QAInfomax: Learning Robust Question Answering System by Mutual Information Maximization

Yi-Ting Yeh   **Yun-Nung Chen**
Department of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan
r07922064@csie.ntu.edu.tw y.v.chen@ieee.org

Abstract

Standard accuracy metrics indicate that modern reading comprehension systems have achieved strong performance in many question answering datasets. However, the extent these systems truly understand language remains unknown, and existing systems are not good at distinguishing distractor sentences, which look related but do not actually answer the question. To address this problem, we propose QAInfomax as a regularizer in reading comprehension systems by maximizing mutual information among passages, a question, and its answer. QAInfomax helps regularize the model to not simply learn the superficial correlation for answering questions. The experiments show that our proposed QAInfomax achieves the state-of-the-art performance on the benchmark Adversarial-SQuAD dataset[1].

1 Introduction

Question answering tasks are widely used for training and testing machine comprehension and reasoning (Rajpurkar et al., 2016; Joshi et al., 2017). However, high performance in standard automatic metrics has been achieved with only superficial understanding, as models exploit simple correlations in the data that happen to be predictive on most test examples. Jia and Liang (2017) addressed this problem and proposed an adversarial version of the SQuAD dataset, which was created by adding a distractor sentence to each paragraph. The distractor sentences challenge the model robustness, and the created Adversarial-SQuAD data shows the inability of a model about distinguishing a sentence that actually answers the question from one that merely has words in common with it, where almost all state-of-the-art machine comprehension systems are significantly degraded on adversarial examples.

Lewis and Fan (2018) argued that over-fitting to superficial biases is partially caused by discriminative loss functions, which saturate when simple correlations allow the question to be answered confidently, leaving no incentive for further learning on the example. Therefore, they designed generative QA models, which use a generative loss function in question answering instead, and showed the improvement on Adversarial-SQuAD.

Instead of regularizing models by generative loss functions, we propose an alternative approach named “QAInfomax” by maximizing mutual information (MI) among passages, questions, and answers, aiming at helping models be not stuck with superficial biases in the data during learning. To efficiently estimate MI, QAInfomax incorporates the recently proposed deep infomax (DIM) in the model (Hjelm et al., 2018), which was proved effective in learning representations for image, audio (Ravanelli and Bengio, 2018), and graph domains (Veličković et al., 2018). In this work, the proposed QAInfomax further extends DIM to the text domain, and encourages the question answering model to generate answers carrying information that can explain not only questions but also itself, and thus be more sensitive to distractor sentences. Our contributions are summarized:

- This paper first attempts at applying DIM-based MI estimation as a regularizer for representation learning in the NLP domain.
- The proposed QAInfomax achieves the state-of-the-art performance on the Adversarial-SQuAD dataset without additional training data, demonstrating its better robustness.

2 Mutual Information (MI) Estimation

In this section, we introduce how scalable estimation of mutual information is performed in terms

[1] The source code is publicly available at https://github.com/MiuLab/QAInfomax.

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 3370–3375,
Hong Kong, China, November 3–7, 2019. ©2019 Association for Computational Linguistics

of practical scenarios via mutual information neural estimation (MINE) (Belghazi et al., 2018) and the deep infomax (DIM) (Hjelm et al., 2018) described below.

The mutual information between two random variable X and Y is defined as:

$$\text{MI}(X, Y) = D_{\text{KL}}(p(X, Y) \parallel p(X)p(Y)),$$

where D_{KL} is the Kullback-Leibler (KL) divergence between the joint distribution $p(X, Y)$ and the product of marginals $p(X)p(Y)$.

MINE estimates mutual information by training a classifier to distinguish between positive samples (x, y) from the joint distribution and negative samples $(x, \bar{y})$ from the product of marginals. Mutual information neural estimation (MINE) uses Donsker-Varadhan representation (DV) (Donsker and Varadhan, 1983) as a lower-bound to estimate MI.

$$\text{MI}(X, Y) \geq \mathbb{E}_{\mathbb{P}}[g(x, y)] - \log(\mathbb{E}_{\mathbb{N}}[e^{g(x,\bar{y})}]),$$

where $\mathbb{E}_{\mathbb{P}}$ and $\mathbb{E}_{\mathbb{N}}$ denote the expectation over positive and negative samples respectively, and g is the discriminator function that outputs a real number modeled by a neural network.

While the DV representation is the strong bound of mutual information shown in MINE, we are primarily interested in maximizing MI but not focusing on its precise value. Thus DIM proposes an alternative estimation using Jensen-Shannon divergence (JS), which can be efficiently implemented using the cross-entropy (BCE) loss:

$$\begin{aligned} \text{MI}(X, Y) &\geq \mathbb{E}_{\mathbb{P}}[\log(g(x, y))] \\ &+ \mathbb{E}_{\mathbb{N}}[\log(1 - g(x, \bar{y}))]. \end{aligned} \tag{1}$$

While two representations should behave similarly, considering that both act like classifiers with objectives maximizing the expected log-ratio of the joint over the product of marginals, it is found that the BCE loss empirically works better than the DV-based objective (Hjelm et al., 2018; Ravanelli and Bengio, 2018; Veličković et al., 2018). The reason may be that the BCE loss is bounded (i.e., its maximum is zero), making the convergence of the network more numerically stable. In our experiments, we primarily use the JS representation to estimate mutual information.

Recently, Tian et al. (2019) showed strong empirical performance through the improved multiview CPC training (Oord et al., 2018), which shares many common ideas as mutual information maximization. Inspired by their work, we modify (1) by first switching the role of x and y and summing them up:

$$\begin{aligned} \text{MI}(X, Y) &\geq \mathbb{E}_{\mathbb{P}}[\log(g(x, y))] \\ &+ \frac{1}{2}\mathbb{E}_{\mathbb{N}}[\log(1 - g(x, \bar{y}))] \\ &+ \frac{1}{2}\mathbb{E}_{\mathbb{N}}[\log(1 - g(\bar{x}, y))], \end{aligned} \tag{2}$$

where $(\bar{x}, y)$ is also the negative sample sampled from the product of marginals.

We empirically find that (2) gives the best performance, and more exploration about parameterization of MI is left as our future work.

3 Methodology

In the extractive question answering dataset like SQuAD, the answer $A = \{a_1, \ldots, a_M\}$ to the question $Q = \{q_1, \ldots, q_K\}$ is guaranteed to be the span $\{p_m, \ldots, p_{m+M}\}$ in the paragraph $P = \{p_1, \ldots, p_N\}$. Given Q and P, the encoded representations from the QA system M can be formulated as:

$$\{r^q, r^p\} = \{r^q_1, \ldots, r^q_K, r^p_1, \ldots, r^p_N\} = M(Q, P),$$

where r^q and r^p are representations of the question and the passage respectively after the reasoning process in the QA system M.

Most models then feed the passage representation r^p to a single-layer neural network, obtain the span start and end probabilities for each passage word, and compute the loss L_{span}, which is the negative sum of log probabilities of the predicted distributions indexed by true start and end indices.

Our QAInfomax aims at regularizing the QA system M to not simply exploit the superficial biases in the dataset for answering questions. Therefore, two constraints are introduced in order to guide the model learning.

1. **Local Constraint (LC)**: each answer word representation r^p_i in the answer representation $r^a = \{r^p_m, \ldots, r^p_{m+M}\}$ should contain information about what the remaining answer words and its surrounding context are.

2. **Global Constraint (GC)**: the summarized answer representation $s = S(r^a)$ should maximize the averaged mutual information to all other question representations in r^q and passage representations in r^p, where S is a summarization function described below.

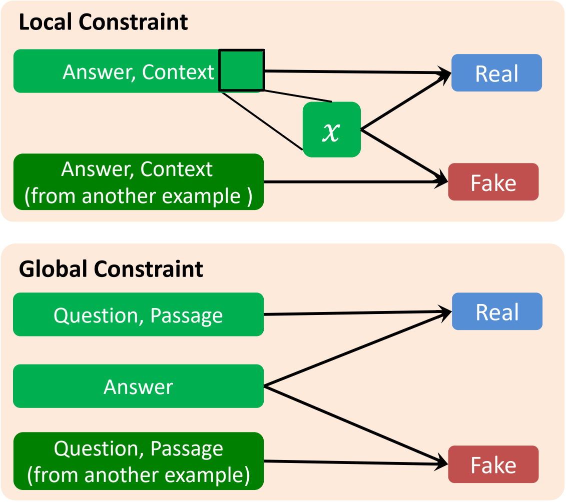

Figure 1: Illustration of the LC and GC.

Intuitively, the model is expected to *choose the answer span after fully considering the entire question and paragraph*. However, traditional QA models suffered the overstability problem, and tended to be fooled by distractor answers, such as the one containing an unrelated human name. As Lewis and Fan (2018) argued, we also believe that the main reason is that QA models are only trained to predict start and end positions of answer spans. Correlation in the dataset allows QA models to find shortcuts and ignore what the answer span looks like. A learned behavior of traditional QA models can be viewed as a simple pattern matching, such as choosing the 5-length span after the word "river" if a question is about a river and the context talks about countries in European.

Following the intuition, two constraints LC and GC are introduced to guide models to learn the desired behaviors. To prevent the model from only learning to match some specific word patterns to find the answer, LC forces the model to generate answer span representations while maximizing mutual information among words in the span and the context words surrounding the span. By maximizing the mutual information between an answer word and *all* of its context words, models need to incorporate the entire context into its decision process while choosing answers, and thus can be more robust to the adversarial sentences. Then we further require models to maximize mutual information among answer words, so models can no longer ignore any word in the chosen answer span.

On the other hand, different from LC , which only focuses on the answer span and its context, GC pushes the model to prefer answer representations carrying information that is globally shared across the whole input conditions Q and P, because shortcuts do not necessarily appear near to the answer. If the model only learns to leverage the correlation specific to the partial input, the MI of any input word without such relationship would *not* increased.

The overview about two proposed constraints is illustrated in Figure 1. The detail of two constraints and our QAInfomax regularizer is described below.

3.1 Local Constraint

As shown in Section 2, the maximization of MI needs positive samples and negative samples drawn from joint distribution and the product of marginal distribution respectively.

In LC , because all answer word representations are expected to carry the information of each other and their contexts, we choose to maximize averaged MI between the sampled answer word representations and the whole answer sequence with its context words. Specifically, a positive sample is obtained by pairing the sampled answer word representation $x \in r^a = \{r^p_m, \dots, r^p_{m+M}\}$ to all other answer and context words $r^c = \{r^p_{m-C}, \dots r^p_{m+M+C}\} \setminus \{x\}$, where C is the hyperparameter defining how many context words for consideration. Negative samples, on the other hand, are obtained by randomly sampling answer representation $\bar{r}^a = \{\bar{r}^p_l, \dots, \bar{r}^p_{l+L}\}$ and the corresponding $\bar{r}^c$ from other training examples. Following (2), the objective for sampled x, r^c, $\bar{x} \in \bar{r}^a$ and $\bar{r}^c$ is formulated.

$$\begin{aligned} \mathrm{LC}(x, r^c, \bar{x}, \bar{r}^c) &= \frac{1}{|r^c|} \sum_{r^c_i \in r^c} \log(g(x, r^c_i)) \qquad (3) \\ &+ \frac{1}{2|\bar{r}^c|} \sum_{\bar{r}^c_j \in \bar{r}^c} \log(1 - g(x, \bar{r}^c_j)) \\ &+ \frac{1}{2|r^c|} \sum_{r^c_i \in r^c} \log(1 - g(\bar{x}, r^c_i)). \end{aligned}$$

3.2 Global Constraint

Different from LC described above, GC forces the learned answer representations r^a to have information shared with all other question and passage representations. Here, we maximize the mutual information between the summarized answer vector $s = S(r^a)$ and $r_l \in r = \{r^q, r^p\} \setminus \{r^a\}$ pairs. In

Model	Original	AddSent	AddOneSent
BiDAF-S (Seo et al., 2016)	75.5	34.3	45.7
ReasoNet-S (Shen et al., 2017)	78.2	39.4	50.3
Reinforced Mnemonic Reader-S (Hu et al., 2017)	78.5	46.6	56.0
QANet-S (Yu et al., 2018)	83.8	45.2	55.7
GQA-S (Lewis and Fan, 2018)	83.7	47.3	57.8
FusionNet-E (Huang et al., 2017)	83.6	51.4	60.7
BERT-S (Devlin et al., 2018)	88.5	51.0	63.4
BERT-S + QAInfomax	**88.6**	**54.5** †	**64.9** †

Table 1: F-measure on AdversarialSquad (S: single, E: ensemble). † indicates the significant improvement over baselines with p-value < 0.05.

the experiments, we use $S(r^a) = \sigma(\frac{1}{M}\sum r_i^a)$ as our summarization function, where σ is the logistic sigmoid nonlinearity.

Specifically, a positive sample here is the pair of a answer summary vector $s = S(r^a)$ and all other word representations in r. Negative samples are provided by sampling question, passage and answer representations $\{\bar{r}^q, \bar{r}^p, \bar{r}^a\}$ from an alternative training example. Then we pair the summary s with $\bar{r} = \{\bar{r}^q, \bar{r}^p\} \setminus \{\bar{r}^a\}$, and $\bar{s} = S(\bar{r}^a)$ with r.

Similar to (3), the objective for the sampled s, r, $\bar{s}$ and $\bar{r}$ is:

$$\begin{aligned} \mathrm{GC}(s, r, \bar{s}, \bar{r}) &= \frac{1}{|r|}\sum_{r_i \in r} \log(g(s, r_i)) \qquad (4) \\ &+ \frac{1}{2|\bar{r}|}\sum_{\bar{r}_j \in \bar{r}} \log((1 - g(s, \bar{r}_j))) \\ &+ \frac{1}{2|r|}\sum_{r_i \in r} \log((1 - g(\bar{s}, r_i))). \end{aligned}$$

3.3 QAInfomax

In our proposed model, we combine two objectives and formulate the model as the complete QAInfomax regularizer. For each training batch consisting of training examples $\{\{Q_1, P_1, A_1\}, \ldots \{Q_B, P_B, A_B\}\}$, we pass the batch into the model M and obtain representations $\{\{r_1^q, r_1^p, r_1^a\}, \ldots, \{r_B^q, r_B^p, r_B^a\}\}$. Note that we abuse the subscripts to denote the example index in the batch for simplicity.

Then we shuffle the whole batch to obtain negative examples $\{\{\bar{r}_1^q, \bar{r}_1^p, \bar{r}_1^a\}, \ldots, \{\bar{r}_B^q, \bar{r}_B^p, \bar{r}_B^a\}\}$. The complete objective L_{info} for QAInfomax becomes:

$$-\frac{1}{B}\sum_{i=1}^{B}(\alpha LC(x_i, r_i^c, \bar{x}_i, \bar{r}_i^c) + \beta GC(s_i, r_i, \bar{r}_i)),$$

where x_i and $\bar{x}_i$ are the representation sampled from r_i^a and $\bar{r}_i^a$, r_i^c and $\bar{r}_i^c$ are r_i^a and $\bar{r}_i^a$ expanded with its context words respectively, s_i and $\bar{s}_i$ are the summary vectors of r_i^a and $\bar{r}_i^a$, and α and β are hyperparameters.

Combined with QAInfomax as a regularizer, the final objective of the model becomes

$$L = L_{span} + \gamma L_{info}, \qquad (5)$$

where L_{span} is the answer span prediction loss and γ is the regularize strength. The objective can be optimized through the simple gradient descent.

4 Experiments

To evaluate the effectiveness of the proposed QAInfomax, we conduct the experiments on a challenging dataset, Adversarial-SQuAD.

4.1 Setup

BERT-base (Devlin et al., 2018) is employed as our QA system M in the experiments, where we set the same hyperparameters as one released in SQuAD training[2].

We set C, α, β and γ to be 5, 1, 0.5, 0.3 respectively in all experiments, and add the proposed QAInfomax into the BERT model as described above. The discriminator function g is the bilinear function similar to the scoring used by Oord et al. (2018):

$$g(x, y) = x^T W y, \qquad (6)$$

where W is a learnable scoring matrix.

We train the BERT model with the proposed QAInfomax on the orignal SQuAD dataset, and

[2] We use PyTorch (Paszke et al., 2017) reimplementation for experiments: `https://github.com/huggingface/pytorch-pretrained-BERT`.

Model	Adversary F1	Speed (iter/s)
BERT	51.0 / 63.4	3.80
+ LC	53.6 / 64.2	3.51
+ GC	52.2 / 63.7	2.75
+ LC + GC	**54.5 / 64.9**	2.72

Table 2: Ablation study with F1 scores on ADDSENT / ADDONESENT. The speed is measured on RTX 2080Ti.

use Adversarial-SQuAD to test the robustness of the augmented model. Only ADDSENT and ADDONESENT metrics are reported for the comparison with previous models, because most previous models did not report their ADDANY and ADDCOMMON scores. Briefly, for each example, ADDSENT runs the model M on every human-approved adversarial sentence, picks the one that makes the model give the worst answer and returns that score. ADDONESENT, on the other hand, only picks a random human-approved adversarial sentence. The numbers reported in all experiments are the best number across at least three runs.

4.2 Results

Table 1 reports model performance on Adversarial-SQuAD. It can be found that QAInfomax yields substantial improvement over the vanilla BERT model, and achieves the state-of-the-art performance on both ADDSENT and ADDONESENT metrics.[3] QAInfomax obtains larger improvement on the ADDSENT, which picks the worst scores of the model. It shows the effectiveness of our QAInfomax in terms of forcing the model to ignore simple correlation in the data and learn the more human-like reasoning processes. It is worth to note that while QAInfomax mitigates the overstability problem and improves the robustness to adversarial examples, it does not hurt the original performance of the QA system, demonstrating the benefit for the practical usage. Some example results from the Adversarial-SQuAD dataset can be found in the Appendix, where adversarial distracting sentences are shown in italic blue fonts.

Function	ADDSENT	ADDONESENT
Mean	**52.2**	**63.7**
Max	52.0	63.3
Sample	**52.2**	63.0

Table 3: Different summarization functions for GC .

Table 2 shows the ablation study of our proposed QAInfomax, where two proposed constraints are both important for achieving such results. We also show the training speed of the proposed method and its limitation, where the GC objective degrades the training speed by 28%. The reason is that GC measures the averaged MI over the *whole* question and passage representations, which may include a long sequence of vectors.

Considering that the summarization function S plays an important role in GC, we explore its different variants in Table 3:

- Mean: $\sigma(\frac{1}{M}\sum r_i^a)$
- Max: $\sigma(maxpool(r^a))$
- Sample: randomly sample one $r_i^a \in r^a$

According to the experimental results, Mean performs the best while Max and Sample has the competitive performance, showing the great robustness of the proposed methods to different architecture choices.

5 Conclusion

This paper presents a novel regularizer based on MI maximization for question answering systems named QAInfomax, which helps models be not stuck with superficial correlation in the data and improves its robustness. The proposed QAInfomax is flexible to apply to different machine comprehension models. The experiments on Adversirial-SQuAD demonstrate the effectiveness of our model, and the augmented model achieves the state-of-the-art results. In the future, we will investigate more methods for reducing the limitations of QAInfomax and improving the capability of generalization in QA systems.

Acknowledgements

We would like to thank reviewers for their insightful comments on the paper. This work was financially supported from the Young Scholar Fellowship Program by Ministry of Science and Technology (MOST) in Taiwan, under Grant 108-2636-E-002-003.

[3] Note that Wang and Bansal modified distractor paragraphs and added them into training data, so we do not compare with them, because we only use the original SQuAD training data.

References

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. 2018. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Monroe D Donsker and SR Srinivasa Varadhan. 1983. Asymptotic evaluation of certain markov process expectations for large time. iv. *Communications on Pure and Applied Mathematics*, 36(2):183–212.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Adam Trischler, and Yoshua Bengio. 2018. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*.

Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. 2017. Reinforced mnemonic reader for machine reading comprehension. *arXiv preprint arXiv:1705.02798*.

Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. 2017. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. *arXiv preprint arXiv:1711.07341*.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. *arXiv preprint arXiv:1707.07328*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Mike Lewis and Angela Fan. 2018. Generative question answering: Learning to answer the whole question.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Mirco Ravanelli and Yoshua Bengio. 2018. Learning speaker representations with mutual information. *arXiv preprint arXiv:1812.00271*.

Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *CoRR*, abs/1611.01603.

Yelong Shen, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. 2017. Reasonet: Learning to stop reading in machine comprehension. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1047–1055. ACM.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. 2019. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*.

Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. 2018. Deep graph infomax. *arXiv preprint arXiv:1809.10341*.

Yicheng Wang and Mohit Bansal. 2018. Robust machine comprehension models via adversarial training. *arXiv preprint arXiv:1804.06473*.

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