Domain Adaptive Adversarial Feature Disentanglement for Neural Question Answering*

Stanford CS224N {Default Robust} Project

Yichen Li, Wei Ren, Xuran Wang
liyichen, weiren, xuranw@stanford.edu

Abstract

Learning-based Question Answering systems have achieved significant success with the help of large language models and pre-trained model weights. However, existing approaches assume that data is drawn *i.i.d* from the same distribution, which violate the more realistic scenario that test-time texts and questions are under different distributions. Deep networks have been used to learn transferable representations for domain adaptation, which has shown success in various vision tasks. In this project paper, we study the problem of domain adaptive question answering leveraging various techniques, ranging from Data Augmentation, Layer Re-initialization and Domain Adversarial Alignment. Our evaluation results on the provided out-of-domain datasets show that our proposed method is able to bring 8.56% performance improvement, compared to the vanilla baseline using DistilBert without any of such domain adaptive designs.

1 Introduction

Modern scientific progress has already successfully built various Questions Answering (QA) systems which are fast, robust and accurate in retrieving information from domain passage and identifying correct text span as the corresponding answer, given a question. However, these QA systems assume training and testing data are sampled *i.i.d* from the same distribution, and thus have been trained to overfit to a specific dataset, which fail to generalize well to out-of-domain (OOD) dataset. Real world applications have revealed the need to build a single QA model applicable to various domains without further fine-tuning to out-of-domain datasets. A robust QA system that is capable of generalizing across different domains not only alleviates the problem of test-time distribution shifts but also is meaningful for low-resource language understanding.

Recently, Domain Adaptation (DA) works are proposed aiming to transfer knowledge learned from one or more labeled source domains to a target domain. Liu et al. [1] tried to learn an interpretable representation using GANs [2, 3]. Ganin et al. [4] applied adversarial training [2] for the domain adaptation purpose. In this project, we aim to leverage existing transfer learning and domain adaptation techniques to study the problem of domain adaptive QA.

In this paper, we aim to leverage the DistilBERT model as a backbone to built a robust QA system that is capable of question-answering for OOD datasets. To this end, we developed a framework consisting of 1) Data Augmentaton, 2) Layer Re-initialization as well as 3) Domain-agnostic Feature Representation learning via an Wasserstein-stabilized adversarial training framework. Our project specifically focuses on the QA task, where our model is given a paragraph, and a question about that paragraph, as input. Then our model will have to select a span of text (predict the start and end of the span) directly from the paragraph as an answer to the question correctly, if the question is answerable. Otherwise, the answer should be N/A. During training, our model is trained on several data-rich in-domain (ID) datasets and also with a few training examples from several smaller datasets of different distribution. Our learned model is able to transfer knowledge learned from the data-rich datasets to smaller/low-resource datasets. The contribution of our work can be summarized as follows:

*We will use 3 late days from our remaining late days.

- We proposed to use an adversarial domain alignment scheme on the DistilBERT backbone with last layer reinitialization to firstly train on both the data-rich ID QA datasets and data augmented OOD datasets, following a finetuning stage on data-augmented OOD datasets to tackle the task of domain-adaptive QA.
- We conducted extensive experiments to demonstrate the effectiveness of our method.
- We analyzed each of our proposed design choices through thorough ablation experiments.

2 Related Work

Our proposed methods are related to three different areas of work, 1) Data Augmentation, 2) Layer Re-initialization and Learning Rate Warm-up, and 3) Domain Adversarial Alignment.

Data Augmentation. Automatic data augmentation is a technique that has already been widely used in the field of computer vision. However, due to the huge differences in various language processing tasks, it can be very challenging to come up with universal augmentation rules that work for general data. One popular data augmentation technique is to use back translation, where new data is generated by translating sentences firstly into some other language and then back into the original language [5]. Other valid techniques include using data noising as smoothing [6], and replacing words with synonyms [7, 8, 9]. However, specifically for the QA task, most of these methods can be expensive to implement. For example, it's challenging to maintain or define the correct answer after applying back translation, which may add unexpected bias and inaccuracy into the newly generated data.

Layer Re-initialization & Learning Rate Warm-up. Re-initialization of pre-trained layers and learning rate warm-up are two ways to improve the performance of fine-tuning so as to increase the robustness of the model. Firstly, researchers have found that the initialization of network parameters has significant effect on the training or fine-tuning of deep neural networks [10, 11, 12]. Tamkin et al [13] investigated the weight re-initialization for layers which provides dominant contribution for transferability. Also as Zhang et. al [11] suggested, simple re-initialization of top pre-trained layers of BERT could boost the validation performance of fine-tuning process and enable the model to generalize better for different NLP tasks without focusing too much on the pre-trained task. Learning rate warm-up, on the other hand, helps overcome the primacy effect from early training examples so as to improve the convergence speed and generalization [14, 15]. It enables the model weights to be trained with equal emphasis on each training batch during the initial stage of training and that turns out to boost the model robustness dramatically. Both methods are general approaches for improving the robustness of deep neural networks, so we applied them on the DistilBERT baseline model seeking for performance enhancement.

Adversarial Alignment. Domain adaptation works are proposed aiming to transfer knowledge learned from one or more labeled source domains to a target domain. Ganin et al. [4] applied adversarial training [2] for the domain adaptation purpose. The method of domain adversarial alignment has also been widely used by other NLP tasks, such as text classification [16, 17], sentiment analysis [18], and relation extraction [19] for its generic mechanism of explicitly dealing with domain shift on a feature level. Chen et al. [18] used it for cross-lingual sentiment classification, in which the adversarial component has a classifier that tries to classify the language of an input sentence. Xu et al. [20] has extended the training mechanism of DANN [4] to conduct relation extraction. Li et al. [21] used domain adversarial alignment to conduct language identification in the scenario that test time languages could have domain gap to the languages used during training. For the task of question answering, Lee et. al [22] built a QA system that can generalize well to various domains with the technique of adversarial training and is applicable to any QA model. Similar to to the method proposed in [22], we adopted the adversarial training scheme to DistilBERT for QA task in our project with various performance enhancing designs.

3 Experiments Overview

Dataset. Six datasets, including three large ID reading comprehension datasets (Natural Questions, NewsQA and SQuAD) each with 50000 training examples and three small OOD datasets (RelationExtraction [23], DuoRC [24], RACE [25]) each with only 127 training examples, are used as the fine-tuning datasets. Model parameters will be tuned via OOD validation sets. The best performance of our QA system on OOD test sets will be reported.

Table 1: An Example of Augmented Context Paragraph by EDA

Operation	Definition	Context Paragraph
None	No operation is performed.	The Seattle Metropolitans were a professional ice hockey team based in Seattle, Washington which played in the Pacific Coast Hockey Association from 1915 to 1924.
SR	Randomly choose words from the sentence that are not stop words or in the answer s pan. Replace each of these words with one of its synonyms chosen at random.	the seattle metropolitans were a professional ***frosting*** hockey team based in seattle washington which ***engage*** in the pacific ***seacoast*** hockey ***connexion*** from 1915 to 1924
RI	Find a random synonym of a random word in the sentence that is not a stop word. Insert that synonym into a random position in the sentence (not in the answer span).	the seattle metropolitans were a professional ***miniature*** ice hockey team based in seattle washington which played in the ***squad*** pacific coast hockey association ***ocean*** from 1915 ***slideway*** to 1924
RS	Randomly choose two words in the sentence (not in the answer span) and swap their positions.	the ***pacific*** metropolitans ***team*** a coast professional ice hockey ***were*** based in seattle washington which played in the ***seattle*** hockey association from 1915 to 1924
RD	Randomly remove words in the sentence (not in the answer span).	the seattle metropolitans were a professional ice hockey in seattle washington played in coast from 1915 to 1924

Evaluation Metric. Exact Match (EM) score and F1 score will be combined for evaluation purpose. These two scores will be averaged across the entire evaluation datasets to get the final reported scores.

Baseline. The baseline QA system finetunes the pre-trained DistilBERT on all ID training sets and is evaluated on OOD validation sets, achieving a F1 score of **47.32** and a EM of 32.98.

4 Out-of-Domain Fine-tuning and Data Augmentation

4.1 Method Description

Recent findings suggested a simple set of universal data augmentation techniques can be surprisingly helpful in boosting model performance, especially for small datasets, on text classification tasks [26]. Here, we explored four easy data augmentation (EDA) techniques for our QA task, including synonym replacement(SR), random insertion(RI), random swap(RS), and random deletion(RD) (Table 1).

For the context paragraph of a given QA sample in the train set, we firstly clean the paragraph by lowercasing every character and removing punctuations. Since long context paragraph have more words than short ones, to compensate, we vary the number of words changed, N, for SR, RI, RS and RD based on context paragraph length l with the formula $N = l\alpha$, where α is a parameter that indicates the percent of words changed in a paragraph. Each augmentation operation has its own parameter α, symbolizing its strength. Furthermore, for every original context paragraph, we generate n_{aug} augmented paragraphs, where each one is obtained by randomly choosing and performing one of the augmentation operations. Particularly, all changes made in a paragraph are performed on parts that do not contain the answer span, so that it's guaranteed the original answer still exists in newly augmented paragraphs. Note our EDA only applies to context paragraphs, while the question and answer per sample remain unmodified. (Our implementation of EDA referenced the official code release from [26], but we have adapted it extensively to fit the need of QA task.)

4.2 Experimental Results

The following experiments are performed under a batch size of 16 and a learning rate of 3e-5. All training and finetuning processes take two epochs and are evaluated on OOD validation sets every 10 batches, during which the best model weights are recorded.

Experiment 1: OOD Fine-tuning. On top of the baseline model, we have continued to fine-tune it with OOD train sets, and obtained a higher F1 score of 49.43 with 4.46% performance boost. It has validated our hypothesis that even small OOD datasets can still be helpful in improving model performance, which makes sense as our model learns what's previously beyond-reach OOD knowledge for the first time.

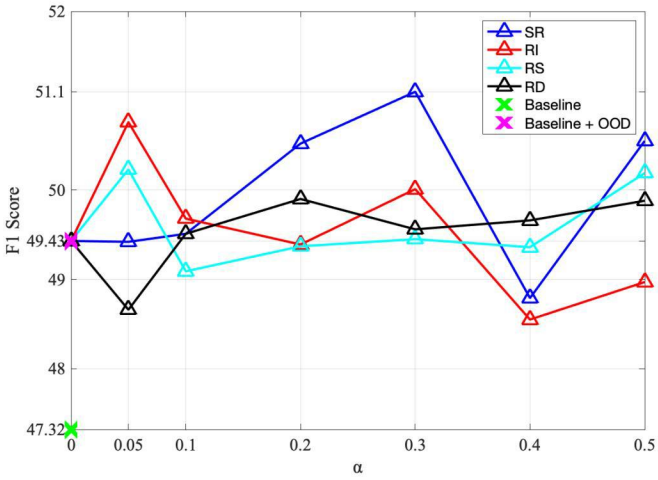

Figure 1: Validation results for ablation study by varying operational strength α ($n_{aug} = 2$). The optimal parameter for each operation is $\alpha_{SR} = 0.3, \alpha_{RI} = \alpha_{RS} = 0.05, \alpha_{RD} = 0.2$.

Table 2: Validation Results for Augmented OOD Fine-tuning on Baseline($\alpha_{all} = 0.1$)

Methods	F1	EM	Performance Gain (F1)
Baseline	47.32	32.98	0
Baseline + OOD Finetune	49.43	36.39	4.46%
Baseline + OOD Finetune + EDA($n_{aug} = 1, \alpha_{all} = 0.1$)	49.62	35.08	4.87%
Baseline + OOD Finetune + EDA($n_{aug} = 2, \alpha_{all} = 0.1$)	**49.74**	35.86	**5.11%**
Baseline + OOD Finetune + EDA($n_{aug} = 4, \alpha_{all} = 0.1$)	48.69	35.34	2.90%
Baseline + OOD Finetune + EDA($n_{aug} = 8, \alpha_{all} = 0.1$)	49.10	35.08	3.76%
Baseline + OOD Finetune + EDA($n_{aug} = 16, \alpha_{all} = 0.1$)	49.27	35.08	4.12%

Experiment 2: Data augmentation on OOD Train Sets. EDA is known to be helpful especially for small datasets [26]. To study the influence of parameter n, the number of generated augmented context paragraphs per original paragraph, we fine-tuned our baseline model on EDA-enhanced OOD train sets, under different parameter $n_{aug} = \{1, 2, 4, 8, 16\}$. Here, all operations share same strength, $\alpha_{SR} = \alpha_{RI} = \alpha_{RS} = \alpha_{RD} = 0.1$. Table 2 summarized evaluation results on OOD validation sets. Here, we observed EDA achieved marginal performance gain for n below 2. However, as n gets larger, model performance suffers compared to no EDA case, which is likely due to overfitting of OOD data.

To further study and understand EDA's boosting effects in the QA task, we directly finetuned DistilBERT on EDA-enhanced OOD train sets, and evaluated different settings on OOD validation sets (Appendix 9.1). We noticed that EDA has led to huge performance gain when applied to OOD datasets (16.88% boost when n=16), after getting rid of the influence of large ID datasets on DistilBERT. We can infer that the ID train sets can suppress EDA's boosting efficiency, as their sizes are overwhelming compared to augmented OOD train sets, therefore making EDA operation less helpful in our previous experiment. These results can help elucidate that EDA is indeed helpful in raising model performance given limited data, and works the best when applied to small datasets.

Experiment 3: Ablation Study. So far, we have seen very encouraging results. In this section, we performed the ablation study to explore effects of each augmentation operation. Specifically, for all four operations, we fine-tuned our baseline model on single-operation-augmented OOD train sets while varying the augmentation parameter $\alpha = \{0.05, 0.1, 0.2, 0.3, 0.4, 0.5\}$, as shown in Fig. 1 (Full results are provided in Appendix 9.2). Our results suggest all four operations contribute to performance gain. It turns out SR contributes to performance gain the most (7.99% gain compared to baseline) when α_{SR}=0.3, but high SR can hurt performance, likely because replacing too many words in paragraph changed the identity of the paragraph. For RI and RS, they boost performance at small α but work worse after $\alpha > 0.05$, as performing too many swaps/insertions is equivalent to shuffling the entire order of the sentence. For RD, performance gains are more stable for different α values, possibly because the relative order of words is maintained in this operation.

Experiment 4: Optimal parameters. The natural next step is to determine what parameters work best collectively on our OOD sets. From our experience, we finetuned baseline on EDA-enhanced OOD train sets based on the optimal parameters ($\alpha_{SR} = 0.3, \alpha_{RI} = \alpha_{RS} = 0.05, \alpha_{RD} = 0.2$) from

Table 3: Validation Results for Augmented OOD Fine-tuning with Optimal Parameters

Methods	F1	EM	Performance Gain (F1)
Baseline	47.32	32.98	0
Baseline + OOD Finetune	49.43	36.39	4.46%
Baseline + OOD Finetune + EDA($n_{aug} = 2$)	49.53	35.08	4.67%
Baseline + OOD Finetune + EDA($n_{aug} = 4$)	49.70	35.67	5.03%
Baseline + OOD Finetune + EDA($n_{aug} = 8$)	**50.30**	36.39	**6.30%**
Baseline + OOD Finetune + EDA($n_{aug} = 16$)	48.88	33.51	3.30%
Baseline + OOD Finetune + EDA($n_{aug} = 32$)	49.65	35.86	4.92%

ablation study, while varying n_{aug}, as shown in Table 3. Though all results look promising, none of them surpass our current best record (F1: 51.10 by applying SR alone), as parameters that work best individually doesn't guarantee they can work best jointly. The randomness in EDA makes parameter hunting more challenging.

4.3 Analysis

Compared to other augmentation techniques, our simple data augmentation strategies are easy to implement while achieving high performance gain. They can be easily adapted and applied to broad text classification tasks, apart from the QA task. Besides, EDA demonstrates particularly strong results for small datasets. It's believed the performance gain comes from the fact that new vocabulary is introduced to the model through SR and RI, allowing the model to generalize to words in OOD test sets that are not in the training sets. Moreover, generating augmented data similar to original data introduces some degree of variation and noise, which may potentially help prevent overfitting.

However, EDA has its limitations too. For large datasets and for models that have already been pre-trained or fine-tuned on massive datasets, EDA probably doesn't help much. Besides, EDA exhibits great randomness, given changes in original data are randomly chosen. Under the same EDA setting, the variation in data quality can lead to fluctuation of model performance. Therefore, it's hard to replicate evaluation results even under the same model, same training data and same augmentation parameters, making EDA-driven performance gain more random. Such randomness is also disadvantageous to locate optimal augmentation parameters, such that the performance boost from EDA gets less controllable.

5 Re-initialization Pre-trained Layers With Warm-up

5.1 Method Description

Previously model fine-tuning is commonly built upon pre-trained weights that transfers pre-training information for better performance. However, researchers have found that re-initializing the pre-trained weights of certain layers could boost fine-tuning optimization process [11]. They reinitialized on both the pooler layers and the top L BERT Transformer blocks, as shown in Fig. 3 in Appendix 9.4. and observed performance enhancement for fine-tuning on either prediction accuracy or the mean/variance of training loss. The high-level intuition behind is that usually the higher pre-trained layers are more specified on the pre-training task and the lower pre-trained weights correspond to more general features. Considering the similarity between DistilBERT and BERT, we tried to reinitialize a proper number of transformer blocks (up to 2) and checked whether using fewer top pre-trained weights could help the model generalize well to different tasks.

Besides re-initialization, we also applied a learning rate warm-up together with re-initialization to see whether there could be a collaborative boost. Learning rate warm-up is used to reduce the dominant effect of early training examples if the dataset is highly differentiated somehow [15]. For example, the model might possibly skew toward certain features or even toward off-topic features due to the uncertainty of data shuffling. By gradually increasing the learning rate during the early training process, the model could generalize better without early over-fitting and converge faster overall.

5.2 Experimental Results

Experiment 1: Revisiting ID Fine-tuning. We firstly fine-tuned the initial DistilBERT model on the ID datasets with re-initialization technique in a default setting of 3 epochs. The validation results on OOD datasets are shown in Table 4. We could observe that with right amount of re-initialization

Table 4: Validation Results for ID Fine-tuning with New Techniques

Methods	F1	EM	Performance Gain (F1)
Baseline	47.32	32.98	0
Re-initialization with $L = 1$	47.51	31.94	0.40%
Re-initialization with $L = 1$ & Learning rate warm-up	**48.48**	31.41	2.45%
Re-initialization with $L = 2$	46.56	31.94	-1.60%
Re-initialization with $L = 2$ & Learning rate warm-up	47.57	32.20	0.53%

($L = 1$), there would be an improvement on F1 score compared to the initial baseline. This implies that proper amount of re-initialization could enable the fine-tuning to go beyond the restriction from the pre-trained tasks and generalize well to new tasks.

We also applied a learning rate warm-up for around one epoch (i.e. a warm-up ratio of 0.3 for a total epochs of 3) together with re-initialization and it's interesting to see a significant performance enhancement on F1 score. It implies the model doesn't need to step back to correct the potential bias learned in the early steps, so it could converge faster. Plus, the learning rate warm-up helps the model to treat the importance of each batch examples equally at the first glance so as to generalize better.

Experiment 2: Reinit-EDA OOD Fine-tuning. Previouly we have achieved a F1 performance gain of 7.99% based on the OOD fine-tuning with EDA. Since layer re-initialization could help the model to be more robust as shown in the previous experiment, we tried OOD fine-tuning combining EDA and re-initialization (Reinit-EDA OOD Fine-tuning) with parameters that achieve the best EDA performance. The validation results are shown in Table 9 in Appendix. Unfortunately there is no performance improvement with the combination of the two techniques. It's probably due to the fact that OOD datasets are too few to train the huge amount of parameters which are reset due to layer re-initialization, even for just one transformer block re-initialization.

6 Domain Adversarial Alignment

6.1 Method Description

Another method we adopted to tackle the domain adaptive QA task is through Adversarial Domain Alignment proposed by [27]. The high level intuition of such a method is through the adversarial game between a feature generator and a domain discriminator, where the feature generator aims to generate features that can confuse the discriminator and the discriminator aims to tell which domain a feature belongs to. Under such a minimax game, feature generator would be able to generate domain-invariant features.

For the QA Setting, we design a network leveraging DistilBERT [28] as a feature generator, $\mathcal{G}$, and a multi layer perceptron (MLP) $\mathcal{D}$ as a domain discriminator. Specifically, we use the weights from the last hidden layer of the DistilBERT model, h^{last}, through a single layer perceptron (SLP), to generate a feature representation f, where $f_{d_i} = MLP(h^{last}(x_{d_i})) = G(x_{d_i})$ and d_i denotes the domain label. The domain discriminator D takes feature f_{d_i} as an input to predict which domain f_{d_i} belongs to, by outputing a k-dimensional one-hot vector.

Our training mechanism involves a two step process. First, we optimize the feature generator with the bi-lateral objective of optimizing the task objective of QA accuracy as well as the DistilBERT-based feature generator objective of producing domain-indiscriminatable feature by minimizing the KL-divergence between different domain features.

$$\mathcal{L}_{KL}(x_{d_A}, x_{d_B}, G) = G(x_{d_B})(\log G(x_{d_B}) - G(x_{d_A})) \tag{1}$$

where x_{d_A}, x_{d_B} represents domain A and domain B respectively. The subsequent step is to optimize the domain discriminator to distinguish between the domain of the features. For this objective, we use cross entropy loss:

$$\mathcal{L}_{ce}(x_{d_A}, x_{d_B}, G, D) = -\mathbb{E}_{x_{d_A}}[\log D(G(\mathbf{x_{d_A}}))] - \mathbb{E}_{\mathbf{x_{d_B}}}[\log(1 - D(G_t(\mathbf{x_{d_A}})))] \tag{2}$$

Wasserstein Stabilization. Wasserstein Stabilization [29] is commonly used with GAN based architectures. It minimizes a reasonable and efficient approximation of the Earth Mover's distance in order to provide balance between the generator and discriminator.

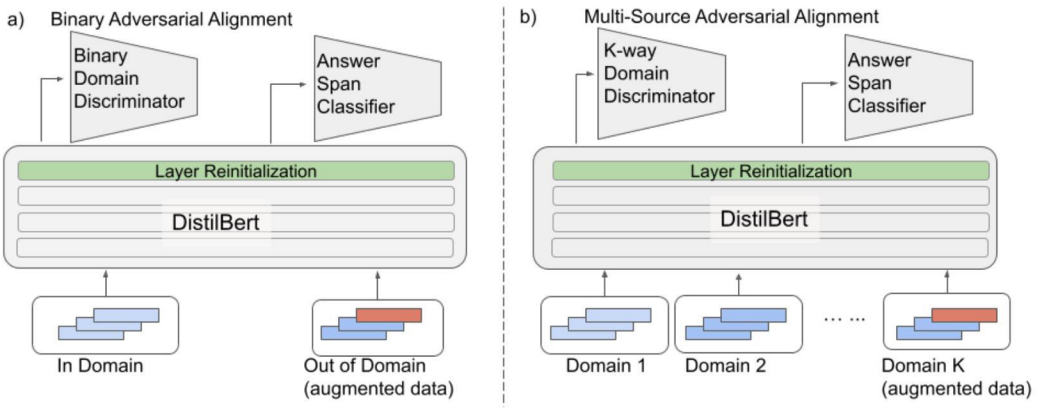

Figure 2: Model Architecture of our proposed approach. Section a) shows the model architecture for the Binary Adversarial Alignment case, where the domain discriminator is trying to distinguish between ID and OOD datasets based on the last layer of the hidden states of the DistilBERT Model. Section b) shows the model architecture for the Multi-Source Adversarial Alignment case, where the domain discriminator is trying to distinguish between the k specific datasets based on the last layer of the hidden states of the DistilBERT Model.

Table 5: Adversarial Domain Alignment Experiment and Ablation Studies

Methods	BDA F1	BDA EM	MSDA F1	MSDA EM
baseline	47.32	32.98	47.32	32.98
Adversarial	48.40	33.51	48.34	33.64
Adversarial + Aug	48.55	33.51	48.26	33.25
Adversarial + Aug + MMD	46.64	32.46	46.03	31.94
Adversarial + Aug + W-Reg	49.21	35.08	49.50	35.08
Adversarial + Aug + W-Reg + Re-init	49.97	33.77	49.64	34.55
Adversarial + Aug + W-Reg + Re-init + warmup	50.79	33.77	50.03	33.51
Adversarial + Aug + W-Reg + Re-init + warmup + EDA	50.28	35.08	**51.16**	36.65
Adversarial + Aug + W-Reg + EDA	**51.37**	**37.70**	49.92	**36.91**

6.2 Experimental Results

Our proposed method is a combination of data augmentation techniques and Wasserstein-regularized Binary Domain Alignment with OOD finetuning. This section entails the 1) model architectures and training schemes adopted for the adversarial domain alignment experiment, 2) a detailed ablation study that provides empirical soundness of each specific design choices. (Our implementation of adversarial DistilBERT referenced the official code release from Lee et al. [22]).

6.2.1 Model Architecture Details

Binary Adversarial Domain Alignment (BDA). As shown in Fig. 2, our first setup treats the task as a single-source, single-target, close-domain adaptation setting, in which we treat all of the three source ID datasets as a single source domain and all the three OOD datasets as a different domain, the target domain, to be classified by the binary source-target domain discriminator. We conducted a hyper-parameter grid search with different learning rate and batch sizes, and observe that a batch size of 32, and learning rate of 5e-5 for DistilBERT weights, and a learning rate of 5e-4 for MLP-discriminator works the best and yields 48.40 and 33.51 of F1 and EM scores respectively. We slightly increased the learning rate for the discriminator because the discriminator does not rely on pre-trained weights and would need a higher learning rate to converge to better results.

Multi-Source Adversarial Alignment (MSDA). We also experimented with directly conducting six-way domain classification on both the ID and OOD datasets, because we think that all the ID and OOD datasets might also be subjected to different domain shift. We observe that this experimental setting is also able improve the performance on the OOD evaluation dataset to F1 as 48.34, EM to 33.64, with the same aforementioned batch size and learning rate.

Table 6: Validation Results after Combining Multiple Strategies

Methods	F1	EM	Performance Gain (F1)
Baseline	47.32	32.98	0
Re-initialization + Warm-up + EDA	49.93	33.25	5.52%
B-Adversarial + Aug + W-Reg + EDA	**51.37**	**37.70**	**8.56%**
MS-Adversarial + Aug + W-Reg + Re-init + Warmup + EDA	51.16	36.65	8.11%

6.2.2 Ablation Experiments

We also conducted several ablation experiments to show the effectiveness of the specific design choices. We summarized our results in Table 5, where Adversarial means the adversarial technique, either Binary Adversarial Alignment (BDA) or Multi-Source Adversarial Alignment (MSDA), Aug refers Data Augmentation Technique described in Section 4 with $\alpha_{SR} = 0.3$, MMD stands for maximum mean discripency loss as shown in eq. 3 in Appendix, W-Reg refers to Wasserstein regularization, described in Sec 6.1, EDA refers to the OOD fine-tuning scheme as described in section 4.2, Re-init means the layer re-initialization and learning rate warm-up technique described in section 5, with last layer re-initialization and warm-up ratio=0.3.

We observe from the table that nearly all of the design choices boost the OOD question answering performance by some amount, except for MMD-loss. Specifically, Wasserstein-stabilization has historically shown success in various GAN-based works [29], and the same idea of preventing influx of large model weight can help stabilize training during the minimax game. Additionally, for the same aforementioned reasons, fine-tuning on OOD datasets can further improve the model's performance.

The idea of leveraging MMD loss for our task stems from vision domain adaptation literatures [30]. However, the loss does not work as expected when adopted for the QA task in NLP. We think that this is due to the fundamental difference between the task of QA and image or language classification, where there exists the notion of a class prior and that distributions of the same class should have minimum mean discrepancy. Nevertheless, for the task of QA, each question, context, answer, from each domain is different. Without such a prior, minimizing the mean discrepancy between different samples in a batch does not provide the same domain adaptation effect as in classification tasks.

7 Multi-Strategy Combination

Up to this point, we have thoroughly studied the mechanisms and potential limitations of EDA, re-initialization, and adversarial alignment. All of the above techniques have been proved to contribute to model performance on a varied scale. The last step is to leverage the advantages of all techniques we have explored so far to build a more robust QA system collectively. Concretely, we combined multiple strategies, by making modifications on training data (EDA), training process (Re-initialization) and model architecture (Adversarial Alignment), while applying parameters that may work best to our knowledge. Here, we report some of the top-performing model settings in Table 6, after successful integration of multiple techniques. To sum up, the best QA system (F1: 51.37 with 8.56% performance gain against baseline on OOD validation set) is achieved via applying Binary Domain Adversarial Alignment (B-Adversarial) with wasserstein-stablization, followed by fine-tuning on EDA-enhanced OOD train sets. Our QA system was eventually evaluated on OOD test set and achieved a F1 score of 58.84, and EM of 41.10.

8 Conclusions

In summary, we proposed a DistilBERT-based method that can tackle the task of Domain Adaptive Question Answering. Specifically, we propose to use a wasserstein-stablized adversarial domain alignment scheme on the DistilBERT backbone with last layer reinitialized, to train on both the data-rich ID QA datasets and data augmented OOD datasets, following a fine-tuning stage on data augmented OOD datasets. We have conducted extensive experiments to demonstrate the effectiveness of our proposed method in bringing significant performance boost for the task of domain-adaptive QA. We also conducted carefully-designed ablation studies to show the performance gain resulted from each of the proposed components. Our proposed model addresses the problem of domain-adaptive QA from various perspectives, including data, model architecture, and training scheme. In terms of future work, we can further explore more complex data augmentation techniques, such as back translation. We can also try few-shot learning, meta learning, and mixture-of-experts technique to build a even more robust QA system.

References

[1] Alexander H. Liu, Yen-Cheng Liu, Yu-Ying Yeh, and Yu-Chiang Frank Wang. A unified feature disentangler for multi-domain image translation and manipulation. *CoRR*, abs/1809.01361, 2018.

[2] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

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

[4] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In Francis Bach and David Blei, editors, *ICML 2015*, volume 37 of *Proceedings of Machine Learning Research*, Lille, France, 07–09 Jul 2015. PMLR.

[5] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. Qanet: Combining local convolution with global self-attention for reading comprehension, 2018.

[6] Ziang Xie, Sida I. Wang, Jiwei Li, Daniel Lévy, Aiming Nie, Dan Jurafsky, and Andrew Y. Ng. Data noising as smoothing in neural network language models, 2017.

[7] Oleksandr Kolomiyets, Steven Bethard, and Marie-Francine Moens. Model-portability experiments for textual temporal analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 271–276, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[8] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[9] William Yang Wang and Diyi Yang. That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2557–2563, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[10] Yann Dauphin and Samuel S Schoenholz. Metainit: Initializing learning by learning to initialize. 2019.

[11] Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q. Weinberger, and Yoav. Artzi. Revisiting few-sample bert fine-tuning. *arXiv preprint arXiv:2006.05987*, 2019.

[12] Hongyi Zhang, Yann N Dauphin, and Tengyu Ma. Residual learning without normalization via better initialization. In *International Conference on Learning Representations*, volume 3, page 2, 2019.

[13] Alex Tamkin, Trisha Singh, Davide Giovanardi, and Noah Goodman. Investigating transferability in pretrained language models. *arXiv preprint arXiv:2004.14975*, 2020.

[14] Priya Goyal, Piotr Dollár, Ross B. Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: training imagenet in 1 hour. *CoRR*, abs/1706.02677, 2017.

[15] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. In *Proceedings of the Eighth International Conference on Learning Representations (ICLR 2020)*, April 2020.

[16] Xilun Chen, Ben Athiwaratkun, Yu Sun, Kilian Q. Weinberger, and Claire Cardie. Adversarial deep averaging networks for cross-lingual sentiment classification. *CoRR*, abs/1606.01614, 2016.

[17] Xilun Chen and Claire Cardie. Multinomial adversarial networks for multi-domain text classification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1226–1240, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[18] Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. Adversarial deep averaging networks for cross-lingual sentiment classification, 2018.

[19] Yi Wu, David Bamman, and Stuart Russell. Adversarial training for relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1778–1783, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[20] Brian Xu, Mitra Mohtarami, and James R. Glass. Adversarial domain adaptation for stance detection. *CoRR*, abs/1902.02401, 2019.

[21] Yitong Li, Timothy Baldwin, and Trevor Cohn. What's in a domain? learning domain-robust text representations using adversarial training. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 474–479, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[22] 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.

[23] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. *CoRR*, abs/1706.04115, 2017.

[24] Amrita Saha, Rahul Aralikatte, Mitesh M. Khapra, and Karthik Sankaranarayanan. Duorc: Towards complex language understanding with paraphrased reading comprehension. *CoRR*, abs/1804.07927, 2018.

[25] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard H. Hovy. RACE: large-scale reading comprehension dataset from examinations. *CoRR*, abs/1704.04683, 2017.

[26] Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks, 2019.

[27] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, January 2016.

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

[29] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017.

[30] Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. Mmd gan: Towards deeper understanding of moment matching network, 2017.

Table 7: Validation Results for Augmented OOD Fine-tuning on DistilBERT($\alpha_{all} = 0.1$)

Methods	F1	EM	Performance Gain (F1)
DistilBERT + OOD Finetune	23.51	13.61	0
DistilBERT + OOD Finetune + EDA($n_{aug} = 4$)	25.18	17.28	7.10%
DistilBERT + OOD Finetune + EDA($n_{aug} = 8$)	25.12	16.49	6.85%
DistilBERT + OOD Finetune + EDA($n_{aug} = 16$)	**27.48**	17.80	**16.88%**

Table 8: Validation Results for Ablation Study by Varying Operational Strength ($n_{aug} = 2$)

Methods	F1	EM	Performance Gain (F1)
Baseline	47.32	32.98	0
Baseline + OOD Finetune	49.43	36.39	4.46%
Baseline + OOD Finetune + EDA($\alpha_{SR} = 0.05$)	49.42	35.60	4.44%
Baseline + OOD Finetune + EDA($\alpha_{SR} = 0.1$)	49.51	35.34	4.63%
Baseline + OOD Finetune + EDA($\alpha_{SR} = 0.2$)	50.52	36.91	6.76%
Baseline + OOD Finetune + EDA($\alpha_{SR} = 0.3$)	**51.10**	37.43	**7.99%**
Baseline + OOD Finetune + EDA($\alpha_{SR} = 0.4$)	48.79	34.29	3.11%
Baseline + OOD Finetune + EDA($\alpha_{SR} = 0.5$)	50.55	37.17	6.83%
Baseline + OOD Finetune + EDA($\alpha_{RI} = 0.05$)	**50.76**	36.13	**7.27%**
Baseline + OOD Finetune + EDA($\alpha_{RI} = 0.1$)	49.68	35.08	4.99%
Baseline + OOD Finetune + EDA($\alpha_{RI} = 0.2$)	49.39	35.08	4.37%
Baseline + OOD Finetune + EDA($\alpha_{RI} = 0.3$)	50.01	36.39	5.87%
Baseline + OOD Finetune + EDA($\alpha_{RI} = 0.4$)	48.55	34.82	2.60%
Baseline + OOD Finetune + EDA($\alpha_{RI} = 0.5$)	48.97	36.65	3.49%
Baseline + OOD Finetune + EDA($\alpha_{RS} = 0.05$)	**50.23**	35.86	**6.15%**
Baseline + OOD Finetune + EDA($\alpha_{RS} = 0.1$)	49.09	35.60	3.74%
Baseline + OOD Finetune + EDA($\alpha_{RS} = 0.2$)	49.37	35.86	4.33%
Baseline + OOD Finetune + EDA($\alpha_{RS} = 0.3$)	49.45	35.34	4.50%
Baseline + OOD Finetune + EDA($\alpha_{RS} = 0.4$)	49.36	36.13	4.31%
Baseline + OOD Finetune + EDA($\alpha_{RS} = 0.5$)	50.19	36.39	6.07%
Baseline + OOD Finetune + EDA($\alpha_{RD} = 0.05$)	48.66	34.03	2.83%
Baseline + OOD Finetune + EDA($\alpha_{RD} = 0.1$)	49.51	34.03	4.63%
Baseline + OOD Finetune + EDA($\alpha_{RD} = 0.2$)	**49.90**	35.60	**5.45%**
Baseline + OOD Finetune + EDA($\alpha_{RD} = 0.3$)	49.56	35.34	4.73%
Baseline + OOD Finetune + EDA($\alpha_{RD} = 0.4$)	49.66	35.08	4.95%
Baseline + OOD Finetune + EDA($\alpha_{RD} = 0.5$)	49.88	36.65	5.41%

Table 9: Validation Results for Reinit-EDA OOD Fine-tuning

Methods	F1	EM
No re-initialization ($L = 0$) + EDA	51.10	37.43
Re-initialization with $L = 1$ + EDA	48.66	31.94
Re-initialization with $L = 2$ + EDA	43.29	29.06

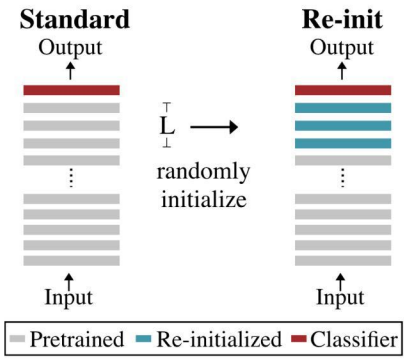

Figure 3: Re-initialization of last L layers for fine-tuning

9 Appendix

9.1 Augmented OOD Fine-tuning on DistilBERT (Table 7)

9.2 Ablation Study on EDA Operational Strength (Table 8)

9.3 Reinitialization with EDA OOD Fine-tuning (Table 9)

9.4 Re-initialization Scheme (Figure 3)

10 Maximum Mean Discrepancy

MMD. We also considered maximum mean discrepancy loss that is commonly used in vision domain adaption literature [30]. Denote by $\mathcal{H}_k$ be the reproducing kernel Hilbert space (RKHS) endowed with a characteristic kernel k. The *mean embedding* of distribution p in $\mathcal{H}_k$ is a unique element $\mu_k(P)$ such that $\mathbf{E}_{\mathbf{x}\sim P} f(\mathbf{x}) = \langle f(\mathbf{x}), \mu_k(P) \rangle_{\mathcal{H}_k}$ for all $f \in \mathcal{H}_k$. The MK-MMD $d_k(P,Q)$ between probability distributions P and Q is defined as the RKHS distance between the mean embeddings of P and Q. The squared formulation of MMD is defined as

$$d_k^2(P,Q) \left\| \mathbf{E}_P \left[\Phi_\alpha(\mathbf{x}^s) \right] - \mathbf{E}_Q \left[\Phi_\alpha\left(\mathbf{x}^t\right) \right] \right\|_{\mathcal{H}_k}^2. \tag{3}$$