A Study on Adversarial Training for Robust Question Answering

Kathleen Pietruska, Laura Fee Nern

Motivation

Question Answering is an established area in the field of NLP. An ideal question answering model does not concentrate on internal particularities of the training data which might not be relevant for the general task of reading comprehension, and instead generalizes well to unseen datasets. A suitable concept that helps increasing the domain invariance of a model is adversarial training. In this poster, we will focus on different approaches for adversarial training to solve domain adaption on the question answering task and compare them empirically.

We consider **6 datasets**

- **in-domain:** Three datasets with large training corpus
- **out-of-domain:** Three datasets with only 127 training examples each

For each sample, the model is provided with a tuple (question, context) and required to correctly predict the start- and end-position of a span within the context containing the answer to the question

GRLA (Gradient Reversal Layer Approach)

This approach connects a domain classifier to the LSTMs output vector of our DistilBERT baseline via a gradient reversal layer (GRL) and evaluates it with a cross entropy loss. The GRL acts as the identity function in the forward step, but multiplies $-\lambda$ to the gradient when performing the gradient computation.

- the discriminator tries to minimize the cross entropy loss
- the DistilBERT tries to maximize the cross entropy loss

ADVA (Adversarial Training Approach)

ADVA also extends an existing question answering model with a discriminator, but now the QA model and the domain classifier are trained separately. The output features of DistilBERT are considered as a fixed input when optimizing the discriminator. The model tries to minimize the loss

$$l = l_{QA} + \lambda l_D$$

for DistilBERT loss l_{QA} and Kullback-Leibler divergence loss l_D between the classifiers output x and the uniform distribution:

$$l_D(\mathbf{x}) = \sum_i \frac{1}{K}\left(\log\left(\frac{1}{K}\right) - \log\left(\operatorname{softmax}(x_i)\right)\right)$$

ALUM (Adversarial training for large neural LangUage Models)

In the ALUM approach, the adversary is a noise vector added to the embeddings of the input sequence. The intuition is that we want to improve performance on the QA task on the one hand and minimize the difference an ϵ-perturbation of the embedding can have on the models output. To do so, we try to minimize the *virtual adversarial training loss* l_{ALUM}

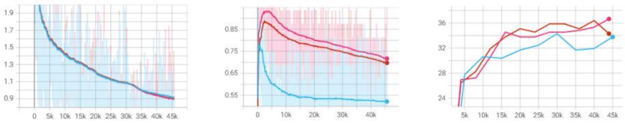
$$l_{ALUM} = l_{QA}(f(e,\theta), y) + \alpha\, l_{QA}(f(e+\delta_{\max},\theta), f(e,\theta)),$$
$$\text{where } \delta_{\max} = \operatorname{argmax}_{\{\delta:\, \|\delta\|_\infty < \epsilon\}} l_{QA}(f(e+\delta,\theta), f(e,\theta)).$$

- f is our QA model with parameters θ and embeddings e as input
- l_{QA} measures the difference between the output of the QA model and the true start and end positions of the answer y
- α is responsible for balancing the two objectives

We integrate the adversarial training only during fine-tuning and use a standard pretrained model.

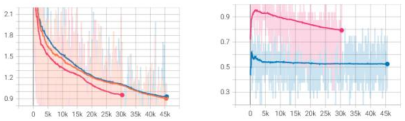
Experimental Results

GRLA The best GRLA experiment has the standard setting of $\lambda = 0.01$ and a 3-Layer MLP for the discriminator to distinguish between 6 classes. Three visualized GRL experiments differ in $\lambda = 0.1$ in blue $\lambda = 0.01$ in pink and a dropout of $p = 0.3$ in red.



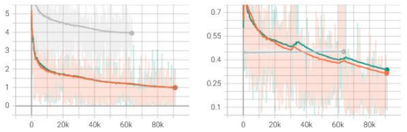
The left plot shows the training loss of the QA task, the middle plot shows the discriminators accuracy in predicting the corresponding dataset, and on the right, the EM score on the ood validation set is displayed.

ADVA In our best ADVA experiment, we used the standard setting of $\lambda = 0.01$ and a 3-Layer MLP for the discriminator to distinguish between 6 classes. The figure below shows the performance of the model in the best experiment in pink in comparison to the baseline in orange and the performance for $\lambda = 1$ in blue.



The left plot shows the training loss of the QA task, while the right plot shows the discriminators accuracy.

ALUM The best performing run, visualized below in green, used $\alpha = 1$ and $\epsilon = 1e-5$ and ood data during training. The orange and green curve differ in the use of ood train data, which barely effects the loss functions but the performance on the ood validation set. The grey line corresponds to $\alpha = 10$, which performs poorly on the ood data.



On the left the training loss of the QA task for ALUM. The second graphic shows the robust error, i.e. the QA loss between the model with and without noise added to the embeddings.

Comparison

Both approaches ADVA und GRLA manage to improve the provided baseline performance on the ood dev dataset. Unfortunately, no improvement in baseline performance in terms of F1 or EM score was evident in our experiments by applying the ALUM approach.

Method	ood dev dataset			ood test dataset	
	EM	F1	↑ F1	EM	F1
baseline	33.51	48.84	0.0%	-	-
GRLA	36.65	51.64	5.73%	40.21	58.58
ADVA	36.39	**51.71**	**5.88**%	40.09	58.47
ADVA	**37.17**	50.35	3.09%	**41.08**	**59.14**

References

[1] Seanie Lee, Donggyu Kim, and Jangwon Park. Domain-agnostic question-answering with adversarial training, 2019.

[2] Motoki Sato, Hitoshi Manabe, Hiroshi Noji, and Yuji Matsumoto. Adversarial training for cross-domain universal dependency parsing, 2017.

[3] Xiaodong Liu, Hao Cheng, Pengcheng He, Weizhu Chen, Yu Wan, Hoifung Poon, and Jianfeng Gao. Adversarial training for large neural language models, 2020.