MRCLens: an MRC Dataset Bias Detection Toolkit

Anonymous ACL submission

Abstract

Many recent neural models have shown remarkable empirical results in Machine Reading Comprehension, but evidence suggests sometimes the models take advantage of dataset biases to predict and fail to generalize on out-of-sample data. While many other approaches have been proposed to address this issue from the computation perspective such as new architectures or training procedures, we believe a method that allows researchers to discover biases, adjust the data or the models in an earlier stage will be beneficial. Thus, we introduce **MRCLens**, a toolkit which detects whether biases exist before users train the full model. For the convenience of introducing the toolkit, we also provide a categorization of common biases in MRC.

1 Introduction

The ability of machines to read and comprehend texts is a critical skill in natural language processing. Recently sophisticated neural network models such as BiDAF (Seo et al., 2016), RNet (Wang et al., 2017) and QANet (Yu et al., 2018) have achieved remarkable accuracies on several benchmark datasets like SQuAD (Rajpurkar et al., 2016). However, some popular datasets contain superficial patterns that can be exploited by models to make predictions without learning much about the contexts. As a result, the models might fail to generalize to out-of-sample datasets (Yogatama et al., 2019; Rimell et al., 2009; Paperno et al., 2016) or in adversarial settings (Jia and Liang, 2017; Wallace et al., 2019).

The community has approached the problem from the modelling perspective (Fisch et al., 2019; Takahashi et al., 2019). For example, a popular example is to first train a bias-only model based, and then combine it with a full model to learn the additional information (Sugawara et al., 2018). In addition, there are also diagnostic tools such as interactive frameworks (Lee et al., 2019) or attention matrix visualizer (Rücklé and Gurevych, 2017; Liu et al., 2018) to evaluate QA models. A common limitation of these approaches is we cannot discover the biases until the models have been trained and evaluated, which posted a challenge for such analysis when computational resources are limited.

Our study contributes to existing work by introducing a toolkit **MRCLens** which detects bias in MRC datasets. This toolkit tests a given dataset against several known biases before training the full model. Our toolkit can be applied to various SQuAD formatted MRC datasets. This also allows researchers to make adjustments to improve the datasets or develop models that target the existing biases. Along our implementation of the toolkit, we find it convenient to categorize the biases. Thus, our second contribution is a summary of common biases in MRC. Through literature reviews, we identify various recurring biases which can fall into three categories. We summarize them as **Similarity Bias**, **Keyword Bias** and **Question Bias**. Furthermore, we introduce the concept of 'distance' as a way to measure MRC bias. These concepts will be discussed in more detail in section 2.2.

2 Background

2.1 Related Work

The MRC task evaluates a system's ability to retrieve information and make meaningful inferences (Sutcliffe et al., 2013). Many recent neural models have shown remarkable results, but some models exploit dataset-specific patterns which fail to generalize (Clark et al., 2019; Talmor and Berant, 2019; Sen and Saffari, 2020). Min *et al*. observed that 92% of answerable questions in SQuAD can be answered only using a single context sentence (Min et al., 2018). When confounding sentences which have semantic overlap with the question were added to a dataset, the MRC model's perfor-

mance dropped significantly (Jia and Liang, 2017). In another experiment, many questions in an easier subset of the dataset had their answers in the most similar sentence and could be answered with word-matching (Sugawara et al., 2018). In Story Cloze Test tasks, recognizing the superficial features is essential for the models to achieve good performance (Schwartz et al., 2017). Consequently, many models lack certain advanced skills such as inference or multiple-sentence reasoning.

Biases can also come from a few informative key words. For example, entailment models trained on MNLI (Bowman et al., 2015) would guess answers based on whether a sentence-question pair contains the same words (McCoy et al., 2019) or solely the existence of keywords (Gururangan et al., 2018; Wang et al., 2019). Weissenborn demonstrated that more than a third of the questions were answered using a simple baseline model which prioritized answers with question words in the surrounding context (Weissenborn et al., 2017). Sugawara showed that certain questions might require specific lexical patterns around the correct answer (Sugawara et al., 2018). Researchers have also found certain important words were ignored by MRC models (Jia and Liang, 2017; Mudrakarta et al., 2018), while other less important patterns were overused (Mudrakarta et al., 2018). For example, when negations were added to the questions, datasets such as NewsQA or TriviaQA failed to update their answers (Sen and Saffari, 2020). Other works also found QA models can achieve good performance with incomplete inputs (Niven and Kao, 2019).

Furthermore, the questions by themselves sometimes contain clues used by models to locate an answer quickly. As early as 1999, the use of bag-of-words, when combined with other heuristics, achieved up to 40% accuracy for answering interrogative queries (Hirschman et al., 1999). Early researchers designed heuristic-rules based systems specifically to answer 'wh' questions (Riloff and Thelen, 2000). In more recent studies, some researchers have found that a notable proportion of the questions were still answerable when incomplete questions were given (Sugawara et al., 2018; Kaushik and Lipton, 2018). Other works showed that the models were not robust when questions were paraphrased (Ribeiro et al., 2018; Gan and Ng, 2019). Chen *et al* also found the existence of spurious correlations in WikiHop which were exploited by the model to achieve good performance using only the questions and answers without the contexts (Chen and Durrett, 2019). These studies suggests that keywords in the question allow the model to locate key information without having the model to read and comprehend the context.

2.2 Categories of dataset bias in MRC

Through the literature review, we observe that the most commonly seen biases in MRC can fall into three main categories. (1) Some biases directly exploit the relationship between the question and sentences similar to the question (that is, question-sentence pairs with high TFIDF scores), and we refer to them as **Similarity Bias**. (2) The biases can take advantage of a few key words in context. We refer to them as **Keyword Bias**. (3) The questions by themselves contain information which can be exploited by models to make predictions without carefully reading the passage. We refer to them as **Question Bias**.

The three types of biases are closely related to one another. The similarity between the question and the context usually refers to the TFIDF score, which can be understood as the *distance* between them. In fact, each category of bias relies on 'distance' at different scales. Similarity bias and keyword bias rely on the sentence-level or the local keyword-level distance from a passage to the targeted question. Likewise, question bias exploits the distance between question tokens and a passage. In fact, this is not a new concept. For example, previous researchers have applied this concept to incorporate distance supervision to enhance their QA models (Cheng et al., 2020). We are inspired by this abstraction to design our experiments and facilitate our discussion.

3 Overview of MRCLens

We are inspired by (Sugawara et al., 2020) to use ablation experiments to test the impact of biases. Perturbing the original dataset and reevaluating models using the perturbed data is a method used frequently in various fields of NLP (Belinkov and Bisk, 2017; Carlini and Wagner, 2018; Glockner et al., 2018). Sugawara and their colleagues presented 12 requisite skills which could be used to evaluate an MRC model. For each skill, they performed one corresponding ablation by perturbing the dataset. A comparison of the performance on the original dataset versus the perturbed dataset would indicate if the specific requisite skill is

needed by the model to answer questions. Their method fits the purpose of our study. However, the key difference is that, while they are interested in if specific requisite skills are needed, we aim to study if specific biases are needed by a model.

Our toolkit MRCLens incorporates existing works into a new tool which can detect if the biases described above exist in a given dataset at an earlier stage of the training process. MRCLens requires data to be SQuAD formatted and will be provided via github. It consists of three main parts:

(1) A preprocessing module which perturbs the original dataset in 8 ways corresponding to different biases, and tokenizes the data. Specifically, we divide the three categories of biases from section 2.2 into 8 bias units indexed from 1 to 8, and we relate each bias unit to one ablation. Define $\mathbf{X}$ as the feature space, $\mathbf{Y}$ as the labels, (x, y) as an (input, label) pair, and f be a model. Let b_i be a potential bias and m_i be a method which ablates the feature that provides the corresponding information n_j. Suppose $f(x) = y$ for some x in $\mathbf{X}$. We are interested in if $f(m_i(x)) = y$, which means x can be solved without information n_i.

(2) A neural-network MRC model which trains a model and evaluates it against both the original test data and the perturbed test data. This model is based on a baseline neural-network model put forward by (Clark et al., 2019). After preprocessing, we train a neural-network baseline model on the original training data. Then for each bias, we test the baseline model against the corresponding perturbed dataset. The model's performance on this new dataset would indicate to what extent the specific bias impacts the result.

(3) An evaluation module which presents the results in an organized format which allows for interpretation. MRCLens compares the performance between the original and the modified dataset. By checking whether the questions are solvable after ablations, we can interpret whether the presence of a specific bias leads to unintended but correct answers. When the performance gap is small, we can infer the bias b_i is used to answer the questions without n_i. If the gap is large, a notable proportion of the solved questions may require n_i.

4 Experiment and Discussion

4.1 Experiment Setup

We use SQuAD (version 1.1) for the experiment. The model is a recurrent co-attention model(Clark et al., 2019; Chen et al., 2016). The model consists of an embedding layer with character CNN, a co-attention layer, and a shared BiLSTM layer as the pooling layer. We use a 0.2 dropout rate, a learning rate decay of 0.999 every 100 steps(Clark et al., 2019). We use a plain loss function which computes the negative log likelihood given the model outputs and the labels. Ideally, MRCLens would be agnostic of the model architectures, since we care most about the changes in accuracy before and after ablation, not the accuracy itself.

4.2 Experiment Results

We performed four experiments to measure **Similarity Bias**, which refers to the similarity between a sentence in context and the question calculated based on TFIDF score. In experiments 1 and 2, we inject noise by adding a part of the question or the full question in front of a sentence that does not contain the original answer. This enhances the similarity score between the question and another sentence. If the model relies heavily on the most similar sentence to make predictions, then this change will misguide the model to look for answer span in the wrong place and lead the accuracy to drop. e_1 and e_2 use a truncated version of the dataset where only one question is kept per passage, because multiple questions are often asked based on one passage but it could be confusing to insert information from all questions.

In e_3, we shuffle the sentence order. If the performance doesn't change significantly, that means the model mainly relies on information from individual sentences, but not heavily on the contextual relationship between them.

Table 1: Similarity Bias - f1 drops after e_1, e_2 and minor change after e_3 suggest the model relies on context-question similarity but not so much on the inter-sentence relationships.

ablation	em	f1	f1 drop
e_1 insert full question	39.72	48.82	30.93
e_2 insert half question	53.36	64.13	15.62
e_3 shuffle sentence order	66.19	74.48	6.13

We performed two experiments to evaluate **Question Bias**. In e_4, we keep only the interrogative words in the question, and in e_5 we shuffle the order of words in the question. Finally, there are three experiments which measure **Keyword Bias**. We consider nouns, verbs and adjectives from questions as potential keywords and we insert them

Table 2: Question Bias - interrogatives alone can still be informative, and the sequence of question words is not essential for making predictions

ablation	em	f1	f1 drop
e_4 interrogatives	17.10	23.62	56.99
e_5 shuffle question words	56.08	64.05	16.56

respectively to a random sentence in the context other than the one containing the true answer. Like e_1 and e_2, we use the truncated dev dataset.

Table 3: Keyword Bias - key nouns from questions bring the more noise to contexts than verbs and adjectives.

ablation	em	f1	f1 drop
e_6 insert key nouns	51.28	62.29	17.46
e_7 insert key verbs	58.68	71.07	8.68
e_8 insert key adj.	59.55	72.29	7.46

e_3, e_4, e_5 use the original dev dataset with 10570 entries whose f1 score is 80.61%, while e_1, e_2, e_6, e_7, e_8 use the truncated dev dataset with 1943 entries and an f1 score of 79.75%. According to Table 1, accuracies dropped notably due to the added contents from the questions even though everything else remains the same. f1 drops from 80% to 64.13% when we insert half of the question, and to 48.82% when we insert the full question. The model is likely looking for answer in the sentence where question words were inserted, as it is now the most similar sentence. The result from e_3 informs us that the sentence order has very little influence on the model's prediction. Thus this dataset is not suitable for evaluating a model's ability to understand 'sentence-level compositionality'(Sugawara et al., 2020).

Results from Table 3 are consistent with those from Table 1. Our changes shortened the local distance between questions and words or short phrases. The drops in accuracies suggest the models were misled to some extent to search for answers around the inserted words. Nouns retain the most information from questions and thus bring most perturbation to the passages, while verbs and adjectives capture similar amount of information.

Finally, Table 2 suggests the questions alone contain indicative information that could be used when not considered in relation to the passages. In 17% of the cases, interrogatives are sufficient for the model to make predictions. e_5 shows the model's performance is affected only slightly after we shuffle the words to make the question nonsensible.

4.3 Discussion

The distances between questions and contexts are indicative of how biased the dataset is. For example, e_3 shuffles the sentence order but preserves the distance between sentences and questions, so it has the least effects on the performance. Through experiments 8,7,6,2,1, the noise we inserted to the original dataset gradually lengthens the relative distance between the correct answer. As we add key words or phrases to other parts of the paragraph, the effects of similarity bias or keyword bias are diluted because we enhance the relevance between the questions and other parts of the passages. The drop in f1 score increases from around 8% to 30.95% as we increase the noise from inserting keywords to inserting the full questions.

Our method also provides another way to interpret the similarity bias. The *distance* between the question and the context is one of the most discussed biases in MRC. Indeed, 80% of our dev dataset has the correct answer in the most similar sentence. e_2 inserted the full question into a random sentence in each passage so that the most similar sentence will always be the one where the question was inserted, but despite this change, the model still reached an exact match score of 39.72%. This suggests the model did not over-rely on the most similar sentence.

5 Conclusion

This study presents a toolkit MRCLens which can be used to detect dataset biases at the early stage of a study. MRCLens can be applied to SQuAD formatted datasets. It outputs helpful interpretations which help researchers to determine to what extent biases exist in the dataset of interest. In future work, we hope to enhance the toolkit to fit datasets of various formats, design methods to quantitatively evaluate the toolkit's outputs, and develop methodologies for other Machine Comprehension Tasks.

References

Yonatan Belinkov and Yonatan Bisk. 2017. Synthetic and natural noise both break neural machine translation. *arXiv preprint arXiv:1711.02173*.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Nicholas Carlini and David Wagner. 2018. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 1–7. IEEE.

Jifan Chen and Greg Durrett. 2019. Understanding dataset design choices for multi-hop reasoning. *arXiv preprint arXiv:1904.12106*.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2016. Enhanced lstm for natural language inference. *arXiv preprint arXiv:1609.06038*.

Hao Cheng, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2020. Probabilistic assumptions matter: Improved models for distantly-supervised document-level question answering. *arXiv preprint arXiv:2005.01898*.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. *arXiv preprint arXiv:1909.03683*.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 1–13, Hong Kong, China. Association for Computational Linguistics.

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.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking nli systems with sentences that require simple lexical inferences. *arXiv preprint arXiv:1805.02266*.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. *arXiv preprint arXiv:1803.02324*.

Lynette Hirschman, Marc Light, Eric Breck, and John D Burger. 1999. Deep read: A reading comprehension system. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pages 325–332.

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

Divyansh Kaushik and Zachary C Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. *arXiv preprint arXiv:1808.04926*.

Gyeongbok Lee, Sungdong Kim, and Seung-won Hwang. 2019. Qadiver: Interactive framework for diagnosing qa models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9861–9862.

Shusen Liu, Tao Li, Zhimin Li, Vivek Srikumar, Valerio Pascucci, and Peer-Timo Bremer. 2018. Visual interrogation of attention-based models for natural language inference and machine comprehension. Technical report, Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States).

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

Sewon Min, Victor Zhong, Richard Socher, and Caiming Xiong. 2018. Efficient and robust question answering from minimal context over documents. *arXiv preprint arXiv:1805.08092*.

Pramod Kaushik Mudrakarta, Ankur Taly, Mukund Sundararajan, and Kedar Dhamdhere. 2018. Did the model understand the question? *arXiv preprint arXiv:1805.05492*.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355*.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The lambada dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031*.

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

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging nlp models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865.

Ellen Riloff and Michael Thelen. 2000. A rule-based question answering system for reading comprehension tests. In *ANLP-NAACL 2000 Workshop: Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding Systems*.

Laura Rimell, Stephen Clark, and Mark Steedman. 2009. Unbounded dependency recovery for parser evaluation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 813–821.

Andreas Rücklé and Iryna Gurevych. 2017. End-to-end non-factoid question answering with an interactive visualization of neural attention weights. In *Proceedings of ACL 2017, System Demonstrations*, pages 19–24.

Roy Schwartz, Maarten Sap, Ioannis Konstas, Leila Zilles, Yejin Choi, and Noah A Smith. 2017. Story cloze task: Uw nlp system. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 52–55.

Priyanka Sen and Amir Saffari. 2020. What do models learn from question answering datasets? *arXiv preprint arXiv:2004.03490*.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Saku Sugawara, Kentaro Inui, Satoshi Sekine, and Akiko Aizawa. 2018. What makes reading comprehension questions easier? *arXiv preprint arXiv:1808.09384*.

Saku Sugawara, Pontus Stenetorp, Kentaro Inui, and Akiko Aizawa. 2020. Assessing the benchmarking capacity of machine reading comprehension datasets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8918–8927.

Richard FE Sutcliffe, Anselmo Penas, Eduard H Hovy, Pamela Forner, Alvaro Rodrigo, Corina Forascu, Yassine Benajiba, and Petya Osenova. 2013. Overview of qa4mre main task at clef 2013. In *CLEF (Working Notes)*.

Takumi Takahashi, Motoki Taniguchi, Tomoki Taniguchi, and Tomoko Ohkuma. 2019. CLER: Cross-task learning with expert representation to generalize reading and understanding. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 183–190, Hong Kong, China. Association for Computational Linguistics.

Alon Talmor and Jonathan Berant. 2019. Multiqa: An empirical investigation of generalization and transfer in reading comprehension. *arXiv preprint arXiv:1905.13453*.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing nlp. *arXiv preprint arXiv:1908.07125*.

Haohan Wang, Da Sun, and Eric P Xing. 2019. What if we simply swap the two text fragments? a straightforward yet effective way to test the robustness of methods to confounding signals in nature language inference tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7136–7143.

W Wang et al. 2017. R-net: machine reading comprehension with self-matching networks. natural language computer group, microsoft reserach. asia, beijing. Technical report, China, Technical Report 5.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural qa as simple as possible but not simpler. *arXiv preprint arXiv:1703.04816*.

Dani Yogatama, Cyprien de Masson d'Autume, Jerome Connor, Tomas Kocisky, Mike Chrzanowski, Lingpeng Kong, Angeliki Lazaridou, Wang Ling, Lei Yu, Chris Dyer, et al. 2019. Learning and evaluating general linguistic intelligence. *arXiv preprint arXiv:1901.11373*.

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*.