QA Domain Adaptation using Data Augmentation and Contrastive Adaptation

Anonymous ACL submission

Abstract

Domain adaptation for question answering (QA) has recently shown impressive results for answering out-of-domain questions. Yet, a common challenge is to build approaches that are effective for niche domains with small text corpora. In this paper, we propose a novel framework called QADA for QA domain adaptation. QADA has two components: (1) A question generation model is used to generate synthetic question-answer samples from the target domain. Different from existing baselines, we enrich the samples via a novel pipeline for data augmentation: for questions, we introduce token-level augmentation (i.e., synonym replacement and token swapping), and, for contexts, we develop hidden-space augmentation which learns to drop context spans via a custom attentive sampling strategy. (2) The QA model is based on transformers. However, unlike existing approaches, we propose to train it via a novel attention-based contrastive adaptation. Here, we use the attention weights to sample informative tokens for discrepancy estimation that helps the QA model separate answers and generalize across source and target domain. To the best of our knowledge, our work is the first in QA domain adaptation to leverage data augmentation and attention-based contrastive adaptation. Our evaluation shows that QADA achieves considerable improvements over state-of-the-art baselines for QA domain adaptation.

1 Introduction

Question answering (QA) is the task of finding answers for a given context and a given question. QA models are typically trained using data triplets consisting of context, question and answer. In the case of extractive QA, answers are represented as subspans in the context defined by a start position and an end position, while question and context are given as running text (e.g., Chen et al., 2017; Devlin et al., 2019; Seo et al., 2016; **?**).

A common challenge in extractive QA is that QA models often suffer from performance deterioration upon deployment and thus make mistakes for user-generated inputs. The underlying reason for such deterioration can be traced back to the domain shift between training data (from the source domain) and test data (from the target domain) (Fisch et al., 2019; Hazen et al., 2019; Miller et al., 2020).

Common approaches to address domain shifts in extractive QA are as follows. One approach is to include target data samples during training (Daumé III, 2007; Kamath et al., 2020). Another approach is to generate synthetic QA samples for the target domain, which are the used additionally during training (Lee et al., 2020; Shakeri et al., 2020). However, these approaches typically require large amounts of target data. As such, they tend to be ineffective in niche domains where corpora sizes are limited (Fisch et al., 2019). Only recently, a contrastive loss has been proposed to handle domain adaptation in QA (Yue et al., 2021).

Several approaches have been used to address issues related to insufficient data and generalization in NLP tasks, yet outside of QA. For example, token-level augmentation (e.g., token swapping) has been to used for generating synthetic samples (Kobayashi, 2018; Wei and Zou, 2019; Yu et al., 2018). Another approach is data augmentation in the hidden space, which is supposed to learn more generalizable features (Chen et al., 2020, 2021; Verma et al., 2019; **?**). However, to the best of our knowledge, no work has previously used data augmentation for QA domain adaptation. For domain adaptation, there are approaches that encourage the model to learn domain-invariant features via feature discriminators (Chen et al., 2018; Lee et al., 2019; Zhang et al., 2017), or adopt contrastive adaptation to regularize the discrepancy between source and target domains (Kang et al., 2019; Yue et al., 2021). However, to the best of our knowledge, no work has integrated the attention mechanism into contrastive adaptation for QA domain adaption.

In this paper, we propose a novel framework for *QA domain adaptation* in the context of small text corpora called QADA. Our QADA framework is designed to handle domain shifts and should thus answer out-of-domain questions. QADA has two components, namely a question generation (QG) model and a QA model: (1) The QG model is used to generate synthetic question-answer samples from the target domain. Here, we integrate novel pipeline for data augmentation to enrich the samples. For questions, we introduce a token-level augmentation (i.e., synonym replacement and token swapping), and, for contexts, we develop a hidden-space augmentation which learns to drop context spans via an attentive sampling strategy. (2) The QA model is implemented via transformers. Here, we propose to train the QA model via a novel attention-based contrastive adaptation. Specifically, we use the attention weights to sample informative tokens that help the QA model separate answers and generalize across source and target domain.

Main contributions of our work are:[1]

1. We propose a novel framework called QADA for domain adaptation in QA. QADA aims at answering out-of-domain question and should thus handle the domain shift upon deployment. Moreover, QADA is specifically designed for niche domains with small text corpora.
2. To the best of our knowledge, QADA is the first work in QA domain adaptation that (i) leverages data augmentation on token-level and in the hidden space; (ii) integrates an attention-based feature sampling in contrastive adaptation.
3. We demonstrate the effectiveness of QADA in settings where corpora are of limited size. Here, QADA achieves a considerably better performance than state-of-the-art baselines for QA domain adaptation.

2 Related Work

Extractive QA has achieved impressive progress in recent years (Devlin et al., 2019; Kratzwald et al., 2019; Lan et al., 2019; Zhang et al., 2020). Yet the accuracy of QA models can drop drastically under domain shifts; that is, when deployed in an unseen domain that differs from the training distribution (Fisch et al., 2019; Talmor and Berant, 2019).

To overcome the above challenge, various approaches for QA domain adaptation have been proposed, which can be categorized as follows. (1) (Semi-)supervised adaptation uses partially labeled data from the target distribution for training (Kratzwald and Feuerriegel, 2019; Yang et al., 2017). (2) Unsupervised adaptation has access to context and question information from the target domain, whereas answers are unavailable (Cao et al., 2020; Chung et al., 2018). (3) Unsupervised adaptation with question generation refers to settings where only context paragraphs in the target domain are available, but QA samples have to be generated separately to train the QA model (Shakeri et al., 2020; Yue et al., 2021). In this paper, we focus on the third category and study the problem of QA domain adaptation via question generation.

Question generation for QA: Several approaches for QG have been developed to generate synthetic questions in an end-to-end fashion (i.e., seq2seq) (Du et al., 2017; Sun et al., 2018). Combining both QG and QA can leverage the similarity among both tasks and thus improve the QA performance (Golub et al., 2017; Tang et al., 2017, 2018). Advanced QG models build upon hierarchical variational autoencoders and transformers (Lee et al., 2020; Shakeri et al., 2020). For example, QAGen-T5 (Raffel et al., 2019; Yue et al., 2021) extends two T5 transformers for generating question-answer pairs. We later use the aforementioned QG models as part of our baselines.

Data augmentation for NLP: Data augmentation for NLP aims at improving the language understanding with diverse data samples. One approach is to apply token-level augmentation and enrich the training data with simple techniques (e.g., synonym replacement, token swapping, etc.) (Wei and Zou, 2019) or custom heuristics (McCoy et al., 2019). Alternatively, augmentation can be done in the hidden space of the underlying model (Chen et al., 2020). For example, one can drop (i.e., cutoff) partial spans hidden layers in hidden space, which aids generalization performance under distributional shifts (Chen et al., 2021) but in NLP tasks outside of QA. To the best of our knowledge, we are the first to propose an augmentation pipeline for QA data in which both token-level and hidden-space augmentation are combined.

Contrastive learning for domain adaptation: Contrastive learning is used to minimize distances of same-class samples and maximize discrepancy among classes (Hadsell et al., 2006). For this, different metrics are adopted to measure pair-wise

[1] The code for our QADA framework is in the supplements. Upon publication, we will make it publicly available.

distances (e.g., triplet loss) or domain distances with the maximum mean discrepancy (Cheng et al., 2016; Schroff et al., 2015). Contrastive learning can also be used for domain adaptation by reducing the domain discrepancy: this “pulls together” intra-class features and “pushes apart” inter-class representations. Here, several applications are in computer vision Kang et al. (2019). In QA domain adaptation, contrastive learning was applied with averaged token features in order to separate answer tokens and minimize the discrepancy between source and target domain (Yue et al., 2021). However, our work is different in that we introduce a novel *attention-based* sampling strategy for contrastive adaptation and in that we propose a finer contrastive loss (see details in Section 4.3).

3 Setup

We consider the following problem setup. As in (Shakeri et al., 2020; Yue et al., 2021), we study domain adaptation for QA with question generation. We further assume a setting with limited text corpora. For notation, we denote the QG model via $\boldsymbol{f}_{\text{qg}}$, and the QA model via $\boldsymbol{f}$.

Training: Our research focuses on question answering under domain shift. Let $\boldsymbol{\mathcal{D}}_s$ denote the source domain, and let $\boldsymbol{\mathcal{D}}_t$ denote the (different) target domain. Then, labeled data from the source domain can be used for training, while, upon deployment, it should perform well on the data from the target domain. Specifically, training is two-fold: we first train a QG model and, following this, a QA model. The input data to each is as follows:

- *Labeled QA data*: Training data is provided by labeled QA data $\boldsymbol{X}_s$ from the source domain $\boldsymbol{\mathcal{D}}_s$. Here, each sample $\boldsymbol{x}_s^{(i)} \in \boldsymbol{X}_s$ is a triplet comprising a question $\boldsymbol{x}_{s,q}^{(i)}$, a context $\boldsymbol{x}_{s,c}^{(i)}$, and an answer $\boldsymbol{x}_{s,a}^{(i)}$. As we consider extractive QA, the answer is represented by the start and end position in the context.

- *Unlabeled target contexts*: We assume partial access to data from the target domain $\boldsymbol{\mathcal{D}}_t$, that is, only unlabeled contexts. The contexts are used later for question generation. Formally, we refer to the contexts via $\boldsymbol{x}_{t,c}^{(i)}$ (with $\boldsymbol{x}_t^{(i)} \in \boldsymbol{X}_t^{'}$ where $\boldsymbol{X}_t^{'}$ is the data from the target domain). We further assume that the corpus of target contexts is of limited size (i.e., $|\boldsymbol{X}_t^{'}|$ is limited).

Objective: Upon deployment, our goal is to maximize the model performance on $\boldsymbol{X}_t$ in the target domain $\boldsymbol{\mathcal{D}}_t$. Mathematically, this corresponds to the optimization problem

$$\boldsymbol{f}^* = \arg\min_{\boldsymbol{f}} \mathcal{L}_{\text{ce}}(\boldsymbol{f}, \boldsymbol{X}_t), \qquad (1)$$

where $\mathcal{L}_{\text{ce}}$ is the cross entropy loss and $\boldsymbol{f}$ represents the QA model.

4 The QADA Framework

4.1 Overview

Our proposed QADA framework has two major components (see Figure 1): (1) **question generation** model with augmentation to enrich learning with synthetic target samples; and a (2) **QA model**. The QA model is trained with attention-based contrastive adaptation.

To address a domain shift upon deployment, we use the two components for QA domain adaptation as follows. In the first component (Sec. 4.2), we use the unlabeled target data $\boldsymbol{X}_t^{'}$ and add synthetic labels (QA pairs) via a QG model. Here, we enrich the set of synthetic data via data augmentation. In the second component (Sec. 4.3), we train the QA model using both the source and the synthetic target data with our attention-based contrastive adaptation, such that the learned features should generalize across the source domain and the target domain.

1. *Question generation (with augmentation)*: We use the unlabeled target contexts $\boldsymbol{X}_t^{'}$, based on which we build synthetic target data $\boldsymbol{X}_t$. Formally, we generate QA pairs $\boldsymbol{x}_{t,q}^{(i)}, \boldsymbol{x}_{t,a}^{(i)}$ using $\boldsymbol{f}_{\text{qg}}$. Each sample $\boldsymbol{x}_t^{(i)} \in \boldsymbol{X}_t$ now contains the original context and a pair of a synthetic question and a synthetic answer. We additionally apply data augmentation to enrich the target data.

2. *QA model (with attention-based contrastive adaptation)*: The QA model $\boldsymbol{f}$ takes the context and question as input and outputs the answer, i.e., it predicts $\boldsymbol{x}_a^{(i)} = \boldsymbol{f}(\boldsymbol{x}_c^{(i)}, \boldsymbol{x}_q^{(i)})$. We train the QA model $\boldsymbol{f}$ with the source data $\boldsymbol{X}_s$ from the source domain $\boldsymbol{\mathcal{D}}_s$ *and* the synthetic target data $\boldsymbol{X}_t$ from the previous step. We impose regularization on the answer extraction and further minimize the discrepancy between source and target domain, so that the learned features generalize well to the target domain.

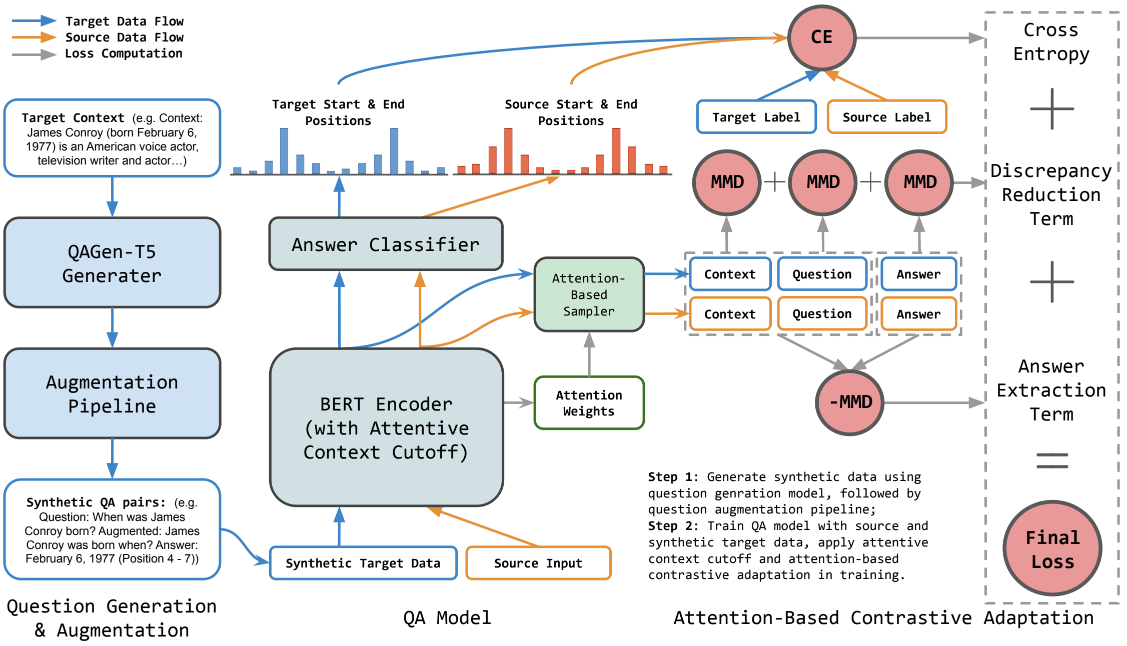
Target Data Flow
Source Data Flow
Loss Computation
Target Context (e.g. Context: James Conroy (born February 6, 1977) is an American voice actor, television writer and actor…)
QAGen-T5 Generater
Augmentation Pipeline
Synthetic QA pairs: (e.g. Question: When was James Conroy born? Augmented: James Conroy was born when? Answer: February 6, 1977 (Position 4 - 7))
Question Generation & Augmentation
Target Start & End Positions
Source Start & End Positions
Answer Classifier
BERT Encoder (with Attentive Context Cutoff)
Attention-Based Sampler
Attention Weights
Synthetic Target Data
Source Input
QA Model
CE
Target Label
Source Label
MMD
MMD
MMD
Context
Question
Answer
Context
Question
Answer
-MMD
Cross Entropy
Discrepancy Reduction Term
Answer Extraction Term
Final Loss
Step 1: Generate synthetic data using question genration model, followed by question augmentation pipeline;
Step 2: Train QA model with source and synthetic target data, apply attentive context cutoff and attention-based contrastive adaptation in training.
Attention-Based Contrastive Adaptation

Figure 1: Overview of our proposed QADA framework. A question generation model is used to generate QA pairs, which are then augmented to enrich the synthetic data. The QA model is trained with attentive contrastive loss designed to improve accuracy on the target domain upon deployment.

4.2 Question Generation (with Augmentation)

QG model: We adopt QAGen-T5 (Yue et al., 2021) and use two separate T5 transformer models (Raffel et al., 2019) for question generation and answer generation. As in (Shakeri et al., 2020; Yue et al., 2021), QAGen-T5 takes a context $\boldsymbol{x}_c$ as input. Then, it first generates a question $\boldsymbol{x}_q$, and then generates an answer $\boldsymbol{x}_a$ conditioned on $\boldsymbol{x}_c$ and $\boldsymbol{x}_q$.

QAGen-T5 is trained on the source dataset with a negative log-likelihood loss using separate output probabilities in each step, i.e.,

$$\mathcal{L}_{\text{qg}}(\boldsymbol{X}) = \sum_{i=1}^{|\boldsymbol{X}|} -\log p_{\theta_{\text{t5}}}\left(\boldsymbol{x}_{\text{out}}^{(i)} \,\middle|\, \boldsymbol{x}_{\text{in}}^{(i)}\right), \qquad (2)$$

where $\boldsymbol{x}_{in}^{(i)}$ and $\boldsymbol{x}_{out}^{(i)}$ refer to the input and output of each QG step, respectively. To select diverse and consistent QA pairs from QAGen-T5, we adopt LM filtering (Shakeri et al., 2020) to select the best k QA pairs for each context (we use $k = 5$ with 10k context paragraphs in our experiments).

Data augmentation: We design a data augmentation pipeline to further enrich the training data based on the generated QA pairs. The augmentation pipeline is divided into two parts (see Figure 2): (i) question augmentation and (ii) context augmentation. The former is done via token-level augmentation, while the latter is done via hidden-space augmentation as described below.

Question augmentation: To perform augmentation of questions, we use synonym replacement and random swaps on certain proportion of tokens. Synonyms are randomly picked from a synonym dictionary (i.e., WordNet; Miller, 1995), while swapping is applied on two randomly chosen words in the question. This simple augmentation is introduced in order to provide syntactically diverse questions. At the same time, by adding noise, it should encourage the QA model to capture robust information in questions (see Wei and Zou, 2019). We control the question augmentation by a token augmentation ratio as a hyperparameter. It determines the percentage of tokens within questions that are changed.

We considered to use the above token-level augmentation also for answers and/or contexts but eventually discarded this idea: (1) token-level augmentation undermines the original text style and the underlying domain characteristics; (2) token changes for contexts are likely to cause shifts among answer spans.

Context augmentation: For context, we adopt augmentation in the hidden space instead of token-level augmentation. Here, we propose to use an attentive context cutoff in the hidden space. Specifically, we zero out sampled context spans in the hidden space after each transformer layer in the QA model. This is illustrated in Figure 2, where all hidden states in the selected span along the input length are dropped (i.e., setting values to zero as shown by the white color). Thereby, our cutoff forces the QA model to attend to context information that is particularly relevant across all input positions and thus hinders it from learning redundant domain information.

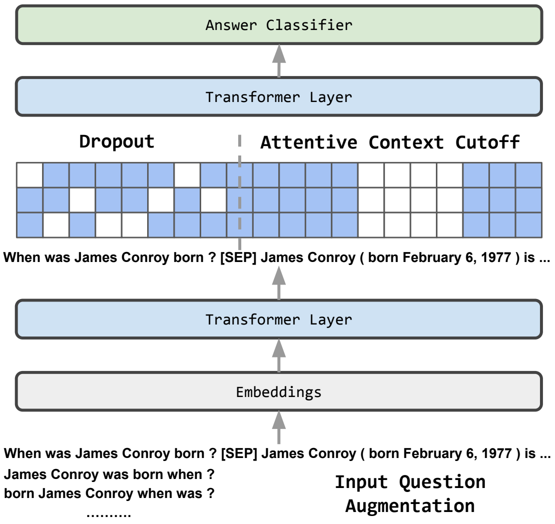
Answer Classifier
Transformer Layer
Dropout
Attentive Context Cutoff
When was James Conroy born ? [SEP] James Conroy (born February 6, 1977) is ...
Transformer Layer
Embeddings
When was James Conroy born ? [SEP] James Conroy (born February 6, 1977) is ...
James Conroy was born when ?
born James Conroy when was ?
....
Input Question Augmentation

Figure 2: Overview of the proposed augmentation pipeline in QADA. Before feeding the input to QA model, we perform (1) token-level augmentation for questions and (2) hidden-space augmentation for contexts. The latter is done via an attentive context cutoff that is performed in the hidden space after every transformer layer.

Formally,our attentive sampling strategy learns to select cutoff spans: we compute a probability distribution and sample a start position using the attention weights $A \in \boldsymbol{R}^{H \times L_c \times L_c}$ in the context span from the previous transformer layer in the QA model. The probability of the i-th position as start position is computed via

$$p_i = \sigma\left(\frac{1}{H}\sum_{j}^{H}\left(\sum_{k}^{L_c} A_{j,k}\right)\right)_i, \tag{3}$$

where H is the number of attention heads, L_c is the context length, and σ denotes the softmax function. We compute the softmax of the averaged weights. We introduce a cutoff context ratio as a hyperparameter. It determines the length of the cutoff (as compared to length of the original context).

Eventually, the above procedure of question and context augmentation should improve the model capacity in question understanding and “cut off” spans in the hidden space. This thus encourages the QA model to reduce redundancy and capture relevant information for QA, i.e., from other positions using self-attention.

4.3 QA Model (with Attention-Based Contrastive Adaptation)

For our QA model, we use BERT-QA, a transformer with the self-attention (Devlin et al., 2019). To train it, we integrate a tailored attention-based contrastive adaptation. The idea is to measure the discrepancy between class features and then reduce the intra-class discrepancy between source and target domains. Unlike domain adaptation in previous work (Kang et al., 2019; Yue et al., 2021), we consider the three sets of tokens (i.e., context, question, and answer) as different classes.

Loss: We perform contrastive adaptation to reduce the intra-class distances between source and target domains. We also maximize the inter-class distances between answer tokens and the other tokens to improve answer extraction. For a mixed batch $\boldsymbol{X}$ with $\boldsymbol{X}_s$ and $\boldsymbol{X}_t$ representing the subset of source and target samples, the loss is

$$\begin{aligned}\mathcal{L}_{\text{QADA}} &= -\mathcal{D}^{\text{MMD}}(\boldsymbol{X}_a, \boldsymbol{X}_{cq}) \\ &\quad + \lambda_{\text{type}} \sum_{cl} \mathcal{D}^{\text{MMD}}(\boldsymbol{X}_{s,cl}, \boldsymbol{X}_{t,cl}) \quad \text{with} \\ \mathcal{D}^{\text{MMD}} &= \frac{1}{|\boldsymbol{X}_s||\boldsymbol{X}_s|}\sum_{i=1}^{|\boldsymbol{X}_s|}\sum_{j=1}^{|\boldsymbol{X}_s|} k(\phi(\boldsymbol{x}_{\text{s}}^{(i)}), \phi(\boldsymbol{x}_{\text{s}}^{(j)})) \\ &\quad + \frac{1}{|\boldsymbol{X}_t||\boldsymbol{X}_t|}\sum_{i=1}^{|\boldsymbol{X}_t|}\sum_{j=1}^{|\boldsymbol{X}_t|} k(\phi(\boldsymbol{x}_{\text{t}}^{(i)}), \phi(\boldsymbol{x}_{\text{t}}^{(j)})) \\ &\quad - \frac{2}{|\boldsymbol{X}_s||\boldsymbol{X}_t|}\sum_{i=1}^{|\boldsymbol{X}_s|}\sum_{j=1}^{|\boldsymbol{X}_t|} k(\phi(\boldsymbol{x}_{\text{s}}^{(i)}), \phi(\boldsymbol{x}_{\text{t}}^{(j)})),\end{aligned} \tag{4}$$

where $\boldsymbol{X}_a$ represents answer tokens in $\boldsymbol{X}$, $\boldsymbol{X}_{cq}$ represents all context and question tokens in $\boldsymbol{X}$, $\boldsymbol{X}_{s,cl}$ and $\boldsymbol{X}_{t,cl}$ denotes tokens of class cl (context, question or answer) in the source batch $\boldsymbol{X}_s$ and target batch $\boldsymbol{X}_t$, respectively. Here, λ_{type} is a hyperparameter. Moreover, $\mathcal{D}^{\text{MMD}}$ computes the discrepancy using our scheme below. In $\mathcal{L}_{\text{ours}}$, the first term maximizes the distance of answer tokens to other input tokens, thereby improving answer extraction (*extraction term*), while the second terms reduce the intra-class discrepancy (*discrepancy term*).

MMD: The maximum mean discrepancy (MMD) computes the proximity between probabilistic distributions in the reproducing kernel

Hilbert space $\mathcal{H}$ using drawn samples (Gretton et al., 2012). In our implementation, we compute the discrepancy between the source and target distributions with empirical kernel mean embeddings using sampled token features, where the Gaussian kernel k is adopted to estimate the distance in $\mathcal{H}$.

In previous research (Yue et al., 2021), the MMD distance was computed with the empirical kernel mean embeddings using a feature mapping ϕ (i.e., BERT encoder), where $\boldsymbol{x}_{\mathrm{s}}^{(i)}$ is the i-th sample from the source batch $\boldsymbol{X}_s$, and $\boldsymbol{x}_{\mathrm{t}}^{(j)}$ is the j-th sample from the target batch $\boldsymbol{X}_t$. However, simply using ϕ as in previous work (Yue et al., 2021) would return the *averaged* feature of all relevant tokens in the sample rather than *informative* class information (i.e., features at the decision boundary which are "hard" to predict).

Instead, we design an attention-based sampling strategy. First, we leverage the attention weights $A \in \boldsymbol{R}^{H \times L_{\boldsymbol{x}} \times L_{\boldsymbol{x}}}$ from the input $\boldsymbol{x}$ of the encoder of the QA model. Based on this, we compute a probability distribution using the softmax σ and sample an index from it. The corresponding feature of the index from the QA encoder is used as the class feature, i.e.,

$$\phi(\boldsymbol{x}) = \boldsymbol{f}_{\mathrm{enc}}(\boldsymbol{x})_i \ \text{ with } \ i \sim \sigma\Big(\frac{1}{H}\sum_{j}^{H}\big(\sum_{k}^{L_{\boldsymbol{x}}} A_{j,k}\big)\Big), \tag{5}$$

where $\boldsymbol{f}_{\mathrm{enc}}$ is the encoder of the QA model. As a result, features are sampled proportionally to the attention weights. This should reflect more representative information of the token class for discrepancy estimation. We apply the aforementioned attention-based sampling to both context and question features. For answers, we use feature averaging, as we expect all answer tokens to be equally important.

Illustration: We visualize an illustrative QA sample in Figure 3 to explain the advantage of our attention-based sampling for domain discrepancy estimation. We visualize all token features and then examine the extraction term from Eq. 4. We further show the feature mapping ϕ from CAQA (Yue et al., 2021) returning the *average feature*. In contrast, our ϕ focuses on the estimation of more informative distances. As a result, our proposed attention-based sampling strategy is more likely to sample "harder" context tokens. These are closer to the decision boundary, as such token positions have higher weights in A. Owing to this choice of ϕ, QADA changes how we measure the answer-context discrepancy, and, therefore, is more effective in separating answer tokens from other tokens.

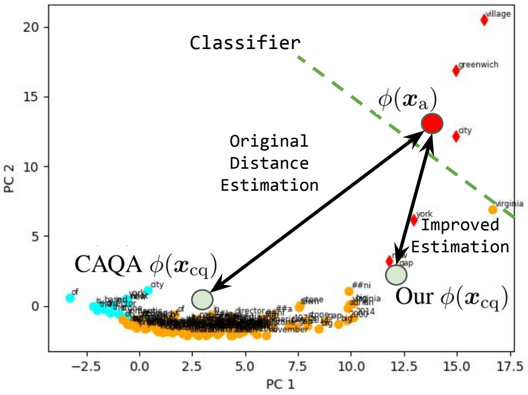

Figure 3: Illustration of QA samples in feature space. The token features are obtained from the last layer in BERT-QA and visualized using principle component analysis (PCA). Question tokens are in cyan, context tokens in orange, and answer tokens in red. Shown are the feature mappings ϕ from CAQA (Yue et al., 2021) vs. ours.

4.4 Learning Algorithm

We incorporate the contrastive adaptation loss from Eq. 4 into the original training objective. This gives our overall loss

$$\mathcal{L} = \mathcal{L}_{\mathrm{qa}} + \lambda_{\mathrm{token}}\mathcal{L}_{\mathrm{QADA}}, \tag{6}$$

where $\mathcal{L}_{\mathrm{qa}}$ denotes the cross entropy loss for training the QA model and λ_{token} is a hyperparameter acting as a scaling factor for the contrastive term.

5 Experiments

Datasets: We use the following datasets (see Appendix A for details):

- For the *source dataset* $\mathcal{D}_s$, we use SQuAD v1.1 (Rajpurkar et al., 2016).

- For *target dataset* $\mathcal{D}_t$, we follow (Yue et al., 2021) and select four datasets: HotpotQA (Yang et al., 2018), Natural Questions (Kwiatkowski et al., 2019), SearchQA (Dunn et al., 2017), and TriviaQA (Joshi et al., 2017) from MRQA 2019 (Fisch et al., 2019). This selection makes our results comparable with other works in QA domain adaptation that also leverage QG (e.g., Lee et al., 2020; Shakeri et al., 2020).

Recall that we are interested in settings with corpora of limited size. Hence, we proceed as follows:

Approach	Training data	**HotpotQA**	**NaturalQ.**	**SearchQA**	**TriviaQA**
		EM / F1	EM / F1	EM / F1	EM / F1
Performance on target dev set w/o domain adaptation					
Source only	SQuAD	41.38/59.15	46.61/60.82	21.55/30.68	50.92/60.56
Performance on target dev set w domain adaptation					
HCVAE	SQuAD + 10k target contexts	39.47/55.60	37.12/51.17	17.41/24.24	45.66/55.28
AQGen	SQuAD + 10k target contexts	45.64/60.81	44.81/58.81	31.75/39.34	51.74/60.75
QAGen2S	SQuAD + 10k target contexts	45.43/60.48	44.68/58.17	30.87/38.06	50.60/59.67
QAGen-T5	SQuAD + 10k target contexts	46.70/61.99	46.99/61.12	32.02/38.58	54.63/63.00
CAQA	SQuAD + 10k target contexts	46.98/62.09	47.66/61.53	35.57/42.86	55.36/63.57
QADA (ours)	SQuAD + 10k target contexts	**47.94/63.45**	**49.03/62.60**	**36.58/44.03**	**55.58/63.72**
Performance on target dev set w/ supervised training					
Supervised (w/ target data)	10k target samples	49.52/66.56	54.88/68.10	60.20/66.96	54.63/60.73
Supervised (w/ target data)	All target samples	57.96/74.76	67.08/79.02	71.54/77.77	64.51/70.27

Table 1: Results of QA domain adaptation on target datasets.

From the target datasets, we randomly select 10k context paragraphs that provide the unlabeled target contexts $\boldsymbol{x}_{t,c}^{(i)}$ for question generation. In QADA, QA pairs are subject to LM filtering, so that a maximum 5 questions per context are kept. For all baselines, the same number of context paragraphs is used with 5 QA pairs for each context.

Baselines: Baselines are chosen that allow for QA domain adaptation with question generation, so that the same data as in QADA are leveraged. For this, we combine models for QG and QA. For QG, we use hierarchical conditional VAE (HCVAE) (Lee et al., 2020), AQGen & QAGen two-step (QAGen2S) (Shakeri et al., 2020), QAGen-T5 & CAQA (Yue et al., 2021). For QA, we use the uncased base BERT-QA (Devlin et al., 2019). Details on the baselines are in Appendix B.

Implementation: To limit the data size, we use our proposed augmentation pipeline to generate one augmented question per QA pair. The other hyperparameters were tuned, that is, by empirically searching for the best combination of all hyperparameters. Details are in appendix C.

Evaluation: To evaluate the predictions, we follow (Lee et al., 2020; Shakeri et al., 2020; Yue et al., 2021) and assess the exact matches (EM) and the F1 score on the test data. All evaluations are performed on the dev sets. Results are reported for the best combination of hyperparameters.

6 Experimental Results

6.1 Overall Performance of Domain Adaptation

Our main results for domain adaptation are in Table 1. We distinguish three major groups: (1) *Without domain adaptation.* Here, we report a naïve baseline called "source only" for which train BERT-QA isolely on SQuAD. (2) *With domain adaptation.* This refers to the above baselines where domain adaptation is achieved by combining both BERT-QA and a QG model. These are also trained jointly using both SQuAD and 10k target contexts. This group also includes QADA. (3) *With supervised learning.* These are trained with target samples that, in an actual production setting, would be unavailable. Hence, this reflects an "upper bound".

Overall, the domain adaptation baselines are outperformed by our QADA across all target datasets. Hence, this confirms the effectiveness of the proposed framework using both data augmentation and attention-based contrastive adaptation. In addition, we observe the following: (1) QADA has performance improvements over CAQA by up to 2.87 % and in 2.73 % in EM and F1 score, respectively. (2) QADA achieves almost a similar magnitude as supervised learning (with target data) for three datasets. For SearchQA, our explanation for the gap is that this is likely caused by the large domain discrepancy and long context paragraphs. (3) QADA has a comparatively small improvement for TriviaQA, but outperforms supervised learning with 10k samples. As a potential reason, this may indicate a small domain discrepancy between SQuAD and TriviaQA, as the adaptation performance may be limited due to the small domain variation (see (Yue et al., 2021)).

Appendix D reports additional results for a sensitivity analysis with different hyperparameters. This confirms the robustness of our results. Appendix E provides a qualitative analysis of QA samples.

6.2 Sensitivity Analysis for Token Augmentation Ratio

Our QADA uses question augmentation where a token augmentation ratio determines the percentage

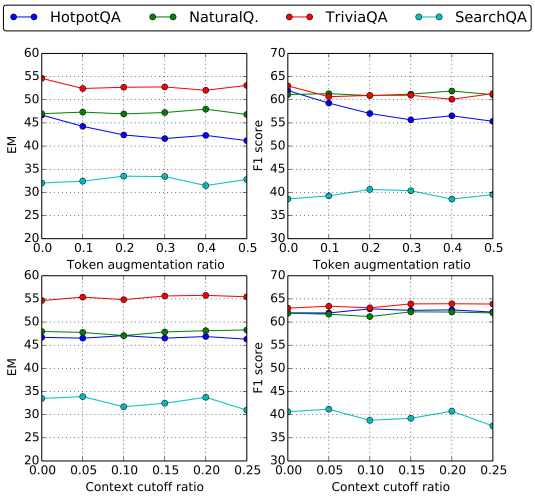

Figure 4: Sensitivity analysis of QA performance for different token augmentation ratios (top) and different context cutoffs (bottom).

of tokens that are changed. Figure 4 compares different token augmentation ratios from 0 to 0.5.

Overall, we observe some variation but, importantly, the performance remains fairly robust for different nonzero ratios. Moreover, we find comparatively large improvements for NaturalQuestions and SearchQA. In the case of SearchQA, the EM increases from 32.02 to 33.50, which corresponds to an improvement by 4.6 %; the F1 score increases from 38.58 to 40.64, which corresponds to an improvement by 5.3 %. In contrast, there are no improvements for HotpotQA and TriviaQA, thereby suggesting that effectiveness of question augmentation may be limited as the discrepancy between source and target domain is too small (cf. (Yue et al., 2021) for a discussion of the datasets).

6.3 Sensitivity Analysis for Context Cutoff Ratio

Now we study the context cutoff ratio (see Figure 4). The context cutoff ratio determines the relative context length of the span that we zero out in the hidden-space augmentation for contexts based on the attention. The range for context cutoff ratio in our experiments is from 0 to 0.25.

Across different context cutoff ratios, we again observe that the performance is – to a large extent – robust. We further find that tuning context cutoff ratios can be particularly powerful for target datasets with comparatively small discrepancy w.r.t. the source dataset. To show this, we use TriviaQA and compare our QADA from above (with a nonzero context cutoff ratio) against an implementation where the context cutoff ratio is set to zero. A nonzero ratio leads to improvements by 1.14 % and 0.95 % in EM and F1, respectively. This suggests that the context cutoff ratio improves the overall capability of the system in retrieving context information.

	Extraction term only	Extraction + discrepancy
Dataset	EM / F1	EM / F1
HotpotQA	47.94/63.26	**47.94/63.45**
NaturalQ.	**49.03/62.60**	48.61/61.81
SearchQA	34.10/40.76	**36.58/44.03**
TriviaQA	55.05/63.31	**55.58/63.72**

Table 2: Results of QA adaptation with the extraction term and the optional discrepancy term.

6.4 Ablation Study for Extraction vs. Discrepancy Term

We now seek to understand the improvements due to our attention-based contrastive adaptation. Here, we perform an ablation study for the extraction term and the discrepancy term (see Eq. 4). We repeat the experiments in two variants (Table 2): attentive-based contrastive adaptation with (1) only the extraction term and (2) both terms.

We find improvements of different magnitude when additionally including the discrepancy term (with exception of a slight performance deterioration for NaturalQ.). The highest improvement using the discrepancy term is obtained for SearchQA. As a result, both EM and F1 score are improved by 7.3 % and 8.0 %, respectively. The results suggest that the discrepancy term is especially effective for settings with a comparatively large discrepancy between source/target datasets. In sum, the results imply that the contrastive adaptation can lead to improvements in discrepancy reduction.

7 Conclusion

In this paper, we propose a novel framework called QADA for QA domain adaptation in the context of small text corpora. QADA introduces: (1) question generation with additional data augmentation to generate and enrich synthetic QA data; and (2) an attention-based contrastive adaptation for training QA models to learn domain-invariant features that generalize across source and target domain. Our experiments demonstrate the effectiveness of QADA: it achieves a superior performance over state-of-the-art baselines in QA domain adaptation.

References

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic QA corpora generation with roundtrip consistency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6168–6173, Florence, Italy. Association for Computational Linguistics.

Yu Cao, Meng Fang, Baosheng Yu, and Joey Tianyi Zhou. 2020. Unsupervised domain adaptation on reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7480–7487.

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, Vancouver, Canada. Association for Computational Linguistics.

Jiaao Chen, Dinghan Shen, Weizhu Chen, and Diyi Yang. 2021. HiddenCut: Simple data augmentation for natural language understanding with better generalizability. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4380–4390, Online. Association for Computational Linguistics.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger. 2018. Adversarial deep averaging networks for cross-lingual sentiment classification. *Transactions of the Association for Computational Linguistics*, 6:557–570.

De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng. 2016. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1335–1344.

Yu-An Chung, Hung-Yi Lee, and James Glass. 2018. Supervised and unsupervised transfer learning for question answering. 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 1585–1594, New Orleans, Louisiana. Association for Computational Linguistics.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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. Association for Computational Linguistics.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352, Vancouver, Canada. Association for Computational Linguistics.

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

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 1–13, Hong Kong, China. Association for Computational Linguistics.

David Golub, Po-Sen Huang, Xiaodong He, and Li Deng. 2017. Two-stage synthesis networks for transfer learning in machine comprehension. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 835–844, Copenhagen, Denmark. Association for Computational Linguistics.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. 2012. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1735–1742. IEEE.

Timothy J Hazen, Shehzaad Dhuliawala, and Daniel Boies. 2019. Towards domain adaptation from limited data for question answering using deep neural networks. *arXiv preprint arXiv:1911.02655*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of* *the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Amita Kamath, Robin Jia, and Percy Liang. 2020. Selective question answering under domain shift. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5684–5696, Online. Association for Computational Linguistics.

Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. 2019. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4893–4902.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. 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 452–457, New Orleans, Louisiana. Association for Computational Linguistics.

Bernhard Kratzwald, Anna Eigenmann, and Stefan Feuerriegel. 2019. RankQA: Neural question answering with answer re-ranking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6076–6085, Florence, Italy. Association for Computational Linguistics.

Bernhard Kratzwald and Stefan Feuerriegel. 2019. Putting question-answering systems into practice: Transfer learning for efficient domain customization. *ACM Trans. Manage. Inf. Syst.*, 9(4).

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. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

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

Dong Bok Lee, Seanie Lee, Woo Tae Jeong, Donghwan Kim, and Sung Ju Hwang. 2020. Generating diverse and consistent QA pairs from contexts with information-maximizing hierarchical conditional VAEs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 208–224, Online. Association for Computational Linguistics.

Seanie Lee, Donggyu Kim, and Jangwon Park. 2019. Domain-agnostic question-answering with adversarial training. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 196–202, Hong Kong, China. Association for Computational Linguistics.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

John Miller, Karl Krauth, Benjamin Recht, and Ludwig Schmidt. 2020. The effect of natural distribution shift on question answering models. In *International Conference on Machine Learning*, pages 6905–6916. PMLR.

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. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Siamak Shakeri, Cicero Nogueira dos Santos, Henghui Zhu, Patrick Ng, Feng Nan, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. End-to-end synthetic data generation for domain adaptation of question answering systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5445–5460, Online. Association for Computational Linguistics.

Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. 2018. Answer-focused and position-aware neural question generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3930–3939, Brussels, Belgium. Association for Computational Linguistics.

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

Duyu Tang, Nan Duan, Tao Qin, Zhao Yan, and Ming Zhou. 2017. Question answering and question generation as dual tasks. *arXiv preprint arXiv:1706.02027*.

Duyu Tang, Nan Duan, Zhao Yan, Zhirui Zhang, Yibo Sun, Shujie Liu, Yuanhua Lv, and Ming Zhou. 2018. Learning to collaborate for question answering and asking. 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 1564–1574, New Orleans, Louisiana. Association for Computational Linguistics.

Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. 2019. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447. PMLR.

Jason Wei and Kai Zou. 2019. 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. Association for Computational Linguistics.

Zhilin Yang, Junjie Hu, Ruslan Salakhutdinov, and William Cohen. 2017. Semi-supervised QA with generative domain-adaptive nets. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1040–1050, Vancouver, Canada. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. *arXiv preprint arXiv:1804.09541*.

Zhenrui Yue, Bernhard Kratzwald, and Stefan Feuerriegel. 2021. Contrastive domain adaptation for question answering using limited text corpora. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9575–9593, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970, Vancouver, Canada. Association for Computational Linguistics.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2020. Retrospective reader for machine reading comprehension. *arXiv preprint arXiv:2001.09694*.

Appendix

A Dataset Details

As the source dataset, we adopt **SQuAD v1.1** (Rajpurkar et al., 2016). SQuAD v1.1 is a question-answering dataset where context paragraphs originate from Wikipedia articles. The QA pairs were then annotated by crowdworkers.

In our experiments, we adopt four other datasets from MRQA (Fisch et al., 2019) as target datasets:

1. **HotpotQA** is a question-answering dataset with multi-hop questions and supporting facts to promote reasoning in QA (Yang et al., 2018).
2. **NaturalQuestions** (Kwiatkowski et al., 2019) builds upon real-world user questions. These were then combined with Wikipedia articles as context. The Wikipedia articles may or may not contain the answer to each question.
3. **TriviaQA** (Joshi et al., 2017) is a question-answering dataset containing evidence information for reasoning in QA.
4. **SearchQA** (Dunn et al., 2017) was built based on an existing dataset of QA pairs. The QA pairs were then extended by contexts, which were crawled through Google search.

B Baseline Details

We introduce five baselines which also involve QA domain adaptation through question generation:

1. **HCVAE** uses a hierarchical variational autoencoder to encode contexts. Moreover, latent variables are sampled in the latent space to generate output questions and answers (Lee et al., 2020).
2. **AQGen** uses a transformer architecture for question generation. Here, answers are generated in the first step, followed by question generation in the second step (Shakeri et al., 2020).
3. **QAGen2S** also uses a transformer model for question generation model. QAGen2S first generates the questions end-to-end and, subsequently, generates answers conditioned on question and context (Shakeri et al., 2020).
4. **QAGen-T5** adopts two T5 transformers to generate question-answer pars in two steps. The first T5 transformer generates questions end-to-end, followed by the answer generation using a second T5 transformer (Yue et al., 2021).
5. **CAQA** leverages QAGen-T5 for question generation but extends the learning algorithm. Specifically, CAQA uses contrastive adaptation to reduce domain discrepancy and promote answer extraction for QA domain adaptation. (Yue et al., 2021).

C Implementation Details

QG model: We train the question generation models on the source dataset. Then, the trained QG models are used to generate QA pairs based on the given input context. We implement and train AQGen and QAGen2S as in (Shakeri et al., 2020; Yue et al., 2021), QG models are trained for 10 epochs with the default settings when available. For generating synthetic data, we apply LM filtering for QAGen-T5 and adopt roundtrip filtering for the other QG models as in (Alberti et al., 2019). The optimizer is set to AdamW without weight decay and warmup. We validate the QG models based on the SQuAD dev set to select the best QG model.

QA model: We train BERT-QA with a learning rate of $3 \cdot 10^{-5}$ for two epochs and a batch size of 16 using both QA data from both source domain and the synthetic target samples. We use the AdamW optimizer without linear warmup. We additionally use Nvidia Apex for mixed precision training (Kamath et al., 2020; Yue et al., 2021). When using HCVAE for question generation, we make use of the results from (Yue et al., 2021).

Hyperparameter for QADA: For our experiments, we first searched for a combination of token augmentation ratio and context cutoff ratio, followed by tuning both λ_{token} and λ_{type}. Specifically, we empirically searched for the best combination across different ranges. For data augmentation, we experimented with different token augmentation ratios r_{token} in $[0, 0.1, 0.2, 0.3, 0.4, 0.5]$, and we experimented with different context cutoff ratios r_{context} in $[0, 0.05, 0.1, 0.15, 0.2, 0.25]$. For contrastive adaptation, we experimented with λ_{token} in the range $[10^{-4}, 5 \cdot 10^{-4}, 10^{-3}, 5 \cdot 10^{-3}, 10^{-2}]$ and λ_{type} from $[0, 0.01, 0.05, 0.1, 0.5]$. For λ_{token} and λ_{type}, details are reported in Table 3 and Table 4. Eventually, the best combination was selected.

D Sensitivity Analysis of Hyperparameters

To better understand the behavior of λ_{token} and λ_{type}, we perform a sensitivity analysis where we vary both hyperparameters. Detailed results for different hyperparameter combinations are presented in Table 5 and Table 6, respectively.

Dataset	**Augmentation**		**QADA**
	r_{token}	r_{context}	EM / F1
HotpotQA	0	0.1	47.94/63.45
NaturalQ.	0.4	0.25	49.03/62.60
SearchQA	0.2	0.05	36.58/44.03
TriviaQA	0	0.2	55.58/63.72

Table 3: Augmentation selection for the main results.

Dataset	**Hyperparam.**		**QADA**
	λ_{token}	λ_{type}	EM / F1
HotpotQA	$5 \cdot 10^{-3}$	0.05	47.94/63.45
NaturalQ.	$5 \cdot 10^{-4}$	0	49.03/62.60
SearchQA	10^{-2}	0.5	36.58/44.03
TriviaQA	$5 \cdot 10^{-4}$	0.05	55.58/63.72

Table 4: Hyperparameter selection for the main results.

First, we experiment with different λ_{token} values. For this, we temporarily exclude the discrepancy term in the attention-based contrastive adaptation loss (i.e., $\lambda_{\text{type}} = 0$). Here, we test λ_{token} in $[10^{-4}, 5 \cdot 10^{-4}, 10^{-3}, 5 \cdot 10^{-3}, 10^{-2}]$. The results are in Table 5. Second, we vary λ_{type}. In doing so, we keep the value of λ_{token} fixed at the best value from the first step. Here, the results are reported in Table 6. We then select λ_{token} with the best performance. This gives the results in the main analysis (Table 1).

E Qualitative Analyis of QA Samples

We performed a qualitative analysis of the data augmentation. For this, we report examples of synthetic QA pairs that were generated during question generation using out augmentation pipeline for a given context. The examples for the different datasets are in Tables 7 to 10. Here, the token augmentation ratio was set to 0.1 for HotpotQA and TriviaQA, and to 0.2 for SearchQA, and to 0.4 for NaturalQuestions.

We make a few interesting observations. First, we see more synonym replacements than token swaps. This may be attributed to the short lengths of questions (since token swaps involves twice as many tokens. Second, the generated QA pairs appear similar to SQuAD in style. However, the augmented questions tend to also incorporate terms that are otherwise less frequent as well as more word orders that are otherwise seen less commonly.

Dataset	λ_{type}	λ_{token}				
		10^{-4}	$5 \cdot 10^{-4}$	10^{-3}	$5 \cdot 10^{-3}$	10^{-2}
HotpotQA	$\lambda_{\text{type}} = 0$	47.94/63.26	47.55/63.03	47.62/63.31	46.53/61.99	46.50/62.02
NaturalQ.	$\lambda_{\text{type}} = 0$	48.91/62.63	49.03/62.60	48.55/62.11	48.78/62.36	48.33/61.87
SearchQA	$\lambda_{\text{type}} = 0$	32.95/39.73	33.92/40.73	31.94/38.76	33.62/40.69.	34.10/40.76
TriviaQA	$\lambda_{\text{type}} = 0$	55.29/63.42	55.05/63.31	54.90/63.31	55.05/63.27	54.96/63.24

Table 5: Results of QADA with different λ_{token} values on target datasets.

Dataset	λ_{token}	λ_{type}				
		0	0.01	0.05	0.1	0.5
HotpotQA	$\lambda_{\text{token}} = 5 \cdot 10^{-3}$	46.53/61.99	46.81/62.22	**47.94/63.45**	46.57/61.89	46.23/61.91
NaturalQ.	$\lambda_{\text{token}} = 5 \cdot 10^{-4}$	**49.03/62.60**	48.61/61.81	47.51/61.38	47.93/61.64	47.55/62.59
SearchQA	$\lambda_{\text{token}} = 10^{-2}$	34.10/40.76	32.96/39.87	34.66/42.02	35.25/42.48	**36.58/44.03**
TriviaQA	$\lambda_{\text{token}} = 5 \cdot 10^{-4}$	55.05/63.31	55.25/63.45	**55.58/63.72**	55.43/63.68	54.59/63.03

Table 6: Results of QADA based on the selected λ_{token} (from above) while varying different λ_{type} values. Best value in bold.

Examples: HotpotQA
Context (given): Jim Conroy [SEP] James Conroy (born February 6, 1977) is an American voice actor, television writer and actor. He is known for appearing on television shows, such as "Celebrity Deathmatch", "Kenny the Shark" and "Fetch! with Ruff Ruffman., radio commercials and video games. He worked for companies such as WGBH, The Walt Disney Company and Discovery Channel. [PAR] [TLE] Kenny the Shark [SEP] Kenny the Shark is an American animated television series produced by Discovery Kids. The show premiered on NBC's Discovery Kids on NBC from November 1, 2003 and ended February 18, 2006 with two seasons and 26 episodes in total having aired. **Question 1**: How many episodes did the show have? **Augmented 1**: How many sequence did the show have? **Answer 1**: 26 **Question 2**: What is Jim Conroy's birth date? **Augmented 2**: What is jim conroys giving birth date? **Answer 2**: February 6, 1977 **Question 3**: What is the name of the American animated television series? **Augmented 3**: What is the name of the american revivify television series? **Answer 3**: Kenny the Shark
Context (given): Gang of Youths [SEP] Gang of Youths are an Australian indie rock group consisting of principal songwriter David Le'aupepe (lead vocals/guitar/piano), Max Dunn (bass guitar), Jung Kim (keyboards/guitar), Joji Malani (lead guitar) and Donnie Borzestowski (drums). Their debut album, "The Positions", peaked at No. 5 on the ARIA Albums Chart in May 2015 and was nominated for multiple ARIA Awards. [PAR] [TLE] Let Me Be Clear [SEP] Let Me Be Clear is the debut extended play by Australian alternative band Gang of Youths. The EP features 5 original tracks and a cover of Joni Mitchell's "Both Sides Now". It was released on 29 July 2016 and debuted at number 2 on the ARIA Charts. **Question 1**: When was the song released? **Augmented 1**: When was the song unblock? **Answer 1**: 29 July 2016 **Question 2**: How many tracks are on the EP? **Augmented 2**: How many trail are on the EP? **Answer 2**: 5 **Question 3**: What is the name of the Australian indie rock group? **Augmented 3**: What is the name of the Australian indie tilt group? **Answer 3**: Gang of Youths

Table 7: Examples of QA pairs generated via our augmentation pipeline for HotpotQA.

Examples: NaturalQuestions

Context (given): <P> Red blood cell distribution width (RDW or RDW - CV or RCDW and RDW - SD) is a measure of the range of variation of red blood cell (RBC) volume that is reported as part of a standard complete blood count . Usually red blood cells are a standard size of about 6 - 8 Ŏ3bcm in diameter . Certain disorders , however , cause a significant variation in cell size . Higher RDW values indicate greater variation in size . Normal reference range of RDW - CV in human red blood cells is 11.5 - 14.5 % . If anemia is observed , RDW test results are often used together with mean corpuscular volume (MCV) results to determine the possible causes of the anemia . It is mainly used to differentiate an anemia of mixed causes from an anemia of a single cause . </P>.

Question 1: What do higher RDW values indicate?
Augmented 1: What do mellow RDW appreciate indicate?
Answer 1: Greater variation in size
Question 2: What is the measure of the range of variation of red blood cell volume?
Augmented 2: Variation is the measure of the range of what volume red blood cell of?
Answer 2: Red blood cell distribution width
Question 3: What is the normal reference range of RDW - CV in human red blood cells?
Augmented 3: What is the convention reference straddle of RDW resume in human being red blood cadre?
Answer 3: 11.5 - 14.5 %

Context (given): <P> The original World Trade Center was a large complex of seven buildings in Lower Manhattan , New York City , United States . It featured the landmark twin towers , which opened on April 4 , 1973 , and were destroyed in 2001 during the September 11 attacks . At the time of their completion , the “ Twin Towers ” – the original 1 World Trade Center , at 1,368 feet (417 m) ; and 2 World Trade Center , at 1,362 feet (415.1 m) – were the tallest buildings in the world . Other buildings in the complex included the Marriott World Trade Center (3 WTC) , 4 WTC , 5 WTC , 6 WTC , and 7 WTC . All were built between 1975 and 1985 , with a cost of $ 400 million ($ 2,300,000,000 in 2014 dollars) . The complex was located in New York City ’s Financial District and contained 13,400,000 square feet (1,240,000 m) of office space . </P>.

Question 1: What was the original height of the 1 World Trade Center?
Augmented 1: What was the master tallness of the globe trade wind center?
Answer 1: 1,368 feet
Question 2: How many meters of office space was in the complex?
Augmented 2: Of many meters the office space was in how complex?
Answer 2: 1,240,000
Question 3: When were the twin towers destroyed?
Augmented 3: When were the twin tower ruin?
Answer 3: 2001

Table 8: Examples of QA pairs generated via our augmentation pipeline for NaturalQuestions.

Examples: SearchQA
Context (given): [DOC] [TLE] jeopardy/1333_Qs.txt at master jedoublen/jeopardy GitHub [PAR] Number: 2. ANIMAL SONGS | British singer Robyn Hitchcock is known for his tunes about these animals, including "Bass" & "Aquarium" | Fish. right: Matt. Wrong:. [DOC] [TLE] Robyn Hitchcock - Wikipedia [PAR] Robyn Rowan Hitchcock (born 3 March 1953) is an English singer-songwriter and guitarist. While primarily a vocalist and guitarist, he also plays harmonica, piano, and bass guitar. ... Hitchcock's lyrics tend to include surrealism, comedic elements, ... Hitchcock released his solo debut, Black Snake Diamond Rle in 1981,... [DOC] [TLE] Positive Vibrations: Softcore - fegMANIA! [PAR] An except from Positive Vibrations' complete guide to the songs of Robyn Hitchcock. ... **Question 1**: What is the dance music of northeastern Argentina known as? **Augmented 1**: What is the terpsichore music of northeasterly argentina known as? **Answer 1**: Chaman **Question 2**: What was Hitchcock's solo debut called? **Augmented 2**: What was Alfred Hitchcock solo debut called? **Answer 2**: Black Snake Diamond Rle **Question 3**: When did Hitchcock release his solo debut? **Augmented 3**: When did Hitchcock release his solo introduction? **Answer 3**: 1981
Context (given): [DOC] [TLE] Battle of Blood River - Wikipedia [PAR] The Battle of Blood River is the name given for the battle fought between 470 Voortrekkers ("Pioneers"), led by Andries Pretorius, and an estimated 15,000 21,000 Zulu attackers on the bank of the Ncome River on 16 December 1838, in what is today KwaZulu-Natal, South Africa. ... Casualties amounted to 3,000 of king Dingane's soldiers dead, including two... [DOC] [TLE] Battle of Blood River | South African history | Britannica.com [PAR] Battle of Blood River, Blood River also known as Ncome River, (Dec. ... 16, 1838, a Boer force led by Andries Pretorius induced a Zulu attack on a Boer laager (protected ... defeated an army of Zulu warriors on the banks of the Ncome River. [DOC] [TLE] The Battle of Blood River | South African History Online [PAR] On 16 December 1838 the Battle of Blood River took place near the Ncome River in KwaZulu Natal. ... Towards a peoples history ... Voortrekkers under the leadership of Andries Pretorius and the Zulu's under the leadership of Dingane the Zulu King. ... **Question 1**: Who led the Boer force in the Battle of Blood River? **Augmented 1**: Who take the afrikander force in the battle of blood river? **Answer 1**: Andries Pretorius **Question 2**: Who led the Boers to a huge victory over the Zulus? **Augmented 2**: Who pass the Boers to a huge victory over the Zulu? **Answer 2**: Andries Pretorius **Question 3**: What was the name of Dingane's generals? **Augmented 3**: What was the name of Dinganes superior general? **Answer 3**: Dambuza

Table 9: Examples of QA pairs generated via our augmentation pipeline for SearchQA.

Examples: TriviaQA
Context (given): [DOC] [TLE] Spiers on Sport: the unjust sacking of Kenny Shiels (From ...Spiers on Sport: the unjust sacking of Kenny Shiels (From HeraldScotland) [PAR] / Spiers on Sport , Graham Spiers [PAR] When a manager wins one of only four trophies collected by a football club in 80 years, there has to be a degree of respect shown towards him, right? [PAR] When he also works slavishly on all aspects of a club due to staffing limitations - training, recruiting, video-editing, youth development etc - wouldn't that admiration for him grow even greater? [PAR] Loading article content [PAR] Kenny Shiels, sacked by Kilmarnock, is by no means perfect. But he has been a pretty good manager at Rugby Park, whose dismissal is hard to fathom. [PAR] It transpires, too, that many Kilmarnock supporters, contrary to what we might have been led to believe, are also peeved at their manager's sacking. A mob of them descended on Rugby Park the other evening to vent their spleen at Michael Johnston, the club's chairman. [PAR] I quite like and admire Johnston. He is a lawyer, a bit old-school, who gets flack galore in his Killie role but always stands his ground. But this decision seems quaint to me. [PAR] First, let's look at Shiels' record. He was Kilmarnock manager for two seasons, during which the club lifted the Scottish League Cup and finished seventh and ninth in the SPL. ... **Question 1**: Where did most of Shiels' felonies occur? **Augmented 1**: Where did most of Shiels felonies go on? **Answer 1**: Rugby Park **Question 2**: What club did he manage? **Augmented 2**: What society did he manage? **Answer 2**: Kilmarnock **Question 3**: Who is the chairman of the rugby club? **Augmented 3**: Who is the president of the rugby club? **Answer 3**: Michael Johnston
Context (given): [DOC] [TLE] Bagpuss and the little girl who owned him meet up for 40th ...Bagpuss and the little girl who owned him meet up for 40th birthday | Daily Mail Online [PAR] A bit looser at the seams... but Emily STILL loves him: Bagpuss is reunited with the little girl who owned him to celebrate the famous programme's 40th birthday [PAR] Emily Firmin was 8-years-old when she starred as the owner of Bagpuss [PAR] Her father Peter Firmin created the TV show which was broadcast in 1974 [PAR] Now aged 48, she is still instantly recognisable as the same little girl from the hit TV programme [PAR] To celebrate the 40th anniversary of the show, Ms Firmin has been reunited with the famous cat at Canterbury Heritage Museum[DOC] [TLE] Bagpuss - The Intro - SmallfilmsBagpuss - The Intro [PAR] The Intro [PAR] There was a little girl and her name was Emily [PAR] And she had a shop [PAR] There it is [PAR] It was rather an unusual shop because it didn't sell anything [PAR] You see, everything in that shop window was a thing that somebody had once lost [PAR] And Emily had found [PAR] And brought home to Bagpuss [PAR] Emily's cat Bagpuss [PAR] Saggy old cloth cat in the whole wide world [PAR] Well now, one day Emily found a thing [PAR] ... **Question 1**: How long was the show repeated in the UK? **Augmented 1**: How recollective was the show repeated in the UK? **Answer 1**: 1974 **Question 2**: What year did the show Bagpuss first air? **Augmented 2**: What yr did the show Bagpuss first air? **Answer 2**: 1974 **Question 3**: When was the series first broadcast? **Augmented 3**: When was the series kickoff broadcast? **Answer 3**: 1974

Table 10: Examples of QA pairs generated via our augmentation pipeline for TriviaQA.