QAGAN: Adversarial Approach To Learning Domain Invariant Language Features

Shubham Shrivastava
Department of Computer Science
Stanford University
shubhams@stanford.edu

Kaiyue Wang
Department of Computer Science
Stanford University
garywang@stanford.edu

Abstract

Training models that are robust to data domain shift has gained an increasing interest both in academia and industry [1]. Question-Answering language models, being one of the typical problem in NLP research, has received much success with the advent of large transformer models [2]. However, existing approaches mostly works under the assumption that data is drawn from same distribution which is unrealistic.

In this paper, we explore adversarial training approach towards learning domain-invariant features so that language models can generalize well to *out-of-domain* datasets. We also inspect various other ways to boost our model performance including data augmentation by paraphrasing sentences, conditioning *end* of answer span prediction on the *start* word, and carefully designed annealing function. Our initial results shows that in combination with these methods, we are able to achieve 15.2% improvement in EM score and 5.6% boost in F1 score on *out-of-domain* validation dataset over the baseline. We also dissect our model outputs and visualize the model hidden-states by projecting them onto a lower-dimensional space, and discover that our specific adversarial training approach indeed encourages the model to learn domain invariant embedding and bring them closer in the multi-dimensional space.

1 Introduction

Question answering in language modeling is a difficult task as it requires language models to understand the context and the question. Transformers based encoder-only models such as BERT [3] has had tremendous success in solving these tasks. These models are pre-trained on a tremendous amount of unlabeled data from BooksCorpus [4] and Wikipedia, and then fine-tuned on downstream tasks like question-answering. During fine-tuning, these models encode both question and context embedding together separated by a special separator token [SEP], and then predicts for each context word probabilities for it being either a *start* word or an *end* word. Fine-tuning these models yield impressive results using only a single classification layer on top of pre-trained BERT.

While these models result in great performance on downstream tasks, they fail to generalize well across datasets. This is attributed by the inherent domain gap between various datasets which causes a large performance drop when tested on an unseen dataset. We attempt to minimize this domain gap and work towards building a model that is robust to domain shifts in dataset by learning domain-invariant features through adversarial training. We show through systematic evaluations that our approach actually helps the network close this gap and generalize better to the out-of-domain datasets.

We also condition our *end_logit* prediction on the *start_logit* through an `MLP` and a `Self-Attention` prediction head, and we find that in both cases it boosts the performance further. Our approach to minimizing the domain gap through adversarial training to confuse a discriminator working towards classifying the domain that each sample came from is intuitive and have proven to work in computer vision literature [5, 6, 7]. Additionally, we tried augmenting our data using paraphrasing by back translation and word substitution with close embedding [8]. We observed that such technique does help with generalization and over fitting issue and has positive improvement on out of domain classification task.

Concretely, our main contributions can be summarized as:

- We show through various qualitative and quantitative evaluations that adversarial training helps the model learn domain-invariant features across various datasets.
- We design an annealing function called *heated `tanh` annealing*, and show that it improves the model performance.
- We present a simple conditional `MLP` and `Self-Attention` prediction head for *start* and *end* logits prediction, and show through various experiments that they further helps the model perform better.

2   Related Work

Domain Adversarial Training: The idea of adversarial training was first proposed by [9] and is mainly used for task generation problem. Since then, learning domain invariant features through adversarial training framework has gain increasing attention in research community.

One method proposed by [10] trained 2 classifiers one to perform task specific classification task and the other, using the hidden state of the first classifier, predicts whether the data belong to source or target domain. Lee et al. [11] implemented a multi-domain classifier for the question-answering task to correctly predict the domain each data sample came from, while the QA model objective was set to minimize the KL divergence between discriminator output and a uniform distribution. Sato et al. [12] expanded on the binary classification discriminator architecture with an additional neural layer for each domain, which captures domain-specific feature representations and applied it to dependency parsing tasks.

Data Augmentation: Data augmentation has been shown to prevent model from over-fitting. Longpre et al. [8] explored data augmentation for QA task by paraphrasing the question context through back translation. Along the same line, Garg et al. [13] used `BERT-MLM` to generate alternatives sentences for masked tokens. Wei et al. [14] proposed some easy data augmentation techniques such as synonym replacement, random insert, random swap and deletion which are shown to improve performance of task with limited small data set.

We take inspiration from these work and build a model that is able to learn domain-invariant features, and further boost its performance through several techniques such as conditional logits prediction, data augmentation, annealing, and fine-tuning.

3   Approach

We set out to explore the domain-gap within various QA datasets [15, 16, 17, 18, 19, 20] by projecting higher-dimensional embedding extracted through a language model trained on humongous volume of unlabeled data onto a lower-dimensional space and then visually inspecting them. Specifically, we use a pre-trained DistilBERT model and project the hidden-states vectors onto a 2-dimensional space using t-SNE [21]. We observe that there indeed exists a domain-gap across these datasets as shown in figure 3a. Furthermore, we see that when these networks are fine-tuned on a downstream task, it further segregates the embeddings farther apart to a point where data from each dataset is clustered in its own island of higher-dimensional space. This motivates our approach of adversarial training regime where a discriminator is trained with an objective of classifying the dataset domain of question-answer pair, and the language model works towards confusing the discriminator. We call our model `QAGAN`, and as shown in figure 3c, it does reduce the domain gap across datasets.

We work with 6 datasets [15, 16, 17, 18, 19, 20] in our work, for three of which (`indomain_train`) we have $50k$ training samples per dataset, and for the other three (`oodomain_train`) we have 127 training samples as described in section 4.1. Goal of our work is to train a model that generalizes well to `oodomain_val` and `oodomain_test` dataset.

3.1 Baseline

Our baseline system fine-tunes a pre-trained DistilBERT [22] on all `indomain_train` dataset. The loss function is set up as a combination of negative log likelihood for the start and end location.

3.2 QAGAN

For building our model, we take the huggingface's [23] implementation of DistilBERT [22] and build a simple MLP-based prediction head to attach on top of the pre-trained DistilBERT's sequence encoder output. We also take the discriminator model implementation from the work by Lee et al. [11] and add support for training the model in adversarial fashion. The discriminator was implemented such that it works towards discriminating the dataset domain by operating on its `[CLS]` representation as it summarizes the whole context of the paragraph.

We also take inspiration from Wang et al. [24] and design a conditional prediction heads to condition our *end_logits* prediction on *start_logits* and notice that it improves our model performance. Two prediction heads designed in our work were linear MLP and self-attention based, and while both improved our model performance, we find the linear conditional head to yield the best results. These two models are shown in figure 1.
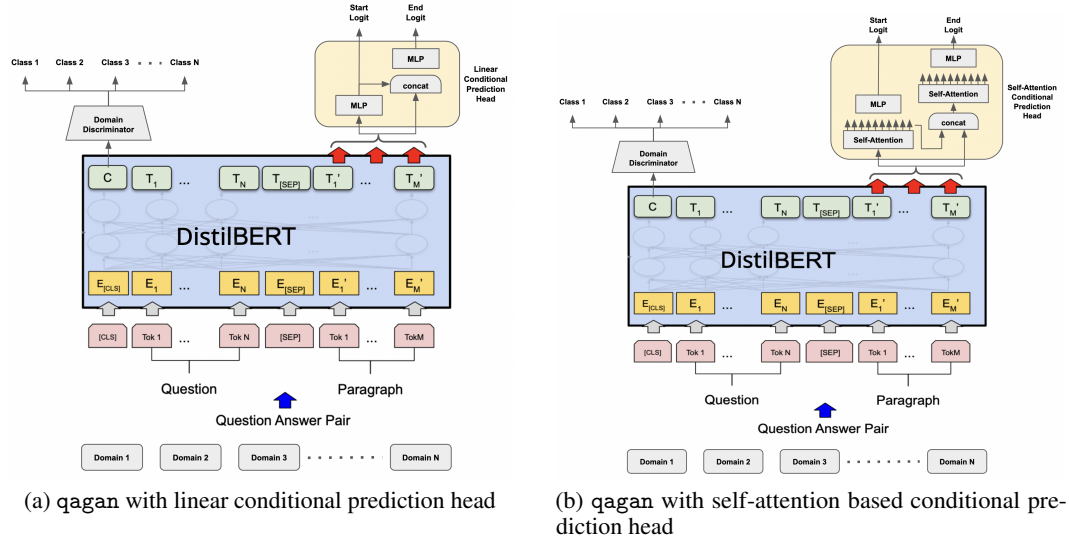

(a) `qagan` with linear conditional prediction head

(b) `qagan` with self-attention based conditional prediction head

Figure 1: Variants of `QAGAN`

We train our *question-answering* (QA) model by minimizing the *negative-log-likelihood* loss between ground-truth and prediction. In addition, we train our discriminator to classify the dataset domain class correctly, while the QA model learns how to fool the discriminator into outputting random dataset class probability.

$$L_{QA} = -\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N_k} [log P_\theta(\hat{y}_{i,s}^{(k)} | x_i^{(k)}, q_i^{(k)}) + log P_\theta(\hat{y}_{i,e}^{(k)} | x_i^{(k)}, q_i^{(k)})] \qquad (1)$$

$$L_{adv} = -\frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N_k} y_{i,rand}^{(k)} log P_\theta(\hat{y}_{i,c}^{(k)} | h_{cls}^{(k)}) \qquad (2)$$

3

$$L_D = -\lambda_2 \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N_k} y_{i,c}^{(k)} log P_\theta(\hat{y}_{i,c}^{(k)} | h_{cls}^{(k)}) \tag{3}$$

During QA model training we optimize the loss function, $L_{qagan} = L_{QA} + \lambda_1 L_{adv}$, and during the discriminator model training we optimize the loss function L_D, where λ_1 and λ_2 is a hyper-parameter chosen to control the importance of adversarial loss. Furthermore, L_{adv} is optimized every n_{fd} steps and L_D is optimized every n_{td} steps.

N, $\hat{y}_{i,s}$, and $\hat{y}_{i,e}$ in equation 1 are the total number of in-domain data, the start position and the end position of answer in the passage respectively. $\hat{y}_{i,c}$ and h_{cls} in equation 2 are the class of the dataset and [CLS] hidden representations respectively. By adding this term in the loss function during training the QA model, it is encouraging the QA model to produce hidden states that would confuse the discriminator model such that it would output random probabilities for each class.

During our initial experiments we did not observe considerable performance improvement, and we hypothesize that at the beginning of training, discriminator does not have much information to go off from, and hence training it to correctly classify the domain while the QA model works towards confusing it ends up worsening the model performance. To counter this, we design an annealing function (*heated tanh annealing*) described in section 3.2.1 which helps improve the network performance and also allows it to learn domain-invariant features. Additionally, we replaced the discriminator input with complete hidden states instead of just the [CLS] representation to examine its effects, and it did not seem to improve the performance.

Lee et al. [11] used Kullback-Leiber divergence between uniform distribution and the distribution of class probability generated by the discriminator during QA model training which helped the model generate a representation such that it confuses the discriminator. In our work, we also replaced this loss function with *negative log-likelihood* between the discriminator output and random classes, and it proved to improve the model performance slightly.

3.2.1 Heated `Tanh` Annealing

(a) n_{ws}=10k, n_{max}=250k   (b) n_{ws}=150k, n_{max}=250k
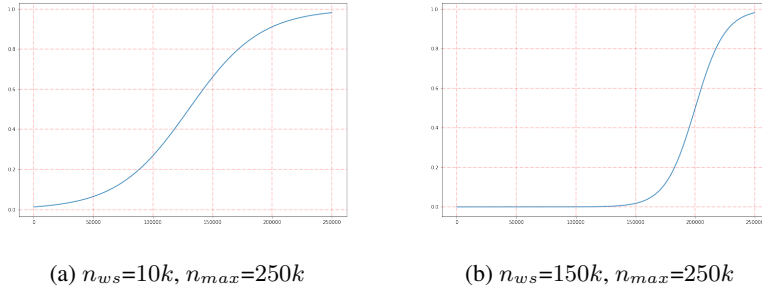
Figure 2: Heated `Tanh` Annealing for optimizing *discriminator*.

We designed an annealing function as given by equation 4 and illustrated in figure 2. This function allows the discriminator to initially learn a good model for domain classification, and slowly after n_{ws} warm-up steps, the model starts using the discriminator output as a proxy for learning domain-invariant features. We find that weighting the *fake discriminator loss* with this function resulted in an improved performance as expected.

$$f_{anneal}(z) = \frac{tanh(2 * \frac{2z - n_{max} - n_{ws}}{n_{max} - n_{ws}}) + 1}{2} \tag{4}$$

Where, n_{max} is maximum number of steps, and n_{ws} is the number of warm-up steps.

3.2.2 Conditional QA Prediction Head

We designed two conditional heads for *start* and *end* logits prediction. These heads are illustrated in figures 1a and 1b respectively.

Linear Conditional Head: Sequence output of DistilBERT is passed through an MLP to produce `start_logit` which is then concatenated with the same sequence and passed through another MLP to produce `end_logit`.

Self-Attention Conditional Head: Sequence output of DistilBERT is passed through a *self-attention* layer followed by an MLP to produce `start_logit`. Output of this *self-attention* layer is concatenated with the sequence output and fed through another *self-attention* layer followed by another MLP layer to then produce `end_logit`.

3.2.3 Data Augmentation

For data augmentation, we tried 2 different techniques to generate more data in order to regularize the model for more robustness.

Back Translation: We used a trained machine translation model MarianMTModel [25] provided by hugging face to translate the context and question into German and translate it back to English as a paraphrasing mechanism. In order to maintain the quality of the back translation, we slice the context around the answer text with an added padding. We also make sure that the sliced index does not happen within a sentence by heuristically looking for sentence ending punctuation. Moreover, to maintain the translated sentence quality, we calculate a perplexity score using pre-trained GPT-2 model [26] and only take the translated text that doesn't deviate from original text.

It is also observed that after translation, the answer text will get morphed and can no longer be found in the translated text. We just skip the generated text under these circumstances.

The paraphrasing reduces the dependency on domain specific sentence structure. For example we see

> a tiny girl the size of the old woman's thumb

Get paraphrased to

> a little girl the size of the thumb of the old woman.

Word Replacement: To increase the robustness of word choices, we also does word replacement where we randomly replace words with its counter parts that is close in embedding space.
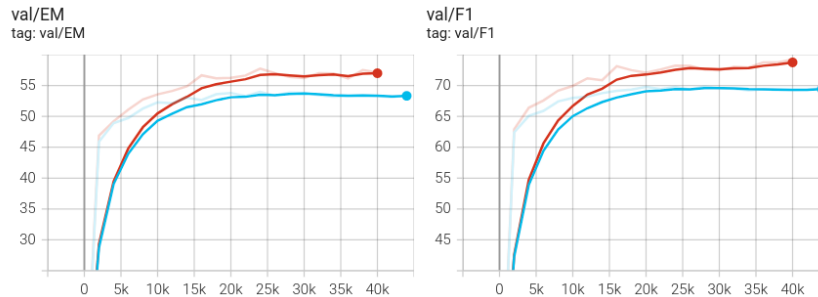
(a) Validation plot on model with data augmentation (red) and model without data augmentation (blue)

3.3 Fine-tuning:

After the models have been trained on `in-domain-train` set, we also fine-tune them on the `out-of-domain-train` set, and that brings about 3.8% boost in EM score and 3.3% boost in F1 score while tested on `out-of-domain`-validation set. This however, has worsens the performance when the model is initially trained on both (`in-domain-train` + `out-of-domain-train`) datasets, and rightfully so, because the model will overfit to the small `out-of-domain-train` split during fine-tuning.

4 Experimental Details

We built various variants of `QAGAN` and performed experiments to test our hypothesis.

- Starting from the discriminator loss function during QA model training, we experimented with minimizing *KL-divergence* between discriminator classification and a uniform distribution, and *negative-log-likelihood* between discriminator classification output and random dataset domain class (D_{obj} column in table 2).
- We designed several prediction heads including a simple MLP ($P_{head} = mlp$), conditional MLP ($P_{head} = cmlp$), and a conditional self-attention ($P_{head} = csat$) based sub-network.
- Another option during our model training is to use either `[CLS]` representation as an input to the discriminator, or use the complete `hidden-states` representation, and in our experiments we found the former to yield better results.
- Using insights from our domain-gap analysis, we will also experiment with enforcing on better generalization by constraining hidden state to conform to normal distribution (h_kld column in table 2).
- We also designed an annealing function to aid the training of our discriminator, and experimented with and without this function to evaluate its effectiveness (*anneal* column in table 2).
- We trained our model by combining the augmented data to evaluate the effectiveness of data augmentation(*aug* column in table 2)
- For some of our experiments we also included the *out-of-domain train* set into the complete training set to analyze its effect on final model performance (*ood_train* column in table 2)

For all our experiments, we used a learning rate of $3e - 5$, and trained our model for a total of 3 epochs. Hyper-parameters used in our loss functions were: $\lambda_1 = 0.5$, $\lambda_2 = 0.5$, $n_{ws} = 1000$, $n_{max} = 250k$, $n_{td} = 1$, $n_{fd} = 2$. Training was done on a machine with single NVIDIA RTX 3090 GPU, and takes about 53 minutes per epoch.

4.1 Dataset Description

Table 1: Statistics for datasets used for building the QA system for this project. **Question Source** and **Passage Source** refer to data sources from which the questions and passages were obtained.

Dataset	Question Source	Passage Source	Train	Validation	Test
in-domain datasets					
SQuAD[17]	Crowdsourced	Wikipedia	50,000	10,507	-
NewsQA[16]	Crowdsourced	News articles	50,000	4,212	-
Natural Questions[15]	Search logs	Wikipedia	50,000	12,836	-
SQuAD Aug	Crowdsourced	Wikipedia	20,000	-	-
NewsQA Aug	Crowdsourced	News articles	20,000	-	-
Natural Questions Aug	Search logs	Wikipedia	20,000	-	-
oo-domain datasets					
DuoRC[19]	Crowdsourced	Movie reviews	127	126	1,248
RACE[20]	Teachers	Examinations	127	126	419
RelationExtraction[18]	Synthetic	Wikipedia	127	126	2,693
DuoRC Aug	Crowdsourced	Movie reviews	314	-	-
RACE Aug	Teachers	Examinations	240	-	-
RelationExtraction Aug	Synthetic	Wikipedia	250	-	-

We used 3 in-domain reading comprehension datasets (Natural Questions [15], NewsQA [16] and SQuAD [17]), each have a sample size of 50K for training the QA system and we used 3 small out-of-domain datasets (RelationExtraction [18], DuoRC [19], RACE [20]) with size of 127 for fine-tuning. To preprocess the data, we converted each (question, paragraph) into multiple chunks of size 384 with a stride of 128 in order for it to fit the distilBERT model input dimension.

As introduced in previous section, we also explored various data augmentation techniques including paraphrasing by back-translation introduced in [27] and semantic-preserving perturbation techniques introduced in [28] which generates an additional 20K for each indomain datasets and 1K for each oodomain datasets.

4.2 Evaluation method

We evaluated our method via two metrics: **Exact Match (EM)** score and **F1** score. Exact Match(EM) is a strict binary measure that tells us whether the predicted span of words exactly match the reference. F1 measure, on the other hand, is a more relaxed metric and is the harmonic mean of precision and recall. Both metrics are widely used in the evaluation of question and answering task. [17]

4.3 Quantitative Results

Table 2: Experimental results of our method on QA *validation* datasets. Blue color highlights best performance without fine-tuning on *oodomain_train* dataset or including *oodomain_train* in the *train* set. $P_{head} \rightarrow$ prediction head, $h\_kld \rightarrow$ KLD loss on hidden-states,$D_{input} \rightarrow$ discriminator input, $D_{obj} \rightarrow$ discriminator objective, $ood\_train \rightarrow$ oodomain_train set included in training, $aug \rightarrow$ data augmentation. Prediction Heads: mlp $\rightarrow$ linear MLP head; cmlp $\rightarrow$ condition MLP head; csat $\rightarrow$ conditional self-attention head.

Method	P_{head}	finetune	anneal	h_kld	ood_train	aug	D_{input}	D_{obj}	ind_val F1	ind_val EM	ood_val F1	ood_val EM
baseline	mlp	✗	-	-	✗	✗	-	-	70.49	54.48	48.29	30.89
baseline	mlp	✓	-	-	✗	✗	-	-	-	-	49.68	34.03
qagan	mlp	✗	✓	✗	✗	✗	[CLS]	KLD	70.10	54.24	46.56	31.15
qagan	mlp	✓	✓	✗	✗	✗	[CLS]	KLD	-	-	47.38	33.25
qagan	mlp	✗	✓	✗	✗	✗	[hidn]	NLL	68.88	52.69	46.95	30.89
qagan	mlp	✓	✓	✗	✗	✗	[hidn]	NLL	-	-	48.46	34.03
qagan	mlp	✗	✓	✗	✗	✗	[CLS]	NLL	69.85	53.84	46.92	31.68
qagan	mlp	✓	✓	✗	✗	✗	[CLS]	NLL	-	-	49.16	34.03
qagan	csat	✗	✓	✗	✗	✗	[CLS]	NLL	69.79	53.67	47.32	31.15
qagan	csat	✓	✓	✗	✗	✗	[CLS]	NLL	-	-	48.87	34.55
qagan	cmlp	✗	✗	✗	✗	✗	[CLS]	NLL	70.01	54.06	49.30	32.98
qagan	cmlp	✗	✓	✓	✗	✗	[CLS]	NLL	69.85	54.09	47.88	30.89
qagan	cmlp	✓	✓	✓	✗	✗	[CLS]	NLL	-	-	49.05	32.20
qagan	cmlp	✗	✓	✗	✗	✗	[CLS]	NLL	70.00	53.84	49.38	34.29
qagan	cmlp	✗	✓	✗	✓	✗	[CLS]	NLL	69.56	53.80	50.25	35.08
qagan	cmlp	✗	✓	✗	✓	✓	[CLS]	NLL	73.21	55.13	50.49	**35.90**
qagan	cmlp	✓	✓	✗	✗	✗	[CLS]	NLL	-	-	**51.00**	35.60

Table 3: Experimental results of our method on QA *test* datasets. $P_{head} \rightarrow$ prediction head, $D_{obj} \rightarrow$ discriminator objective, $ood\_train \rightarrow$ oodomain_train set included in training, $aug \rightarrow$ data augmentation. Prediction Head: cmlp $\rightarrow$ condition MLP head. *heated tanh annealing* was used for all experiments.

Method	P_{head}	finetune	ood_train	aug	D_{input}	D_{obj}	oodomain_test F1	oodomain_test EM
qagan	cmlp	✓	✗	✗	[CLS]	NLL	58.193	40.665
qagan	cmlp	✗	✓	✗	[CLS]	NLL	58.671	41.009
qagan	cmlp	✗	✓	✓	[CLS]	NLL	**58.898**	**42.362**

Adversarial training helps our model learn domain-invariant features as evident by figure 3c. These domain-invariant features help the model generalize well to *out-of-domain* dataset and hence perform better on the validation set. We train our network with various configurations of parameters and

notice a consistent improvement in F1 and EM score of *out-of-domain* validation dataset (ood_val) as we keep adding components introduced in our work including *conditional prediction head*, *heated* `tanh` *annealing*, *negative-log-likelihood* training of the discriminator during QA model training, and data augmentation. While we do not notice any improvement in F1 and EM score for *in-domain* validation dataset, the scores for our best model on *out-of-domain* validation set after finetuning our model on *out-of-domain* train set is pretty significant (5.6% improvement in F1 score and 15.2% improvement in EM score).

5   Qualitative Analysis

Effectiveness of Adversarial Training: Figure 3a illustrates the domain-gap across various datasets, and is extracted through a pre-trained *DistilBERT* [22] model. We train a *baseline* QA model on *indomain_train* dataset which is formulated by adding an `MLP` prediction head on top of the *DistilBERT* model and analyze the embeddings. As seen in the plot (figure 3b, embeddings for each dataset coming from this trained model is clustered individually and clearly shows that the domain-gap is accentuated even more with this setting. Lastly, with our approach, QAGAN, we find that although multi-modal, various dataset embeddings coexists in similar high-dimensional space as shown in figure 3c. This solidifies our hypothesis that the adversarial training setting indeed helps the model learn domain-invariant features
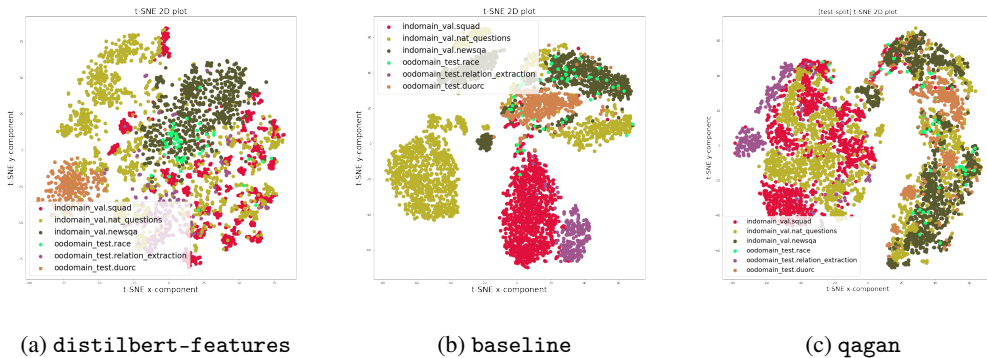

(a) `distilbert-features`   (b) `baseline`   (c) `qagan`

Figure 3: Domain-gap across various QA dataset embeddings and the effectiveness of applying our method towards learning domain-invariant features.

Quality of Predicted Answers: We show a few random qualitative samples predicted by our model and further analyze the quality of predictions in appendix (section 7). In particular, we observe that the model does extremely well on *named entities*, but suffers badly with contexts requiring some common reasoning.

6   Conclusion

We presented a method for training a question-answering language model in an adversarial fashion and showed through various experiments that it helps the model generalize well to *out-of-domain* dataset. We supplemented the discriminator model training with an annealing function, carefully designed to first let it learn correct classification task and then slowly help the language model learn domain-invariant features. Furthermore, we condition *end* logits prediction on *start* logits which facilitates a great deal of model improvement. Through systematic experiments with various configurations, we find that our model, QAGAN, with a conditional linear prediction head provides us the best results and achieves 5.6% improvements in F1 score and 15.2% improvement in EM score over the baseline. We analyze the higher-dimensional embeddings produced by our model and confirm that our method of adversarial training indeed helps the language model learn domain-invariant features and generalizes well towards *out-of-domain* datasets.

References

[1] Muhammad Awais, Fengwei Zhou, Hang Xu, Lanqing Hong, Ping Luo, Sung-Ho Bae, and Zhenguo Li. Adversarial robustness for unsupervised domain adaptation. 2021.

[2] Kate Pearce, Tiffany Zhan, Aneesh Komanduri, and Justin Zhan. A comparative study of transformer-based language models on extractive question answering. 2021.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[4] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[5] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.

[6] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 79–88, 2018.

[7] Nikita Jaipuria, Xianling Zhang, Rohan Bhasin, Mayar Arafa, Punarjay Chakravarty, Shubham Shrivastava, Sagar Manglani, and Vidya N. Murali. Deflating dataset bias using synthetic data augmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pages 3344–3353. Computer Vision Foundation / IEEE, 2020.

[8] Shayne Longpre, Yi Lu, Zhucheng Tu, and Chris DuBois. An exploration of data augmentation and sampling techniques for domain-agnostic question answering. *CoRR*, abs/1912.02145, 2019.

[9] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv 1412.6572*, 12 2014.

[10] Yaroslav Ganin, E. Ustinova, Hana Ajakan, Pascal Germain, H. Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky. Domain-adversarial training of neural networks. In *J. Mach. Learn. Res.*, 2016.

[11] Seanie Lee, Donggyu Kim, and Jangwon Park. Domain-agnostic question-answering with adversarial training. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*. Association for Computational Linguistics, 2019.

[12] Motoki Sato, Hitoshi Manabe, Hiroshi Noji, and Yuji Matsumoto. Adversarial training for cross-domain Universal Dependency parsing. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 71–79, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[13] Siddhant Garg and Goutham Ramakrishnan. Bae: Bert-based adversarial examples for text classification. pages 6174–6181, 01 2020.

[14] Jason Wei and Kai Zou. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China, November 2019. Association for Computational Linguistics.

[15] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019.

[16] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[17] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[18] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[19] Amrita Saha, Rahul Aralikatte, Mitesh M. Khapra, and Karthik Sankaranarayanan. DuoRC: Towards Complex Language Understanding with Paraphrased Reading Comprehension. In *Meeting of the Association for Computational Linguistics (ACL)*, 2018.

[20] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[21] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[22] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020.

[23] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics.

[24] Shuohang Wang and Jing Jiang. Machine comprehension using match-lstm and answer pointer. *CoRR*, abs/1608.07905, 2016.

[25] Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[26] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[27] Shayne Longpre, Yi Lu, Zhucheng Tu, and Chris DuBois. An exploration of data augmentation and sampling techniques for domain-agnostic question answering. volume abs/1912.02145, 2019.

[28] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865, Melbourne, Australia, July 2018. Association for Computational Linguistics.

7 Appendix

7.1 Qualitative Samples

We show a few randomly selected qualitative sample predictions from our model below. The model seems to perform well in most cases, particularly doing pretty well for named entities such as "Gupta Empire" and "Emily Perkins". Some of the failure cases include large contexts with references that need prior knowledge built into the system, an example of which would be for the question "when was the first wonder woman comic released" where the context refers to two dates for two events - (1) character first appeared in December 1941 (2) character's first cover was in January 1942. This is a particularly difficult question to answer as the model should understand that comic release date refers to when the character was first appeared on Sensation Comics cover.

Question: the golden age of india took place during the rule of the
Context: BPB The Gupta Empire was an ancient Indian empire , which existed at its zenith from approximately 319 to 485 CE and covered much of the Indian subcontinent . This period is called the Golden Age of India . The ruling dynasty of the empire was founded by Sri Gupta ; the most notable rulers of the dynasty were Chandragupta I , Samudragupta , and Chandragupta II . The 5th - century CE Sanskrit poet Kalidasa credits the Guptas with having conquered about twenty - one kingdoms , both in and outside India , including the kingdoms of Parasikas , the Hunas , the Kambojas , tribes located in the west and east Oxus valleys , the Kinnaras , Kiratas , and others . EEPE
Answer: Gupta Empire
Prediction: Gupta Empire

Question: Other than the motion picture and television industry, what other major industry is centered in Los Angeles?
Context: The motion picture, television, and music industry is centered on the Los Angeles in southern California. Hollywood, a district within Los Angeles, is also a name associated with the motion picture industry. Headquartered in southern California are The Walt Disney Company (which also owns ABC), Sony Pictures, Universal, MGM, Paramount Pictures, 20th Century Fox, and Warner Brothers. Universal, Warner Brothers, and Sony also run major record companies as well.
Answer: music
Prediction: music industry

Question: Where was a lab for Tesla set up?
Context: In late 1886 Tesla met Alfred S. Brown, a Western Union superintendent, and New York attorney Charles F. Peck. The two men were experienced in setting up companies and promoting inventions and patents for financial gain. Based on Tesla's patents and other ideas they agreed to back him financially and handle his patents. Together in April 1887 they formed the Tesla Electric Company with an agreement that profits from generated patents would go to Tesla, to Peck and Brown, and to fund development. They set up a laboratory for Tesla at 89 Liberty Street in Manhattan where he worked on improving and developing new types of electric motors, generators and other devices.
Answer: Manhattan
Prediction: 89 Liberty Street in Manhattan

Question: who plays unis in she 's the man
Context: BLiB Emily Perkins as Eunice Bates , Olivia 's nerdy , eccentric friend and Duke 's lab partner EELiE
Answer: Emily Perkins
Prediction: Emily Perkins

Question: Why did Berlin Huguenots switch to German from French in their services?
Context: In Berlin, the Huguenots created two new neighbourhoods: Dorotheenstadt and Friedrichstadt. By 1700, one-fifth of the city's population was French speaking. The Berlin Huguenots preserved the French language in their church services for nearly a century. They ultimately decided to switch to German in protest against the occupation of Prussia by Napoleon in 1806-07. Many of their descendents rose to positions of prominence. Several congregations were founded, such as those of Fredericia (Denmark), Berlin, Stockholm, Hamburg, Frankfurt, Helsinki, and Emden.
Answer: in protest against the occupation of Prussia by Napoleon
Prediction: protest against the occupation of Prussia by Napoleon in 1806-07

Question: How was scarcity managed in many countries?
Context: Price controls exacerbated the crisis in the US. The system limited the price of "old oil" (that which had already been discovered) while allowing newly discovered oil to be sold at a higher price to encourage investment. Predictably, old oil was withdrawn from the market, creating greater scarcity. The rule also discouraged development of alternative energies. The rule had been intended to promote oil exploration. Scarcity was addressed by rationing (as in many countries). Motorists faced long lines at gas stations beginning in summer 1972 and increasing by summer 1973.
Answer: rationing
Prediction: rationing

Question: In which year did the newspaper define southern California?
Context: In 1900, the Los Angeles Times defined southern California as including "the seven counties of Los Angeles, San Bernardino, Orange, Riverside, San Diego, Ventura and Santa Barbara." In 1999, the Times added a newer county—Imperial—to that list.
Answer: 1900
Prediction: In 1900

Question: when was the first wonder woman comic released
Context: BPB Wonder Woman is a fictional superhero appearing in American comic books published by DC Comics . The character is a founding member of the Justice League , a goddess , and Ambassador - at - Large of the Amazon people . The character first appeared in All Star Comics 8 in December 1941 and first cover - dated on Sensation Comics 1 , January 1942 . In her homeland , the island nation of Themyscira , her official title is Princess Diana of Themyscira , Daughter of Hippolyta . When blending into the society outside of her homeland , she adopts her civilian identity Diana Prince . The character is also referred to by such epithets as the Amazing Amazon " , the Spirit of Truth " , Themyscira 's Champion " , the God - Killer " , and the " Goddess of Love and War " . EEPE
Answer: January 1942
Prediction: December 1941