SQuAD2-CR: Semi-supervised Annotation for Cause and Rationales for Unanswerability in SQuAD 2.0

Gyeongbok Lee[1], **Seung-won Hwang**[2], **Hyunsouk Cho**[1]
[1]Knowledge AI Lab., NCSOFT Co., [2]Yonsei University
gbpalace@ncsoft.com, seungwonh@yonsei.ac.kr, dakgalbi@ncsoft.com

Abstract

Existing machine reading comprehension models are reported to be brittle for adversarially perturbed questions when optimizing only for accuracy, which led to the creation of new reading comprehension benchmarks, such as SQuAD 2.0 which contains such type of questions. However, despite the super-human accuracy of existing models on such datasets, it is still unclear how the model predicts the answerability of the question, potentially due to the absence of a shared annotation for the explanation. To address such absence, we release SQuAD2-CR dataset, which contains annotations on unanswerable questions from the SQuAD 2.0 dataset, to enable an explanatory analysis of the model prediction. Specifically, we annotate (1) explanation on why the most plausible answer span cannot be the answer and (2) which part of the question causes unanswerability. We share intuitions and experimental results that how this dataset can be used to analyze and improve the interpretability of existing reading comprehension model behavior.

Keywords: SQuAD 2.0, Machine Reading Comprehension, Corpus Annotation, Model Interpretability, Evaluation

1. Introduction

The machine reading comprehension (MRC) task aims to find useful information from unstructured text queried in the form of natural language. To solve this task, a model needs the ability to find the context associated with the question and infer the correct answer. Recently, data-driven learning methods are being actively studied, as many large-scale benchmark data are released and various resources on the web can be easily accessed and utilized.

Among MRC benchmarks, Stanford Question Answering Dataset (SQuAD) is the most widely adopted for evaluating the reading comprehension capabilities of a model, which evaluates how well the model predicts the answer span for a paragraph, given a natural language problem. As it is a large-scale, high-quality set of annotations obtained from crowdsourcing, many state-of-the-art methods use this dataset to train their models and show their effectiveness compared to previous approaches.

However, the first version of SQuAD was designed to have an answer span for all problems, which trained models to find the most relevant span regardless of whether the correct answer was actually inferred from the question. This bias is reported to degrade the robustness of the model for adversarial perturbed questions or paragraph. To solve this problem, SQuAD 2.0 was released, to include unanswerable problems obtained by crowdsourcing human perturbations, such as changing the word in a question or adding a question that is not related to the problem.

Although pretrained contextualized embedding, obtained through language modeling from large corpora, has enabled superhuman performance for both SQuAD 1.0 and 2.0, their robustness with respect to model behavior has been understudied. To illustrate, Figure 1 shows an existing MRC model, that can find the most plausible answer span (green span) and predict whether the question is answerable. However, we cannot evaluate whether the model identifies the right reason why green span cannot be the answer, for which we add blue (cause) and red (rationale) annotations in our dataset. This would enable a new analysis, such as Figure 2, comparing models in terms of which cause of unanswerability leads to their best predictions, examining six unanswerability causes we will explain later. Although existing papers present partial statistics or selected examples to show the robustness of the model over samples, these results cannot be directly compared as in Figure 2 because the number of samples is very small and the examples used in each paper are not identical.

We find that the lack of such analysis stems from the absence of a shared dataset with gold standard annotations. We thus build the extended dataset **SQuAD2-CR** (Cause and Rationales) based on unanswerable questions in the SQuAD 2.0 dataset to help researchers understand the RC model's behavior toward perturbed (thus unanswerable)

Article: Super Bowl 50
Passage (from Wikipdia): "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager.*
Question 1: "*What is the name of the quarterback who was 37 in Super Bowl XXXIII?*"
Answerability for Q1: Unanswerable (Number Swap)
Question 2: "*Who is the youngest quarterback ever to play in a Super Bowl?*"
Answerability for Q2: Unanswerable (Antonym)

Figure 1: Two examples from the SQuAD 2.0 dataset. Each green span shows the plausible answer to each question. In SQuAD2-CR, **Cause** annotates the words as blue in the question that explain unanswerability. **Rationales** annotates fine-grained reason as red why the plausible answer cannot be entailed by the question.

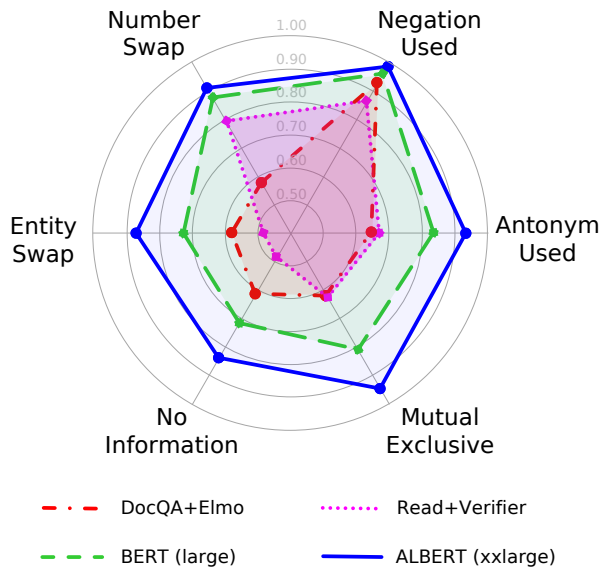

Figure 2: Spidergram for analysis of four MRC models for the aspect of question unanswerability.

questions. This dataset consists of two annotation sets for unanswerable question instances from SQuAD 2.0 in two contexts: **Cause** and **Rationales**. The former labels the reason class capturing why the plausible span cannot be the answer to the problem, and the latter offers the word-level reason why the question is not answered in the paragraph from the question. Some examples of SQuAD2-CR dataset annotations are shown in Figure 1. In the case of **Question 1**, *Number Swap* and *who was 37* becomes the cause and rationales, respectively, of the question's unanswerability. Beyond the binary class, these fine-grained annotations enable quantitative comparison and analysis of the robustness of models trained on the SQuAD 2.0, such as that in Figure 2, and can be useful indicators for tuning existing models. For example, one might notice the weakness of some models in addressing the *Entity Swap* cause and decide to augment training resources for such classes.

In this paper, we describe how we collect such annotation from the SQuAD 2.0 dataset and from our annotation and some preliminary experimental results show how our dataset can be used to analyze the model results. We also show how we can extend a dataset in scale for unlabeled question answer pairs using a semi-supervised approach to keep annotation overhead realistic. We release the dataset in `https://antest1.github.io/SQuAD2-CR/`.

2. Background and Related Works

As described in the previous section, the SQuAD 2.0 dataset aims to test the performance and robustness of MRC models by (a) understanding the question, (b) determining whether there is an answer span in the passage, and (c) predicting the most plausible answer span if one exists. This dataset consists of 97K answerable questions and 54K unanswerable questions about passages in Wikipedia.

In addition to the answer span information, this dataset also contains binary labels that indicate whether the question is answerable or not for the given passage. Compared to previous datasets such as (Clark and Gardner, 2017; Jia and Liang, 2017), SQuAD 2.0 has advantages in evaluating model robustness since it 1) contains rich adversarial perturbation made by humans, 2) pairs answerable and unanswerable questions in the same context, and 3) also marks an answer candidate for unanswerable questions. Existing works deal with the answerability of the model by adding a special loss function (Levy et al., 2017; Clark and Gardner, 2017) and/or extra classifier for answerability (Hu et al., 2019; Sun et al., 2018) that is incorporated into existing answer span finding architectures. Recent state-of-the-art works based on pretrained contextual embedding BERT (Devlin et al., 2018) utilize a classification (CLS) token to determine whether the question is answerable or not that is inserted as the beginning of the input text.

One distinction of our dataset is that it extends binary labels into the perturbation type. The authors of SQuAD 2.0 present seven categories of question unanswerability, including answerable noise, from 100 randomly sampled unanswerable questions to show the diversity of the dataset. Some works (Hu et al., 2019; Zhu et al., 2019) follow or modify these categories to analyze the robustness of their model. (Yatskar, 2018) classifies 230 unanswerable questions with different categories to compare SQuAD 2.0 and other question answering datasets. However, these are neither scalable nor reproducible since (1) the sample size is too small, as they are all less than 1K, and (2) there is no available public information on what instances they used for their analysis.

In contrast, another distinction of our dataset lies **in scale**: (1) SQuAD2-CR contains approximately 10K human-labeled annotations about cause in total, and these are propagated to all unanswerable questions on SQuAD 2.0 by semi-supervised learning. Additionally, (2) SQuAD2-CR shares the identifier information of SQuAD 2.0, making it easy to reproduce existing results and to make comparisons between models.

Our work is also related to the existing efforts to make MRC models that allow interpretation of their behavior. To analyze model behavior, (Wallace et al., 2019) provides gradient-based saliency maps and adversarial attacks for instance-level model interpretation as well as a suite of various interpretation techniques. (Lee et al., 2019), which targets the SQuAD 2.0 dataset, provides information on how the QA model contributes to the performance of the model by integrating visualizations and analysis tools for an explanation. (Wu et al., 2019) supports rule-based data grouping and counterfactual error analysis for effective error analysis of the model. These tools can provide some interpretable hints as to why the model works well, but they still lack an explicit explanation of the model's robustness or require manual definition.

Our dataset is complementary to these tools because it provides such explicit labels for explanations to extend their functionality. It can be used as a metric for evaluating model robustness with model attention and prediction results, as a training source to automatically perform data grouping or as a source to create adversarial examples of the desired type.

Table 1: Description, statistics, and examples for fine-grained unanswerable causes in SQuAD2-CR.

Name (Abbr.) Number of instances	**Description and Examples**
Entity Swap (**E**) Train 5818 / Dev 1122 (43.8% / 36.1%)	Entity replaced with other entity.
	P: The **USGS** has released a California Earthquake forecast which models ...
	Q: What did the **UGSS** release ?
Number Swap (**#**) Train 1642 / Dev 254 (12.3% / 8.2%)	Number or date replaced with other number or date.
	P: Internet2 announced a partnership ... boosting its capacity from **10** Gbit/s to **100** Gbit/s.
	Q: Who did Internet2 partner with to boost their capacity from **100** Gbit/s to **1000** Gbit/s?
Negation (**N**) Train 1860 / Dev 506 (14.0% / 16.3%)	Negation word inserted or removed.
	P: The principles of European Union law are rules of law which **have been developed** by the European Court of Justice, ...
	Q: Which entity **did not develop** the principles of European Union law?
Antonym (**A**) Train 2818 / Dev 593 (21.2% / 19.1%)	Antonym word for context is used in the question.
	P: Within two months of the launch, BSkyB **gained** 400,000 new subscribers, ...
	Q: How many subscribers were **lost** within two months of launch from BSkyB?
Mutual Exclusion (**X**) Train 318 / Dev 256 (2.4% / 8.2%)	Word or phrase is mutually exclusive with something for which an answer is present.
	P: CYP27B1, which is the gene responsible for converting the pre-hormone version of vitamin D, **calcidiol into the steroid hormone version, calcitriol**. ...
	Q: What gene **converts calcitriol into calcidiol?**
No Information (**I**) Train 841 / Dev 375 (6.3% / 12.1%)	Asks for condition that is not satisfied by anything in the paragraph, or paragraph does not imply any answer.
	P: **The state symbols include the pink heath (state flower)**, Leadbeater's possum ...
	Q: What is the **Victoria state color**?

3. Dataset Collection

SQuAD2-CR consists of two annotation sets for the **cause** and **rationale** of unanswerability. These are based on the questions that are marked as unanswerable.

3.1. Annotation on Cause

Description This annotation identifies why the question is not answerable based on the question and the most plausible answer span from passage. This is the common approach taken to offer examples demonstrating model robustness. Based on (Rajpurkar et al., 2018), we define six unanswerable reasons as follows:

- *Entity Swap* changes the entity in the question to another one, breaking the connection between the question and the passage.
- *Number Swap* changes the number or date in the question to another number or date. While the entity perturbation usually replaces one entity with another in the paragraph, the number perturbation replaces numbers with other values that do not exist.
- *Negation* inserts or removes negation words such as "not" in the question. This is the easiest example to generate and thus can be most easily determined by the model.
- *Antonym* replaces the word in the question with its antonym. This approach has the same effect as *Negation* but is more challenging to address if the model does not use a representation that can effectively separate the opposite words.
- *Mutual Exclusion* uses a word or phrase that is mutually exclusive with something for which the answer is present. It is different from *Antonym* because it does not simply use the opposite word but broadly changes the expression used in the question.
- *No Information* asks for a condition that is not satisfied by any information in the paragraph, or the paragraph does not imply any answer. This category usually indicates that the cause is not part of any other category, and questions tend to be entirely new instead of existing answerable questions that have been perturbed.

Some examples and statistics are described in Table 1. There are two differences between these categories and those in (Rajpurkar et al., 2018):

- We separated *Number Swap* from *Entity Swap*, since numeric values have different semantics than entities, as described above.
- We merge *Contradiction* and *Other Neutral* into a single category *No Information*, since there was large disagreement from annotators in the appropriate label between two classes.

Collection We manually annotate 16,403 questions with three annotators. We provide a word difference between the current question and the answerable question with the same answer span if possible to easily determine the perturbation of the question. We use a majority vote to merge the annotation results into a single annotation by taking the label confirmed by more than two annotators. For instance, when the same number is given different labels, the authors manually checked them and assigned one of the three labels based on the above definition. This usually occurred in the *No Information* class.

Table 2: Three types of examples for rationales annotation.

Simple Word Perturbation (Train 49.7% / Dev 52.4%)
What district of Warsaw chose the President between **1990** and **1993** ?
In what constituent country of the United Kingdom is **Trevithick** located ?
What is one **not** common example of a critical complexity measure ?
Phrase Perturbation (Train 16.2% / Dev 18.1%)
How many US Presidents **once campaigned in Cambridge** ?
What architecture type **came after Early Gothic** ?
When did the Sierra Sky Park **fall out of use** ?
Others: Complex Perturbation, Unrelated Question (Train 34.1% / Dev 29.5%)
Where is **Los Angeles a district of** ?
When was the **settlement which would become Boleslaw established** ?
What service did BSkyB **give away** for **free unconditionally** ?

What is the least used type of reduction
What is the most frequently employed type of reduction
0 0 0 1 1 1 0 0 0

Figure 3: Example of automatic question annotation

3.2. Annotation on Rationales

Description This annotation assigns a binary label to each word in the question to mark whether it contributes the question being unanswerable for the given passage and is inspired by the attention visualization of the neural network model.

Table 2 shows some examples of question labels on unanswerable questions. The bold-faced words are labeled as making questions unanswerable for the given passage. These labels indicate that the words play a decisive role when the MRC model determines whether the problem is answerable or not. Some common patterns would be single word replacement by other entities, antonym words or the insertion of negation words such as "not". More complex cases partially or completely alter the expression present in the paragraph, and these cases usually appear only in human perturbations.

Collection To generate such labels at scale, we first automatically annotate questions by 1) extracting answerable and unanswerable question pairs from SQuAD 2.0 sharing the same context and answer span and then 2) marking their intersection words as 0 and 1, with the assumption that questions sharing the context and exact answer span tend to contain similar intent regardless of answerability. While these methods are efficient for labeling many easy cases, some noise may exist, such as determiner changes, so we extract common conversion patterns and then refine some errors. We also manually annotate questions that do not have such a pair. Three annotators independently evaluate each question-answer pair. To merge annotation results into a single annotation, we use a majority vote: for each word, we label it as 1 only if more than two annotators mark the word because it is the word or part of the phrase that make the question unanswerable.

In this way, we annotate 24,771 and 3,695 instances from the SQuAD 2.0 training and development set. The limitation of this schema is that we cannot represent a removing perturbation on the question, such as removing "not". In this case, we do not assign 1 to all words in the question. One alternative way to represent word removal is adding extra slots, but we observed that this information is not well learned when expanding existing annotations.

4. Analysis of Existing MRC Models

Using our dataset, we analyze the output of the MRC models: DocQA+ELMO (Clark and Gardner, 2017), Read+Verifier (Hu et al., 2019), BERT (Devlin et al., 2018) and ALBERT (Lan et al., 2019). We also visualize attention from Read+Verifier and the ALBERT model for interpretation.

For the non-BERT models, DocQA utilizes the loss value from the answer span prediction to check answerability, while Read+Verifier introduces a new classifier for verifying the question and answer pair. In contrast, BERT-based models first pretrain deep bidirectional representations from large-scale unlabeled text without any explicit modeling for a specific task.

ALBERT is one of the variants of the BERT models, and it is currently the state-of-the-art model for various language understanding tasks, including SQuAD 2.0. This model uses two parameter-reduction techniques to reduce the parameters of the model and introduces sentence-order prediction loss to focus on modeling intersentence coherence. We expect other variants, such as RoBERTa (Liu et al., 2019), to show similar behaviors, as they share similar structures and training methods.

4.1. Cause Analysis

For all models, we classify the prediction results for unanswerable questions and then calculate no-answer accuracy (how well did the model identify the question's answerability) for each question. Table 3 and Figure 2 summarize the results of the models described above. Note that we used only human-labeled annotation for evaluation.

Table 3: Prediction accuracy for each unanswerability class evaluated by SQuAD2-CR (cause).

Model (EM / F1 / NoAns Acc)	NoAns Acc in each class					
	E	#	N	A	X	I
DocQA+ELMo 65.1 / 67.6 / 71.0	58.5	59.4	93.7	65.1	62.9	61.6
Read+Verifier 72.3 / 74.8 / 74.6	49.2	79.9	87.9	68.0	62.1	48.0
BERT (large) 71.7 / 81.9 / 84.7	72.5	89.4	98.4	84.7	80.5	71.7
ALBERT (xxlarge) 86.8 / 89.8 / 92.4	88.2	91.9	99.2	93.7	94.5	83.2

We can see that the DocQA+ELMo and Read+Verify have a significant performance degradation due to their failure in some cases. In BERT and ALBERT, the gap in the overall accuracy is significantly reduced in all unanswerable cases. This result indicates that the contextual representation from the pretrained BERT model plays an important role in determining not only the answer span but also the answerability of the question.

For **Negation** and **Number Swap**, both BERT and non-BERT models perform classifications relatively well, suggesting that these types of perturbation are an easy problem to solve with the model.

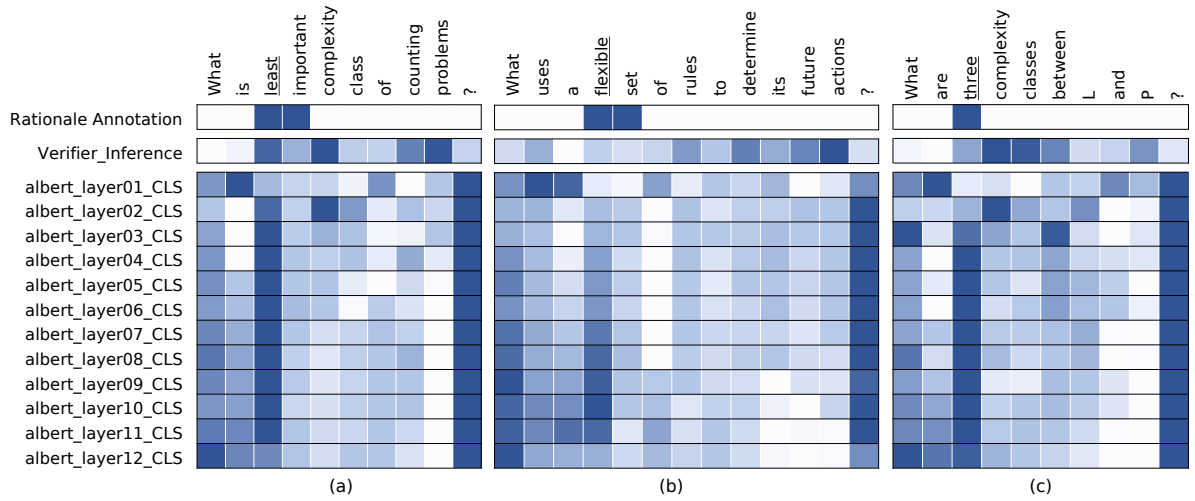

Figure 4: Rationale annotation in SQuAD2-CR and attention visualization for the Read+Verifier and ALBERT.

For **Antonym** and **Mutual Exclusive**, one possible source of the difference between the BERT and non-BERT models is the embedding space the model uses. Context-free word representations, trained with unsupervised learning such as GloVE, assume that semantically similar or related words appear in similar contexts– This may contribute to the failure to distinguish antonym words from synonyms (Mohammad et al., 2008; Hill et al., 2015). Using existing embeddings mixed with other embeddings considering antonyms, such as (Mrkšić et al., 2016), may solve this issue, especially in non-BERT models. The contextual representations used in BERT naturally solve this problem by using a multilayer architecture with a high capacity and training on a large corpus.

For **Entity Swap**, low performance can be associated with the limited discernment of the representation, resulting in the model mismatching the perturbed word in the question with the corresponding text in the passage. In non-BERT models, this problem would be alleviated by changing the tokenization method to cover more words, increasing vocabulary size, or increasing the dimension or the embedding size. The contextual representation performed by BERT will naturally solve this issue, as the representation of the word dynamically changes depends on the other words in the context.

For **No Information**, low performance indicates that the architecture in non-BERT models failed to capture the meaningful signature of contradiction or classify neutral relations to entailment. We guess that BERT has potentially learned to do this well when pretraining on two language modeling tasks. Its state-of-the-art performance in various natural language inference tasks, such as MNLI-m and QNLI, supports this conjecture.

4.2. Rationale Analysis

While disagreement exists about whether the standard attention modules provide meaningful explanations for model output (Jain and Wallace, 2019; Wiegreffe and Pinter, 2019), visualization of the attention layer output is still a common approach for explaining the model behavior. Our rationale annotations can guide an evaluation of whether a model places weight on the critical parts of a question as humans do when predicting the answerability of a question. For Verifier, we extract the attention weights, since each attention value matches the corresponding word from the question. For ALBERT, we follow (Tang et al., 2018) and visualize model attention. Specifically, we compare the scaled dot-product attention on the CLS token with that from each transformer layer of the model. We aggregate attention from each head with an element-wise average. As BERT-based models frequently use byte-pair encoding (Sennrich, 2016) or sentencepiece tokenizer (Kudo and Richardson, 2018) instead of word tokenizer for model input and are sometimes does not matched with a word-level token, we average the values from the subword tokens. We only consider the tokens from the question to obtain the word-level attention value.

Figure 4 visualizes the result for attention from the model and the corresponding rationale label. Read+Verifier predicted the answerability of the question correctly for (a) and (b) and incorrectly for (c), while ALBERT correctly predicted answerability for all problems.

In general, attention follows the rationale annotation for cases when the model is correct, as in (a). However, even if the answer is correct, Read+Verifier often does not follow the rationale, such as (b), and similar trends were observed in the early layers of BERT. This tendency suggests that it is difficult to predict unanswerability with single context matching; thus, the use of multilayer attention rather than single attention is crucial for such reasoning. In ALBERT, we can see this tendency, especially in the latter layers close to the final prediction, except for the last layer.

In ALBERT, the model gives high attention to structural words such as ? (question mark) since the model receives the concatenation of the question and the context as the input. While not shown in the figure, we can observe that special tokens such as [CLS] and [SEP] have the strongest attention value in the entire input. (Clark et al., 2019) Relative clauses such as *What* or Where also receive relatively high attention, which indicates that these words affect the prediction of both answer span (Palangi et al., 2018) and question answerability.

Although not all examples follow exactly the above observations, we can assume that the attention mechanism, when compared with the annotated rationale, helps to explain that the model actually concentrates well where the question is perturbed or makes the question unanswerable.

5. Semi-supervised Dataset Expansion

This section discusses an automated way to expand our annotation to cover all remaining unanswerable questions in SQuAD 2.0 or other benchmarks containing different passages. An intuitive way is to provide pseudolabels to unlabeled data using our annotations and semi-supervised approaches. Specifically, we apply tri-training (Zhou and Li, 2005), which is one of the strong baselines for neural semi-supervised learning for natural language processing (Ruder and Plank, 2018). In this algorithm, each initial unanswerable reason classifier is trained independently on bootstrapped samples (random sample with replacement); then, these classifiers are refined in the tri-training process, leveraging the agreement of three independent models for the final hypothesis to reduce the bias of predictions on unlabeled data.

Classifier Architecture Inspired by the recent success of the BERT, we employ existing MRC model layers as the base layer (pretrained layer) to obtain contextual representations from the question and passage pair, followed by feed-forward task-specific layers, e.g., cause prediction and rationale labeling. We observed that pretraining the base layer with the question answerability classification task before fine-tuning the model can lead to significant performance improvement compared to training from scratch.

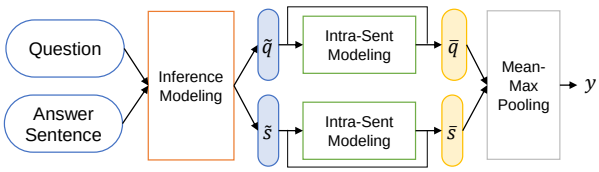

Figure 5: Brief overview of the interaction-based verifier. For more details on this model, please refer to (Hu et al., 2019).

In this paper, we show the result when using an interaction-based verifier model[1] as a base layer. Any other MRC models can be used as well if the model produces the probability of answerability when the question is given. For example, we can use the final embedding of the classification (CLS) token when using BERT (Devlin et al., 2018) fine-tuned on SQuAD 2.0, which is expected to have higher performance. We also release the extended dataset from the classification model described in later sections. This dataset covers all questions in the SQuAD 2.0 dataset. While these are more noisy than human annotation, we expect these can be used in distant learning. We are planning to update the extended dataset with the better model when available.

[1]We use our implementation of the model. The "no answer accuracy" (NoAns ACC) of our verifier implementation is 73.6%, while the reported performance is 74.6%. We neither use ELMo representation nor data augmentation for training the base verifier model.

Table 4: Unanswerable reason classification result given various settings, measured by micro-F1 score.

Model	Ratio of Answerable Questions (O)				
	0%	**10%**	**20%**	**33%**	**50%**
Majority	37.6	34.1	30.1	33.0	50.0
Scratch	64.1	58.3	55.7	50.9	52.8
FixBase	60.1	55.0	56.4	57.0	**61.5**
FixBase+	62.1	56.6	60.4	59.2	**63.4**
TuneBase	**70.9**	**67.8**	**60.6**	**61.9**	60.3
TuneBase+	**72.7**	**69.2**	**64.6**	**62.9**	62.9

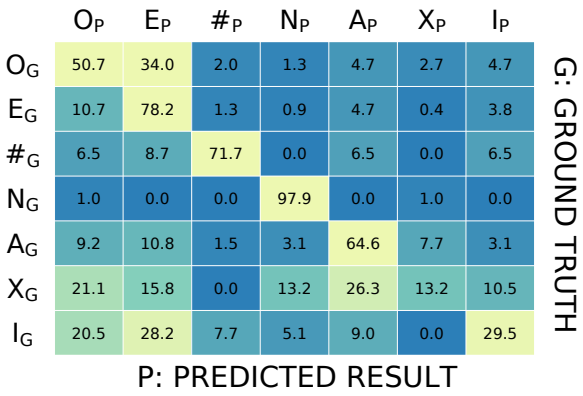

Figure 6: Confusion matrix for reason classification

5.1. Reason Classification for Cause Annotation

This task extends a binary question classification in SQuAD 2.0, which distinguishes only whether the question is answerable or not, by providing the cause label if the question is not answerable. To evaluate performance, we calculate the micro-F1 score to measure how well the model allocates an appropriate label to the question. We also randomly add answerable (**O**) questions from the original dataset to assess whether each unanswerable case is easily distinguished from the answerable cases.

We adopt majority selection as a baseline and compare reason classification models trained on the following settings:

- **Scratch**: Initialize only the word embedding layer with GloVe (Pennington et al., 2014) and train the model from scratch.
- **FixBase**: initialize the weights of the base layers with the pretrained model in the SQuAD 2.0 dataset and freeze weight.
- **TuneBase**: Similar to **FixBase**, but do not fix the base layer weights and keep whole layers trainable.
- **FixBase+**: same as **FixBase** but trained on extended dataset by semi-supervision.
- **TuneBase+**: same as **TuneBase** but trained on extended dataset by semi-supervision.

Table 4 shows the evaluation results for all methods described above. We observe an overall improvement in performance when using the semi-supervised method, indicating that incorporating unlabeled data can help to expand the small labeled data effectively. Additionally, better performance was achieved with a pretrained MRC model (TuneBase) for representation, but simply using the representation as an embedding is not effective (FixBase). When setting the portion of the answerable questions as half of the instances, the performance tendency was reversed, but we observe that there exists a bias toward predicting all questions as answerable due to a heavy class imbalance.

Figure 6 shows a confusion matrix on the model trained with 20% answerable questions. Simple pattern matching categories, such as number swap, negation, and antonym, are classified relatively well (high numbers in diagonals), while the other cases such as *Mutual Exclusion* (abbreviated as **X**) are relatively confused with some other cases.

In particular, we can observe that answerable questions and *no information* questions are frequently misclassified as *entity swap*. While the other unanswerable causes can be solved by finding a mismatching context between the question and passage, *Entity Swap* (especially when other entities in the passage are used) and *No Information* case need more than word matching to figure out since the important words in the question usually can be aligned to the words in passage, thus hard to be detached if the base model lacks the ability of textual entailment over simple matching. As *Entity Swap* is the majority label of the dataset, some borderline cases may be misclassified for this cause. This result is consistent with Section 4.1., which shows the importance of the performance of the base model.

5.2. Binary Question Labeling for Rationales Annotation

We treat this task similarly to a POS tagging problem with the binary label. With this setting, the feed-forward network predicts the binary label of each word, so the output is the sequence of probabilities that the word will be labeled as true (1). We provide word- and question-level evaluation: word-level accuracy checks whether word-level labels are correct, and the false negative rate (FNR) measures the ratio of true labels predicted as false (0). It is desirably for a model to have high accuracy and low FNR. We set the threshold as 0.5 for each word and apply similar settings with Section 5.1. to evaluate the model.

As the output of the question labeling model is the sequence of probabilities, the existing agreement schema used in tri-training is not applicable. We therefore calculate agreement by calculating the Euclidean distance instead of the majority vote and define agreement when two outputs have distances less than the threshold. We then make binary labels from the averaged output with the same threshold.

Table 5 shows the evaluation results for all methods described above. Similar to reason classification, initializing with a pretrained verifier gives better accuracy and a better FNR score than training from scratch, and the semi-supervised approach contributes to the overall accuracy of the model.

We illustrate prediction examples from the best model on the test set with various unanswerable reasons in Figure 7 to verify that this information can be used as a proxy label for the rationale behind question unanswerability. We can observe that the question labeling model can find both (a) simple patterns such as negation word insertion or number swap and (b) complex mutual exclusion. All of the questions in the first group were correctly predicted as unanswerable, while the prediction for the second group is relatively low. The example of mutual exclusion is a pair of questions and answer spans that disagree semantically but consist of words of similar meaning: "give away for free" and "charged additional subscription fees".

Table 5: Results with question labeling network

Model	Acc.	FNR	Model	Acc.	FNR
Majority (0)	73.3	100	Scratch	78.5	37.0
FixBase	79.8	51.6	TuneBase	80.1	38.6
FixBase+	81.5	47.2	TuneBase+	**82.3**	**36.7**

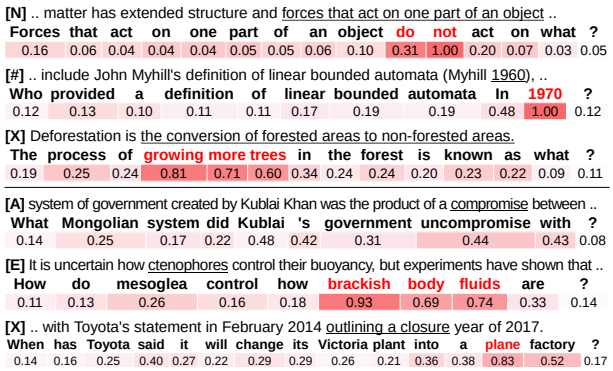

Figure 7: Visualization of question labeling prediction

6. Conclusion

In this paper, we release SQuAD2-CR, the largest dataset to our knowledge, which is annotated causes of and rationales for question unanswerability in the SQuAD 2.0 dataset. We annotate to indicate why the question is not answerable for the given passage considering two aspects: sentence-level cause and word-level rationale. Using these annotations, we interpreted how the existing MRC model predicts the answerability of the question with the output and attention weights. We also present some baseline classifier models for expanding annotations to unlabeled passage-question pairs using semi-supervised learning.

We think using a better MRC model (such as ALBERT in Section 4.) as a base layer or dealing with imbalanced classes will play a major role in further improving the performance of the model, and we left this as future work. Another possible future research direction is using this resource for training through the targeted augmentation of training resources for weak causes or by providing attention supervision to minimize the gap between the model and user desired attention (Das et al., 2017; Liu and Zhang, 2017). We hope that SQuAD2-CR can help to promote research on the explainability of reading comprehension models.

7. Bibliographical References

Clark, C. and Gardner, M. (2017). Simple and effective multi-paragraph reading comprehension. *arXiv preprint arXiv:1710.10723*.

Clark, K., Khandelwal, U., Levy, O., and Manning, C. D. (2019). What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.

Das, A., Agrawal, H., Zitnick, L., Parikh, D., and Batra, D. (2017). Human attention in visual question answering: Do humans and deep networks look at the same regions? *Computer Vision and Image Understanding*, 163:90–100.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Hill, F., Reichart, R., and Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Hu, M., Peng, Y., Huang, Z., Yang, N., Zhou, M., et al. (2019). Read+ verify: Machine reading comprehension with unanswerable questions. *AAAI*.

Jain, S. and Wallace, B. C. (2019). Attention is not explanation. In *Proceedings of the 2019 Conference of the NAACL*, pages 3543–3556.

Jia, R. and Liang, P. (2017). Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on EMNLP*.

Kudo, T. and Richardson, J. (2018). Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on EMNLP: System Demonstrations*, pages 66–71.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Lee, G., Kim, S., and Hwang, S.-w. (2019). Qadiver: Interactive framework for diagnosing qa models. In *AAAI*.

Levy, O., Seo, M., Choi, E., and Zettlemoyer, L. (2017). Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342.

Liu, J. and Zhang, Y. (2017). Attention modeling for targeted sentiment. In *Proceedings of the 15th Conference of the EACL*, pages 572–577.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Mohammad, S., Dorr, B., and Hirst, G. (2008). Computing word-pair antonymy. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 982–991. Association for Computational Linguistics.

Mrkšić, N., Séaghdha, D. O., Thomson, B., Gašić, M., Rojas-Barahona, L., Su, P.-H., Vandyke, D., Wen, T.-H., and Young, S. (2016). Counter-fitting word vectors to linguistic constraints. *arXiv preprint arXiv:1603.00892*.

Palangi, H., Smolensky, P., He, X., and Deng, L. (2018). Question-answering with grammatically-interpretable representations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Rajpurkar, P., Jia, R., and Liang, P. (2018). Know what you don't know: Unanswerable questions for squad. *Proceedings of the 56th Annual Meeting of the ACL*.

Ruder, S. and Plank, B. (2018). Strong baselines for neural semi-supervised learning under domain shift. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1044–1054. Association for Computational Linguistics.

Sennrich, R. (2016). How grammatical is character-level neural machine translation? assessing mt quality with contrastive translation pairs. *arXiv preprint arXiv:1612.04629*.

Sun, F., Li, L., Qiu, X., and Liu, Y. (2018). U-net: Machine reading comprehension with unanswerable questions. *arXiv preprint arXiv:1810.06638*.

Tang, G., Sennrich, R., and Nivre, J. (2018). An analysis of attention mechanisms: The case of word sense disambiguation in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 26–35.

Wallace, E., Tuyls, J., Wang, J., Subramanian, S., Gardner, M., and Singh, S. (2019). Allennlp interpret: A framework for explaining predictions of nlp models. *arXiv preprint arXiv:1909.09251*.

Wiegreffe, S. and Pinter, Y. (2019). Attention is not not explanation. 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 11–20.

Wu, T., Ribeiro, M. T., Heer, J., and Weld, D. S. (2019). Errudite: Scalable, reproducible, and testable error analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 747–763.

Yatskar, M. (2018). A qualitative comparison of coqa, squad 2.0 and quac. *arXiv preprint arXiv:1809.10735*.

Zhou, Z.-H. and Li, M. (2005). Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on knowledge and Data Engineering*, 17(11):1529–1541.

Zhu, H., Dong, L., Wei, F., Wang, W., Qin, B., and Liu, T. (2019). Learning to ask unanswerable questions for machine reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4238–4248. Association for Computational Linguistics.