

Learning to Ignore Adversarial Attacks

Yiming Zhang[†] Yangqiaoyu Zhou[†] Samuel Carton[‡] Chenhao Tan[†]

[†] University of Chicago [‡] University of New Hampshire

{yimingz0, zhouy1, chenhao}@uchicago.edu samuel.carton@unh.edu

Abstract

Despite the strong performance of current NLP models, they can be brittle against adversarial attacks. To enable effective learning against adversarial inputs, we introduce the use of rationale models that can explicitly learn to ignore attack tokens. We find that the rationale models can successfully ignore over 90% of attack tokens. This approach leads to consistent and sizable improvements ($\sim 10\%$) over baseline models in robustness on three datasets for both BERT and RoBERTa, and also reliably outperforms data augmentation with adversarial examples alone. In many cases, we find that our method is able to close the gap between model performance on a clean test set and an attacked test set and hence reduce the effect of adversarial attacks.

1 Introduction

Adversarial robustness is an important issue in NLP, asking how to proof models against confounding tokens designed to maliciously manipulate model outputs. As such models become more powerful and ubiquitous, research continues to discover surprising vulnerabilities (Wallace et al., 2019), demanding improved robustness methods.

A common defense method to combat adversarial attacks is adversarial training. Given knowledge of attack strategies, it constructs synthetic adversarial examples to augment clean examples during training (Zhang et al., 2020). Intuitively, the model will *implicitly* learn to ignore attacking tokens and become robust to that type of attack. In practice, however, this goal can be challenging through data augmentation alone.

In this study, we propose a simple yet effective adversarial training schema for additive attacks: *explicitly* training the model to ignore adversarial tokens. We do this by augmenting the underlying model with a rationale extractor (Lei et al., 2016) to serve as an input filter, and then training this

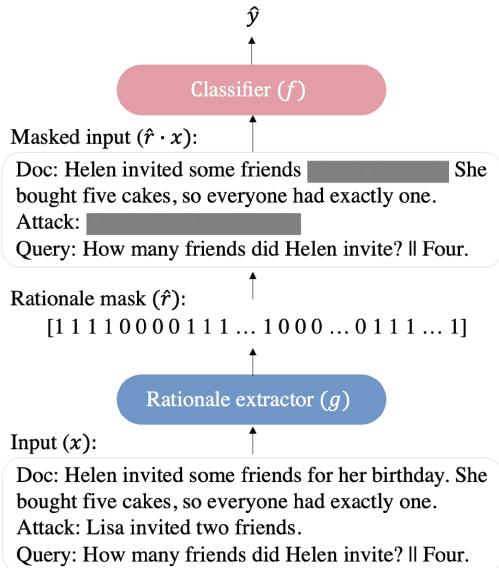


Figure 1: Example illustration of an ideal rationale model that is robust to added attack tokens. The rationale extractor filters out the confounding sentence (“Lisa invited two friends”) added by the adversary, and extracts the supporting spans (“Helen invited some friends” and “She bought five cakes, so everyone had exactly one”) to help the model deduce the correct answer (“Four”).

extractor to ignore attacking tokens as an additional joint objective to overall label accuracy (Fig. 1).

In addition to training the extractor to distinguish the attacking/non-attacking token dichotomy, we also explore the utility of human-provided explanations in this regard. In doing so, we ask: does learning from human rationales help the model avoid attending to attacking tokens?

Fine-tuning BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) on multiple datasets, we demonstrate that the additive attack proposed by Jia and Liang (2017) does reduce model accuracy, and that data augmentation with adversarial examples provides limited benefit in defending these models from this attack in most cases.

Our main results are that rationale-style models learn to ignore these attacks more effectively than only with data augmentation, leading to an improvement of $\sim 10\%$ in accuracy on attacked examples compared to baseline models and an advantage of 2.4% over data augmentation alone, mostly recovering clean test performance. While human explanations may potentially improve the interpretability of these models, they are of limited use in improving this defense even further.

In summary, we offer three main contributions:

- We show that explicitly training an extractive rationale layer to ignore attack tokens is more effective than implicitly training a model via data augmentation with adversarial examples.
- We assess whether human-annotated rationales augment this defense, showing that they have only a limited benefit.
- We conduct an in-depth error analysis of differences between models, explaining some of the patterns we observe in our main results.

Our code is available at <https://github.com/ChicagoHAI/rationalization-robustness>.

2 Related Work

We build on prior work on adversarial robustness and learning from explanations.

Adversarial robustness. Adversarial attacks against NLP models seek to maliciously manipulate model output by perturbing model input. [Zhang et al. \(2020\)](#) present a survey of both attacks and defenses. Example attacks include character-level manipulations ([Gao et al., 2018](#); [Li et al., 2019](#)), input removal ([Li et al., 2017](#); [Feng et al., 2018](#)), synonym substitutions ([Ren et al., 2019](#)), and language model-based slot filling ([Li et al., 2020](#); [Garg and Ramakrishnan, 2020](#); [Li et al., 2021](#)). A distinction in attack types is whether the attack requires access to the model ([Ebrahimi et al., 2018](#); [Yoo and Qi, 2021](#); [Wallace et al., 2019](#)) or not ([Alzantot et al., 2018](#); [Jin et al., 2020](#)). TextAttack ([Morris et al., 2020](#)) is a framework and collection of attack implementations. Our work focuses on the ADDSENT attack proposed by [Jia and Liang \(2017\)](#) in reading comprehension.

As interest in adversarial attacks has increased, so has interest in developing models robust to these attacks. A popular defense method is adversarial training via data augmentation, first proposed by [Szegedy et al. \(2014\)](#) and employed by [Jia and](#)

[Liang \(2017\)](#) to bring their model *almost* back to clean test performance. A recent example in this vein is [Zhou et al. \(2020\)](#), which proposes Dirichlet Neighborhood Ensemble as a means for generating dynamic adversarial examples during training. Another popular approach is knowledge distillation ([Papernot et al., 2016](#)), which trains an intermediate model to smooth between the training data and the final model. Our work explores a new direction that explicitly learns to ignore attacks.

Learning from explanations. Recent work has sought to collect datasets of human-annotated explanations, often in the form of binary *rationales*, in addition to class labels ([DeYoung et al., 2019](#); [Wiegreffe and Marasović, 2021](#)), and to use these explanations as additional training signals to improve model performance and robustness, sometimes also known as *feature-level feedback* ([Hase and Bansal, 2021](#); [Beckh et al., 2021](#)).

An early work is [Zaidan et al. \(2007\)](#), which uses human rationales as constraints on an SVM. More recently, [Ross et al. \(2017\)](#) uses human rationales to penalize neural net input gradients showing benefits for out-of-domain generalization, while [Erion et al. \(2021\)](#) use a similar method based on “expected gradients” to produce improvements in in-domain test performance in certain cases. [Katakka et al. \(2021\)](#) evaluate feature feedback for two attention-style models, finding, again, gains in out-of-domain performance, while [Han and Tsvetkov \(2021\)](#) use influence functions ([Koh and Liang, 2017](#)) to achieve a similar outcome. Where our study differs from most previous work is in using feature feedback for adversarial rather than out-of-domain robustness. A concurrent work by [Chen et al. \(2022\)](#) uses rationalization to improve robustness. The proposed method is similar to our work, but we explore supervision with attack tokens and achieve stronger robustness to additive attacks.

3 Adversarial Attacks and Datasets

In this paper, we focus on model robustness against the ADDSENT additive attack proposed by [Jia and Liang \(2017\)](#). The attack is designed for reading comprehension: consider each instance as a tuple of document, query, and label (d, q, y) , where y indicates whether the query is supported by the document. The attack manipulates the content of the query to form an attack sentence (A) and adds A to the document to confuse the model. Specifically, ADDSENT proceeds as follows:

Query q: FC Bayern Munich was founded in 2000.
Mutated Query \hat{q}: DYNAMO Leverkusen Cologne was founded in 1998.
Modified Document d' ...has won 9 of the last 13 titles. DYNAMO Leverkusen Cologne was founded in 1998. They have traditional local rivalries with ...

Figure 2: An example of the ADDSENT attack.

1. We modify the query q by converting all named entities and numbers to their nearest neighbor in the GloVe embedding space (Pennington et al., 2014). We flip all adjectives and nouns to their antonyms using WordNet (Miller, 1995) and yield a mutated query \hat{q} . If we fail to mutate the query due to not being able to find matching named entities or antonyms of adjectives and nouns, we skip the example.
2. We convert the mutated query \hat{q} into an adversarial attack A using CoreNLP (Manning et al., 2014) constituency parsing, under a set of about 50 rules enumerated by Jia and Liang (2017). This step converts it into a factual statement that resembles but is not semantically related to the original query q .
3. The adversarial attack A is inserted at a random location within the original document and leads to a new tuple (d', q, y) .¹

The key idea behind the ADDSENT attack is that the mutations alter the semantics of the query by mutating the named entities and numbers, so that the attack contains words or phrases that are likely confusing to the model without changing the true semantics of the input. An example of the ADDSENT attack is given above.

The original approach includes an additional step of using crowdsourced workers to filter ungrammatical sentences. We do not have access to this manual validation process in all datasets. Occasionally, ADDSENT generates ungrammatical attacks but it nevertheless proves empirically effective in reducing the performance of our models.

Datasets. To evaluate our hypotheses on learning to ignore adversarial attacks, we train and evaluate models on the Multi-Sentence Reading Comprehension (MULTIRC; Khashabi et al., 2018) and Fact Extraction and VERification (FEVER; Thorne et al., 2018) datasets. Both are reading comprehension datasets, compatible with the ADDSENT attack. For MULTIRC, the query consists of a question and potential answer about the document, labeled as true or false, while for FEVER it is a factual claim about the document labeled as “supported” or “unsupported”. Both datasets include *human rationales*, indicating which tokens are pertinent to assessing the query. Table 1 summarizes their basic statistics.

Dataset	Text length	Rationale length	Total size
MULTIRC	336.0	52.0	32,088
FEVER	335.9	47.0	110,187
SQuAD	119.8	—	87,599

Table 1: Basic statistics of MULTIRC, FEVER, and SQuAD.

In modeling these two datasets, we follow standard practice in appending the query to the end of the document with [SEP] tokens. We use train/validation/test splits prepared by the ERASER dataset collection (DeYoung et al., 2019). Because we are interested in relative differences between training regimes rather than absolute performance, we subsample the FEVER training set to 25% so that it is comparable to MULTIRC for the sake of training efficiency.

Directly applying the synthetic ADDSENT attack to MULTIRC and FEVER leads to occasionally ungrammatical adversarial examples due to incorrectly applied conversion heuristic or errors in constituency parsing. To alleviate this concern, we further evaluate on SQuAD (Rajpurkar et al., 2016), for which Jia and Liang provide an ADDSENT-attacked *evaluation* set that is re-written and approved by human workers. However, this dataset does not have human rationales. We again use the train/validation/test splits provided by Jia and Liang in our experiments.

4 Modeling

Our study assesses whether adding an explicit rationale extractor to a model and training it to ignore attack tokens results in a more effective defense than simply adding attacked examples to the training set. This comparison results in several combinations of model architectures and training regimes.

We denote each training instance as $(\mathbf{x}, \mathbf{r}, y)$: a text sequence \mathbf{x} consisting of the concatenated document and query, a ground-truth binary rationale sequence \mathbf{r} , and a binary label y .

¹We experimented with variants of inserting only at the beginning or the end. The results are qualitatively similar, so we only report random in this paper.

Human rationale & attack	ADV. + ATK. SUP.	ADV. + HUMAN SUP.
... and 18 national cups. FC Bayern was founded in 1900 by 11 football players, led by Franz John. Although Bayern won ... European Cup three times in a row (1974 – 1976). DYNAMO Leverkusen Cologne was founded in 1998. Overall, Bayern has reached ten European and 18 national cups. FC Bayern was founded in 1900 by 11 football players, led by Franz John. Although Bayern won ... European Cup three times in a row (1974 – 1976). DYNAMO Leverkusen Cologne was founded in 1998. Overall, Bayern has reached ten European and 18 national cups. FC Bayern was founded in 1900 by 11 football players, led by Franz John. Although Bayern won ... European Cup three times in a row (1974 – 1976). DYNAMO Leverkusen Cologne was founded in 1998. Overall, Bayern has reached ten European ...

Table 2: An example from FEVER illustrating different modes of adversarial training with rationale supervision. Human rationales are colored yellow, and attack tokens are colored red. Rationale models are supervised to extract the blue tokens, while ignoring the gray tokens.

Data augmentation?	Rationale?
No data augmentation	None Human (HUMAN SUP.)
Augmented with attack data (Adv.)	None Non-attack (ADV. + ATK. SUP.) Human (ADV. + HUMAN SUP.)

Table 3: Summaries of rationale model setups.

Baseline models and training. We use BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) as basis models. In the baseline training condition we fine-tune these models as normal, evaluating them on both the original test set and a version of the test set where each item has been corrupted with the ADDSENT attack described above. We denote this condition as “NO ADV.”

In the baseline adversarial training via data augmentation condition (denoted ADV.), we add ADDSENT-attacked versions of each training example to the training set on a one-to-one basis, allowing the model to train for the presence of such attacks. This represents a fairly standard baseline defense in the literature (Zhang et al., 2020).

Following prior adversarial robustness literature (Jia et al., 2019), we also consider a stronger baseline by augmenting the training set with K perturbed examples for each training example. For our main experiments, we use $K = 10$. This setting (denoted ADV.-10X) measures whether the baseline method implicitly adapts to the ADDSENT attack when abundant signal is provided.

Rationale model. To lend the baseline model an extractor capable of filtering out confounding tokens, we use the rationale model proposed by Lei et al. (2016). It comprises a rationale extractor g and a label predictor f (Fig. 1). The rationale extractor generates a binary predicted rationale \hat{r} , which is applied as a mask over the input to the predictor via

masking function m , producing a predicted label:

$$\begin{aligned} g(\mathbf{x}) &\rightarrow \hat{r} \\ f(m(\mathbf{x}, \hat{r})) &\rightarrow \hat{y} \end{aligned} \quad (1)$$

The two components are trained together to optimize predicted label accuracy as well as loss associated with the predicted rationale. In an unsupervised scenario, this loss punishes the norm of the predicted rationale, encouraging sparsity on the (heuristic) assumption that a sparse rationale is more interpretable. In this study, we rather consider the supervised scenario, where we punish \hat{r} ’s error with respect to a ground-truth rationale r . However, we find empirically that the rationale sparsity objective is useful in combination with the rationale supervision objective, leading to the following joint objective function using cross-entropy loss \mathcal{L}_{CE} with hyperparameter weights λ_1 and λ_2 :

$$\mathcal{L}_{CE}(\hat{y}, y) + \lambda_1 \mathcal{L}_{CE}(\hat{r}, r) + \lambda_2 \|\hat{r}\|. \quad (2)$$

Adversarial training with rationale supervision.

To introduce rationale supervision, we augment the training set with attacked examples on a one-to-one basis with original examples, similar to adversarial training. Moreover, we can change the ground-truth rationale to reflect the desired behavior for the model. We consider two options for this new ground-truth r : (1) a binary indicator of whether a token is adversarial or not (ADV. + ATK. SUP.), and (2) the human-annotated rationale (ADV. + HUMAN SUP.), which also filters adversarial tokens. Table 2 contains an example illustrating the distinction between ADV. + ATK. SUP. and ADV. + HUMAN SUP.

Table 3 summarizes all the combinations of setups that we use in our study. For each of these setups, we test one rationale model using independent BERT modules for g and f , and one using in-

dependent RoBERTa modules for both. We present additional implementation details in Appendix A.

Taken together, these conditions address our three research questions: (1) Is adversarial training via rationale supervision more effective than via attacked examples? (2) Does training the model to emulate human explanation make it intrinsically more robust to attacks? (3) Do human explanations improve upon adversarial training with non-attack tokens as rationale supervision?

5 Experimental Setup and Results

We start by describing our experimental setup and evaluation metrics. We then investigate model performance with different training regimes and conduct an in-depth error analysis.

5.1 Experimental Setup

Our study compares whether rationale-style models are better at learning to explicitly ignore adversarial tokens than standard models via adversarial training. As we describe above, we train three variants of the standard classification model (NO ADV., ADV., ADV.-10X), and three variants of the rationale model (ADV. + ATK. SUP., HUMAN SUP., ADV. + HUMAN SUP.).

Exploring these 6 architecture/training combinations for three datasets (MULTIRC, FEVER, and SQuAD) and two underlying models (BERT and RoBERTa), we report results from all trained models in Table 4. We report relevant metrics on both the clean test set and the attacked test set for each model. For MULTIRC and FEVER, the metric we use is accuracy. Since SQuAD is a span extraction task, we report the Span F1 score instead. Performance on the attacked test set is our key measure of robustness.

Additionally, for the rationale models, we report the mean percentage of attack and non-attack tokens included in each predicted rationale, two metrics that help explain our accuracy results. The mean percentage of attack tokens included in the predicted rationale indicates the effectiveness of ignoring attack tokens: the lower the better.

5.2 Main Results

We focus our analysis on three questions:

1. Does adversarial rationale supervision on augmented data improve robustness over adversarial data augmentation alone?

2. Does human rationale supervision improve adversarial robustness over a standard model?
3. Does the addition of human rationales to adversarial training further improve robustness?

Table 4 summarizes the main results of the paper, showing the accuracy of each combination of architecture, training regime, underlying model and dataset. Looking at the attacked versus clean test set performance for the standard model, we see that **the ADDSENT attack is effective**, reducing accuracy on MULTIRC ($\sim 6\%$), FEVER ($\sim 10\%$), and SQuAD ($\sim 12\text{-}24\%$).

Adversarial rationale supervision (ADV. + ATK. SUP.). Rationale models provide an interface for explicitly supervising the model to ignore attack tokens. Our *key* question is whether they can be used to improve the effectiveness of adversarial training. We first discuss the effect of data augmentation and then show that rationale models are indeed more effective at ignoring attack tokens.

Data augmentation with adversarial examples works, mostly. In almost all cases, data augmentation does result in improved performance on the attacked test set, improving +5.9% (FEVER) and +17.6% (SQuAD) for BERT, as well as +6.4% (MULTIRC), +9.7% (FEVER), and +9.4% (SQuAD) for RoBERTa. The exception is BERT on MULTIRC, where it causes a decrease of -1.0%. However, in only one case out of six does data augmentation with adversarial examples bring the model back to clean test performance (RoBERTa on MULTIRC, +0.3%).

Surprisingly, BERT on MULTIRC is the only scenario where the ADV.-10X augmentation significantly improves attack accuracy (4.3% improvement over ADV.). In all the other cases, adding more adversarial examples does not improve robustness and even leads to a 3.5% drop in SQuAD for RoBERTa. This result demonstrates that BERT and RoBERTa may not learn from adversarial examples alone.

Adversarial rationale supervision improves on adversarial training baselines in all cases. We see an improvement of +4.6% for BERT on MULTIRC, +2.9% for BERT on FEVER, +2.7% for BERT on SQuAD, +2.2% for RoBERTa on MULTIRC, +0.7% for RoBERTa on FEVER, and +1.0% for RoBERTa on SQuAD (2.4% on average). For the one case where adversarial data augmentation recovered clean test performance (RoBERTa on MULTIRC), adversarial rationale supervision actu-

Model	Architecture	Training	MULTIRC (Acc.)		FEVER (Acc.)		SQuAD (Span F1)	
			Clean	Attacked _S	Clean	Attacked _S	Clean	Attacked _H
BERT	Standard	NO ADV.	68.6	62.6	88.2	78.9	86.4	62.8
		ADV.	67.3	61.6	88.5	84.8	86.0	80.4
		ADV.-10X	66.2	65.9	86.3	84.5	82.2	78.0
	Rationale	ADV. + ATK. SUP.	69.6	66.2	87.1	87.7	86.5	83.1
		HUMAN SUP.	70.0	64.4	88.0	76.7	---	---
		ADV. + HUMAN SUP.	70.5	69.4	87.5	87.5	---	---
RoBERTa	Standard	NO ADV.	82.6	76.5	93.5	83.0	93.2	81.0
		ADV.	84.4	82.9	93.2	92.7	92.9	90.4
		ADV.-10X	83.5	82.1	93.5	93.2	89.9	86.9
	Rationale	ADV. + ATK. SUP.	85.2	85.1	93.4	93.4	93.3	91.4
		HUMAN SUP.	84.0	74.9	94.1	85.7	---	---
		ADV. + HUMAN SUP.	85.0	82.5	93.4	93.4	---	---

Table 4: Model performance on clean and attacked test sets for MULTIRC, FEVER, and SQuAD. Attacked_S are synthetic attacks produced by ADDSENT, and Attacked_H are attacks generated by human workers. We vary the level of augmentation for the standard classification models (NO ADV., ADV., ADV.-10X). For rationale models, we control for the presence of adversarial training data and the type of rationale supervision: ADV. + ATK. SUP. treats non-attack tokens as rationale, and HUMAN SUP. does not use adversarial training. Rationale models outperform the baseline classifiers across all attacked datasets.

Model	Training	MULTIRC		FEVER		SQuAD	
		Attack %	NON-A %	Attack %	NON-A %	Attack %	NON-A %
BERT	ADV. + ATK. SUP.	1.4	98.4	0.2	96.7	27.8	99.7
	HUMAN SUP.	87.5	8.2	66.7	17.8	---	---
	ADV. + HUMAN SUP.	9.5	14.4	0.5	24.4	---	---
RoBERTa	ADV. + ATK. SUP.	6.0	96.7	0.9	95.8	16.1	99.0
	HUMAN SUP.	92.4	12.6	60.0	12.2	---	---
	ADV. + HUMAN SUP.	32.1	15.6	0.1	23.0	---	---

Table 5: Percentage of attack and non-attack (NON-A) tokens *included* in the predicted rationales. Lower is better for attack tokens. Arguably, a lower percentage of non-attack tokens is also better as it improves interpretability.

ally improves clean test performance by +2.5%.

The effectiveness of ADV. + ATK. SUP. is even more salient if we compare with NO ADV. on attacked test: 3.6%, 8.8%, and 20.3% for BERT on MULTIRC, FEVER, and SQuAD, 8.6%, 10.4%, and 10.4% for RoBERTa on MULTIRC, FEVER, and SQuAD (10.4% on average).

The above findings remain true even when we compare our methods against the theoretically stronger baseline of ADV.-10X, where the training dataset is augmented with 10 perturbed examples for every training example. Our models trained with adversarial rationale supervision outperforms ADV.-10X across all datasets and models, and our best model outperforms the ADV.-10X baseline by 3.3% on average. This result highlights both the efficiency and the effectiveness of our method: with adversarial rationale supervision, BERT and RoBERTa achieve greater defense against the ADDSENT attack using 10% of the ad-

versarial examples.

Interestingly, the adversarially-supervised rationale model demonstrates a strong ability to generalize knowledge learned from synthetic attacks to tune out human-rewritten attacks (+20.3% on SQuAD; recall we do not have human-rewritten attacks during training), indicating the potential of our method in a real-world scenario.

Table 5 explains this success. The adversarially-supervised rationale model includes 6% or fewer attacking tokens on MULTIRC and FEVER, indicating that it did largely succeed in learning to occlude these tokens for the predictor. Additionally, both BERT and RoBERTa rationale models are able to tune out most human-generated attack tokens, ignoring over 70% of attack tokens while keeping 99% of the original text for both models.

Effect of human rationale supervision alone (HUMAN SUP.). We find mixed evidence for whether human rationale supervision alone im-

proves adversarial robustness. For BERT on MULTIRC and RoBERTa on FEVER, human rationale outperforms the standard classification model, but the opposite occurs for the other two model/dataset combinations.

Table 5 contextualizes this mixed result: the rationale model supervised solely on human rationales includes 60.0% to 92.4% of attack tokens in its rationale (compared to between 8.2% and 17.8% of non-attack tokens), indicating that it is largely fooled by the ADDSENT attack into exposing the predictor to attack tokens.

This result may be explained by the fact that human rationales for these datasets identify the part of the document that pertains particularly to the query, while the ADDSENT attack crafts adversarial content with a semantic resemblance to that same query. Hence, it is understandable that human rationale training would not improve robustness.

Human and adversarial rationale supervision (ADV. + HUMAN SUP.). Although human rationales alone may not reliably improve model robustness, a final question is whether human rationales can serve as a useful addition to adversarial training. Does training the model to both ignore adversarial tokens and emulate human explanations further improve robustness against the ADDSENT attack?

In two out of four cases, the performance of ADV. + HUMAN SUP. is equal to that of ADV. + ATK. SUP. Only for BERT on MULTIRC does ADV. + HUMAN SUP. result in an improvement, being the only configuration that brings performance back to that of clean test for that model, and dataset. For RoBERTa on MULTIRC, it actually weakens attacked test performance.

While these results are mixed, Table 5 shows that the model does at least achieve this result at a much lower included percentage of non-attack tokens ($\sim 20\%$ vs. $> 95\%$), a concession toward model interpretability.

Overall, our results suggest that human rationales have limited effect in defending against adversarial attacks, but can be important in developing sparse (and potentially interpretable) models.

5.3 Error Analysis

To better understand the behavior of the models, we examine mistakes from BERT compared to explicitly training a rationale extractor on MULTIRC. We start with a qualitative analysis of example errors, and then discuss general trends, especially on why

human rationales only provide limited benefits over ADV. + ATK. SUP. More in-depth analyses can be found in the appendix for space reasons, including a Venn diagram of model mistakes.

Qualitative analysis. We look at example errors of ADV. to investigate attacks that are confusing even after adversarial augmentation. Table 6 shows example outputs of the rationale models based on either non-attack tokens or human rationales.

Example 1 shows a case where models with explicit rationale extractors ignore attacks more effectively than ADV. In the attack sentence, “*tete didn’t stay in*” is highly similar to the query, so a model likely predicts True if it uses the attack information. In comparison, both rationale models ignore the attack in label prediction, which enables them to make correct predictions.

Example 2 demonstrates that ADV. + HUMAN SUP. makes mistakes when it fails to include crucial information in rationales while avoiding attack tokens. ADV. + HUMAN SUP. predicts the wrong label because it misses information for the number of friends in its rationale. ADV. + ATK. SUP. gets this example correct because it can both ignore the attack and include the necessary information.

Finally, Example 3 shows an example where ADV. + HUMAN SUP. is better than ADV. + ATK. SUP. when generating rationales to ignore noises. ADV. + HUMAN SUP. includes attacks in rationale, but it is still able to predict the label because the attack is not confusing given the selected rationale. The generated rationale helps ADV. + HUMAN SUP. to avoid unnecessary information that may confuse the model. For example, the sentence with “picts” could confuse the model to predict True. On the other hand, ADV. + ATK. SUP. gets this example wrong, despite occluding all attack tokens.

More generally, we find that ADV. + HUMAN SUP. tends to have high false negative rates. When ADV. + HUMAN SUP. fails to extract good rationales, it tends to predict False due to missing information from the rationale. In contrast, ADV. + ATK. SUP. rarely occludes necessary information, so it does not suffer from the same issue.

ADV. + ATK. SUP. is better than ADV. + HUMAN SUP. when human rationales are denser and passage length is longer (see Table 9 in the appendix). We observe that denser human rationales usually comprise evidence from different parts of the passage. Since ADV. + ATK. SUP. predicts almost all non-attack tokens as rationale, they have

Human rationale & attack	ADV. + ATK. SUP.	ADV. + HUMAN SUP.
(A) Example 1, true label: False		
[CLS] ... in may 1904 , the couple ' s first son , hans albert einstein , was born in bern , switzerland . their second son , eduard , was born in zurich in july 1910 . in 1914 , the couple separated ; einstein moved to berlin and his wife remained in zurich with their sons . they divorced on 14 february 1919 , having lived apart for five years a - tête did n ' t stay in basel after charles and houben separated [SEP] who did n ' t stay in zurich after albert and marie separated ? tête [SEP] ADV. prediction: True	[CLS] ... in may 1904 , the couple ' s first son , hans albert einstein , was born in bern , switzerland . their second son , eduard , was born in zurich in july 1910 . in 1914 , the couple separated ; einstein moved to berlin and his wife remained in zurich with their sons . they divorced on 14 february 1919 , having lived apart for five years a - tête did n ' t stay in basel after charles and houben separated [SEP] who did n ' t stay in zurich after albert and marie separated ? tête [SEP] ADV. + ATK. SUP. prediction: False	[CLS] ... in may 1904 , the couple ' s first son , hans albert einstein , was born in bern , switzerland . their second son , eduard , was born in zurich in july 1910 . in 1914 , the couple separated ; einstein moved to berlin and his wife remained in zurich with their sons . they divorced on 14 february 1919 , having lived apart for five years a - tête did n ' t stay in basel after charles and houben separated [SEP] who did n ' t stay in zurich after albert and marie separated ? tête [SEP] ADV. + HUMAN SUP. prediction: False
(B) Example 2, true label: True		
[CLS] ... on the day of the party , all five friends showed up . each friend had a present for susan . 6 thank - you cards did helen send . susan was happy and sent each friend a thank you card the next week . [SEP] how many thank - you cards did susan send ? 5 [SEP] ADV. prediction: False	[CLS] ... on the day of the party , all five friends showed up . each friend had a present for susan . 6 thank - you cards did helen send . susan was happy and sent each friend a thank you card the next week . [SEP] how many thank - you cards did susan send ? 5 [SEP] ADV. + ATK. SUP. prediction: True	[CLS] ... on the day of the party , all five friends showed up . each friend had a present for susan . 6 thank - you cards did helen send . susan was happy and sent each friend a thank you card the next week . [SEP] how many thank - you cards did susan send ? 5 [SEP] ADV. + HUMAN SUP. prediction: False
(C) Example 3, true label: False		
[CLS] ... roman legions encountered the strongholds of the castle rock and arthur ' s seat , held by a tribe of ancient britons known as the votadini . the mercians were probably the ancestors of the manaw . little is recorded about this group , but they were probably the ancestors of the gododdin , whose feats are told in a seventh - century old welsh manuscript the god ... din ... [SEP] who were probably the ancestors of the gododdin ? the pict [SEP] ADV. prediction: True	[CLS] ... roman legions encountered the strongholds of the castle rock and arthur ' s seat , held by a tribe of ancient britons known as the votadini . the mercians were probably the ancestors of the manaw . little is recorded about this group , but they were probably the ancestors of the gododdin , whose feats are told in a seventh - century old welsh manuscript the god ... din ... [SEP] who were probably the ancestors of the gododdin ? the pict [SEP] ADV. + ATK. SUP. prediction: True	[CLS] ... roman legions encountered the strongholds of the castle rock and arthur ' s seat , held by a tribe of ancient britons known as the votadini . the mercians were probably the ancestors of the manaw . little is recorded about this group , but they were probably the ancestors of the gododdin , whose feats are told in a seventh - century old welsh manuscript the god ... din ... [SEP] who were probably the ancestors of the gododdin ? the pict [SEP] ADV. + HUMAN SUP. prediction: False

Table 6: Example outputs from ADV. + ATK. SUP. and ADV. + HUMAN SUP. with BERT in MULTIRC. Attack tokens are marked in red. True human rationales are marked in yellow, and predicted rationales are marked in blue. We only show tokens where generated rationales disagree with each other or with the human rationale/attack.

higher human rationale recall (98.6%) than ADV. + HUMAN SUP. (57.6%). Thus, ADV. + ATK. SUP. generates higher quality rationales when human rationales are dense. Similarly, long passages prove difficult for ADV. + HUMAN SUP.

In summary, these analyses highlight the challenges of learning from human rationales: it requires precise occlusion of irrelevant tokens while keeping valuable tokens, and must account for variance in human rationale and input lengths. These challenges partly explain the limited benefit of ADV. + HUMAN SUP. over ADV. + ATK. SUP.

6 Concluding Discussion

In this study, we find that adding an explicit extractor layer helps a model learn to ignore additive adversarial attacks produced by the ADDSENT attack more effectively than conventional adversarial training via data augmentation.

This is an exciting result because it defeats an attack which is otherwise stubbornly effective against even copious adversarial data augmentation. It is a novel use for this type of explicit token relevance representation, which is more typically applied for model interpretability (Lei et al., 2016).

This makes it related to defenses like Cohen et al. (2019) which allow the model to reject inputs as out-of-distribution and abstain from prediction, but it differs in rejecting only part of the input, making a prediction from the remainder as usual.

Generality. As Carlini et al. (2019) note, it is easy to overstate claims in evaluating adversarial defenses. Hence, we note that our results pertain only to the ADDSENT attack, and perform favorably against a baseline defense in adversarial training via data augmentation. Since most adversarial training approaches assume the ability to *generate* a large number of synthetic attack examples, it is reasonable to further assume that we have access to the positions of the attacks. However, such knowledge about attacks may not be available in general. Nevertheless, the success of the rationale model architecture in learning to occlude adversarial tokens does hold promise for a more general defense based on a wider range of possible attacks and possible defenses by the extractor layer.

Utility of human rationales. We explore the possibility that human-provided explanations may make the model more robust against adversarial attacks. We mostly find that they do not, with the notable exception of BERT on MULTiRC, the only case in which the augmentation brings the model back to clean test accuracy. While it does provide an advantage of sparsity over supervision with non-attack tokens, this advantage alone may not justify the cost of collecting human explanations for robustness. Further understanding of human rationales and novel learning strategies are required for improving model robustness.

Future directions. A generalization of our approach might convert the “extractor” layer into a more general “defender” layer capable of issuing a wider range of corrections in response to a wider range of attacks. It could, for example, learn to defend against attacks based on input removal (e.g. Feng et al. (2018)) by training to recognize gaps in the input and fill them via generative closure. This defender could be coupled with a self-supervision style approach (e.g., Hendrycks et al. (2019)) involving an “attacker” capable of levying various types of attack against the model. We leave such a generalization for future work.

Limitations

Our work focuses on improving model robustness by explicitly ignoring adversarial attacks. In this

work, we only explore a known type of adversarial attack (ADDSENT), and the performance of our method against unknown attacks is yet to be validated. Since our method uses rationalization as the underlying mechanism for ignoring tokens, it would take non-trivial work to make our method compatible with attacks in the form of token removal and flipping. Finally, we limit our experiments to the domain of QA, where the ADDSENT attack is naturally applicable.

Ethics Statement

Our work contributes to the line of research that focuses on improving the adversarial robustness of language models. We also explore novel ways to integrate human explanations into the training paradigm. We believe robustness to adversarial attacks is essential to the deployment of trustworthy models in the wild, and we hope this work brings current research a step closer to this objective. To avoid ethical concerns related to over-claiming results, we emphasize in both our concluding discussion and the limitations section that our work builds on the assumption that we know the type of attack and only experiments with ADDSENT. Furthermore, our approach tends to increase the computational cost compared to adversarial training both during training and inference. One should consider the tradeoff between robustness and computation.

Acknowledgements

We would like to thank members of the Chicago Human+AI Lab for feedback on early versions of this work, and Howard Chen for fruitful discussions on additive adversarial attacks and sharing findings of his concurrent work. We further thank all anonymous reviewers for their insightful suggestions and comments. This work was supported in part by an NSF grant, IIS-2126602, and gifts from Google and Amazon.

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. *Generating Natural Language Adversarial Examples*. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2890–2896, Brussels, Belgium. Association for Computational Linguistics.

Katharina Beckh, Sebastian Müller, Matthias Jakobs, Vanessa Toborek, Hanxiao Tan, Raphael Fischer, Pascal Welke, Sebastian Houben, and Laura von Ruecken. 2021. *Explainable Machine Learning with Prior Knowledge: An Overview*. *arXiv:2105.10172 [cs]*. ArXiv: 2105.10172.

Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. 2019. *On Evaluating Adversarial Robustness*. *arXiv:1902.06705 [cs, stat]*. ArXiv: 1902.06705.

Howard Chen, Jacqueline He, Karthik Narasimhan, and Danqi Chen. 2022. Can rationalization improve robustness? In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. 2019. *Certified Adversarial Robustness via Randomized Smoothing*. In *Proceedings of the 36th International Conference on Machine Learning*, pages 1310–1320. PMLR. ISSN: 2640-3498.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. *arXiv:1810.04805 [cs]*. ArXiv: 1810.04805.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2019. *ERASER: A Benchmark to Evaluate Rationalized NLP Models*. *arXiv preprint*. ArXiv: 1911.03429.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. *HotFlip: White-Box Adversarial Examples for Text Classification*. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36, Melbourne, Australia. Association for Computational Linguistics.

Gabriel Erion, Joseph D. Janizek, Pascal Sturmels, Scott M. Lundberg, and Su-In Lee. 2021. *Improving performance of deep learning models with axiomatic attribution priors and expected gradients*. *Nature Machine Intelligence*, 3(7):620–631. Bandiera_abtest: a Cg_type: Nature Research Journals Number: 7 Primary_atype: Research Publisher: Nature Publishing Group Subject_term: Computer science;Regulatory networks;Risk factors Subject_term_id: computer-science;regulatory-networks;risk-factors.

William Falcon. 2019. Pytorch lightning. *GitHub Note: <https://github.com/PyTorchLightning/pytorch-lightning>*, 3:6.

Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber. 2018. *Pathologies of Neural Models Make Interpretations Difficult*. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium. Association for Computational Linguistics.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. *Black-Box Generation of Adversarial Text Sequences to Evade Deep Learning Classifiers*. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 50–56.

Siddhant Garg and Goutham Ramakrishnan. 2020. *BAE: BERT-based Adversarial Examples for Text Classification*. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6174–6181, Online. Association for Computational Linguistics.

Xiaochuang Han and Yulia Tsvetkov. 2021. *Influence Tuning: Demoting Spurious Correlations via Instance Attribution and Instance-Driven Updates*. *arXiv:2110.03212 [cs]*. ArXiv: 2110.03212.

Peter Hase and Mohit Bansal. 2021. *When Can Models Learn From Explanations? A Formal Framework for Understanding the Roles of Explanation Data*. *arXiv:2102.02201 [cs]*. ArXiv: 2102.02201.

Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. 2019. *Using Self-Supervised Learning Can Improve Model Robustness and Uncertainty*. *arXiv:1906.12340 [cs, stat]*. ArXiv: 1906.12340.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. *Categorical Reparameterization with Gumbel-Softmax*.

Robin Jia and Percy Liang. 2017. *Adversarial Examples for Evaluating Reading Comprehension Systems*. *arXiv:1707.07328 [cs]*. ArXiv: 1707.07328.

Robin Jia, Aditi Raghunathan, Kerem Göksel, and Percy Liang. 2019. *Certified Robustness to Adversarial Word Substitutions*. *arXiv:1909.00986 [cs]*. ArXiv: 1909.00986.

Di Jin, Zhijing Jin, Joey Tianyi 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(05):8018–8025. Number: 05.

Anurag Katakkan, Weiqin Wang, Clay H. Yoo, Zachary C. Lipton, and Divyansh Kaushik. 2021. *Practical Benefits of Feature Feedback Under Distribution Shift*. *arXiv:2110.07566 [cs]*. ArXiv: 2110.07566.

Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. *Looking Beyond the Surface: A Challenge Set for Reading Comprehension over Multiple Sentences*. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 252–262, New Orleans, Louisiana. Association for Computational Linguistics.

Pang Wei Koh and Percy Liang. 2017. *Understanding black-box predictions via influence functions*. *arXiv preprint arXiv:1703.04730*. 00009.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. **Rationalizing neural predictions**. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas. Association for Computational Linguistics.

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2021. **Contextualized Perturbation for Textual Adversarial Attack**. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5053–5069, Online. Association for Computational Linguistics.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2019. **TextBugger: Generating Adversarial Text Against Real-world Applications**. *Proceedings 2019 Network and Distributed System Security Symposium*. ArXiv: 1812.05271.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2017. **Understanding Neural Networks through Representation Erasure**. *arXiv:1612.08220 [cs]*. ArXiv: 1612.08220.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. **BERT-ATTACK: Adversarial Attack Against BERT Using BERT**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6193–6202, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. **RoBERTa: A Robustly Optimized BERT Pre-training Approach**. *arXiv:1907.11692 [cs]*. ArXiv: 1907.11692.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. **The Stanford CoreNLP Natural Language Processing Toolkit**. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

George A. Miller. 1995. **WordNet: a lexical database for English**. *Communications of the ACM*, 38(11):39–41.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. **TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126, Online. Association for Computational Linguistics.

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. 2016. **Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks**. *arXiv:1511.04508 [cs, stat]*. ArXiv: 1511.04508.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. **GloVe: Global Vectors for Word Representation**. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. **SQuAD: 100,000+ Questions for Machine Comprehension of Text**. *arXiv:1606.05250 [cs]*. ArXiv: 1606.05250.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. **Generating Natural Language Adversarial Examples through Probability Weighted Word Saliency**. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1085–1097, Florence, Italy. Association for Computational Linguistics.

Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. 2017. **Right for the Right Reasons: Training Differentiable Models by Constraining their Explanations**. *arXiv preprint arXiv:1703.03717. 00000*.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. **Intriguing properties of neural networks**. *arXiv:1312.6199 [cs]*. ArXiv: 1312.6199.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. **FEVER: a large-scale dataset for Fact Extraction and VERification**. *arXiv:1803.05355 [cs]*. ArXiv: 1803.05355.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. **Universal Adversarial Triggers for Attacking and Analyzing NLP**.

Sarah Wiegreffe and Ana Marasović. 2021. **Teach Me to Explain: A Review of Datasets for Explainable NLP**. *arXiv:2102.12060 [cs]*. ArXiv: 2102.12060.

Ronald J. Williams. 1992. **Simple statistical gradient-following algorithms for connectionist reinforcement learning**. *Machine learning*, 8(3-4):229–256.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pieric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2020. **HuggingFace’s Transformers: State-of-the-art Natural Language Processing**. *arXiv:1910.03771 [cs]*. ArXiv: 1910.03771.

Jin Yong Yoo and Yanjun Qi. 2021. **Towards Improving Adversarial Training of NLP Models**. *arXiv:2109.00544 [cs]*. ArXiv: 2109.00544.

Omar Zaidan, Jason Eisner, and Christine Piatko. 2007. [Using “Annotator Rationales” to Improve Machine Learning for Text Categorization](#). In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 260–267, Rochester, New York. Association for Computational Linguistics.

Wei Emma Zhang, Quan Z. Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. [Adversarial Attacks on Deep-learning Models in Natural Language Processing: A Survey](#). *ACM Transactions on Intelligent Systems and Technology*, 11(3):1–41.

Yi Zhou, Xiaoqing Zheng, Cho-Jui Hsieh, Kai-wei Chang, and Xuanjing Huang. 2020. [Defense against Adversarial Attacks in NLP via Dirichlet Neighborhood Ensemble](#). *arXiv:2006.11627 [cs]*. ArXiv: 2006.11627.

A Design choices and implementation details

We use the HuggingFace ([Wolf et al., 2020](#)) distributions of BERT and RoBERTa, and Pytorch Lightning ([Falcon, 2019](#)) for model training. Models are trained for a minimum of 3 epochs with early stopping based on a patience of 5 validation intervals, evaluated every 0.2 epochs.

In practice, we find it useful to pretrain the predictor layer f of the rationale model on full input before jointly training it with the extractor g . We observe that this trick stabilizes training and helps prevent mode collapse. In producing the predicted rationale, we automatically assign a 1 (indicating relevance) to every token in the query, so that they are always fully visible to the predictor and the effect of the extractor is in adjudicating which tokens of the document are used or ignored.

Traditionally, this style of rationale model produces binary predicted rationales via either reinforcement learning ([Williams, 1992](#)) or categorical reparameterization such as Gumbel Softmax ([Jang et al., 2016](#)). One argument for this approach is that binary rationales are more interpretable, leaving less ambiguity about the precise role of a given token in the model’s output. Another argument is that transformer-based models like BERT don’t have a native interpretation for partially-masked input, whereas fully-masked input can represent in-distribution modifications such as the [MASK] token substitution used in masked-LM pretraining.

However, we find that relaxing this binary constraint leads to better outcomes for adversarial training. Thus, our model produces predicted rationale

\hat{r} by passing predicted rationale logits ϕ through a sigmoid function. The masking function m we use is simply to multiplicatively weight x by predicted rationale \hat{r} during training (we discretize r during testing),

$$m(\mathbf{x}, \hat{\mathbf{r}}) = \hat{\mathbf{r}} \cdot \mathbf{x} \quad (3)$$

From a theoretical perspective, jointly optimizing the rationale extractor g and label predictor f should allow the model to predict rationale \hat{r} that is more adapted to the predictor. Separately optimizing both components implies that the rationale extractor does not get penalized for poor label prediction performance, and often leads to predicted rationale that is closer to human rationale r . In our experiments, we include both training setups as a hyperparameter.

B Hyperparameters

For our experiments, we fine-tune both the rationale extractor g and predictor f for the rationale models from a pretrained language model. We fine-tune BERT components from a pre-trained *bert-base-uncased* model, and RoBERTa from a pre-trained *roberta-large* model. We use an Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ for all experiments.

We find gradient accumulation helps with training stability of BERT and RoBERTa, and we report gradient accumulation as a hyperparameter for both models. Table 7 describes a list of hyperparameters we use for both BERT and RoBERTa. We do a grid search over all combinations of hyperparameters listed in table 7, and we report results of the model that achieves the highest performance on the original dev set.

C Computation Details

We ran our experiments on a mix of RTX 3090, A30 and A40 GPUs. All experiments combined take less than 300 GPU hours.

The rationale model has about two times the parameters of its base model. The BERT-based rationale model has 220 million parameters and RoBERTa-based rationale model has 708 million parameters. Both models can be trained on a GPU with 24GB of memory. Training the rationale model typically takes double the training time compared to the standard model. On a RTX 3090 GPU, training a BERT Rationale model for SQuAD takes

Parameter	BERT Rationale	RoBERTa Rationale
Batch Size	8	8
Learning Rate	2e-5	5e-6
Gradient Accumulation	10 batches	8 batches
Masking Strategy m	$m_{\text{zero}}, m_{\text{mask}}$	$m_{\text{zero}}, m_{\text{mask}}$
Prediction Supervision Loss Weight	1.0	1.0
Rationale Supervision Loss Weight λ_1	1.0	1.0
Sparsity Loss Weight λ_2	0.0, 0.1, 0.2, 0.3	0.0, 0.1, 0.2, 0.3
Jointly Optimized	True, False	True, False

Table 7: Hyperparameters used in parameter search and training.

about 4 hours, while training the standard BERT model takes 2.5 hours.

D Statistical Significance

Due to limited computational resources and a large number of experiment conditions, our experiments are not repeated across multiple random seeds. To verify the statistical significance of improvements of the top-performing rationale models against the strongest baselines, we report Wilcoxon signed-rank test results in Table 8. Note that we do not report SQuAD results due to the incompatibility of the statistical test with the metric (Span-F1). We find that for 3 out of 4 model-dataset pairs, the observed improvements of the rationale models over the baselines are highly significant.

E More Error Analysis

Easy examples have high jaccard similarity between human rationale and QUERY+ANSWER. All three models excel at these examples. High similarity should help models to find human rationale or generate rationales that mimic human rationale easily, but we also observe that the generated rationales do not necessarily provide the greatest alignment with human rationale for examples BERT rationale models get correct. For instance, rationale F1 is 53.9 for examples that human-supervised BERT gets correct and BERT gets wrong, which is smaller than rationale F1 (56.2) for examples both models get wrong. Notice that attack and human rationale are similar due to the attack generation technique, but this does not affect model performance because training with augmentation allows the rationale models to ignore attack tokens (attack recall = 89.3 and 97.4 for BERT rationale models). Likewise, we think BERT (ADV.) also benefits from the high similarity to identify important text

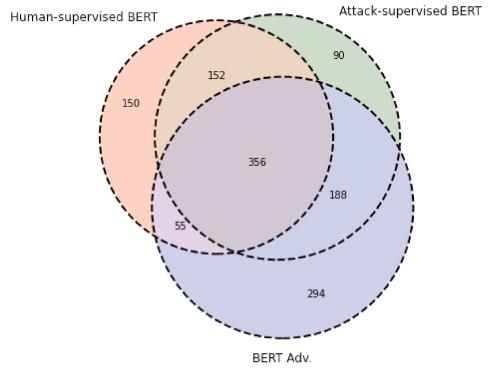


Figure 3: Venn diagram for errors by BERT (ADV.), human-supervised BERT, and attack-supervised BERT.

areas and learns to ignore attacks from training augmentation.

BERT rationale models handle denser human rationale slightly better than BERT (ADV.). We define sparsity of X as the number of tokens in X divided by the total number of tokens in the input, so larger sparsity correspond to dense rationales. Counter-intuitively, all three models are bad at examples with the most dense human rationale. This can be accounted for by the fact that these are also examples where QUERY+ANSWER and human rationale have the least jaccard similarity: human rationale sparsity and the jaccard similarity has a Pearson’s coefficient of 0.25 ($p < 0.001$). Thus, examples with denser human rationale are likely to contain confusing information for models. We find BERT rationale models can resist this confusion better than BERT (ADV.). For instance, human rationale sparsity = 0.167 when human-supervised BERT is correct but BERT is wrong, and it is 0.165 when BERT is correct but BERT rationale is wrong.

Model	Dataset	Baseline Variant (Acc)	Rationale Variant (Acc)	p-value
BERT	MULTIRC	ADV.-10X (65.9%)	ADV. + HUMAN SUP. (69.4%)	1.0×10^{-4}
BERT	FEVER	ADV. (84.8%)	ADV. + ATK. SUP. (87.7%)	8.1×10^{-13}
RoBERTa	MULTIRC	ADV. (84.8%)	ADV. + ATK. SUP. (85.1%)	9.4×10^{-4}
RoBERTa	FEVER	ADV.-10X (93.4%)	ADV. + ATK. SUP. (93.2%)	0.18

Table 8: Wilcoxon signed-rank test for statistical significance of improvements of the top-performing rationale models over the strongest baselines across models and datasets.

	Input Length	Human Rationale Length
human-supervised BERT correct, attack-supervised BERT wrong	357.097	360.278
attack-supervised BERT correct, human-supervised BERT wrong	81.191	79.098

Table 9: Input and human rationale length of mistakes by attack-supervised BERT and human-supervised BERT.

F More Results on SQuAD

In Table 10, we report the F1 scores of BERT Classification and BERT Rationale models on four different evaluation sets: the original SQuAD development set dev, the synthetic attack set ADDSENTS, the human re-written and filtered attack set ADDSENTH and the human-generated, model-free attack baseline ADDONESENTH.

Similar to §5.2, we find the performances on the clean set (SQuAD dev) to be approximately equal across models and training schemes. We observe a slight drop (-0.4%) on dev accuracy when adding adversarial training to the BERT classification, which points to compromised learning on the original SQuAD task after adding adversarial examples. Without adversarial training, we observe roughly 38%, 24%, 15% performance decreases for ADDSENTS, ADDSENTH, ADDONESENTH, respectively. All three attacks lead to much more significant performance drops than the ADDSENT attack on MULTIRC and FEVER, which yields an approximate 6% performance drop. This observation is likely due to the differences in how the tasks are formulated across datasets: it is plausible that an additive attack such as ADDSENT is more effective on a span-extraction style QA task (SQuAD) than on answer classification style QA tasks (MULTIRC and FEVER).

Surprisingly, the synthetic attack ADDSENTS is more effective than human generated ADDSENTH prior to adversarial training. Since the ADDSENT attack works by mutating the query and adding a

fake answer, the synthetic attack often appears syntactically similar to the query. On the other hand, human generated attacks in ADDSENTH often fits more naturally in the document and grammatically correct, but does not mirror the structure of the query. For a model that solves the QA task by simply looking for the best match of the query inside a document while skipping complex reasoning, it's conceivable that ADDSENTS leads to the greatest performance drop.

Since ADDSENTH and ADDONESENTH are attack examples re-written and filtered by humans, we use them as a proxy for understanding the model behavior in a real-world setting. We find the BERT Rationale model with attack rationale supervision significantly outperforms the BERT Classification baseline trained with adversarial augmentation (+2.7% on ADDSENTH, +2.2% on ADDONESENTH). Similar to findings in §5.2, we observe attack rationale supervision (ADV. + ATK. SUP.) as a more effective adversarial training method than adversarial data augmentation (ADV.). It is worth noting that despite training on the synthetic attacks, the rationale model demonstrates strong ability to generalize knowledge learned from synthetic attacks to tune out human-rewritten attacks, which explains the strong performance on ADDSENTH and ADDONESENTH.

An apparent anomaly in Table 10 is the strong performance of BERT Classification on ADDSENTS (93.3%), which is even greater than the performance on the clean development set

Architecture	Training	dev	ADDSENTS	ADDSENTH	ADDONESENTH
Bert Classification	NO ADV.	86.4	48.7	62.8	71.2
	ADV.	86.0	93.3	80.4	81.9
	ADV.-10x	82.2	95.9	78.0	79.7
Bert Rationale	NO ADV.	86.6	47.7	62.0	70.4
	ADV. + ATK. SUP.	86.5	88.3	83.1	84.1

Table 10: F1 scores of BERT Rationale and Classification models on the SQuAD task.

(86.0%). During the ADDSENT attack, the answer is mutated into an incorrect, but similar phrase (e.g. Dallas Cowboys → Michigan Vikings). The presence of a mutated answer in the passage likely gives the model additional information on what the correct answer looks like, while the rationale model avoids utilizing this information to a much higher degree (88.3%) due to attack rationale supervision. This "mutated answer" signal is akin to spurious correlations in datasets, and our method helps the BERT rationale model ignore such spurious correlations a lot more effectively than the baseline BERT model.

Overall, these analyses shine light on the benefits of including rationale supervision in adversarial training. Our method achieves greater adversarial robustness in a close-to-real-world setting (ADDSENTH and ADDONESENTH) by generalizing from synthetic attacks.