DAML-ST5: Low Resource Style Transfer via Domain Adaptive Meta Learning

Anonymous ACL submission

Abstract

Text style transfer (TST) without parallel data has achieved some practical success. However, most of the existing unsupervised text style transfer methods suffer from (i) requiring massive amounts of nonparallel data to guide the transferring of different text styles. (ii) huge performance degradation when fine-tuning the model in new domains. In this work, we propose DAML-ST5, which consists of two parts, DAML and ST5. DAML is a domain adaptive meta-learning approach to refine general knowledge in multi-heterogeneous source domains, which is capable of adapting to new unseen domains with a small amount data. Moreover, we propose a new unsupervised TST model Style-T5 (ST5), which is composed of a sequence-to-sequence pre-trained language model T5 and uses style adversarial training for better content preservation and style transfer. Results on multi-domain datasets demonstrate that our approach generalize well on unseen low-resource domains, achieving state of the art results against ten strong baselines.

1 Introduction

Text style transfer (TST) aims to change the style of the input text and keep its content unchanged, which has been applied successfully to text formalization (Jain et al., 2019) , text rewriting (Nikolov and Hahnloser, 2018) , personalized dialogue generation (Niu and Bansal, 2018) and other stylized text generation tasks (Gao et al., 2019; Cao et al., 2020; Syed et al., 2020).

Text style transfer has been explored as a sequence-to-sequence learning task using parallel datasets (Jhamtani et al., 2017; Wang et al., 2020b; Pryzant et al., 2020). However, parallel datasets are difficult to obtain due to expensive manual annotation. The recent surge of deep generative methods (Hu et al., 2017a; Zhao et al., 2017; Li et al., 2018) has spurred progress in text style transfer without parallel data. However, these methods typically require large amounts of nonparallel data and may not perform well in some low-resource domain scenarios.

One typical method is to resort to massive data from different domains, which has been studied as an effective solution to address the above data insufficiency issue (Glorot et al., 2011; Wang et al., 2017). However, directly leveraging large amounts of data from other domains for TST task is problematic due to the differences in data distribution over different domains, as different domains usually use their own domain-specific lexica (Li et al., 2019a). For instance, if we use the TST model trained on high-resource movie domain (source domain) and fine-tune it on low-resource restaurant domain (target domain), we may get unreasonable sentences like "the food is dramatic", where the sentiment word "dramatic" is typically used in movie domain. This is the domain adaption issue that often occurs in text style transfer due to inconsistency between source domain and target domain.

In this work, we tackle the problem of domain adaptation in the scenarios where the target domain data is scarce and misaligned with the distribution in the source domain. Recently, model-agnostic meta-learning (MAML) has received resurgence in the context of few-shot learning scenario (Lin et al., 2019; Gu et al., 2018; Li et al., 2020; Nooralahzadeh et al., 2020). Inspired by the essence of MAML (Qian and Yu, 2019), we propose a new meta learning training strategy named domain adaptive meta learning (DAML). Different from MAML, DAML adopts a domain adaptive approach to construct meta tasks which would be more suitable to learn a robust and generalized initialization for low-resource TST domain adaption.

With the DAML strategy, we design a TST model for each domain. Usually, if a TST model tries to decouple style information from the semantics of a text, it tends to produce content loss during style transfer (Hu et al., 2017b; Dai et al., 2019; Carlson et al., 2018). Thus, we propose a new style transfer model Style-T5 (ST5), which is composed of a sequence-to-sequence pre-trained language model T5 (Raffel et al., 2019) and uses style adversarial training for style transfer. In this way, ST5 can better preserve the content information without disentangling content and style in the latent space.

Combining DAML and ST5, in this paper, we propose the method named DAML-ST5, which extends traditional meta-learning to a domain adaptive method combined with a sequence-to-sequence style transfer model. DAML is trained in two alternating phases, during the meta-training phase, a series of meta-tasks are constructed from a large pool of source domains for balanced absorption of general knowledge, resulting in domain specific temporary model. In the meta validation stage, the temporary model is evaluated on the meta validation set to minimize domain differences and realize meta knowledge transfer across different domains. In ST5, a pre-training language model based TST model is used to improve text content retention. Moreover, we propose a two-stage training algorithm to better combine DAML training method and ST5 model.

In summary, the main contributions in this paper are three-fold: (i) We propose a new unsupervised TST model, which achieves sota performance without disentangling content and style latent representations compared to other models. (ii) We extend the traditional meta-learning strategy to domain adaptive meta transfer method, which effectively alleviate the domain adaption problem in TST. (iii) We propose a two-stage training algorithm to train DAML-ST5, which achieves state-of-the-art performance against multiple strong baselines.

2 Related Work

2.1 Text Style Transfer

Text style transfer based on deep learning has been extensively studied in recent years. A common pattern is to first separate the latent space as content and style features, and then adjust the style-related features and generate stylistic sentences through the decoder. (Hu et al., 2017a; Fu et al., 2017; Li et al., 2019a)assumes that the separation can be achieved through appropriate style regularization in an automatic encoding process, which is achieved by adversarial discriminator or style classifier. However, these style transfer paradigms use large amounts of annotation data to train models for specific tasks. Obviously, if we already have a model for a similar task, it is unreasonable to still need a lot of data to train the model from scratch.

On the other hand, some of previous work learn to do TST without manipulate the style of the generated sentence based on this learned latent space. (Dai et al., 2019)uses the transformer architecture language model to introduce attention mechanism, but they do not make full use of the prior knowledge of sequence to sequence pre-trained language model, such as Bart (Lewis et al., 2019) and T5 (Raffel et al., 2019), which have made great progress in text generation tasks. In this paper, we not only proposed DAML training method to solve the domain shift problem in TST, but also proposed a new TST model architecture named Style-T5, which makes no assumption about the latent representation of source sentence and takes the proven sequence-to-sequence pre-trained language model.

2.2 Domain adaptation

Domain adaptation has been studied in various natural language processing tasks, such as sentiment classification (Glorot et al., 2011), dialogue systems (Qian and Yu, 2019), machine translation (Wang et al., 2017), etc. However, there is no recent work about domain adaptation for a TST, except DAST (Li et al., 2019a). DAST is a semi-supervised learning method that adapts domain vectors to adapt models learned from multiple source domains to a new target domain via domain discriminator. Different From DAST, we propose to combine meta-learning and adversarial networks to achieve similar domain adaption ability, and our model exceeds the performance of DAST without domain discriminator.

2.3 Model-Agnostic Meta-Learning

Model-agnostic meta-learning (MAML) (Finn et al., 2017) provides a general method to adapt to parameters in different domains. MAML solves few shot learning problems by learning a good parameter initialization. During testing, such initialization can be fine-tuned through a few gradient steps, using a limited number of training examples in the target domain. Although there have been some researches (Qian and Yu, 2019; Li et al., 2020; Wu et al., 2020) on MAML in natural language processing, it is still scarce compared to computer vision. Different from the above research on classification under a few-shot learning, our research focuses on text style transfer based on text generation. In this paper, we are seeking a new meta-learning strategy combined with adversarial networks, which is more suitable for encouraging robust domain representation. As far as we know, we are the first to try to adopt meta-learning in text style transfer domain adaptation.

3 Methodology

In this section, we first define the problem of domain adaptive learning for TST. Then we describe our approach DAML-ST5 in detail.

3.1 Task Definition

Let $D_S = \{D_1, ..., D_N\}$ be N source domains in the training phase, where $D_n (1 \leq n \leq N)$ is the n-th source domain containing style-labelled non-parallel data $D_n = \{(X_i, l_i)\}_{i=1}^{L_n}$, where L_n is the total number of sentences, X_i denotes the i^{th} source sentence, and l_i denotes the corresponding style label, which belongs to a source style label set: $l_i \in L_S$ (e.g., positive/negative). Likewise, there are K target domains $D_T = \{D_1, ..., D_K\}$ which are unseen in D_S. Our task is to transfer a sentence X_i with style l_i in the target domain to another sentence $Y_i^{'}$ sharing the same content while having a different style $\tilde{l}_i$ from l_i and domain-specific characteristics of the target domain.

To make domain adaptation in TST, we propose a two-stage algorithm: pre-training learning strategy and domain adaptive meta learning strategy. In pre-training learning, our objective is to make the model more able to preserve content information and distinguish between different text styles. In domain adaptive meta learning, our objective is to learn a meta-knowledge learner for the sequence-to-sequence model by leveraging sufficient source data D_s. Given a new unseen domain from D_{new} , the new learning task of TST can be solved by fine-tuning the learned sequence-to-sequence model (domain-invariant parameters) with only a small number of training samples.

3.2 DAML-ST5 Approach

3.2.1 Overview of Domain Adaptive Meta-Learning

Model-agnostic meta-learning can utilize a few training samples to train a model with good generalization ability. However, since it is based on the assumption that the meta tasks are from the same distribution (Figure 2, left), simply feeding all the sources data into it might get sub-optimal results (Chen and Zhu, 2020). Therefore, we propose a modified way to construct meta tasks (Figure 2, right). Different from MAML, for DAML, in one batch, the data in each meta task comes from the same source domain and each meta task comes from a different domain. In this way, we can guarantee that DAML can learn generic representations from different domains in a balanced way. During each iteration, we randomly split all source domains into a meta-training set D_{tr} and a meta-validation set D_{val}, where $D_S = D_{tr} \cup D_{val}$ and $D_{tr} \cap D_{val} = \emptyset$. A meta-training task T_i is sampled from D_{tr} and is composed of n instances from a specific domain. Likewise, a meta-validation task T_j is sampled from D_{val}. The validation errors on D_{val} should be considered to improve the robustness of the model. In short, with DAML, the parameters learned by the model in the parameter space are not biased towards any one particular domain s with as little data as possible during model updating as shown in Figure 2(right).

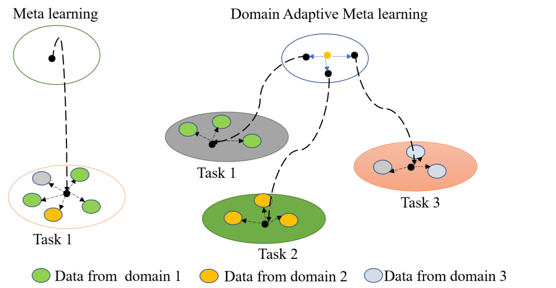

Figure 1: Comparison of meta learning and domain adaptive meta transfer learning (DAML). In DAML, each meta task contains n sentences from the same domain. In MAML, the data in each meta task comes from different domains.

In the final evaluation phase, the meta-knowledge learned by the sequence-to-sequence mode can be applied to new domains. Given a new unseen domain $D_{new} = (T_{tr}, T_{te})$, the learned sequence-to-sequence model and the discriminator are fine-tuned on T_{tr} and finally tested on T_{te}.

3.2.2 Style-T5 model

In this section, we give a brief introduction to our proposed model: **Style-T5** , which combine sequence-to-sequence pre-trained model T5 (Raffel et al., 2019) with a discriminator model. (1) For the content preservation, we train the sequence-to-sequence T5 model θ to reconstruct original input

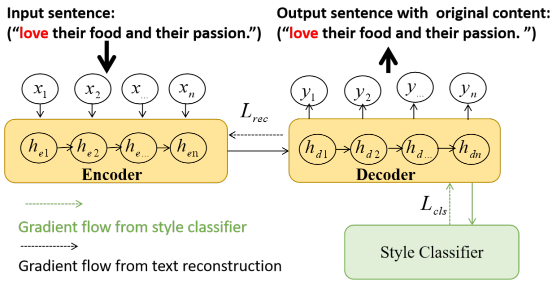

Figure 2: The basic structure of our TST model Style-T5 (ST5) with first stage training procedure. The green dashed line represents the loss of style classification to ensure that the style classifier can distinguish between different text styles. The black dotted line rerents text reconstruction loss to ensure the generated sentence has a similar semantic meaning as the input sentence.

sentence X with the original style label l. (2) For the style controlling, we train a discriminator network γ to assist the sequence-to-sequence model network to better control the style of the generated sentence. The structure of the model is shown in Figure 2.

T5-model To ease the explanation, we start with sequence-to-sequence model T5 here. T5 is a sequence-to-sequence pre-trained language model proposed by (Raffel et al., 2019), which follows the standard transformer (Vaswani et al., 2017) encoder-decoder architecture. Explicitly, for a input sentence $X = (x_1, x_2, ..., x_n)$ of length n, $X \in D$, the T5 encoder $Enc(X; \theta_E)$ maps inputs to a sequence of continuous hidden representations $H = (h_1, h_2, ..., h_n)$. Then, the T5 decoder $Dec(H; \theta_D)$ estimates the conditional probability for the output sentence $Y = (y_1, y_2, ..., y_n)$ by auto-regressively factorized its as:

$$p_\theta(Y|X) = \prod_{t=1}^{n} p_\theta(y_t|H, y_1, ..., y_{t-1}) \quad (1)$$

At each time step t, the probability of the next token is computed by a softmax classifier:

$$p_\theta(y_t|H, y_1, ..., y_{t-1})) = softmax(o_t) \quad (2)$$

where o_t is logit vector outputted by decoder network. For T5 model, which let the output sequence Y to be the same as the input sequence X by teacher forcing.

Discriminator Model However,by teacher forcing, T5 model tends to ignoring the style labels and collapses to a reconstruction model, which might simply copy the input sentence, hence fails to transfer the style. Therefore, to make the model learn meaningful style information, we apply a style discriminator γ for the style regularization. In summary, we use a style discriminator to provide the direction (gradient) for TST so that it conforms to the target style. Our discriminator is a multi-layer perceptron with a sigmoid activation function to predict style labels. Our model training involves two stages: pre-training learning strategy and domain adaptive meta learning strategy.

3.2.3 First Stage: Pre-training Learning Strategy

In the first stage, we train the discriminator model so that it can classify different text styles. In this stage, the discriminator models are equivalent to a text classifier. Inspired by (Lewis et al., 2019), we feed the hidden states from the last layer of the decoder into the classifier instead of the gumble-softmax trick (Jang et al., 2017) for gradient backpropagation, which is more stable than gumble-softmax. The loss function for the discriminator is simply the cross-entropy loss of the classification problem:

$$\mathcal{L}_{cls}(\gamma) = -\mathop{\mathbb{E}}_{X_i \sim D_S}[logP(l_i|X_i, l_i; \theta, \gamma)] \quad (3)$$

For sequence-to-sequence model, we pre-train the encoder and the decoder to allow the generation model to learn to copy an input sentence X using teacher forcing. The loss function of the sequence-to-sequence model minimizes the negative log-likelihood of the training data:

$$\mathcal{L}_{rec}(\theta) = -\mathop{\mathbb{E}}_{X_i \sim D_S}[logP(Y_i|X_i; \theta)] \quad (4)$$

In summary, in the first stage, we train the sequence model and the style classification model separately on source domain so that they learn content preservation and style discrimination respectively. The first stage training procedure of the Style-T5 is summarized in Algorithm 1.

3.2.4 Second Stage: Domain Adaptive Meta Learning Strategy

After the first stage of training, the style classifier has learned how to distinguish between different text styles. Therefore, in the second stage we use the trained text classifier to provide the direction for TST. For style controlling, we adopt method of adversarial training to avoid disentangling the

Algorithm 1 Style-T5 Pre-traing Learning

Input: sequence-to-sequence model f_θ,discriminator γ,and a dataset D_i with style l_i belong to L_s
Output: well-trained parameter θ, γ

Sample a batch of m sentences $X_1, X_2, ...X_m$ from D_i.
while in first stage and not convergence **do**
  Use f_θ to generate new sentence
  $Y_i = f_\theta(X_i, l_i)$
  Compute $\mathcal{L}_{cls}(\gamma)$ for Y_i by Eq. (4) ;
  Compute $\mathcal{L}_{rec}(\theta)$ for Y_i by Eq. (3) ;

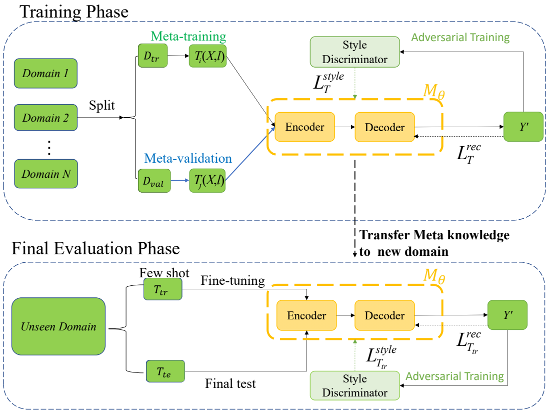

Figure 3: Overview of our proposed DAML-ST5 with second stage training strategy. In the meta-training phase, a temporary model $(\theta_{old}, \theta_{new})$ is learned from D_{tr} . In the meta-validation phase, the base model is updated by gradient descent with respect to the parameters θ on D_{val} . In the final evaluation phase, the learned sequence encoder is fine-tuned on T_{tr} and tested on T_{te} from a unseen domain D_{new}.

content and style in the latent space, the discriminator model aims to minimize the negative loglikelihood of opposite style $\tilde{l}_i$ when feed to the sequence model sentence X_i with the style label l_i :

$$\mathcal{L}_{style}(\theta) = -\mathop{\mathbb{E}}_{X_i \sim D}[logP(\tilde{l}_i|X_i, l_i; \theta, \gamma)] \quad (5)$$

In the second stage, we use DAML algorithm for domain adaptive TST, so the text reconstruction loss and the style discriminator loss are calculated over the meta-training samples in task T_i from D_{tr}. These two losses can be written as

$$\begin{aligned}\mathcal{L}^{rec}_{T_i}(\theta) &= -\mathop{\mathbb{E}}_{X_i \sim T_i}[logP(Y_i|X_i; \theta)] \\ \mathcal{L}^{style}_{T_i}(\theta) &= -\mathop{\mathbb{E}}_{X_i \sim T_i}[logP(\tilde{l}_i|X_i, l_i; \theta, \gamma))]\end{aligned} \quad (6)$$

The second stage of the algorithm is called domain adaptive meta strategy, which consists of two core phase: a meta-training phase and a meta-validation phase, as shown in Figure 3.

Domain Adaptive Meta-Training. In the meta-training phase, our objective is to learn different domain specific temporary models for each domain that are capable of learning the general knowledge of each domain. Inspired by feature-critic networks (Li et al., 2019b), we use a a similar manner to adapt the parameters of domain specific temporary model:

$$\begin{aligned}\theta^{old}_i &= \theta_{i-1} - \alpha\nabla\theta_{i-1}\mathcal{L}^{rec}_{T_i}(\theta_{i-1}, \gamma_{i-1}) \\ \theta^{new}_i &= \theta^{old}_{i-1} - \alpha\nabla\theta_{i-1}\mathcal{L}^{style}_{T_i}(\theta_{i-1}, \gamma_{i-1})\end{aligned} \quad (7)$$

where i is the adaptation step in the inner loop, and α is the learning rate of the inner optimization. At each adaptation step, the gradients are calculated with respect to the parameters from the previous step. For each domain of D_{tr}, it has different θ^{old} and θ^{new} . The base model parameters θ_0 should not be changed in the inner loop.

Algorithm 2 The training procedure of DAML-ST5

Input: $\mathcal{D} = \{\mathcal{D}_1, ..., \mathcal{D}_K\}, \alpha, \beta$
Output: optimal meta-learned model θ

Initialize the base sequence-to-sequence model θ and discriminator model γ by algorithm 1
while not converge **do**
  Randomly split $\mathcal{D} = \mathcal{D}_{tr} \cup \mathcal{D}_{val}$ and $\mathcal{D}_{tr} \cap \mathcal{D}_{val} = \emptyset$
  Meta-training:
  for j in meta batches **do** //Outer loop
    Sample a task T_j from D_{val}
    for i in adaptation steps **do** //Inner loop
      Sample a task T_i from D_{tr}
      Compute meta-training rec loss $\mathcal{L}^{rec}_{T_i}$
      Compute meta-training style loss $\mathcal{L}^{style}_{T_j}$
      Compute adapted parameters with gradient descent for θ_{i-1}
      $\theta^{old}_i = \theta_{i-1} - \alpha\nabla\theta_{i-1}\mathcal{L}^{tr}_{T_i}(\theta_{i-1}, \gamma_{i-1})$
      $\theta^{new}_i = \theta^{old}_{i-1} - \alpha\nabla\theta_{i-1}\mathcal{L}^{style}_{T_i}(\theta_{i-1}, \gamma_{i-1})$
    Meta-validation:
    Compute meta-validation loss on T_j: $\mathcal{L}^{val}_{T_j}$
  Meta-optimization:
  Perform gradient step w.r.t. θ
  $\theta_0 = \theta_0 - \beta\nabla_{\theta_0}\mathbb{E}_{T_j}\mathcal{L}^{val}_{T_j}(\theta^{old}_i, \theta^{new}_i, \gamma)$

Domain Adaptive Meta-Validation After meta-training phase, DAML-ST5 has already learned a temporary model$(\theta^{old}_i, \theta^{new}_i)$ in the meta-training domains D_{tr}. The meta-validation phase tries to minimize the distribution divergence between the source domains D_{tr} and simulated target domains D_{val} using the learned temporary model. In the meta-validation phase, each temporary model is calculated on the meta-validation domain D_{val} to get meta validation losses.

$$\mathcal{L}^{val}_{T_j} = \mathcal{L}^{rec}_{T_j}(\theta^{old}_i, \gamma_0) + \mathcal{L}^{style}_{T_j}(\theta^{new}_i, \gamma_0) \quad (8)$$

Thus, the base model θ is updated by gradient descent

$$\theta_0 = \theta_0 - \beta \nabla \theta_0 \mathcal{L}_{T_j}^{val} \quad (9)$$

where β is the meta-learning rate. Unlike the ordinary gradient descent process, the update mechanism of Eq. (9) involves updating one gradient by another gradient (w.r.t. the parameters of the temporary model), this process requires a second-order optimization partial derivative.

3.2.5 Final Evaluation Phase of DAML-ST5

In the final evaluation phase, we first initialize the model with the parameters learned during the above algorithm 2. Then, the model takes input as a new adaptation task T, which consists of a small in-domain data S_{tr} for fine-tuning the model and a test set S_{te} for testing. The procedure is summarized in Algorithm 3.

Algorithm 3 The Final Evaluation Procedure of DAML-ST5

Input: θ, γ learned from Algorithm 2, low resource training set S_{tr} and test set S_{te} of an unseen domain D_{new}
Output: Performance on S_{te}

```
while not convergence do
    Serialize a task T_tr from the unseen domain S_tr
    Update θ = θ − β∇_θ Σ_{T_tr} (L^rec_{T_tr}(θ) + L^style_{T_tr}(θ))
return optimal θ* for S_te
Style accuracy, bleu, domain accuracy = f_{T_te}(θ)
```

Dataset	Domain	Train	Dev	Test	Human Reference
Yelp	Restaurant	444k	4k	1k	1k
Amazon	Product	554k	2k	1k	1k
IMDB	Movie	341k	2k	1k	No
Yahoo!	Q & A	5k	1k	1k	No

Table 1: Statistics of source and target datasets(non-parallel data). The style label set is {negative, positive}.

4 Experiment

In this section, we first detail the experimental setups. Then, we present our experimental results over multiple target domains.

4.1 Datasets and Experimental Setups

In this experiment, we use the following four datasets from different domains: (i) IMDB movie review corpus (Diao et al., 2014). (ii) Yelp restaurant review dataset (Li et al., 2018). (iii) Amazon product review dataset (Li et al., 2018). (iv) YAHOO! Answers dataset (Li et al., 2019a), the amazon and yelp test sets each have 1k human annotations.The statistics of these corpora are summarized in Table 1.

T5 model is implemented by Huggingface Transformers (Wolf et al., 2020), taking the T5 base model (220MB) for our experiments. For style discriminator, we use 4-layer fully connect neural networks. We train our framework using the Adam optimizer (Kingma and Ba, 2014)with the initial learning rate 1e-5, the epoch is set to 50 for both stage 1 and stage 2. The inner learning rate α is 0.0001 and outer learning rate β is 0.001. Following (Shankar et al., 2018; Li et al., 2020), we use leave-one-out evaluation method by picking a domain as the target domain D_{new} for the final evaluation. For each iteration of the training phase, two source domains are randomly selected as the meta-training domain D_{tr} and the remaining domains as the meta-validation domain D_{val}.

In order to evaluate the model performance, we use three popular and widely adopted automatic metrics following previous work (Li et al., 2019a; Fu et al., 2017; Hu et al., 2017a) and a human metric . **BLEU** verifies whether the generated sentences retain the original content (Papineni et al., 2002). While IMDB and Amazon have no manual references, we compute the BLEU scores w.r.t the input sentences. **Style Control** (S-Acc) measures the style accuracy of the transferred sentences with a style classifier that is pre-trained on the datasets. **Domain Control** (D-Acc) verifies whether the generated sentences have the characteristics of the target domain with a pre-trained domain classifier to measure the percentage of generated sentences belonging to the target domain. **Human Evaluation** Following (Madotto et al., 2019), We randomly sampled 100 sentences generated on the target domain and distributed a questionnaire at Amazon Mechanical Turk asking each worker to rank the content retention (0 to 5), style transfer(0 to 5) and fluency(0 to 5): $human\ score = Average(\sum score_{style} + \sum score_{content} + \sum score_{fluency}), human\ score \in [0, 100]$. Five workers are recruited for human evaluation.

4.2 Baselines

In our experiments, for ST5 model, we adopt five state-of-the-art TST models for comparison: CrossAlign (Shen et al., 2017), ControlGen (Hu et al., 2017a), DAST (Li et al., 2019a), CatGen (Wang et al., 2020a) and FGIM (Wang et al.,

	Restaurant(1% target domain data)					Restaurant(100% target domain data)				
Model/Training method	S-Acc	BLEU	G-score	Human	D-Acc	S-Acc	BLEU	G-score	Human	D-Acc
CrossAlign	78.4	4.5	18.7	14.6	76.8	88.3	5.6	22.2	70.3	83.5
ControlGen	80.1	6.7	23.2	15.4	80.4	90.6	25.5	22.5	78.9	87.9
FGIM	83.1	4.6	19.6	16.4	82.0	90.4	24.6	48.6	69.4	85.2
DAST	88.3	17.5	39.3	19.5	**90.5**	91.2	26.5	49.2	79.4	92.6
CatGen	85.4	18.5	39.7	29.4	80.5	88.4	27.9	49.7	65.7	86.0
ST5(ours)	**89.6**	**20.1**	**42.4**	**30.1**	89.2	**93.3**	**30.3**	**53.2**	**85.2**	**93.4**
In-Domain	87.4	9.7	29.1	16.4	87.3	94.5	20.4	43.9	78.4	93.6
Joint-Training	82.3	8.4	26.2	18.7	84.6	85.4	21.6	42.9	73.6	93.4
Fine-Tuning	65.2	2.8	13.5	12.6	79.8	92.8	24.2	47.3	73.7	93.7
D-Shift	79.3	10.4	28.7	15.4	79.8	91.2	23.4	46.1	73.7	93.7
MAML	88.2	18.6	40.5	24.8	74.5	90.4	20.1	42.6	70.4	92.1
DAML(ours)	**90.0**	**21.4**	**43.8**	**25.1**	**89.9**	**96.7**	**32.1**	**55.7**	**80.2**	**94.7**
DAML-ST5(ours)	**94.5**	**25.4**	**48.9**	**34.2**	**92.9**	**97.8**	**35.5**	**58.9**	**83.1**	**96.4**

Table 2: Evaluation results on restaurant domain(Yelp). The restaurant domain is used as the target domain and the other three domains as the source domain. G-score is the geometric mean of S-Acc and BLEU.

	Yelp(negative-to-positive)	Yelp(positive-to-negative)
Input	there chips are ok , but their salsa is really bland.	love their food and their passion.
Joint-Training	there are good , but their food is really good. .	laughable their food and bad food.
Fine-Tuning	there chips act very well.	their food is hard to use.
D-Shift	there are usually dramatic exhibits.	my husband and toilet smelled.
MAML	there chips are bad,but there salsa is really good.	hate their food and their passion
DAML-ST5(ours)	there chips are surprised, and their salsa is really nice.	hard to swallow food and serious discrespect.

Table 3: Transferred sentences on Yelp(few shot), where red denotes successful style transfers, blue denotes content losses, violet denotes domain errors and green denotes grammar errors, better looked in color. More examples are in the appendix.

Movie	In-Domain	Fine-Tuning	D-Shift	MAML	**DAML**
S-Acc	70.4	59.3	74.4	79.8	**81.5**
BLEU	23.1	25.4	27.4	26.9	**31.2**
D-Acc	87.3	75.2	72.2	74.5	**92.3**
Product	In-Domain	Fine-Tuning	D-Shift	MAML	**DAML**
S-Acc	84.1	80.2	83.5	84.6	**87.0**
BLEU	14.0	14.5	17.8	18.1	**19.9**
D-Acc	80.5	75.4	73.5	79.4	**84.1**
Q & A	In-Domain	Fine-Tuning	D-Shift	MAML	**DAML**
S-Acc	94.1	90.1	92.1	89.6	**95.5**
BLEU	12.8	13.7	14.5	18.7	**20.5**
D-Acc	80.6	70.0	72.5	76.5	**86.7**

Table 4: Results on each of the remaining domains treated as target domain,every target domains using 1% data for fine-tuning, base model is ST5.

2019). They are jointly trained on the source domains and fine-tuned on the target domain.

To well analyze our training method DAML, following (Li et al., 2020), we also use five simple and effective domain adaptation settings with ControlGen (Hu et al., 2017a) structure as DAML: (1) **In-Domain** method is trained on the training set of the target domain; (2) **Joint-Training** method combines all the training sets of the source and target domains and performs a joint-training on these datasets; (3) **Fine-Tuning** method is trained on the training sets of the source domains and then fine-tuned on the training set of the target domain; (4) **D-Shift** This is trained on the combination of training sets from all source domains. Then, the evaluation is conducted on the test set of a target domain using the direct domain shift strategy; (5) **MAML** method uses classical model agnostic meta-learning algorithm (Finn et al., 2017).

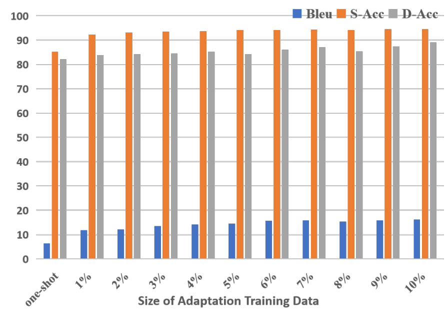

Figure 4: The system performance on amazon improves when the size of the target data increases. Even the one-shot learning achieves decent performance.

4.3 Results and Analysis

For DAML-ST5, we first choose restaurant as the target domain and the other three as the source domains for observation. Table 2 reports the results of different methods and models under both the full-data and few-shot settings. From this table, we can see that DAML-ST5 outperforms all baselines

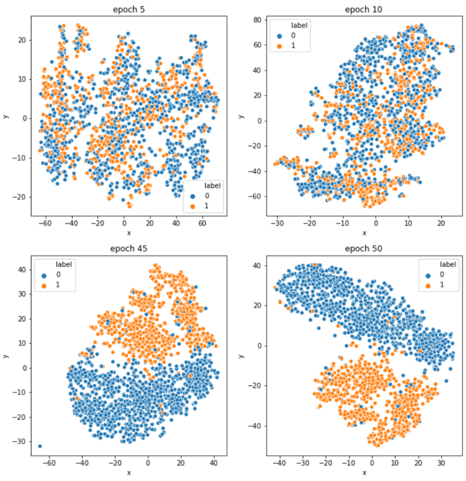

Figure 5: The t-sne plots of source domain sentences and generated target domain sentence in different DAML training epochs. The labels 0 and 1 represent the source domain sentence embedding and generated target domain sentence embedding respectively.

in terms of *S-Acc*, *BLEU*, *D-Acc* and human evaluation. We attribute this to the fact that DAML-ST5 explicitly simulates the domain shift during training via DAML, which helps to adapt to the new target domain. We can also see that in the case of few-shot setting, the results of *Fine-tuning* and *Joint training* are even worse than *In-domain* and DAML. The reason may be that the data size of the source domain is much larger than target domain, so that the model tend to remember the characteristics of the source domain. MAML achieves good performance in most metrics, however, it does not balance meta-tasks across different source domains so that it performs badly on D-acc.

Further, in order to verify the robustness of our method under the low-resource setting, we select the other three domains as the target domain respectively. As shown in Table 4, our approach has achieved good performance on different target domains.

We also provide some examples in Table 3 . From the example, we can see intuitively that *D-shift* and *Fine-tuning* will lead to the misuse of domain-specific words due to lack of target domain information. In addition, compared with *Joint-training*, the sentences generated by DAML-ST5 are more consistent with the human reference. Compared to MAML, DAML generates sentences that are more diverse and vivid due to the more balanced absorption of information from multiple domains. Figure 4 shows the system performance positively correlates with the amount of training data available in the target domain. To visualize how well DAML-ST5 performs on the new unseen domain, we use t-SNE (Van der Maaten and Hinton, 2008) plots to analyze the degree of separation between the source domain sentences and the generated target domain sentences. Figure 5 shows that as the training epoch increases, the sentences generated by DAML-ST5 in the target domain are completely separated from the source domain in the latent space.

4.4 Ablation Study

To study the impact of different components on the overall performance, we further did an ablation study for our model, and the results are shown in Table 5. After we disabled the reconstruction loss, our model failed to learn meaningful outputs and only learned to generate a word for any combination of input sentences and styles. Then, when the discriminator loss is not used, the model degrades rapidly, which will simply copy the original sentence without any style modification. After not using the pretraining language model weights, the performance of the model is reduced in the metric of content preservation. When using gumble-softmax instead of hidde states for gradient descent, the model performs poorly in terms of style accuracy because of the instability of gumble-softmax. In summary, each of these factors plays an important role in the DAML-ST5 training stage.

Model	S-Acc	BLEU	D-Acc
DAML-ST5	**94.5**	**25.4**	**92.9**
w/o reconstruction loss	50.0	0	50.0
w/o discriminator loss	2.1	21.6	93.4
w/o language model weights	87.4	17.3	90.3
w/ gumble-softmax	85.6	18.3	91.0

Table 5: Model ablation study results on Yelp dataset. The size of adaptation training data is 1 %.

5 Conclusion

In this paper, We propose DAML-ST5, a novel training strategy combined with a new TST model for domain adaptation, which can be easily adapted to new domains with few shot data. On four popular TST benchmarks, we found significant improvements against multiple baselines, verifying the effectiveness of our method. In future work, we explore to extend this approach for other low resource tasks in NLP.

References

Yixin Cao, Ruihao Shui, Liangming Pan, Min-Yen Kan, Zhiyuan Liu, and Tat-Seng Chua. 2020. Expertise style transfer: A new task towards better communication between experts and laymen. *arXiv preprint arXiv:2005.00701*.

Keith Carlson, Allen Riddell, and Daniel Rockmore. 2018. Evaluating prose style transfer with the bible. *Royal Society open science*, 5(10):171920.

Xiwen Chen and Kenny Q Zhu. 2020. ST^2: Small-data text style transfer via multi-task meta-learning. *arXiv preprint arXiv:2004.11742*.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. 2019. Style transformer: Unpaired text style transfer without disentangled latent representation. *arXiv preprint arXiv:1905.05621*.

Qiming Diao, Minghui Qiu, Chao-Yuan Wu, Alexander J Smola, Jing Jiang, and Chong Wang. 2014. Jointly modeling aspects, ratings and sentiments for movie recommendation (jmars). In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 193–202.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2017. Style transfer in text: Exploration and evaluation.

Xiang Gao, Yizhe Zhang, Sungjin Lee, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2019. Structuring latent spaces for stylized response generation. *arXiv preprint arXiv:1909.05361*.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *ICML*.

Jiatao Gu, Yong Wang, Yun Chen, Kyunghyun Cho, and Victor O. K. Li. 2018. Meta-learning for low-resource neural machine translation.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. 2017a. Toward controlled generation of text.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017b. Toward controlled generation of text. In *International Conference on Machine Learning*, pages 1587–1596. PMLR.

Parag Jain, Abhijit Mishra, Amar Prakash Azad, and Karthik Sankaranarayanan. 2019. Unsupervised controllable text formalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6554–6561.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. 2017. Shakespearizing modern language using copy-enriched sequence-to-sequence models. *arXiv preprint arXiv:1707.01161*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Dianqi Li, Yizhe Zhang, Zhe Gan, Yu Cheng, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2019a. Domain adaptive text style transfer. *arXiv preprint arXiv:1908.09395*.

Jing Li, Shuo Shang, and Ling Shao. 2020. Metaner: Named entity recognition with meta-learning. In *Proceedings of The Web Conference 2020*, pages 429–440.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: A simple approach to sentiment and style transfer. *arXiv preprint arXiv:1804.06437*.

Yiying Li, Yongxin Yang, Wei Zhou, and Timothy M. Hospedales. 2019b. Feature-critic networks for heterogeneous domain generalization.

Zhaojiang Lin, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2019. Personalizing dialogue agents via meta-learning.

Andrea Madotto, Zhaojiang Lin, Chien-Sheng Wu, and Pascale Fung. 2019. Personalizing dialogue agents via meta-learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5459.

Nikola I Nikolov and Richard HR Hahnloser. 2018. Large-scale hierarchical alignment for data-driven text rewriting. *arXiv preprint arXiv:1810.08237*.

Tong Niu and Mohit Bansal. 2018. Polite dialogue generation without parallel data. *Transactions of the Association for Computational Linguistics*, 6:373–389.

Farhad Nooralahzadeh, Giannis Bekoulis, Johannes Bjerva, and Isabelle Augenstein. 2020. Zero-shot cross-lingual transfer with meta learning. *arXiv preprint arXiv:2003.02739*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Reid Pryzant, Richard Diehl Martinez, Nathan Dass, Sadao Kurohashi, Dan Jurafsky, and Diyi Yang. 2020. Automatically neutralizing subjective bias in text. In *Proceedings of the aaai conference on artificial intelligence*, volume 34, pages 480–489.

Kun Qian and Zhou Yu. 2019. Domain adaptive dialog generation via meta learning. *arXiv preprint arXiv:1906.03520*.

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*.

Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. 2018. Generalizing across domains via cross-gradient training. *arXiv preprint arXiv:1804.10745*.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in neural information processing systems*, pages 6830–6841.

Bakhtiyar Syed, Gaurav Verma, Balaji Vasan Srinivasan, Anandhavelu Natarajan, and Vasudeva Varma. 2020. Adapting language models for non-parallel author-stylized rewriting. In *AAAI*, pages 9008–9015.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Ke Wang, Hang Hua, and Xiaojun Wan. 2019. Controllable unsupervised text attribute transfer via editing entangled latent representation.

Rui Wang, Masao Utiyama, Lemao Liu, Kehai Chen, and Eiichiro Sumita. 2017. Instance weighting for neural machine translation domain adaptation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1482–1488.

Tianlu Wang, Xuezhi Wang, Yao Qin, Ben Packer, Kang Li, Jilin Chen, Alex Beutel, and Ed Chi. 2020a. Cat-gen: Improving robustness in nlp models via controlled adversarial text generation. *arXiv preprint arXiv:2010.02338*.

Yunli Wang, Yu Wu, Lili Mou, Zhoujun Li, and Wenhan Chao. 2020b. Formality style transfer with shared latent space. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2236–2249.

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

Qianhui Wu, Zijia Lin, Guoxin Wang, Hui Chen, Börje F Karlsson, Biqing Huang, and Chin-Yew Lin. 2020. Enhanced meta-learning for cross-lingual named entity recognition with minimal resources. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9274–9281.

Jake Zhao, Yoon Kim, Kelly Zhang, Alexander M. Rush, and Yann LeCun. 2017. Adversarially regularized autoencoders.

A Appendix

A.1 Human Evaluation Questionnaire

For the results generated by each method, we randomly selected 100 sentences to be placed in the Amazon Mechanical Turk[1] questionnaire. As shown in Figure 6, the questionnaire asked to judge the generated sentences on three dimensions: strength of style transfer, degree of content retention, and text fluency. To minimize the impact of spamming, we require each worker to be a native English speaker.

A.2 More Generation Examples

To demonstrate more examples of generation to verify the effectiveness of the model, we selected 10 generated sentences from amazon and yelp each, as shown in Table 6 and Table 7.

[1] https://www.mturk.com/

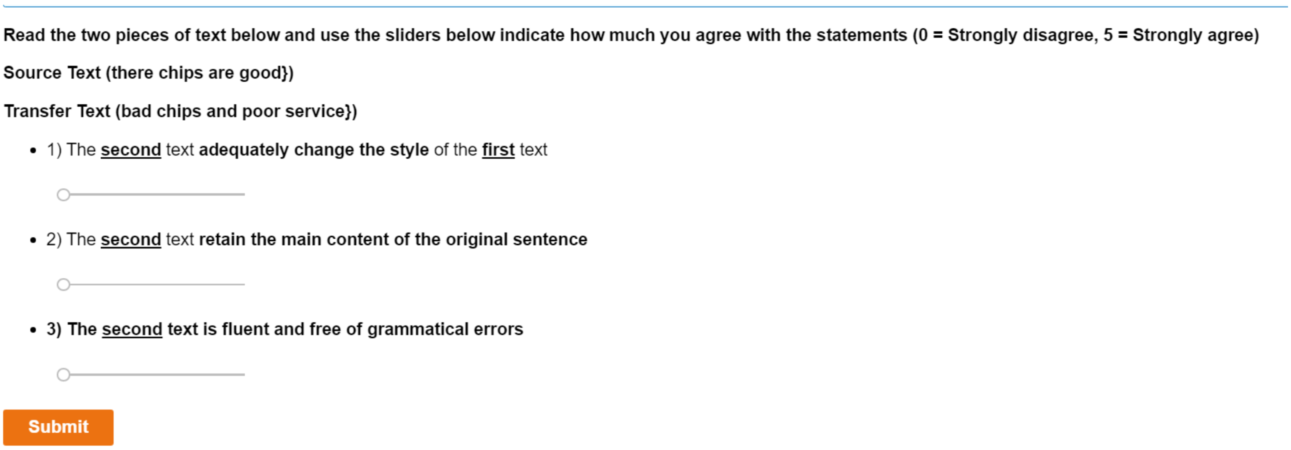

Figure 6: Human evaluation questionnaire.

	Amazon (negative-to-positive) or (positive-to-negative)
Input	anker, don't like it.
DAML-ST5	anker is a great product, i like it.
Human Ref.	anker is a fine product though i d never heard of it.
Input	but the back side looks terrible as well.
DAML-ST5	but the back side looks great as well.
Human Ref.	but the back side looks nice as well .
Input	the hot dogs come out bad
DAML-ST5	the hot dogs come out great.
Human Ref.	we often cook sausages , hot dogs and hamburgers in it as well .
Input	i will buy hamilton beach products again .
DAML-ST5	i will not buy hamilton beach products again.
Human Ref.	i will never buy hamilton beach products again .
Input	synching is not too hard , with no passwords or codes required
DAML-ST5	synching is easy, with no passwords or codes required.
Human Ref.	synching is simple , with no passwords or codes required .
Input	they were flimsy
DAML-ST5	they were a bit flimsy, but they were ok.
Human Ref.	was concerned for the price that these measuring cups would be flimsy .
Input	it does not move much air.
DAML-ST5	it moves air very fast.
Human Ref.	it moves plenty of air , oscillating between two rooms .
Input	very ugly unit, hard to assemble and difficult to operate.
DAML-ST5	very nice unit, easy to assemble and operate.
Human Ref.	very nice unit , easy to assemble and operate .
Input	i would recommend this money case
DAML-ST5	i would not recommend this money case to anyone.
Human Ref.	i would never recommend this terrible waste of money case.
Input	And this is not what you would need.
DAML-ST5	and this is what you would need.
Human Ref.	and that would be what you would need .

Table 6: More Transferred sentences on Amazon(few shot setting).

	Yelp (negative-to-positive) or (positive-to-negative)
Input	Ever since joes has changed hands it's gotten better and better.
DAML-ST5	ever since joes has changed hands it's gotten worse and worse.
Human Ref.	ever since joes has changed hands it 's just gotten worse and worse .
Input	There is so much room in that part of the venue
DAML-ST5	there is not enough room in that part of the venue.
Human Ref.	there is definitely not enough room in that part of the venue .
Input	It didn't taste watered down at all.
DAML-ST5	it tasted watered down.
Human Ref.	so basically tasted watered down .
Input	This pharmacy is really considerate.
DAML-ST5	this pharmacy is really a shithole.
Human Ref.	i ca n't believe how inconsiderate this pharmacy is .
Input	definitely not disappointed that i could use my birthday gift !
DAML-ST5	definitely disappointed that i could not use my birthday gift!
Human Ref.	definitely disappointed that i could not use my birthday gift !
Input	but it probably doesn't suck too !
DAML-ST5	but it probably does suck too!
Human Ref.	but it probably sucks too !
Input	the service was quick and responsive
DAML-ST5	the service was slow and not responsive.
Human Ref.	we sit down and we got some really slow and lazy service .
Input	they said we could sit at the table with no hesitation
DAML-ST5	they said we could not sit at the table.
Human Ref.	said we could n't sit at the table if we were n't ordering dinner .
Input	the wine was above average and the food was even better
DAML-ST5	the wine was average and the food was even worse.
Human Ref.	the wine was very average and the food was even less .
Input	i would not visit this place again
DAML-ST5	i would definitely visit this place again.
Human Ref.	one of my favorite chinese place to eat !

Table 7: More Transferred sentences on Yelp(few shot setting).