel if we want to do whole graph propagation (Veličković et al., 2018).

6 Results:

Attention Strategies: Both the AoA and AD strategies provide a meaningful $(0.7 - 0.9\%)$ improvement over a baseline BERT_W model as shown in Table 1. Note that our baseline BERT_W already achieves a stronger baseline than previously published SOTA by (Alberti et al., 2019a) by relying on the stronger whole-word-masking pre-training mechanism for the underlying BERT model. Combining both attention strategies with the SQUAD 2 PT yields the best single model performance, though the improvements are primarily on LA performance rather than SA. Exploring why only LA improves is left as part of our future work once we start with even larger, better pre-trained models.

Data Augmentation: As seen in Table 2, using a (well aligned) crowd-sourced dataset (SQuAD 2) for pre-training proves to be quite effective. It provides the largest data augmentation gain in SA F1, $\sim$1.6%, as well as a $\sim$1% gain in LA F1. Employing 4 million synthetic question answer pairs also provide similar gains in SA F1 and an even better gain ($\sim$2.3%) in LA F1. From an efficiency perspective, however, SQuAD 2 PT only introduces 130K additional examples to the training process, whereas synthetic data augmentation requires training over 4M additional examples (on top of the training required for the data generator). We also find that it was unhelpful to combine SQUAD 2 PT with 4M synthetic examples for improving single model performance; so we evaluate our best performing model architectures only using the SQuAD 2 PT strategy.

	Short Answer F1	**Long Answer F1**
Human Performance		
Single human	57.5	73.4
Super-annotator	75.7	87.2
Prior Work (Industry Research on NQ)		
DecAtt + Doc Reader (Parikh et al., 2016)	31.4	54.8
BERT_L w/ SQuAD 1.1 PT (Alberti et al., 2019b)	52.7	64.7
BERT_L w/ 4M Synthetic (Alberti et al., 2019a)	55.1	65.9
This Work		
GAAMA: BERT_W + AoA + AD + SQuAD 2 PT	**57.0**	**68.6**

Table 1: **Comparison of GAAMA vs prior work in the industry.** GAAMA clearly outperforms the competitors in both short and long answer F1.

Data Augmentation on BERT	Short Answer F1	Long Answer F1
BERT_W + SOS	55.8	66.7
BERT_W + 1M Synthetic	56.6	67.9
BERT_W + 4M Synthetic	56.9	**68.3**
BERT_W + SQuAD 2 PT	**57.0**	67.3
BERT_W + SQuAD 2 PT + 4M Synthetic	56.8	67.6

Table 2: **Performance of various Data Augmentation strategies.** SQuAD helps short but synthetic helps long answers.

6.1 ZSTL Experiments

We create a random train (75%) and test split (25%) of the BioASQ 8b annotated questions in order to assess the performance of GAAMA with and without training. Comparison with more heavily fine-tuned prior art (Yoon et al., 2019) is left as part of future work and beyond the scope of this work as they focus on fine-tuned large language models e.g. BioBERT (Lee et al., 2020) with more extensive vocabularies. Note again, that our work focuses on minimizing these steps for new target domains. Hence, since our objective is not to keep retraining GAAMA for every new domain, we refrain from changing the underlying pre-trained LM. We observe that the GAAMA's ZSTL config performs competitively (0.56 lower on MRR) on BioASQ showing that there is hope of transferring models zero-shot to entirely unseen domains.

On CovidQA, we predict the sentence that contains our predicted answers. Table 6 shows the results. GAAMA performs quite competitively to a BioBERT baseline and outperforms it on all the three metrics. This amplifies the fact that it is not always necessary to start with a domain-specific LM. We note that GAAMA gets better P@1, slightly lower R@3, and the same MRR and hence it still gives a tough competition to a system trained on empirically a much better performing pre-trained LM than BERT: T5 (Raffel et al., 2019). We also note that both the T5 and BioBERT baselines are trained specifically to do sentence classification whereas GAAMA performs reading comprehension to extract answer spans and we predict the sentence that contains the spans. So no new "task-specific' training is involved in this process.

Components	SAF1	LA F1
BERT_W + AoA + AD + SQuAD 2 PT	57.0	**68.6**
- AD	**57.2**	68.2
- AoA	57.0	68.5
- SQuAD 2 PT	56.7	67.5
Effects of Attention & Data Augmentation		
- AoA - SQuAD 2 PT	56.3	66.7
- AD - SQuAD 2 PT	56.1	67.3
- AoA - AD - SQuAD 2 PT	55.4	66.0

Table 3: Ablation study of GAAMA's various components.

Model	$F1$	T^G_{50}	T^G_{95}	T^C_{50}	T^C_{95}
Base	42.5	0.05	0.49	0.53	2.32
Large	50.8	0.10	0.66	1.51	6.00

Table 4: F1 and inference times for BERT base and large models running on GPU and CPU for a subset of the NQ dev set. T^D_K is the K-th percentile inference in seconds when running on device D (GPU or CPU).

GAAMA Configs	Factoid MRR
Trained on target domain	**24.93**
ZSTL	24.37

Table 5: Results on our test split of the BioASQ 8b dataset. GAAMA with ZSTL is highly competitive.

	P@1	R@3	MRR
Prior Work (Tang et al., 2020)			
BioBERT + MS MARCO	0.194	0.313	0.312
T5 + MS MARCO	0.282	**0.404**	**0.415**
This Work			
GAAMA (ZSTL)	**0.306**	0.377	0.414

Table 6: ZSTL performance of GAAMA vs. the prior work on the CovidQA dataset.

6.2 Efficiency

Inference: Inference efficiency is a crucial requirement of industry-scale systems. We investigate the inference times of both base and large models; while large models are ideal for academic benchmarks, the faster inference times of base models can be worth the reduction in accuracy in industrial settings. Measurements are carried out using a random sample of examples from the NQ dev set with a Nvidia® Tesla® P100 GPU and 8 threads from an Intel® Xeon® E5-2690 16-core CPU. In order to decrease inference time, we simulate passage retrieval to send the model the most relevant passage by selecting the first correct top level candidate if there is one and the first (incorrect) top level candidate if there is not. We find in Table 4 that switching from base to large yields an 8.3% absolute increase in F1 in exchange for 1.3x to 2.8x increases in inference time. When running the model on a GPU these result in manageable 95th percentile inference times of less than a second; whereas on the CPU the 95th percentile times are multiple seconds. We conclude that either of these models could be deployed in production environments on GPU only. In future work we intend to explore network pruning or knowledge distillation techniques for potential speedups with the large model.

Training: Efficient training is also an important component of industry-scale systems. To this end we consider both the number of model parameters and the amount of PT data. Our AoA implementation adds less than 1% to BERT_W's parameters and AD does not add any as it is implemented in the loss. Similarly, by using a well-aligned PT dataset (SQuAD 2.0) we are able to rival the performance of the much larger 4M synthetically generated corpus (Alberti et al., 2019a) with only 130K examples as seen in Table 2.

7 Analysis of GAAMA's Components

Table 3 shows the ablation study of GAAMA's components. Note that our best model's performance on short answers (57.2) almost matches a single human performance[7]. When doing manual error analysis on a sample of the NQ dev set, we do observe patterns suggesting that each of GAAMA's components do bring different strengths over just the best final combination (BERT_w + AoA + AD + SQuAD2 PT) e.g. the Wikipedia article for *Salary Cap* contains multiple sentences related to the query *"when did the nfl adopt a salary cap"*:

> The new Collective Bargaining Agreement (CBA) formulated in **2011** had an initial salary cap of $120 million...The cap was first introduced for the **1994** season and was initially $34.6 million. Both the cap and...

The later sentence contains the correct answer, *1994*, since the question is asking for when the salary cap was initially adopted. The SOS augmented model correctly makes this prediction whereas our SQUAD 2 augmented models predict *2011* from the earlier sentence. There are also cases where the correct answer span appears in the middle or later part of a paragraph and, though our SQUAD 2 augmented models predict the spans correctly, they assign a lower score (relative to its optimal threshold) than the SOS augmented model. The position bias, therefore, appears to hurt the performance of the system in certain situations where location of the answer span relative to the paragraph is not a useful signal of correctness.

On average, of course, the BERT_W+SQUAD2 PT + AoA + AD configuration performs the best and manual error analysis indicates some ability to better attend to supporting evidence when it is further out from the correct answer span. For example, the correct answer in example 1 from figure 1 is *December 24, 1836* which the AoA + AD model correctly identifies the answer span despite the question's and context's lack of lexical and grammatical alignment. While the base BERT_W models fail at extracting the date (instead predicting a span more closely associated with the keywords in the query such as *seven years earlier*).

8 Conclusion

Although large pre-trained language models have shown super-human performance on benchmark datasets like SQuAD, we show that there is plenty of room to make improvements on top of BERT_{QA}. Specifically, we outline prior strategies that do not work on a real benchmark consisting of "natural

[7](Kwiatkowski et al., 2019) notes that human performance was measured on a random sample of NQ dev.

questions" showing the difficulty of the dataset and need for better algorithms. We introduce GAAMA and outline several strategies that are broadly classified under attention and data augmentation and show how effective it can be to attain competitive performance on NQ compared to other industry baselines. We also outline GAAMA's OOTB zero-shot transfer on two unseen datsets and show optimistic performance. Our future work will involve adding larger pre-trained language models like T5 and also exploring multi-lingual QA.

9 Acknowledgement

We would like to thank the multilingual NLP team at IBM Research AI and the anonymous reviewers for their helpful suggestions and feedback.

References

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019a. Synthetic QA corpora generation with roundtrip consistency. *CoRR*, abs/1906.05416.

Chris Alberti, Kenton Lee, and Michael Collins. 2019b. A BERT baseline for the natural questions. *arXiv preprint arXiv:1901.08634*, pages 1–4.

Vittorio Castelli, Rishav Chakravarti, Saswati Dana, Anthony Ferritto, Radu Florian, Martin Franz, Dinesh Garg, Dinesh Khandelwal, Scott McCarley, Mike McCawley, et al. 2020. The TechQA Dataset. *Association for Computational Linguistics (ACL)*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879.

Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2017. Attention-over-attention neural networks for reading comprehension. In *Proc. of ACL (Volume 1: Long Papers)*, pages 593–602. ACL, July.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Bhuwan Dhingra, Danish Pruthi, and Dheeraj Rajagopal. 2018. Simple and effective semi-supervised question answering. *CoRR*, abs/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. In *NeurIPS*.

Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint*.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *TACL*, 6:317–328.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of bert. In *EMNLP*, pages 4356–4365.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: a benchmark for question answering research. *TACL*.

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.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. *arXiv preprint arXiv:1906.00300*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Jian Li, Zhaopeng Tu, Baosong Yang, Michael R Lyu, and Tong Zhang. 2018. Multi-head attention with disagreement regularization. In *EMNLP*, pages 2897–2903.

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. *CoRR*, abs/1907.11692.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems 32*, pages 14014–14024. Curran Associates, Inc.

Sewon Min, Danqi Chen, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2019. A discrete hard em approach for weakly supervised question answering. In *EMNLP*.

Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *EMNLP*.

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, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. *EMNLP*.

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

Sebastian Ruder, Matthew E. Peters, Swabha Swayamdipta, and Thomas Wolf. 2019. Transfer learning in natural language processing. In *Proc. of NAACL: Tutorials*, pages 15–18, Minneapolis, Minnesota, June. ACL.

Md Arafat Sultan, Shubham Chandel, Ramón Fernandez Astudillo, and Vittorio Castelli. 2020. On the importance of diversity in question generation for QA. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5651–5656, Online, July. Association for Computational Linguistics.

Raphael Tang, Rodrigo Nogueira, Edwin Zhang, Nikhil Gupta, Phuong Cam, Kyunghyun Cho, and Jimmy Lin. 2020. Rapidly Bootstrapping a Question Answering Dataset for COVID-19.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):138.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008. Curran Associates, Inc.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.

Wei Wang, Ming Yan, and Chen Wu. 2018. Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering. *ACL*.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Darrin Eide, Kathryn Funk, Rodney Michael Kinney, Ziyang Liu, William. Merrill, Paul Mooney, Dewey A. Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex D. Wade, Kuansan Wang, Christopher Wilhelm, Boya Xie, Douglas M. Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. 2020. Cord-19: The covid-19 open research dataset. *ArXiv*.

Yichong Xu, Xiaodong Liu, Yelong Shen, Jingjing Liu, and Jianfeng Gao. 2018. Multi-task learning for machine reading comprehension. *CoRR*, abs/1809.06963.

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*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.

Mark Yatskar. 2018. A qualitative comparison of CoQA, SQuAD 2.0 and QuAC. *CoRR*, abs/1809.10735.

Wonjin Yoon, Jinhyuk Lee, Donghyeon Kim, Minbyul Jeong, and Jaewoo Kang. 2019. Pre-trained language model for biomedical question answering. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 727–740. Springer.

Shiyue Zhang and Mohit Bansal. 2019. Addressing semantic drift in question generation for semi-supervised question answering. In *EMNLP*.

Zhuosheng Zhang, Yuwei Wu, Junru Zhou, Sufeng Duan, and Hai Zhao. 2019. SG-Net: Syntax-guided machine reading comprehension. *arXiv preprint arXiv:1908.05147*.

Bo Zheng, Haoyang Wen, Yaobo Liang, Nan Duan, Wanxiang Che, Daxin Jiang, Ming Zhou, and Ting Liu. 2020. Document modeling with graph attention networks for multi-grained machine reading comprehension. *ACL*.

Wen Zhou, Xianzhe Zhang, and Hang Jiang. 2019. Ensemble BERT with data augmentation and linguistic knowledge on SQuAD 2.0.