QA Domain Adaptation using Hidden Space Augmentation and Self-Supervised Contrastive Adaptation

Zhenrui Yue[*]
UIUC
zhenrui3@illinois.edu

Huimin Zeng[*]
UIUC
huiminz3@illinois.edu

Bernhard Kratzwald
EthonAI
bernhard.kratzwald@ethon.ai

Stefan Feuerriegel
LMU Munich
feuerriegel@lmu.de

Dong Wang
UIUC
dwang24@illinois.edu

Abstract

Question answering (QA) has recently shown impressive results for answering questions from customized domains. Yet, a common challenge is to adapt QA models to an unseen target domain. In this paper, we propose a novel self-supervised framework called QADA for QA domain adaptation. QADA introduces a novel data augmentation pipeline used to augment training QA samples. Different from existing methods, we enrich the samples via hidden space augmentation. For questions, we introduce multi-hop synonyms and sample augmented token embeddings with Dirichlet distributions. For contexts, we develop an augmentation method which learns to drop context spans via a custom attentive sampling strategy. Additionally, contrastive learning is integrated in the proposed self-supervised adaptation framework QADA. Unlike existing approaches, we generate pseudo labels and propose to train the model via a novel attention-based contrastive adaptation method. The attention weights are used to build informative features for discrepancy estimation that helps the QA model separate answers and generalize across source and target domains. To the best of our knowledge, our work is the first to leverage hidden space augmentation and attention-based contrastive adaptation for self-supervised domain adaptation in QA. Our evaluation shows that QADA achieves considerable improvements on multiple target datasets over state-of-the-art baselines in 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., Seo et al., 2016; Chen et al., 2017; Devlin et al., 2019; Kratzwald et al., 2019).

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; Miller et al., 2020; Zeng et al., 2022b).

Existing approaches to address domain shifts in extractive QA can be grouped as follows. One approach is to include labeled target examples or user feedback during training (Daumé III, 2007; Kratzwald and Feuerriegel, 2019a; Kratzwald et al., 2020; Kamath et al., 2020). Another approach is to generate labeled QA samples in the target domain for training (Lee et al., 2020; Yue et al., 2021a, 2022a). However, these approaches typically require large amounts of annotated data or extensive computational resources. As such, they tend to be ineffective in adapting existing QA models to an unseen target domain (Fisch et al., 2019). Only recently, a contrastive loss has been proposed to handle domain adaptation in QA (Yue et al., 2021b).

Several approaches have been used to address issues related to insufficient data and generalization in NLP tasks, yet outside of QA. For example, augmentation in the hidden space encourages more generalizable features for training (Verma et al., 2019; Chen et al., 2020, 2021). For domain adaptation, there are approaches that encourage the model to learn domain-invariant features via a domain critic (Lee et al., 2019; Cao et al., 2020), or adopt discrepancy regularization between the source and target domains (Kang et al., 2019; Yue et al., 2022b). However, to the best of our knowledge, no work has attempted to build a smooth and generalized feature space via hidden space augmentation and self-supervised domain adaption.

[*]Both authors contributed equally to this research.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 2308–2321
December 7-11, 2022 ©2022 Association for Computational Linguistics

In this paper, we propose a novel *self-supervised QA domain adaptation* framework for extractive QA called QADA. Our QADA framework is designed to handle domain shifts and should thus answer out-of-domain questions. QADA has three stages, namely pseudo labeling, hidden space augmentation and self-supervised domain adaptation. First, we use pseudo labeling to generate and filter labeled target QA data. Next, the augmentation component integrates a novel pipeline for data augmentation to enrich training samples in the hidden space. For questions, we build upon multi-hop synonyms and introduce Dirichlet neighborhood sampling in the embedding space to generate augmented tokens. For contexts, we develop an attentive context cutoff method which learns to drop context spans via a sampling strategy using attention scores. Third, for training, we propose to train the QA model via a novel attention-based contrastive adaptation. Specifically, we use the attention weights to sample informative features that help the QA model separate answers and generalize across the source and target domains.

Main contributions of our work are:[1]

1. We propose a novel, self-supervised 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 in an unseen domain.

2. To the best of our knowledge, QADA is the first work in QA domain adaptation that (i) leverages hidden space augmentation to enrich training data; and (ii) integrates attention-based contrastive learning for self-supervised adaptation.

3. We demonstrate the effectiveness of QADA in an unsupervised setting where target answers are not accessible. Here, QADA can considerably outperform state-of-the-art baselines on multiple datasets for QA domain adaptation.

2 Related Work

Extractive QA has achieved significantly progress recently (Devlin et al., 2019; Kratzwald et al., 2019; Lan et al., 2020; 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 (Yang et al., 2017; Kratzwald and Feuerriegel, 2019b; Yue et al., 2022a). (2) Unsupervised adaptation with question generation refers to settings where only context paragraphs in the target domain are available, QA samples are generated separately to train the QA model (Shakeri et al., 2020; Yue et al., 2021b). (3) Unsupervised adaptation has access to context and question information from the target domain, whereas answers are unavailable (Chung et al., 2018; Cao et al., 2020; Yue et al., 2022d). In this paper, we focus on the third category and study the problem of unsupervised QA domain adaptation.

Domain adaptation for QA: Several approaches have been developed to generate synthetic QA samples via question generation (QG) in an end-to-end fashion (i.e., seq2seq) (Du et al., 2017; Sun et al., 2018). Leveraging such samples from QG can also improve the QA performance in out-of-domain distributions (Golub et al., 2017; Tang et al., 2017, 2018; Lee et al., 2020; Shakeri et al., 2020; Yue et al., 2022a; Zeng et al., 2022a). Given unlabeled questions, there are two main approaches: domain adversarial training can be applied to reduce feature discrepancy between domains (Lee et al., 2019; Cao et al., 2020), while contrastive adaptation minimizes the domain discrepancy using maximum mean discrepancy (MMD) (Yue et al., 2021b, 2022d). We later use the idea from contrastive learning but tailor it carefully for our adaptation framework.

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 partial spans in the 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 a hidden space augmentation pipeline tailored for QA

[1] The code for our QADA framework is publicly available at https://github.com/Yueeeeeeee/Self-Supervised-QA.

data in which different strategies are combined for question and context augmentation.

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 distances (e.g., triplet loss) or domain distances with MMD (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 to separate answer tokens and minimize the discrepancy between source and target domain (Yue et al., 2021b, 2022d). However, our work is different in that we introduce a novel *attention-based* strategy to construct more informative features for discrepancy estimation and contrastive adaptation.

3 Setup

We consider the following problem setup for QA domain adaptation, where labeled source data and *unlabeled* target data are available for training. Our goal is to train a QA model $\boldsymbol{f}$ that maximizes the performance in the target domain using both source data and unlabeled target data (Cao et al., 2020; Shakeri et al., 2020; Yue et al., 2021b, 2022d).

Data: 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 pretrain a QA model on the source domain $\boldsymbol{\mathcal{D}}_s$ and, following this, the pretrained QA model is adapted to the target domain $\boldsymbol{\mathcal{D}}_t$. The input data to each domain is as follows:

- *Labeled source data*: Training data is provided by labeled QA data $\boldsymbol{X}_s$ from the source domain $\boldsymbol{\mathcal{D}}_s$. Here, each sample $(\boldsymbol{x}_{s,c}^{(i)}, \boldsymbol{x}_{s,q}^{(i)}, \boldsymbol{x}_{s,a}^{(i)}) \in \boldsymbol{X}_s$ is a triplet comprising a context $\boldsymbol{x}_{s,c}^{(i)}$, a question $\boldsymbol{x}_{s,q}^{(i)}$, 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 data*: We assume partial access to data from the target domain $\boldsymbol{\mathcal{D}}_t$, that is, only contexts and unlabeled questions. The contexts and questions are first used for pseudo labeling, followed by self-supervised adaptation. Formally, we refer to the contexts and questions via $\boldsymbol{x}_{t,c}^{(i)}$ and $\boldsymbol{x}_{t,q}^{(i)}$, with $(\boldsymbol{x}_{t,c}^{(i)}, \boldsymbol{x}_{t,q}^{(i)}) \in \boldsymbol{X}_t'$ where $\boldsymbol{X}_t'$ is the unlabeled data from the target domain.

Model: The QA model can be represented with function $\boldsymbol{f}$. $\boldsymbol{f}$ takes both a question and context as input and predicts an answer, i.e., $\boldsymbol{x}_a^{(i)} = \boldsymbol{f}(\boldsymbol{x}_q^{(i)}, \boldsymbol{x}_c^{(i)})$. 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 of $\boldsymbol{f}$ over target data $\boldsymbol{X}_t$:

$$\min_{\boldsymbol{f}} \mathcal{L}_{\text{ce}}(\boldsymbol{f}, \boldsymbol{X}_t), \tag{1}$$

where $\mathcal{L}_{\text{ce}}$ is the cross-entropy loss.

4 The QADA Framework

4.1 Overview

Our proposed QADA framework has three stages to be performed in each epoch (see Fig. 1): (1) **pseudo labeling**, where pseudo labels are generated for the unlabeled targeted data; (2) **hidden space augmentation**, in which the proposed augmentation strategy is leveraged to generated virtual examples in the feature space; and (3) **contrastive adaptation** that minimizes domain discrepancy to transfer source knowledge to the target domain.

To address the domain shift upon deployment, we use the aforementioned stages as follows. In the first stage, we generate pseudo labels for the unlabeled target data $\boldsymbol{X}_t'$. Next, we enrich the set of training data via hidden space augmentation. In the adaptation stage, we train the QA model using both the source and the target data with our attention-based contrastive adaptation. We summarize the three stages in the following:

1. *Pseudo labeling*: First, we build labeled target data $\hat{\boldsymbol{X}}_t$ via pseudo labeling. Formally, a source-pretrained QA model $\boldsymbol{f}$ generates a (pseudo) answer $\boldsymbol{x}_{t,a}^{(i)}$ for context $\boldsymbol{x}_{t,c}^{(i)}$ and question $\boldsymbol{x}_{t,q}^{(i)}$, $i = 1, \ldots$ Each sample $\boldsymbol{x}_t^{(i)} \in \hat{\boldsymbol{X}}_t$ now contains the original context, the original question, and a predicted answer. We additionally apply confidence thresholding to filter the pseudo labels.
2. *Hidden space augmentation*: The QA model $\boldsymbol{f}$ takes a question and context pair as input.

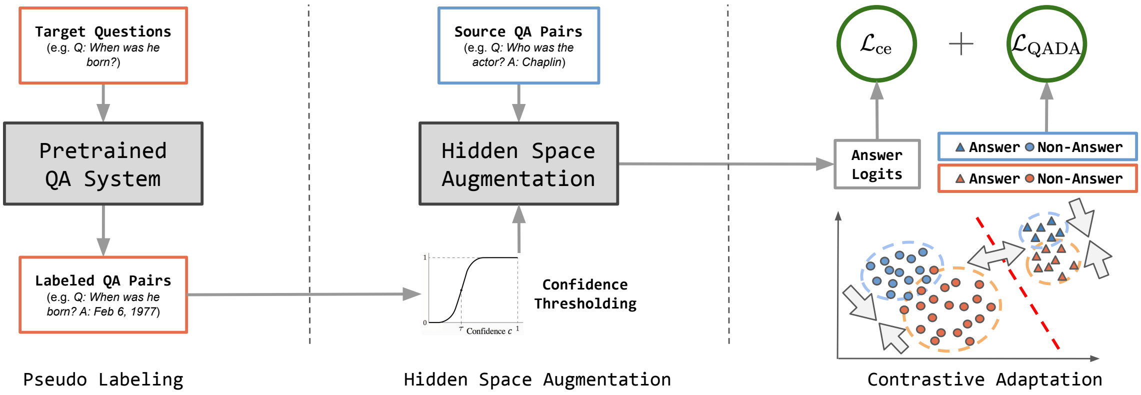
Target Questions
(e.g. Q: When was he born?)
Pretrained QA System
Labeled QA Pairs
(e.g. Q: When was he born? A: Feb 6, 1977)
Source QA Pairs
(e.g. Q: Who was the actor? A: Chaplin)
Hidden Space Augmentation
Confidence Thresholding
Confidence c
τ
$\mathcal{L}_{\text{ce}}$
+
$\mathcal{L}_{\text{QADA}}$
Answer Logits
▲ Answer ● Non-Answer
▲ Answer ● Non-Answer
Pseudo Labeling
Hidden Space Augmentation
Contrastive Adaptation

Figure 1: Overview of our proposed QADA framework. QADA generates pseudo labels for the unlabeled target data, followed by hidden space augmentation. The QA model is trained via attention-based contrastive adaptation.

For questions, we perform Dirichlet neighborhood sampling in the word embedding layer to generate diverse, yet consistent query information. We also apply a context cutoff in the hidden space after transformer layers to reduce the learning of redundant domain information.

3. *Contrastive adaptation*: We train the QA model $\boldsymbol{f}$ with the source data $\boldsymbol{X}_s$ from the source domain $\boldsymbol{\mathcal{D}}_s$ and the target data $\hat{\boldsymbol{X}}_t$ with pseudo labels from the previous stage. We impose regularization on the answer extraction and further minimize the discrepancy between the source and target domain, so that the learned features generalize well to the target domain.

4.2 Pseudo Labeling

Provided with the access to labeled source data, we first pretrain the QA model to answer questions in the source domain. Then, we can use the pretrained model to predict target answers for self-supervised adaptation (Cao et al., 2020). The generated pseudo labels are filtered via confidence thresholding, in which the target samples above confidence threshold τ $(= 0.6$ in our experiments) are preserved for the later stages. We repeat the pseudo-labeling step in each epoch to dynamically adjust the target distribution used for self-supervised adaptation.

The QA model $\boldsymbol{f}$ is pretrained on the source dataset $\boldsymbol{X}_s$ via a cross-entropy loss $\mathcal{L}_{\text{ce}}$, i.e., $\min_{\boldsymbol{f}} \mathcal{L}_{\text{ce}}(\boldsymbol{f}, \boldsymbol{X}_s)$. When selecting QA pairs from the target domain, we further use confidence thresholding for filtering and, thereby, build a subset of target data with pseudo labels $\hat{\boldsymbol{X}}_t$, i.e.,

$$\begin{aligned}\hat{\boldsymbol{X}}_t = \{(\boldsymbol{x}_{t,c}^{(i)}, \boldsymbol{x}_{t,q}^{(i)}, \boldsymbol{f}(\boldsymbol{x}_{t,c}^{(i)}, \boldsymbol{x}_{t,q}^{(i)}) \mid \\ \sigma(\boldsymbol{f}(\boldsymbol{x}_{t,c}^{(i)}, \boldsymbol{x}_{t,q}^{(i)})) \geq \tau, (\boldsymbol{x}_{t,c}^{(i)}, \boldsymbol{x}_{t,q}^{(i)}) \in \boldsymbol{X}_t'\},\end{aligned} \quad (2)$$

where σ computes the output answer confidence (i.e, softmax function).

4.3 Hidden Space Augmentation

We design a data augmentation pipeline to enrich the training data based on the generated QA pairs. The augmentation pipeline is divided into two parts: (i) *question augmentation* via Dirichlet neighborhood sampling in the embedding layer and (ii) *context augmentation* with attentive context cutoff in transformer layers. Both are described below.

Question augmentation: To perform augmentation of questions, we propose *Dirichlet neighborhood sampling* (see Fig. 2) to sample synonym replacements on certain proportion of tokens, such that the trained QA model captures different patterns of input questions. Dirichlet distributions have been previously applied to find adversarial examples (Zhou et al., 2021; Yue et al., 2022c); however, different from such methods, we propose to perform question augmentation in the embedding layer. We first construct the multi-hop neighborhood for each token in the input space. Here, 1-hop synonyms can be derived from a synonym dictionary, while 2-hop synonyms can be extended from 1-hop synonyms (i.e., the synonyms of 1-hop synonyms).

For each token, we compute a convex hull spanned by the token and its multi-hop synonyms (i.e., vertices), as shown in Fig. 2. The convex hull is used as the sampling area of the augmented token embedding, where the probability distribution in the sampling area can be computed using a Dirichlet distribution. That is, the sampled token embedding is represented as the linear combinations of vertices in the convex hull. Formally, for a token x and the set of its multi-hop synonyms $\mathcal{C}_x$,

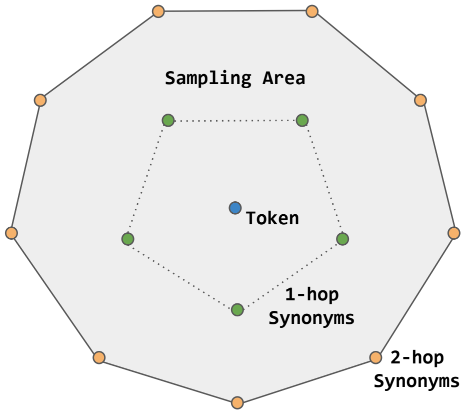

Figure 2: The proposed Dirichlet neighborhood sampling in QADA. We sample coefficients from a Dirichlet distribution and represent the augmented embedding as the linear combination of multi-hop synonyms.

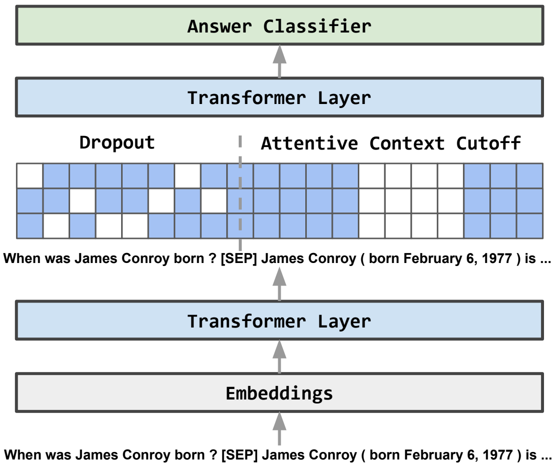
Answer Classifier
Transformer Layer
Dropout
Attentive Context Cutoff
When was James Conroy born ? [SEP] James Conroy (born February 6, 1977) is ...
Transformer Layer
Embeddings
When was James Conroy born ? [SEP] James Conroy (born February 6, 1977) is ...

Figure 3: The proposed context cutoff in QADA. Context cutoff drops context subspans in the length dimension and is performed after transformer layers via an attentive sampling strategy.

we denote the coefficients of the linear combination by $\boldsymbol{\eta}_x = [\eta_{x,1}, \eta_{x,2}, \dots, \eta_{x,|\mathcal{C}_x|}]$. We sample coefficients $\boldsymbol{\eta}_x$ from a Dirichlet distribution:

$$\boldsymbol{\eta}_x \sim \text{Dirichlet}(\alpha_1, \alpha_2, \dots, \alpha_{|\mathcal{C}_x|}), \tag{3}$$

where α values are selected differently for the original token and its multi-hop synonyms. Using the sampled $\boldsymbol{\eta}_x$, we can compute the augmented token embedding with the embedding function $\boldsymbol{f}_e$ via

$$\boldsymbol{f}_e(\boldsymbol{\eta}_x) = \sum_{j \in \mathcal{C}_x}^{|\mathcal{C}_x|} \eta_{x,j}\, \boldsymbol{f}_e(j). \tag{4}$$

Dirichlet distributions are multivariate probability distributions with $\sum \boldsymbol{\eta}_x = 1$ and $\boldsymbol{\eta}_{x,j} \geq 0, \forall j \in \mathcal{C}_x$. The augmented embedding is therefore a linear combination of vertices in the convex hull. By adjusting α values, we can change the probability distribution in the sampling area, that is, how far the sampled embedding can travel from the original token. For example, with increasing α values, we can expect the sampled points approaching the center point of the convex hull.

The above augmentation is introduced in order to provide semantically diverse yet consistent questions. At the same time, by adding noise locally, it encourages the QA model to capture robust information in questions (see Zhou et al., 2021). We control the question augmentation by a token augmentation ratio, ζ, to determine the percentage of tokens within questions that are augmented.[2]

Context augmentation: For contexts, we adopt augmentation in the hidden space of the transformer layers instead of the embedding layer. 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 shown in Fig. 3, 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.

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

$$p_i = \sigma\left(\frac{1}{H}\sum_{j}^{H}\left(\sum_{k}^{L_c} A_{j,k}\right)\right)_i, \tag{5}$$

where H is the number of attention heads, L_c is the context length, and σ denotes the softmax function. Once the cutoff midpoint is sampled, we introduce a context cutoff ratio, φ, as a hyperparameter. It determines the cutoff length (as compared to length

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

of the original context). We avoid context cutoff in the final transformer layer to prevent important answer features from being zeroed out.

Eventually, the above procedure of question augmentation should improve the model capacity in question understanding. Combined with context cutoff, the QA model is further forced to attend context information globally in the hidden space. This thus encourages the QA model to reduce redundancy and capture relevant information, i.e., from all context positions using self-attention.

4.4 Contrastive Adaptation

To adapt the QA model to the target domain, we develop a tailored attention-based contrastive adaptation. Here, our idea is to regularize the intra-class discrepancy for knowledge transfer and increase the inter-class discrepancy for answer extraction. We consider answer tokens and non-answer tokens as different classes (Yue et al., 2021b).

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

$$\begin{aligned}
\mathcal{L}_{\text{QADA}} &= \mathcal{D}(\boldsymbol{X}_{s,a}, \boldsymbol{X}_{t,a}) + \mathcal{D}(\boldsymbol{X}_{s,n}, \boldsymbol{X}_{t,n}) \\
&\quad - \mathcal{D}(\boldsymbol{X}_a, \boldsymbol{X}_n) \quad \text{with} \\
\mathcal{D} &= \frac{1}{|\boldsymbol{X}_s||\boldsymbol{X}_s|} \sum_{i=1}^{|\boldsymbol{X}_s|} \sum_{j=1}^{|\boldsymbol{X}_s|} k(\phi(\boldsymbol{x}_{\text{s}}^{(i)}), \phi(\boldsymbol{x}_{\text{s}}^{(j)})) \\
&\quad + \frac{1}{|\boldsymbol{X}_t||\boldsymbol{X}_t|} \sum_{i=1}^{|\boldsymbol{X}_t|} \sum_{j=1}^{|\boldsymbol{X}_t|} k(\phi(\boldsymbol{x}_{\text{t}}^{(i)}), \phi(\boldsymbol{x}_{\text{t}}^{(j)})) \\
&\quad - \frac{2}{|\boldsymbol{X}_s||\boldsymbol{X}_t|} \sum_{i=1}^{|\boldsymbol{X}_s|} \sum_{j=1}^{|\boldsymbol{X}_t|} k(\phi(\boldsymbol{x}_{\text{s}}^{(i)}), \phi(\boldsymbol{x}_{\text{t}}^{(j)})),
\end{aligned} \tag{6}$$

where $\boldsymbol{X}_a$ represents answer tokens, $\boldsymbol{X}_n$ represents non-answer tokens in $\boldsymbol{X}$. $\boldsymbol{x}_{\text{s}}^{(i)}$ is the i-th sample from $\boldsymbol{X}_s$, and $\boldsymbol{x}_{\text{t}}^{(j)}$ is the j-th sample from $\boldsymbol{X}_t$. $\mathcal{D}$ computes the MMD distance with empirical kernel mean embeddings and Gaussian kernel k using our scheme below. In $\mathcal{L}_{\text{QADA}}$, the first two terms reduce the intra-class discrepancy (*discrepancy term*), while the last term maximizes the distance of answer tokens to other tokens, thereby improving answer extraction (*extraction term*).

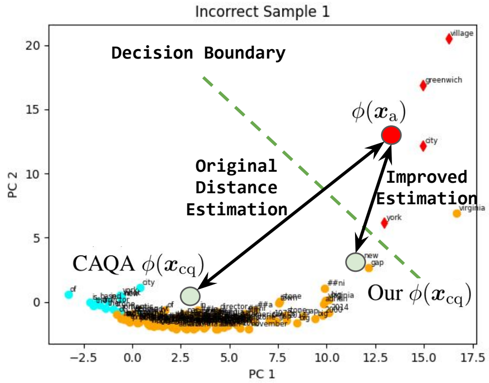

Figure 4: Illustration of QA samples in feature space. The token features are obtained from the last layer in BERT and visualized using principle component analysis (PCA). Question tokens are in cyan, context tokens in orange, and answer tokens in red. We compared the feature mappings ϕ between (i) Yue et al. (2021b) (called CAQA) vs. (ii) our QADA framework.

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 previous research (Yue et al., 2021b), the MMD distance $\mathcal{D}$ was computed using the BERT encoder. However, simply using ϕ as in previous work would return the *averaged* feature of relevant tokens in the sample rather than more *informative* tokens (i.e., tokens near the decision boundary which are "harder" to classify).

Unlike previous methods, we design an attention-based sampling strategy. First, we leverage the attention weights $A \in \boldsymbol{R}^{H \times L_{\boldsymbol{x}} \times L_{\boldsymbol{x}}}$ of input $\boldsymbol{x}$ using the encoder of the QA model. Based on this, we compute a probability distribution for tokens of the relevant class (e.g., non-answer tokens) using the softmax function σ and sample an index. The corresponding token feature from the QA encoder is used as the class feature, i.e.,

$$\phi(\boldsymbol{x}) = \boldsymbol{f}_{\text{enc}}(\boldsymbol{x})_i \text{ with } i \sim \sigma\left(\frac{1}{H} \sum_{j}^{H} \sum_{k}^{L_{\boldsymbol{x}}} A_{j,k}\right), \tag{7}$$

where $\boldsymbol{f}_{\text{enc}}$ 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 attention-based sampling to both answer and non-answer features.

Illustration: We visualize an illustrative QA sample in Fig. 4 to explain the advantage of our

Model	HotpotQA EM / F1	NaturalQ. EM / F1	NewsQA EM / F1	SearchQA EM / F1	TriviaQA EM / F1
(I) Zero-shot target performance					
BERT	43.34/60.42	39.06/53.7	39.17/56.14	16.19/25.03	49.70/59.09
(II) Target performance w/ domain adaptation					
DAT (Lee et al., 2019)	44.25/61.10	44.94/58.91	38.73/54.24	22.31/31.64	49.94/59.82
CASe (Cao et al., 2020)	47.16/63.88	46.53/60.19	43.43/59.67	26.07/35.16	54.74/63.61
CAQA (Yue et al., 2021b)	46.37/61.57	48.55/62.60	40.55/55.90	36.05/42.94	55.17/63.23
CAQA* (Yue et al., 2021b)	48.52/64.76	47.37/60.52	44.26/60.83	32.05/41.07	54.30/62.98
QADA (ours)	**50.80/65.75**	**52.13/65.00**	**45.64/61.84**	**40.47/48.76**	**56.92/65.86**
(III) Target performance w/ supervised training					
BERT w/ 10k Annotations	49.52/66.56	54.88/68.10	45.92/61.85	60.20/66.96	54.63/60.73
BERT w/ All Annotations	57.96/74.76	67.08/79.02	52.14/67.46	71.54/77.77	64.51/70.27

Table 1: Main results of QA adaptation performance on target dataset.

attention-based sampling for domain discrepancy estimation. We visualize all token features and then examine the extraction term from Eq. 6. We further show the feature mapping ϕ from Yue et al. (2021b), which, different from ours, returns 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 our choice of ϕ, QADA improves the measure of answer-context discrepancy and, therefore, is more effective in separating answer tokens.

4.5 Learning Algorithm

We incorporate the contrastive adaptation loss from Eq. 6 into the original training objective. This gives our overall loss

$$\mathcal{L} = \mathcal{L}_{\text{ce}} + \lambda \mathcal{L}_{\text{QADA}}, \tag{8}$$

where λ is a weighting factor for the contrastive loss.

5 Experiments

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

- For the *source domain* $\mathcal{D}_s$, we use SQuAD v1.1 (Rajpurkar et al., 2016).
- For *target domain* $\mathcal{D}_t$, we select MRQA Split I (Fisch et al., 2019): HotpotQA (Yang et al., 2018), Natural Questions (Kwiatkowski et al., 2019), NewsQA (Trischler et al., 2016), SearchQA (Dunn et al., 2017), and TriviaQA (Joshi et al., 2017). This selection makes our results comparable with other works in QA domain adaptation (e.g., Lee et al., 2020; Shakeri et al., 2020; Cao et al., 2020; Yue et al., 2021b, 2022d).

Baselines: As a naïve baseline, we pretrain a BERT on the source dataset as our base model and evaluate on each target dataset with zero knowledge of the target domain. In addition, we adopt three state-of-the-art baselines: domain-adversarial training (DAT) (Lee et al., 2019), conditional adversarial self-training (CASe) (Cao et al., 2020), and contrastive adaptation for QA (CAQA) (Yue et al., 2021b). For a fair comparison, we adopt both the original CAQA and CAQA with our self-supervised adaptation framework (= CAQA*).[3] Baseline details are reported in Appendix B.

Training and Evaluation: We use the proposed method to adapt the pretrained QA model, augmentation hyperparameters are tuned empirically by searching for the best combinations. To evaluate the predictions, we follow (Lee et al., 2020; Shakeri et al., 2020; Yue et al., 2021b) and assess the exact matches (EM) and the F1 score on the dev sets. Implementation details are in Appendix C.

[3] For CAQA*, we exclude question generation and adopt the same process of pseudo labeling and self-supervised adaptation as QADA. Different from QADA, hidden space augmentation is not applied and we use the same objective function as in the original CAQA paper.

Model	HotpotQA EM / F1	NaturalQ. EM / F1	NewsQA EM / F1	SearchQA EM / F1	TriviaQA EM / F1
QADA (ours)	**50.80/65.75**	**52.13/65.00**	**45.64/61.84**	**40.47/48.76**	**56.92/65.86**
w/o Dirichlet sampling	49.57/64.71	51.15/64.24	45.27/61.44	35.90/44.28	56.83/65.51
w/o context cutoff	50.36/65.71	50.30/62.98	45.39/61.47	33.94/42.43	56.04/64.87
w/o contrastive adaptation	48.21/64.54	48.35/61.76	44.35/60.66	30.85/39.42	55.42/64.38

Table 2: Ablation study on different components of QADA.

6 Experimental Results

6.1 Adaptation Performance

Our main results for domain adaptation are in Table 1. We distinguish three major groups: (1) *Zero-shot target performance.* Here, we report a naïve baseline (BERT) for which the QA model is solely trained on SQuAD. (2) *Target performance w/ domain adaptation.* This refers to the methods where domain adaptation techniques are applied. This group also includes our proposed QADA. (3) *Target performance w/ supervised training.* Here, training is done with the original target data. Hence, this reflects an "upper bound".

Overall, the domain adaptation baselines are outperformed by 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) All adaptation methods achieve considerable improvements in answering target domain questions compared to the naïve baseline. (2) QADA performs the best overall. Compared to the best baseline, QADA achieves performance improvements by 6.1% and 4.9% in EM and F1, respectively. (3) The improvements with QADA are comparatively larger on HotpotQA, Natural Questions, and SearchQA ($\sim 8.1\%$ in EM) in contrast to NewsQA and TriviaQA ($\sim 3.1\%$ in EM). A potential reason for the gap is the limited performance of BERT in cross-sentence reasoning, where the QA model often fails to answer compositional questions in long input contexts. (4) QADA can perform similarly or outperform the supervised training results using 10k target data. For example, QADA achieve 56.92 (EM) and 65.86 (F1) on TriviaQA in contrast to 54.63 and 60.73 of the 10k supervised results, suggesting the effectiveness of QADA.

6.2 Ablation Study for QADA

We evaluate the effectiveness of the proposed QADA by performing an ablation study. By comparing the performance of QADA and CAQA* in Table 1, we yield an ablation quantifying the gains that should be attributed to the combination of all proposed components in QADA. We find distinctive performance improvements due to our hidden space augmentation and contrastive adaptation. For example, we observe that EM performance can drop up to 20.8% without QADA, suggesting clear superiority of the proposed QADA.

We further evaluate the effectiveness of the individual components in QADA. We remove the proposed Dirichlet neighborhood sampling, attentive context cutoff and attention-based contrastive adaptation in QADA separately and observe the performance changes. The results on target datasets are reported in Table 2. For all components, we observe consistent performance drops when removed from QADA. For example, the performance of QADA reduces, on average, by 3.3%, 4.5%, and 8.3% in EM when we remove Dirichlet sampling, context cutoff, and contrastive adaptation, respectively. The results suggest that the combination of question and context augmentation in the hidden space is highly effective for improving QA domain adaptation. Moreover, the performance improves clearly when including our attention-based contrastive adaptation.

6.3 Sensitivity Analysis for Hidden Space Augmentation

Our QADA uses question augmentation (i.e., Dirichlet neighborhood sampling) and context augmentation (i.e., context cutoff), where the augmentation ratios determine the percentage of tokens that are augmented. Figure 5 compares different augmentation ratios from 0 to 0.4 on HotpotQA, Natural Questions, and NewsQA. Overall, we observe some variations but, importantly, the performance of QADA improves adaptation performance

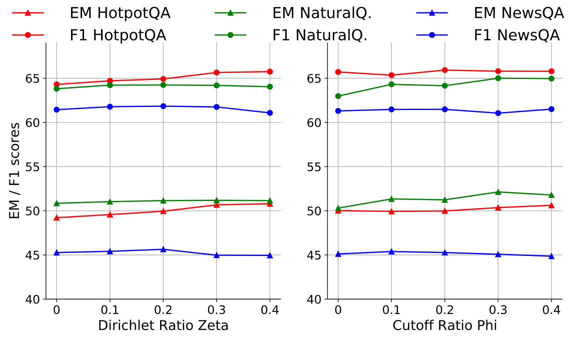

Figure 5: Sensitivity analysis for different Dirichlet sampling ratios (left) and context cutoff ratios (right).

and remains fairly robust for different nonzero ratios. Moreover, we find comparatively large improvements for HotpotQA by introducing Dirichlet neighborhood sampling (2.5% in EM), while Natural Questions benefits more from context cutoff (3.6% in EM). A potential reason for such improvements is that HotpotQA has more complex questions that need potential matching and reasoning, while Natural Questions provides longer unstructured text as contexts, thereby requiring improved understanding of long paragraphs.

7 Conclusion

In this paper, we propose a novel self-supervised framework called QADA for QA domain adaptation. QADA introduces: (1) hidden space augmentation tailored for QA data to enrich target training corpora; and (2) an attention-based contrastive adaptation 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.

8 Limitations

Despite having introduced hidden space augmentation in QADA, we have not discussed different choices of α values for multi-hop synonyms to exploit the potential benefits of the Dirichlet distribution. For context cutoff, dropping multiple context spans in each QA example may bring additional benefits to improve context understanding and the answer extraction process of the QA model. Combined with additional question value estimation in pseudo labeling, we plan to explore such directions in adaptive QA systems as our future work.

Acknowledgments

This research is supported in part by the National Science Foundation under Grant No. IIS-2202481, CHE-2105005, IIS-2008228, CNS-1845639, CNS-1831669. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

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. Mix-Text: 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.

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.

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.

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. 2019a. Learning from on-line user feedback in neural question answering on the web. In *The World Wide Web Conference*, pages 906–916.

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

Bernhard Kratzwald, Stefan Feuerriegel, and Huan Sun. 2020. Learning a Cost-Effective Annotation Policy for Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3051–3062, Online. Association for Computational Linguistics.

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. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

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.

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.

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.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*.

Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176.

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.

Xiang Yue, Ziyu Yao, and Huan Sun. 2022a. Synthetic question value estimation for domain adaptation of question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1340–1351, Dublin, Ireland. Association for Computational Linguistics.

Xiang Yue, Xinliang Frederick Zhang, Ziyu Yao, Simon Lin, and Huan Sun. 2021a. Cliniqg4qa: Generating diverse questions for domain adaptation of clinical question answering. In *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 580–587. IEEE.

Zhenrui Yue, Bernhard Kratzwald, and Stefan Feuerriegel. 2021b. 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.

Zhenrui Yue, Huimin Zeng, Ziyi Kou, Lanyu Shang, and Dong Wang. 2022b. Contrastive domain adaptation for early misinformation detection: A case study on covid-19. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 2423–2433.

Zhenrui Yue, Huimin Zeng, Ziyi Kou, Lanyu Shang, and Dong Wang. 2022c. Defending substitution-based profile pollution attacks on sequential recommenders. In *Proceedings of the 16th ACM Conference on Recommender Systems*, pages 59–70.

Zhenrui Yue, Huimin Zeng, Ziyi Kou, Lanyu Shang, and Dong Wang. 2022d. Domain adaptation for question answering via question classification. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1776–1790, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Huimin Zeng, Zhenrui Yue, Ziyi Kou, Lanyu Shang, Yang Zhang, and Dong Wang. 2022a. Unsupervised domain adaptation for covid-19 information service with contrastive adversarial domain mixup. In *arXiv preprint arXiv:2210.03250*.

Huimin Zeng, Zhenrui Yue, Yang Zhang, Ziyi Kou, Lanyu Shang, and Dong Wang. 2022b. On attacking out-domain uncertainty estimation in deep neural networks. In *IJCAI*.

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

Yi Zhou, Xiaoqing Zheng, Cho-Jui Hsieh, Kai-Wei Chang, and Xuanjing Huang. 2021. Defense against synonym substitution-based adversarial attacks via Dirichlet neighborhood ensemble. 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 5482–5492, Online. Association for Computational Linguistics.

Appendix

A Dataset Details

For the source domain, we adopt **SQuAD v1.1** (Rajpurkar et al., 2016) following (Cao et al., 2020; Lee et al., 2020; Shakeri et al., 2020; Yue et al., 2021b). 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 all datasets from MRQA Split I (Fisch et al., 2019) for the target domains:

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. **NewsQA** (Trischler et al., 2016) provides news as contexts and challenging questions beyond simple matching and entailment.
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.
5. **TriviaQA** (Joshi et al., 2017) is a question-answering dataset containing evidence information for reasoning in QA.

B Baseline Details

As a naïve baseline, we adopt BERT (uncased base version with additional batch normalization layer) and train on the source dataset (Devlin et al., 2019; Cao et al., 2020). Additionally, we implemented the following three baselines for unsupervised QA domain adaptation:

1. **Domain adversarial training (DAT)** (Tzeng et al., 2017; Lee et al., 2019) consists of a QA system and a discriminator using `[CLS]` output in BERT. The QA system is first trained on labeled source data. Then, input data from both domains is used for domain adversarial training to learn generalized features.

2. **Conditional adversarial self-training (CASe)** (Cao et al., 2020) leverages self-training with conditional adversarial learning across domains. CASe iteratively perform pseudo labeling and domain adversarial training to reduce domain discrepancy. We adopt the entropy weighted CASe+E in our work as baseline.

3. **CAQA** (Yue et al., 2021b) leverages QAGen-T5 for question generation but extends the learning algorithm with a contrastive loss on token-level features for generalized QA features. Specifically, CAQA uses contrastive adaptation to reduce domain discrepancy and promote answer extraction.

4. **Self-supervised contrastive adaptation for QA (CAQA*)** (Yue et al., 2021b) is a modified self-supervised baseline based on CAQA. We exclude question generation and adopt the same process of pseudo labeling and self-supervised adaptation as in QADA. Unlike QADA, hidden space augmentation and attention-based contrastive loss are removed.

C Implementation Details

QA model: We adopt BERT with an additional batch norm layer after the encoder for QA domain adaptation, as in (Cao et al., 2020). We first pretrain BERT with a learning rate of $3 \cdot 10^{-5}$ for two epochs and a batch size of 12 on the source dataset. We use the AdamW optimizer with 10% linear warmup. We additionally use Apex for mixed precision training.

Adaptation: For the baselines, we use the original BERT architecture and follow the default settings provided in the original papers. For QADA, adaptation is performed 4 epochs with the AdamW optimizer, learning rate of $2 \cdot 10^{-5}$, and 10% proportion as warmup in each epoch (as training data changes after pseudo labeling). In the pseudo labeling stage, we perform inference on unlabeled target data and preserve the target samples with confidence above the threshold $\tau = 0.6$. For batching in self-supervised adaptation, we perform hidden space augmentation and sample 12 target examples and another 12 source examples.

QADA: For our experiments, the scaling factor λ for the adaptation loss is chosen from $[0.0001, 0.0005]$ depending on the target dataset. For Dirichlet neighborhood sampling, we use $\alpha = 1$ for the original token and a decay factor of 0.1 for multi-hop synonyms (i.e., 0.1 for 1-hop synonyms and 0.01 for 2-hop synonyms). For hyperparameters in hidden space augmentation, we search for a combination of question augmentation ratio ζ and context cutoff ratio φ. Specifically, we empirically search for the best combination in the range of $[0.1, 0.2, 0.3, 0.4]$ for both ζ and φ. Eventually, the best hyperparameter combination is selected.