Learning Invariant Representation Improves Robustness for MRC Models

Hai Yu[1], **Liang Wen**[1], **Haoran Meng**[1], **Tianyu Liu**[2] **and Houfeng Wang**[1]
[1] MOE Key Laboratory of Computational Linguistics, Peking University, China
[2] Tencent Cloud Xiaowei
{yuhaia, yuco, wanghf}@pku.edu.cn; {rogertyliu}@tencent.com;{haoran}@stu.pku.edu.cn

Abstract

The prosperity of Pretrained Language Models (PLM) has greatly promoted the development of Machine reading comprehension (MRC). However, these models are vulnerable and not robust to adversarial examples. In this paper, we propose Stable and Contrastive Question Answering (SCQA) to improve invariance of model representation to alleviate these robustness issues. Specifically, we first construct positive example pairs which have same answer through data augmentation. Then SCQA trains enhanced representations with better alignment between positive pairs by introducing stability and contrastive loss. Experimental results show that our approach can boost the robustness of QA models cross different MRC tasks and attack sets significantly and consistently.[1]

1 Introduction

Machine reading comprehension (MRC) (Zeng et al., 2020) aims to answer the question based on a passage as context. It has experienced rapid development due to the evolution of deep neural networks (Minjoon Seo, 2017; Devlin et al., 2019) and release of large-scale and high-quality datasets (Rajpurkar et al., 2016; Dzendzik et al., 2021). By introducing different kinds of attack sets, however, a considerable amount of literature (Jia and Liang, 2017; Gan and Ng, 2019; Liu et al., 2020b; Si et al., 2021) has shown that the result on in-domain test set tends to overestimate the models' performance. For example, Jia and Liang (2017) exposed models' over-stability issue by adding one distracting sentence, which suggests the models' inability to distinguish a sentence that actually answers the question from one that merely share sufficient words with it but semantically changed. Gan and Ng (2019) showed that a question paraphrased in a slightly different but semantically similar way can mislead the model to output a wrong answer.

Up to now, researchers have proposed several solutions to alleviate these robustness issue, including utilizing external knowledge to create adversarial examples to enrich training data (Wang and Bansal, 2018; Zhou et al., 2020) and adversarial training based methods (Yang et al., 2019; Liu et al., 2020a; Yang et al., 2021). However, recent literature (Liu et al., 2020c,a; Si et al., 2021) suggests that models trained with specific augmented data are still easily attacked by other unseen perturbations. Adversarial training can improve models' robustness under general attacks without requiring any explicit adversarial examples, but at the cost of iterative training schedule.

To be capable of handling more general attacks rather than just a certain type attack without laborious iterative schedule, we propose SCQA to addresses the above robustness issue by learning invariant representations of similar examples inspired by Le-Khac et al. (2020). In detail, the data augmentation module first constructs an example similar to the input example to form a positive pair. Then SCQA utilizes stability loss to scale down the change of probability distribution caused by small label-preserving perturbations. In addition, SCQA introduces contrastive loss to pull semantically close pairs together to further improve the alignment property in the representation space.

In the experimental part, we have organized different MRC tasks and several attack tests as a benchmark for MRC robustness, including span-based extractive, multiple choice and Yes/No MRC. The results show that SCQA with dropout noise as implicit data augmentation can reduce the distance between embeddings of paired examples, and therefore improve the robustness of the MRC models over different types of adversarial perturbations consistently and significantly. Moreover, it is worth to note that our approach can further boost the models' performance with specific explicit data augmentation strategies.

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

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 3306–3314
December 7-11, 2022

2 Methodology

2.1 SCQA Architecture

As shown in Figure 1, SCQA has five modules:

- A data augmentation module that construct positive example pairs.
- An encoder which learns contextual representations for input sequence.
- A contrastive loss layer on top of the encoder, it aims to pull positive pairs together and keep one representation distant from other negative representations in the same batch.
- A predictor that maps the contextual representation into probability distribution to predict the answer.
- A stability calculator that quantifies the change of probability distribution caused by small label-preserving perturbations.

Given each input sequence $x_i \in \mathbb{R}^{L*d}$ consists of question q_i and context c_i, L is the sequence length and d is the hidden dimension, we first apply data augmentation module to generate semantically similar samples $x_i^{'}$ corresponding to x_i to form positive pairs $(x_i, x_i^{'})$, then both x_i and $x_i^{'}$ are feed into the encoder $\mathbf{E}$ to generate the contextual representation h_i and $h_i^{'}$, $h_i, h_i^{'} \in \mathbb{R}^{1*d}$. After that, h_i, $h_i^{'}$ and representations of other examples in the same batch will be passed into contrastive layer to compute contrastive loss $l_{contrastive}$. The predictor module $\mathbf{P}$ will compute probability distribution $\mathbb{P}_i$ and $\mathbb{P}_i^{'}$ of answers. Then, MRC task-specific loss l_{mrc} and stability loss $l_{stability}$ are calculated separately. Finally, all losses are combined to be the optimization objective of model parameters. The model is expected to learn invariant representations of similar input sequences and be robust against label-preserving attacks.

2.2 Data Augmentation Strategies

The purpose of data augmentation module is to generate semantically similar example pairs which are used for calculating the contrast and stability loss. We apply two different data augmentation strategies into SCQA framework to transform training batch size from n to $2 \times n$.

Dropout as Implicit Augmentation Following Gao et al. (2021), we first investigate dropout noise as implicit data augmentation for MRC. Specifically, there are standard dropout masks z in Transformers (Vaswani et al., 2017), so we can just encoder the same input x_i twice with independently sampled dropout masks $z_i, z_i^{'}$ and then get h_i and $h_i^{'}$ to represent x_i and $x_i^{'}$ respectively.

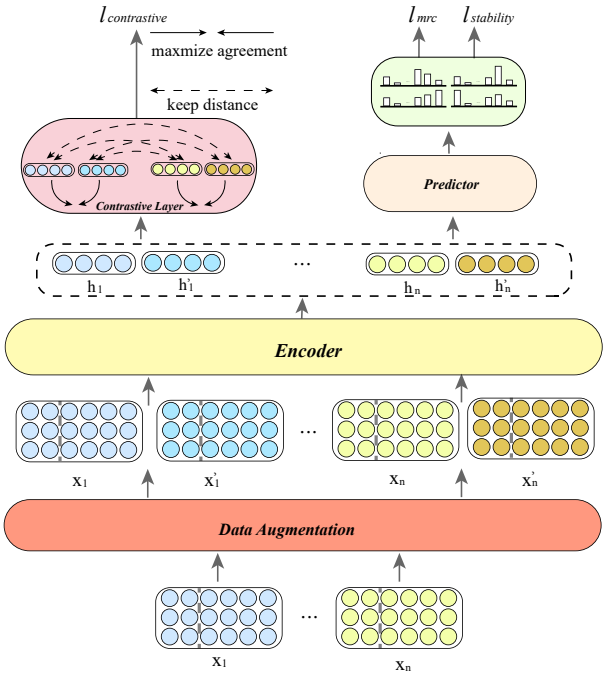

Figure 1: A schematic view of SCQA architecture.

Adversarial Examples as explicit Augmentation Augmenting original training data with adversarial examples created by the same rules as attacks is utilized to improve the models' robustness (Wang and Bansal, 2018), although which can only defend the specific attack (Liu et al., 2020a). For this setting, we follow the strategy in Wang and Bansal (2018) and mix as many adversarial examples as 25% of the original training data. For those instances x_i without corresponding adversarial examples, we take the dropout approach to supplement data.

2.3 Training Loss

Our training loss is a weighted sum of the MRC task-specific loss l_{mrc}, contrastive loss $l_{contrastive}$ and stability loss $l_{stability}$. This section describe stability and contrastive loss in detail. For completeness, we provide description of MRC task-specific loss in Appendix A.1.

$$l_{total} = l_{mrc} + w_1 \cdot l_{contrastive} + w_2 \cdot l_{stability} \quad (1)$$

Stability Loss The aim of stability loss is to scale down change of probability distribution caused by small label-preserving perturbations in data augmentation module. Given the probability distribution $\mathbb{P}_i$ and $\mathbb{P}_i^{'}$ output by predictor $\mathbf{P}$, the stability

Model	Dev	Test	CS	PQ	AS	DE	DG	WR	Average
BERT	65.46	64.82	49.01	58.82	24.06	29.14	51.70	62.87	50.75
SC-BERT	**68.00**	**66.54**	**50.16**	**59.42**	**28.31**	**30.97**	**54.03**	**64.49**	**52.74** (+1.99)
ALBERT	67.81	66.64	49.39	57.09	31.41	36.26	57.82	63.68	53.76
SC-ALBERT	**70.35**	**68.30**	**50.67**	**60.56**	**34.03**	**39.79**	**58.31**	**67.13**	**56.14** (+2.38)
RoBERTa	71.00	70.21	54.20	62.79	28.19	38.87	58.39	67.82	56.43
SC-RoBERTa	**72.66**	**71.30**	**55.65**	**63.90**	**29.14**	**40.15**	**58.51**	**68.61**	**57.49** (+1.06)

Table 1: Results on RACE develop, test set and adversarial test sets cited from Si et al. (2021). All scores are *Accuracy* in percentage. **CS** means CharSwap, **PQ** means ParaQues, **AS** means AddSent, **DE** means Distractor Extraction, **DG** means Distractor Generation and **WR** means Word Replace. p value for t-test is less than 0.005.

loss is calculated as follow:

$$l^i_{stability} = \left\| \mathbb{P}_i - \mathbb{P}'_i \right\|_2 \quad (2)$$

Dev	Dev	AddOneSent	AddSent
Baseline systems			
BERT	88.48	74.40	66.11
ALBERT	89.76	77.88	70.48
RoBERTa	91.95	80.73	75.34
Related works			
BiDAF (2017)	77.3	45.7	34.3
QANet (2018)	83.8	55.7	45.2
R.M-Reader (2018)	86.3	67.0	58.5
QAInfomax (2019)	88.6	64.9	54.5
BERT-AT† (2020a)	87.8	-	81.7
PQAT (2021)	92.3	73.6	64.7
BERT†	88.20	86.68	85.72
ALBERT†	89.35	88.07	**87.68**
RoBERTa†	91.79	90.93	90.17
Our approach			
SC-BERT	**88.68**	75.16	66.48
SC-BERT†	88.56	**87.07**	**86.48**
SC-ALBERT	**90.4**	78.01	69.77
SC-ALBERT†	89.58	**88.43**	87.66
SC-RoBERTa	**92.47**	81.60	75.87
SC-RoBERTa†	92.13	**91.48**	**90.61**

Table 2: Experiment results on SQuAD 1.1 develop set and adversarial set. All scores are *F1* in percentage. † means mixing AddSentDiverse data.

Contrastive Loss We adopt the simple but widely employed normalized temperature-scaled cross-entropy loss in Gao et al. (2021) for constrstive learning. h_i and $h_i^{'}$ denote the representation of x_i and $x_i^{'}$ which are semantically similar, the contrastive loss for $(x_i, x_i^{'})$ within a mini-batch of N pairs is:

$$l^i_{contrastive} = -\log \frac{e^{sim(h_i,h_i^{'})/\tau}}{\sum_{j=1}^{N} e^{sim(h_i,h_j^{'})/\tau}} \quad (3)$$

where τ is the temperature hyper-parameter and $sim(h_1, h_2)$ is the cosine similarity

3 Experimental Settings

Datasets Following previous research (Jia and Liang, 2017; Gan and Ng, 2019; Si et al., 2021), we organize a benchmark for MRC models' robustness by integrating different forms of reading comprehension tasks and different adversarial attacks. The datasets contain span-based extractive SQuAD 1.1 (Rajpurkar et al., 2016) and SQuAD 2.0 (Rajpurkar et al., 2018), multiple choice RACE (Lai et al., 2017) and ReClor (Yu et al., 2019), and Yes/No BoolQ (Clark et al., 2019) and NP-BoolQ (Khashabi et al., 2020). Types of adversarial attacks include CharSwap, ParaQues, AddSent and other dataset-specific attacks. We choose AddSentDiverse (Wang and Bansal, 2018) and training set of NP-BoolQ (Khashabi et al., 2020) as explicit data augmentation to demonstrate that our SCQA approach can further boost the models' performance against specific attack. The details of datasets, attacks and augmented adversarial examples are presented in Appendix A.2.

Models and Metrics We apply the SCQA approach on base version of BERT (Devlin et al., 2019), ALBERT (Lan et al., 2019) and RoBERTa (Liu et al., 2019). The parameters are presented in Appendix A.3. We run experiments three times with different random seeds to report the mean result and do t-test to ensure that the improvement was statistically significant. We follow the existing evaluation indicators. For multiple choice and Yes/No MRC task, the *Accuracy* results are reported. *F1* value is measured for span-based extractive task.

4 Results and Analysis

4.1 Performance against Attacks

With dropout as data augmentation, the results of SCQA method on RACE develop, test set and dif-

Model	Dev	Test	CS	PQ	AS	DE	DG	WR	Average
BERT	65.46	64.82	49.01	58.82	24.06	29.14	51.70	62.87	50.75
+ Dropout	66.20	64.88	48.91	57.94	27.83	29.85	52.80	62.95	51.42
+ SL	67.34	66.23	49.13	**59.63**	27.40	32.12	53.20	64.11	52.40
+ CL	66.89	65.59	49.61	58.63	26.29	30.04	53.06	63.42	51.69
+ SL then CL	66.36	65.14	49.27	59.49	**28.35**	**31.03**	53.06	63.15	51.98
+ CL then SL	67.16	65.83	49.66	59.30	26.79	30.24	53.08	63.70	51.97
+ SL and CL	**68.00**	**66.54**	**50.16**	59.42	28.31	30.97	**54.03**	**64.49**	**52.74**

Table 3: Ablation study of BERT on RACE develop, test and different adversarial test sets. SL means stability loss. CL means training with contrastive loss. SL then CL means training 3 epochs with stability loss then training another 3 epochs with contrastive loss. SL and CL means training with stability and contrastive loss simultaneously.

ferent adversarial sets are summarized in Table 1. Compared with baseline models, SCQA is able to boost model robustness against all attack test sets significantly, without sacrificing the performance on in-domain develop and test sets. According to the relationship between the number of model parameters and improvement of the results, we can find that SCQA can better improve the model with a small number of parameters.

Table 2 reports the robustness of models against AddSent and AddOneSent attack built on SQuAD 1.1, under the explicit and implicit data augmentation strategies respectively. With dropout as implicit augmentation, SCQA improves the performance by about 0.6% consistently. Training with adversarial samples can effectively improve the robustness against specific attack, and SCQA can further boost the performance by about 0.5%.

Overall results on other datasets and adversarial attack sets are presented in Appendix A.4

4.2 Ablation Study

We investigate ablation experiments to observe the impact of stability and contrastive loss. Additionally we explore how to combine stability and contrastive loss to better optimize a robust model. Table 3 shows the results of BERT on RACE dataset, which suggests that just training with stability, contrastive loss or combination serially can either improve model performance, and the combination of the two objective at the same time can achieve the best overall effect.

4.3 Analysis of Embedding Space

To prove the hypothesis that SCQA can train enhanced contextual representations with better invariance, we compute the alignment property l_{align} between representations of example x in test set and corresponding label-preserving adversarial example $x^{'}$. A lower alignment value represents a better representation. The results on Figure 2 shows that SCQA can amends model alignment and therefore improve the robustness of QA models against test-time label-preserving perturbations. We can also observe from the value of alignment that the AddSent attack is more challenging than CharSwap.

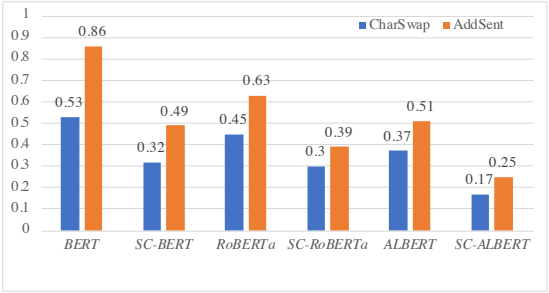

Figure 2: Alignment of models on CharSwap and AddSent attacks of RACE test set.

$$l_{align} \triangleq \mathbb{E}_{(x,x')\in p_{pos}} ||f(x) - f(x^{'})||^2 \quad (4)$$

5 Conclusion

A number of studies have exposed the robustness issues of MRC by introducing different types of adversarial examples. In this paper, we integrate several MRC datasets and adversarial attacks to construct a benchmark for evaluating the robustness of MRC models. Then we propose SCQA to train robust MRC models by learning invariant representation. Experimental results show that, with dropout as implicit data augmentation, our novel approach can improve the alignment attribute and elevate the robustness of MRC models generally and consistently. Moreover, with explicit adversarial examples augmented, SCQA can further boost model performance.

6 Limitations

Current data augmentation strategy just contains creating positive training pairs to defense label-preserving perturbations. SCQA pays little attention to the construction of high-quality negative sample pairs, which need to be further explored to fully utilize the effect of contrastive learning.

Acknowledgements

We thank all the anonymous reviewers for their insightful feedback. The work is supported by National Natural Science Foundation of China under Grant No.62036001 and PKU-Baidu Fund (No.2020BD021). The corresponding author of this paper is Houfeng Wang.

References

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. 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 2924–2936.

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.

Daria Dzendzik, Jennifer Foster, and Carl Vogel. 2021. English machine reading comprehension datasets: A survey. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8784–8804.

Wee Chung Gan and Hwee Tou Ng. 2019. Improving the robustness of question answering systems to question paraphrasing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6065–6075.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, et al. 2020. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031.

Daniel Khashabi, Tushar Khot, and Ashish Sabharwal. 2020. More bang for your buck: Natural perturbation for robust question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 163–170.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. 2020. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934.

Kai Liu, Xin Liu, An Yang, Jing Liu, Jinsong Su, Sujian Li, and Qiaoqiao She. 2020a. A robust adversarial training approach to machine reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8392–8400.

Tianyu Liu, Zheng Xin, Baobao Chang, and Zhifang Sui. 2020b. HypoNLI: Exploring the artificial patterns of hypothesis-only bias in natural language inference. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6852–6860, Marseille, France. European Language Resources Association.

Tianyu Liu, Zheng Xin, Xiaoan Ding, Baobao Chang, and Zhifang Sui. 2020c. An empirical study on model-agnostic debiasing strategies for robust natural language inference. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 596–608, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zhen Huang Xipeng Qiu Furu Wei Minghao Hu, Yuxing Peng and Ming Zhou. 2018. Reinforced mnemonic reader for machine reading comprehension. In *In Proceedings of the Twenty-Seventh In- ternational Joint Conference on Artificial Intelli- gence*, page 4099–4106.

Ali Farhadi1 Hananneh Hajishirzi Minjoon Seo, Aniruddha Kembhavi. 2017. Bidirectional attention flow for machine comprehension. In *International Conference on Learning Representations*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789.

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.

Chenglei Si, Ziqing Yang, Yiming Cui, Wentao Ma, Ting Liu, and Shijin Wang. 2021. Benchmarking robustness of machine reading comprehension models. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 634–644.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Yicheng Wang and Mohit Bansal. 2018. Robust machine comprehension models via adversarial training. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 575–581.

Ziqing Yang, Yiming Cui, Wanxiang Che, Ting Liu, Shijin Wang, and Guoping Hu. 2019. Improving machine reading comprehension via adversarial training. *arXiv preprint arXiv:1911.03614*.

Ziqing Yang, Yiming Cui, Chenglei Si, Wanxiang Che, Ting Liu, Shijin Wang, and Guoping Hu. 2021. Adversarial training for machine reading comprehension with virtual embeddings. In *Proceedings of* SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 308–313.

Yi-Ting Yeh and Yun-Nung Chen. 2019. Qainfomax: Learning robust question answering system by mutual information maximization. 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 3370–3375.

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. In *International Conference on Learning Representations*.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2019. Reclor: A reading comprehension dataset requiring logical reasoning. In *International Conference on Learning Representations*.

Changchang Zeng, Shaobo Li, Qin Li, Jie Hu, and Jianjun Hu. 2020. A survey on machine reading comprehension—tasks, evaluation metrics and benchmark datasets. *Applied Sciences*, 10(21):7640.

Mantong Zhou, Minlie Huang, and Xiaoyan Zhu. 2020. Robust reading comprehension with linguistic constraints via posterior regularization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2500–2510.

A Appendix

A.1 MRC Task-specific Loss

Different forms of machine reading comprehension tasks require task-specific output heads and loss functions, which are as follows:

- **Span-based Extractive MRC** task requires model to predict the start/end position probability distributions of answer. And the training loss l_{span} is the negative log-likelihoood of correct start and end boundaries:

$$l^i_{span} = -y^s_i \log \mathbb{P}^s_i - y^e_i \log \mathbb{P}^e_i \qquad (5)$$

  where y^s_i and y^e_i are the ground truth start and end positions of input example x_i.
- **Span-based Extractive RC with Unanswerable Questions** requires the model not only to predict the correct span answer when the question is answerable, but also to identify when no answer can be inferred from the context. Therefore, we use the negative log-likelihoood of correct start and end position as training objective, and if the question has no answer, we will simply predict both the start and end position as 0. During the inference phase, if the best candidate span answer has a score that is less than the score of the no-answer (sum of start and end probability in position 0) minus a threshold, the no-answer is selected for this example.
- **Multiple Choice RC** requires the model to find the only correct option in the given candidate options based on the context. Given a input example x_i consists of question q_i, context

c_i and k options $o_i^{(1)}, o_i^{(2)}, ..., o_i^{(k)}$, we first create k input sequences where $x_i^(j)$ is composed by q_i, c_i and $o_i^{(j)}$, $1 <= j <= k$. We feed the k sequences into encoder $\mathbf{E}$ to get contextual representations $h_i = h_i^{(1)}, h_i^{(2),...,h_i^{(k)}} \in \mathbb{R}^{k \times d}$ where d is the hidden size. The final predictor module $\mathbf{P}$ then calculate the probability of which option the answer is. The loss function is cross-entropy loss generally used in multiple classification problems.

- **Yes/No MRC** expect the model to answer yes or no when given Yes/No questions and related context. It can be modeled as a binary classification problem and the training objective is cross-entropy loss.

A.2 Details of Datasets

Datasets Table 4 represents the dataset statistics in detail.

- **SQuAD 1.1** Rajpurkar et al. (2016) proposed the dataset in which the 100k+ questions were created by crowdworkers on 536 Wikipedia articles.
- **SQuAD 2.0** was released by Rajpurkar et al. (2018) which combines exsiting SQuAD 1.1 data with over 50k unanswerable questions written adversarially by crowdworkers to look similar to answerable ones.
- **RACE** was presented by Lai et al. (2017) which contains near 100k questions of Chinese middle and high school students' English exams. More reasoning ability of the model is required to answer the question because these questions were carefully designed for evaluating the students' ability in understanding and reasoning.
- **ReClor** was constructed by Yu et al. (2019) from standardized graduate admission examinations. What makes ReClor challenging is that every sentence in the context is important. Therefore the model should not only extract relevant information from the context, but also have the logical reasoning ability.
- **BoolQ** was created by Clark et al. (2019) in unprompted and unconstrained ways and focus on Yes/No questions. The questions are written by people who did not know the answer to the question they were asking. And each question is paired with a paragraph from Wikipedia that an independent annotator has marked if the context contains the answer.
- **NP-BoolQ** proposed by Khashabi et al. (2020) also requires the machine to understand what facts can be inferred to be true or false from the context. It was constructed by applying human-driven natural perturbations to BoolQ (Clark et al., 2019).

Adversarial Attacks We test the robustness of models on following different types of label-preserving perturbations which attack MRC models from their unique perspectives:

- **CharSwap (CS)** attack on multiple choice MRC task was proposed by Si et al. (2021) to show that models' performance drops a lot when there are spelling errors in the data. We expand CharSwap attack to spand-based extractive and Yes/No MRC tasks by randomly swap two adjacent letters in the non-stopwords in the question and context without altering the first or last letters.
- **ParaQues (PQ)** attack on SQuAD 1.1 dataset was proposed by Gan and Ng (2019) to expose the models' over-sensitivity issue of being puzzled by a paraphrased label-preserving question. To investigate other types of MRC models' performance on ParaQues attack, we use fine-tuned T5 model (Raffel et al., 2020) by Questgen [2] based on Quora Question Pairs dataset [3] to generating the paraphrased questions. Then we manually check and reserve the paraphrased questions which have the same meaning of the original question and are written in fluent English.
- **AddSent (AS)** was first introduced by (Jia and Liang, 2017) to investigate the inability of a MRC model to defend the perturbation of a distractor sentence. Inspired by Jia and Liang (2017), Si et al. (2021) proposed to make use of the human-written distractors in multiple choice RACE dataset to create strong distracting setences which are then inserted into the context randomly. Following those work, we create AddSent attack test set on SQuAD 2.0 and ReClor datasets respectively.
- **ContrastSet (ConS)** was proposed by Gardner et al. (2020) to accurately evaluate models' true linguistic capabilities. Different from other label-preserving attacks, they manually

[2] https://github.com/ramsrigouthamg/Questgen.ai

[3] https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs

type	dataset	q/a source	#q	#c	avg |q|	avg |c|	avg |a|	|V|
Extractive	SQuAD 1.1	Crowdworker	108k	20k	11.4	137.1	3.5	87k
Extractive	SQuAD 2.0	Crowdworker	151k	20k	11.2	137.0	3.5	88k
Multiple Choice	RACE	Experts	98k	28k	12.0	329.5	6.3	98k
Multiple Choice	ReClor	Experts	6k	6k	17.0	73.6	20.6	26k
Yes/No	BoolQ	User/Crowdworker	16k	13k	8.8	109.4	n/a	49k
Yes/No	NP-BoolQ	User/Crowdworker	17k	4k	9.1	95.7	n/a	62k

Table 4: Statistics of datasets. **q**, **a** and **c** means question, answer and context respectively. **#q** and **#c** means the number of question and context. The average length of question, context and answer are noted as **avg |q|**, **avg |a|** and **avg |a|**. **|V|** means the vocab size.

perturb the test instances of BoolQ and nine remaining NLP datasets in small but meaningful ways that typically change the gold label, creating contrast sets looks not explicitly adversarial but significantly reduce the performance.

Augmented Adversarial Examples One of the straightforward ways to defend attack is augmenting the training dataset with adversarial examples generated by the same rules as the attacks. To further illustrate the effectiveness of our approach, we train our model on the datasets mixed with the adversarial examples.

- **AddSentDiverse** was proposed by Wang and Bansal (2018) based on the observation that retraining models with data generated by AddSent (Jia and Liang, 2017) has limited effect on the robustness. They further enriched SQuAD 1.1 training data by dynamically generating the fake answers and varying the locations where the distractor sentences are placed. The mixed adversarial examples accounts for 20% of the total SQuAD 1.1 dataset.
- **NP-BoolQ** was proposed by Khashabi et al. (2020) to focus on the value of natural perturbations for robust model design. They asked the workers to change the question by adding or removing up to four terms, resulting a modified question challenging for RoBERTa trained on BoolQ dataset. The mixed adversarial examples accounts for 5% of the total dataset. We utilize training part of NP-BoolQ to augment BoolQ training data.

A.3 Parameters Settings

We use the $base$ version of BERT, RoBERTa and ALBERT. During training, the batch size and epoch varies according to the task, and all models have the same batch size and epoch in the same dataset.

Model	Dev	CS	PQ	ConS	NP-BoolQ$_{dev}$
BERT	71.4	69.7	71.1	58.0	46.8
SC-BERT	73.9	69.5	74.0	58.4	52.2
BERT†	74.5	**70.0**	74.0	61.6	56.5
SC-BERT†	**76.0**	69.9	**76.1**	**63.1**	**56.5**
RoBERTa	75.4	71.5	75.1	60.9	49.7
SC-RoBERTa	77.3	**72.0**	77.0	**62.1**	49.8
RoBERTa†	78.2	71.0	76.4	60.0	53.4
SC-RoBERTa†	**79.1**	71.8	**78.6**	61.4	**57.1**
ALBERT	77.7	65.2	77.0	60.2	55.9
SC-ALBERT	**79.2**	67.3	**78.4**	59.9	56.0
ALBERT†	77.9	**70.0**	77.3	60.0	**56.9**
SC-ALBERT†	78.3	68.8	77.9	**60.4**	56.5

Table 5: Experimental results of models on BoolQ develop set and different kinds of adversarial test sets.† means training with mixed NP-BoolQ train set.

For SQuAD 1.1 and SQuAD 2.0, the batch size and epoch are 12 and 3 respectively. For BoolQ and NP-BoolQ, the batch size and epoch are 12 and 10 separately. The batch size for RACE and ReClor is 6 since each question has 4 options, and the epoch is 3. We set the learning rate lr to $3e-5$ for all models on all datasets, excepting that lr of all models is $2e-5$ for ReClor, lr of RoBERTa is 3e-6 on RACE and NP-BoolQ, and ALBERT has lr 1e-5 for RACE, BoolQ and NP-BoolQ. We keep the other hyper-parameters of models default. The weights in combined loss are simply set $1e-4$ for w_1 and $3e-5$ for w_2. The temperature τ is 0.05.

A.4 Detailed Results

Table 5-8 represents the experimental results of models on BoolQ, NP-BoolQ, SQuAD 2.0, ReClor and corresponding adversarial sets respectively.

Model	Dev	CS	PQ	ConS	BoolQ$_{dev}$
BERT	65.8	**58.6**	62.9	52.1	61.7
SC-BERT	**67.2**	57.5	**64.5**	**58.9**	**65.1**
RoBERTa	69.0	56.3	65.7	55.8	61.1
SC-RoBERTa	**70.6**	**58.4**	**66.0**	**58.2**	**69.7**
ALBERT	70.5	**59.7**	67.1	51.59	62.7
SC-ALBERT	**70.8**	58.0	**67.4**	**57.2**	**67.3**

Table 6: Experimental results of models on NP-BoolQ develop set and different kinds of adversarial test sets.

Model	Dev	CS	PQ	AS
BERT	69.7/76.9	79.8/53.6	69.7/71.5	**74.3**/57.1
SC-BERT	**70.2/77.3**	**80.7/53.9**	**70.2/71.9**	73.8/**57.9**
RoBERTa	76.0/82.8	72.4/60.0	85.5/70.8	87.7/64.9
SC-RoBERTa	**78.0/84.0**	**75.4/61.1**	**86.9/71.2**	**88.0/65.3**
ALBERT	82.5/81.6	**83.8**/55.4	**78.0**/75.8	84.9/60.1
SC-ALBERT	**82.8/82.5**	81.7/**56.5**	77.8/**76.7**	**86.9/60.3**

Table 7: Experimental results of models on SQuAD 2.0 develop set and adversarial test sets. The slash is preceded and followed by the F1 value of examples with no answer and overall examples, respectively.

Model	Dev	CS	PQ	AS
BERT	53.6	35.6	52.0	23.4
SC-BERT	**54.9**	**47.0**	**55.8**	**43.4**
RoBERTa	54.8	45.0	51.6	27.0
SC-RoBERTa	**55.4**	**46.4**	**56.0**	**35.0**
ALBERT	57.0	**46.6**	54.2	36.6
SC-ALBERT	**58.0**	45.6	**56.0**	**48.8**

Table 8: Experimental results of models on ReClor develop set and different kinds of adversarial test sets.