**Problem:**

Robust QA

- State of the art transformer-based QA models fail to generalize well for doing QA on different domains that those they were trained on.
- To avoid solving this by computationally expensive and data inefficient ways we need to find training techniques that allow the model to attain domain-invariant representational power that will allow it to leverage the features that unite language across different domains of discourse.

**Past Approach:**

Domain-Adversarial Learning

- Adversarial learning was introduced in Goodfellow (2014) for generative models.
- Adapted for RobustQA in Lee (2019).
- Add a feed-forward adversarial discriminator network that is simultaneously trained to read BERT’s hidden states to guess the domain of the encoded input data. In both the earlier paper and this one, it only reads the last hidden state of the “CLS” token.
- Add a term in the QA model’s loss function that motivates it to “fool” the adversarial networks and make it unable to guess better than random.
- A good candidate is the KL-divergence between a uniform distribution over domains based on their size and the SoftMax predictions of the adversary.
- The QA model is thus forced to learn hidden representations that are not recognizably linked to the properties of a particular domain, because that would provide information to the adversary that helps it guess correctly.

**Potential Issue:**

Question-Type Blindness

- Observation: question types are not similarly distributed for each dataset.
- Hypothesis 1: learning to leverage the data’s question types is beneficial to the robustness of the model, because the relation of question type and answer type is to a great extent domain-invariant.
- Hypothesis 2: the observation suggests that the model chooses representations based on the input’s question type, the adversarial model could use their correlation to better predict the input’s domain.
- Conclusion: domain adversarial training may be teaching the QA model to become question-type “blind” and thus indirectly damage its robustness.

**New Method:**

Add “Friendly” Question-Type Discriminator

- The data used for the model are triples $\langle q, a, d \rangle$: denoting the $i^{th}$ context question, open-domain ground-truth to the $i^{th}$ training domain and out of total of $K$ training domains.
- $q$ is a feed of a question $q_i$, and $a_i$ and $d_i$ denoting respectively the correct and the position of the gold-standard answer span.
- QA-specific loss function used in the paper in the average of the natural logarithms log likelihood for the predicted start and end position over all data points. It is quite similar to the domain.

- $L_{q,a,d} = \frac{1}{K} \sum_{i=1}^{K} \log P(q_i, a_i, d_i) \quad \text{(1)}$

- QA model must additionally minimize adversarial’s objective by minimizing the KL-divergence of the adversarial discriminator’s output with the uniform distribution over domain frequency for the data.

- $L_{q,a,d} = \frac{1}{K} \sum_{i=1}^{K} KL(P(d_i)||P(d)) \quad \text{(2)}$

- In the case of domain adversarial, the QA model must additionally learn to fool the friendly QA network to make incorrect predictions by maximizing its KL divergence with the uniform distribution over question type frequencies, where $\text{If}$ is the number of question types and $\phi$ is the $k^{th}$ question label.

- $L_{q,a,d} = \frac{1}{K} \sum_{i=1}^{K} KL(P(d_i)||P(d)) \quad \text{(3)}$

- These three loss functions will now combine into one among the hyperparameters $\lambda_q, \lambda_a, \lambda_d$ and $\lambda_f$.

- $L = L_{q,a,d} + \lambda_q L_q + \lambda_a L_a + \lambda_d L_d + \lambda_f L_f$

**References**


**Development Experiments**

<table>
<thead>
<tr>
<th>One Shot OOD</th>
<th>Fine-Tune OOD</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Baseline</strong></td>
<td><strong>Final</strong></td>
</tr>
<tr>
<td><strong>Baseline</strong></td>
<td><strong>Final</strong></td>
</tr>
<tr>
<td><strong>Baseline</strong></td>
<td><strong>Final</strong></td>
</tr>
</tbody>
</table>

- Adversarial training performs badly without extra fine-tuning on the OOD data (worse than baseline) but improves after fine-tuning.
- Might be because it obstructs the QA model from gaining specialized enough representations that are adequate to the task.
- At the same time may be the reason affording greater flexibility in acquiring. Adding the friendly component to the training gives a jump to both for one-shot and fine-tuning possibly by allowing it to leverage useful but domain invariant relationships between types of question and answer.
- But how could it do this if the objectives of the two networks are contradictory? If friend can predict the question can’t Adversary increase its chances using the correlation between domain and question types?
- Post-dive: when we changed the friendly part of the QA net’s objective from maximizing the friend’s KL-divergence from the uniform distribution to minimizing its actual KL. Loss all performance gains likewise disappeared.

**Analysis:**

- As hypothesized, the model naturally tends towards representations linked to question types and adversarial training disincentivizes it from doing so.
- but adding what was meant as a “friendly” objective to dampen this effect estimates it further. The question classifier performs even worse.
- KL divergence objective forces the QA model to “show” Friend representations that lead to make a determinate choice, but not necessarily the correct one.
- Possible Exploration: QA Net has found a compromise: if helping the friendly network implicitly helps the adversary, fail the friend as well.
- Choose representations that point the question classifier to a particular wrong question type. This simultaneously maximizes the Friend’s KL-divergence while giving no indirect hints to the adversary about the data’s domain.
- The increase in OOD performance suggests that the QA net might effectively be learning to leverage certain connections between different question types that allow it to leverage its knowledge in answering question of one type to answering questions of the other.
- Sound particularly promising for achieving robust performance in new domains with different question type distributions from the training data.