What do we expect from Multiple-choice QA Systems?

Krunal Shah and **Nitish Gupta** and **Dan Roth**
University of Pennsylvania
{shahkr, nitishg, danroth}@seas.upenn.edu

Abstract

The recent success of machine learning systems on various QA datasets could be interpreted as a significant improvement in models' language understanding abilities. However, using various perturbations, multiple recent works have shown that good performance on a dataset might not indicate performance that correlates well with human's expectations from models that "understand" language. In this work we consider a top performing model on several Multiple Choice Question Answering (MCQA) datasets, and evaluate it against a set of expectations one might have from such a model, using a series of zero-information perturbations of the model's inputs. Our results show that the model clearly falls short of our expectations, and motivates a modified training approach that forces the model to better attend to the inputs. We show that the new training paradigm leads to a model that performs on par with the original model while better satisfying our expectations.[1]

1 Introduction

Question answering (QA) has been a prevalent format for gauging advances in language understanding. Recent advances in contextual language modelling have led to impressive results on multiple NLP tasks, including on several multiple choice question answering (MCQA, depicted in Fig. 1) datasets, a particularly interesting QA task that provides a flexible space of candidate answers along with a simple evaluation.

However, recent work (Khashabi et al., 2016; Jia and Liang, 2017; Si et al., 2019; Gardner et al., 2019, inter alia) has questioned the interpretation of these QA successes as progress in natural language understanding. Indeed, they exhibit, in various task

[1]Resources for this work are available at:
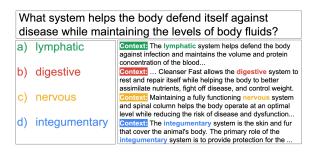http://cogcomp.org/page/publication_view/913

Figure 1: An example from ARC Easy dataset (Clark et al., 2018) showing the three MCQA task inputs.

settings, the brittleness of neural models to various perturbations. They also show (Kaushik and Lipton, 2018; Gururangan et al., 2018) how models could learn to latch on to spurious correlations in the data to achieve high performance on a given dataset. In this paper we continue this line of work with a careful analysis of the extent to which the top performing MCQA model satisfies one's expectation from a model that "understands" language.

We formulate the following set of (non-exhaustive) *expectation principles* that a MCQA model should satisfy.

Monotonicity Expectation: Model performance should not drop if an *incorrect* option is changed to make it even less likely to be correct.

Sanity Expectation: Model should perform poorly given trivially insufficient input.

Reading Expectation: Model should only choose an answer that is supported by the provided context (and thus perform poorly in the absence of informative context).

While we view the first two expectation principles as necessary axioms, the third could depend on one's definition of the MCQA task. An alternate definition could expect the MCQA model to answer questions using the provided context or, in its absence, using its internal knowledge. In this work, however, we use the Reading Expectation as

3547

phrased above; we believe that requiring a model to rely on externally supplied context better gauges its language understanding abilities, and levels the playing field among models with varying levels of internal knowledge.

Guided by these expectation principles we formulate concrete input perturbations to evaluate whether a model satisfies these expectations. We show that the top MCQA model fails to meet any of the expectation principles described above. Our results point to the presence of dataset artifacts which the model uses to solve the datasets, rather than the underlying task.

With goals and insights, we then propose (a) a different training objective – which encourages the model to score each candidate option on its own merit, and (b) an unsupervised data augmentation technique – which aims at "explaining" to the model the necessity of simultaneously attending to all inputs, to help the model solve the task. Our experiments on three popular MCQA datasets indicate that a model trained using our proposed approach better satisfies the expectation principles described above, while performing competitively as compared to the baseline model.

2 Multi-choice Question Answering

In this section, we briefly describe the multiple-choice question answering (MCQA) task, and the model and datasets we use in this work.[2]

MCQA Task   In a k-way MCQA task, a model is provided with a question q, a set of candidate options $\mathcal{O} = \{O_1, \ldots, O_k\}$, and a supporting context for each option $\mathcal{C} = \{C_1, \ldots, C_k\}$. The model needs to predict the correct answer option that is best supported by the given contexts. Figure 1 shows an example of a 4-way MCQA task.

Datasets   We use the following MCQA datasets:

1. **RACE** (Lai et al., 2017): A reading comprehension dataset containing questions from the English exams of middle and high school Chinese students. The context for all options is the same input paragraph.

2. **QASC** (Khot et al., 2020): An MCQA dataset containing science questions of elementary and middle school level, which require composition of facts using common-sense reasoning.

3. **ARISTO**: A collection of non-diagram science questions from standardized tests as used by Clark et al. (2019).[3]

For QASC and ARISTO, the context for an option is a set of top retrieved sentences as suggested by Khot et al. (2020) and Clark et al. (2019).

Baseline Model   We use the RoBERTa large model (Liu et al., 2019) for our experiments. Given the task inputs, the model learns to predict a distribution over the candidate options $\mathcal{O}$; which is computed by normalizing the scores for each candidate (using softmax) and the model is trained using cross entropy loss. To compute the score for the i-th candidate option O_i, the RoBERTa model is fed with the sequence "[CLS] C_i [SEP] q [SEP] O_i [SEP]" as input, and the representation of the [CLS] token is projected to a logit score (Clark et al., 2019).

For ARISTO and QASC, we first fine-tune the RoBERTa model on RACE, as suggested by Clark et al. 2019; Khot et al. 2020, and then on the respective datasets. More details on the training procedure can be found in the appendix.

3 Model vs. Our Expectations

In this section, we define the perturbations we design to evaluate a model against our expectation principles (defined in §1). We then analyze how well the baseline model satisfies these expectations.

Monotonicity Expectation: The following setting tests whether a model is fooled by an obviously incorrect option, one with high word overlap between its inputs.

- **Perturbed Incorrect Option (`PIO`):** The option description for an incorrect option is changed to the question itself and its corresponding context is changed to 10 concatenations of the question.[4]

Sanity Expectation: The following settings test how the model's performance changes when given an unreasonable input, for which it should not be possible to predict the correct answer.

- **No option (`NO`):** The option descriptions for all candidate options is changed to empty, "<s>".

- **No question (`NQ`):** The question (for all its candidate options $\mathcal{O}$) is changed to empty , "<s>".

Reading Expectation: The following setting tests how crucial the context is for the model to correctly answer the questions.

- **No context** (`NC`): The contexts for all candidate options is changed to empty, "<s>".

Baseline model performance Table 1 shows that the model achieves impressive accuracy on all three dataset; RACE (84.8), QASC (85.2), and ARISTO (78.3), which suggests that the model should satisfy the expectations laid out for a good MCQA model.

Evaluating expectations When evaluating the model by modifying an incorrect option and its context (`PIO`), we find that its performance drops notably across all three datasets, for example, from $85.2 \rightarrow 7.9$ for QASC. This shows that the model is not able to cope with an incorrect option containing high word overlap with the question, even when it is trivially wrong and the correct option and its context are present. The baseline model thus fails to satisfy the Monotonicity Expectation.

Given an unreasonable input, where a pivotal component of the input is missing, we find that the baseline model still performs surprisingly well. For example, in ARISTO, removal of the question (`NQ`) only leads to a performance drop from $78.3 \rightarrow 55.3$, and removal of the options (`NO`), from $78.3 \rightarrow 46.8$. This suggests that the datasets contain unwanted biases that the model relies on to answer correctly. This shows that the baseline model fails to satisfy the Sanity Expectation.

The model achieves reasonable performance on the removal of the contexts; thus failing our Reading Expectation, e.g., performance only drops from $78.3 \rightarrow 63.8$ in ARISTO. To achieve this performance the model must rely on its inherent knowledge (Petroni et al., 2019) or, more likely, on dataset artifacts as suggested previously.

4 Proposed Training Approach

To address the aforementioned limitations, and reduce the tendency of the model to exploit dataset artifacts, we propose the following modifications to the training methodology.

4.1 MCQA as Binary Classification

Treating MCQA as a multi-class classification problem requires the model to *minimally* differentiate the correct option from the incorrect options, thus

Eval. Setting	ARISTO	RACE	QASC
Original ($\uparrow$)	78.3	84.8	85.2
Perturbed Incorrect Option ($\uparrow$)	25.4	45.8	7.9
No Option ($\downarrow$)	46.8	–	50.2
No Question ($\downarrow$)	55.3	62.8	34.3
No Context ($\downarrow$)	63.8	49.1	55.8

Table 1: Accuracy of the respective RoBERTa models on RACE, ARISTO and QASC datasets for the different evaluation settings detailed in Section A. The No Option setting is not applicable for RACE as all options would have the same inputs. The arrows denote the expected performance where $\uparrow$ denotes higher is better and $\downarrow$ denotes that lower performance is better.

making the training sensitive to the relative difficulty between the options. We propose to prevent this by training the model to predict the *correctness* of each candidate option separately, by converting the k-way MCQA task into k binary classification tasks. The model is trained to predict a high probability for the correct option triplet (q, O_g, C_g), and low for the other $k - 1$ options.

4.2 Unsupervised data augmentation

We introduce an unsupervised data augmentation technique to discourage the model from exploiting spurious correlations between pairs of inputs and encourage it to read *all* the inputs. During training, given an MCQA instance $(q, \mathcal{O}, \mathcal{C})$, for each of the option triplet (q, O_i, C_i), we generate new examples (each with negative label) by performing one of the following perturbations with a certain probability (details in the appendix):

Option: O_i is changed to one of (a) empty ("<s>") or (b) $O_j \in \mathcal{O}; j \neq g$.

Context: C_i is changed to one of (a) empty ("<s>") or (b) $C_j \in \mathcal{C}; j \neq g$.

Question: q, for all options $\mathcal{O}$ is changed to one of (a) empty ("<s>") or (b) another question from the training set.

No change: The triple is left as is.

This is an automatic data augmentation and requires no manual annotation.

5 Results

The performance of our proposed training approach (+ Our Training) along with the baseline model are presented in Table 2. The new model performs

Dataset	Model	Original (O) (↑)	Perturbed Incorrect Option (PIO) (↑)	No Option (NO) (↓)	No Question (NQ) (↓)	No Context (NC) (↓)
ARISTO	RoBERTa	**78.3**	25.4	46.8	55.3	63.8
	+ Our Training	75.8	**55.5**	**26.9**	**35.4**	**42.4**
RACE	RoBERTa	**84.8**	45.8	–	62.8	49.1
	+ Our Training	83.9	**72.4**	–	**12.4**	**20.6**
QASC	RoBERTa	**85.2**	7.9	50.2	34.3	55.8
	+ Our Training	82.6	**38.0**	**13.7**	**12.3**	**34.7**

Table 2: Comparison of a model trained using our proposed training approach with the baseline model on RACE, ARISTO and QASC datasets. The evaluation settings used are described in Section A.

competitively (within 2.6 points) with the baseline on all three datasets suggesting that our proposed training approach only has minor impact on the overall model performance.

In our **PIO** setting, the new model outperforms the baseline on all three datasets by a large margin (55.5 compared to baseline's 25.4 on ARISTO indicating an improvement over the baseline with regard to our Monotonicity Expectation. Even though the data augmentation did not augment examples with this perturbation, our training approach helps the model better read the inputs and avoid distractor options.

When evaluating over unreasonable inputs in the **NO** and **NQ** settings, the resulting model performs poorly compared to the baseline (13.7 vs. 50.2 and 12.3 vs. 34.3 on QASC), showing that our training approach helps the model to not rely on dataset bias and satisfy the Sanity Expectation.

Finally, the new model performs poorly when we remove the contexts (e.g 20.6 on RACE), indicating how it is able to meet our Reading Expectations. The results also show the resulting model's reliance on the context for information required to correctly answer questions. Moreover, it implies that the resulting model is able to achieve performance similar to the baseline by heavily relying on information from the contexts, as opposed to the baseline that exploits dataset artifacts (as previously shown).

Results showing the performance of the model trained using binary classification loss, without the data augmentation, are attached in the appendix.

6 Related work

Our work builds on numerous recent works that challenge the robustness of neural language models (Jin et al., 2020; Si et al., 2019) or, more generally,

neural models (Kaushik and Lipton, 2018; Jia and Liang, 2017; Khashabi et al., 2016). Our evaluation settings – hiding one of the three inputs to the MCQA models – are similar to Kaushik and Lipton 2018's partial input settings which were designed to point out the existence of dataset artifacts in reading comprehension datasets. However, we argue that our results additionally point to a need for more robust training methodologies and propose an improved training approach. Our data augmentation approach builds on recent works (Khashabi et al., 2020; Kobayashi, 2018; Kaushik et al., 2020; Cheng et al., 2018; Andreas, 2020) that try to leverage augmenting training data to improve the performance and/or robustness of models. However most of these works are semi-automatic or require human annotation while our augmentation approach requires no additional annotation.

7 Conclusion

We formulated three *expectation principles* that a MCQA model must satisfy, and devised appropriate settings to evaluate a model against these principles. Our evaluations on a RoBERTa-based model showed that the model fails to satisfy any of our expectations, and exposed its brittleness and reliance on dataset artifacts. To improve learning, we proposed a modified training objective to reduce the model's sensitivity to the relative difficulty of candidate options, and an unsupervised data augmentation technique to encourage the model to rely on *all* the input components of a MCQA problem. The evaluation of our proposed training approach showed that the resulting model performs competitively with the original model while being robust to perturbations; hence, closer to satisfying our *expectation principles*.

References

Jacob Andreas. 2020. Good-enough compositional data augmentation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7556–7566, Online. Association for Computational Linguistics.

Yong Cheng, Zhaopeng Tu, Fandong Meng, Junjie Zhai, and Yang Liu. 2018. Towards robust neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1756–1766, Melbourne, Australia. Association for Computational Linguistics.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the AI2 reasoning challenge. *CoRR*, abs/1803.05457.

Peter Clark, Oren Etzioni, Daniel Khashabi, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, Niket Tandon, Sumithra Bhakthavatsalam, Dirk Groeneveld, Michal Guerquin, and Michael Schmitz. 2019. From 'f' to 'a' on the N.Y. regents science exams: An overview of the aristo project. *CoRR*, abs/1909.01958.

Matt Gardner, Jonathan Berant, Hannaneh Hajishirzi, Alon Talmor, and Sewon Min. 2019. On making reading comprehension more comprehensive. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 105–112, Hong Kong, China. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. 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 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

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, Copenhagen, Denmark. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:8018–8025.

Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. *International Conference on Learning Representations (ICLR)*.

Divyansh Kaushik and Zachary C. Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015, Brussels, Belgium. Association for Computational Linguistics.

Daniel Khashabi, Tushar Khot, and Ashish Sabharwal. 2020. Natural perturbation for robust question answering. *CoRR*, abs/2004.04849.

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, Peter Clark, Oren Etzioni, and Dan Roth. 2016. Question Answering via Integer Programming over Semi-Structured Knowledge. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. Qasc: A dataset for question answering via sentence composition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:8082–8090.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. 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 452–457, New Orleans, Louisiana. Association for Computational Linguistics.

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, Copenhagen, Denmark. 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. *CoRR*, abs/1907.11692.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Chenglei Si, Shuohang Wang, Min-Yen Kan, and Jing Jiang. 2019. What does BERT learn from multiple-choice reading comprehension datasets? *CoRR*, abs/1910.12391.

A   Training and Evaluation Details

We note the following points about the details of our experiments:

1. For our experiments on the modified training approach, only the final dataset specific fine-tuning step is modified to our approach.

2. For the proposed training method on RACE dataset, the augmented training approach was only applied after the first epoch of training.

3. Since our augmentation approach only modifies examples on the go, the number of examples the model sees in a single epoch remains the same. The baseline models were all fine-tuned for 4 epochs and the models using our proposed approach were finetuned for 1 extra epoch in all cases.

4. Training on NVIDIA TITAN RTX GPUs with 24GB of memory, one epoch on RACE, ARISTO and QASC required 4 hours, 40 minutes and 55 minutes respectively.

B   Data augmentation steps

During training (or prior to it), each example would be modified using the following steps:

```
For option opt in all options:
  If isCorrect(opt) and prob(0.2):
      Flip label of option opt
      With equal probability:
        1. With equal probability:
            context = "<s>"
            context = incorrect context
        2. With equal probability:
            option = "<s>"
            option = incorrect option
        3. With equal probability:
            question = "<s>"
            question = previous question
  If isIncorrect(opt) and prob(0.8)
      With equal probability:
        1. With equal probability:
            context = "<s>"
            context = incorrect context
        2. With equal probability:
            option = "<s>"
            option = incorrect option
```

C   Additional Results

Results showing the performance of the baseline model trained using binary classification loss are described in Table 3.

Dataset	Model	Original (O) ($\uparrow$)	Perturbed Incorrect Option (PIO) ($\uparrow$)	No Option (NO) ($\downarrow$)	No Question (NQ) ($\downarrow$)	No Context (NC) ($\downarrow$)
ARISTO	RoBERTa	**78.3**	25.4	46.8	55.3	63.8
	+ Binary Classification	75.5	41.6	39.7	51.8	61.3
	+ Our Training	75.8	**55.5**	**26.9**	**35.4**	**42.4**
RACE	RoBERTa	**84.8**	45.8	–	62.8	49.1
	+ Binary Classification	83.9	**75.1**	–	60.0	49.4
	+ Our Training	83.9	72.4	–	**12.4**	**20.6**
QASC	RoBERTa	**85.2**	7.9	50.2	34.3	55.8
	+ Binary Classification	84.1	11.1	45.8	39.2	54.6
	+ Our Training	82.6	**38.0**	**13.7**	**12.3**	**34.7**

Table 3: Results contrasting the performance of the baseline model trained using binary classification loss to the baseline model and the model trained using our proposed training approach on RACE, ARISTO and QASC datasets. The evaluation settings used are described in the paper.