Learning to Ignore Adversarial Attacks

Anonymous ACL submission

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 ignore over 90% of attack tokens. This approach leads to consistent sizable improvements (∼8%) over baseline models in robustness, for both BERT and RoBERTa, on MULTIRC and FEVER, 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, eliminating the effect of adversarial attacks.

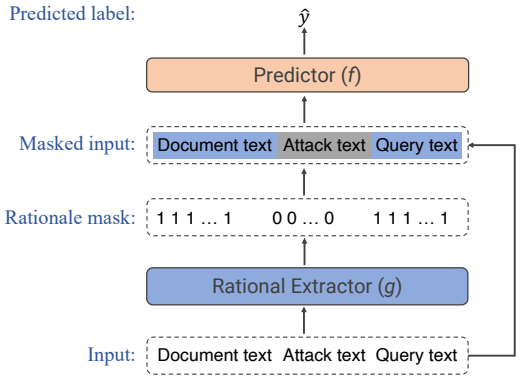

Figure 1: In addition to the typical predictor, an ideal rationale model identifies the "relevant" tokens in the input with a rationale extractor and only present relevant tokens to the predictor after filtering the attack text. The main goal of this work is to explore whether such rationale models can be effectively used to ignore 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 output. As such models become more powerful and ubiquitous, research continues to discover surprising vulnerabilities (e.g., Wallace et al. (2019)), demanding improved robustness methods.

Given a specific attack method, a straightforward way to improve model robustness is to incorporate adversarial examples, attacked using that method, during training in addition to clean examples (Zhang et al., 2020). The goal in doing this is that the model will implicitly learn to ignore attacking tokens and become more robust to that attack type. However, in practice this can be a challenging learning objective.

In this study we explore an alternative method of leveraging such data augmentation: 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 extractor to ignore attacking tokens as an additional joint objective to overall label accuracy (Fig. 1).

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

Training BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) as underlying models on the MultiRC (Khashabi et al., 2018) and FEVER (Thorne et al., 2018) datasets, we demonstrate that the additive attack of Jia and Liang (2017) do reduce model accuracy, and that data augmentation with adversarial examples provides benefit in defending these models from this attack in most cases.

Our main results are that rationale-style models are better able to learn to ignore these attacks than only with data augmentation, leading to an improvement of ∼8% in accuracy on attacked examples compared to baseline models and an advantage of 2.6% over data augmentation alone,

mostly returning to 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 at defending from an otherwise-effective attack 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 primary results.

2 Related work

Our work builds 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. Researchers have explored 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), language model-based slot filling (Li et al., 2020; Garg and Ramakrishnan, 2020; Li et al., 2021). Another distinction is based on 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). Morris et al. (2020) presents the TextAttack framework and provides a collection of attack implementations. Our work focuses on the ADDSENT attack proposed by Jia and Liang (2017) in the context of 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 signal to improve model performance and robustness, sometimes also known as *feature-level feedback* (Hase and Bansal, 2021; Beckh et al., 2021).

An early work in this vein 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. Katakkar et al. (2021) evaluates feature feedback for two attention-style models, finding, again, gains in out-of-domain performance, while Han and Tsvetkov (2021) uses 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.

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 following:

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. If the query is the concatenation of a question and an answer, we convert the mutated query

Query c:
FC Bayern Munich was founded in 2000.
Mutated Query $\hat{c}$:
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.

$\hat{q}$ 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. An example of the ADDSENT attack is given below.

The original approach includes an additional step of using crowdsourced workers to filter ungrammatical attacks . For the sake of simplicity, we skip this manual validation process. Qualitatively, we observe that the attack do result in a significant number of ungrammatical attacks, but the attacks prove 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 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* for all examples, indicating which tokens in the document 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

Table 1: Basic statistics of the datasets.

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). For the sake of training efficiency, and 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.

4 Modeling

Our study assesses whether adding an explicit rationale extractor to a fine-tuned model and training it to ignore adversarial tokens results in a more effective defense than simply adding attacked examples to the training set. This comparison results in several combinations of model architecture and training regime.

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

Baseline models and training. We use BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) as basic 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).

Rationale model. To lend the baseline model an extractor capable of filtering out confounding to-

[1] We 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.

kens, 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 $\hat{\mathbf{r}}$, which is applied as mask over the input to the predictor via masking function m, producing a predicted label:

$$g(\mathbf{x}) \rightarrow \hat{\boldsymbol{r}}$$
$$f(m(\mathbf{x}, \hat{\boldsymbol{r}})) \rightarrow \hat{y} \quad (1)$$

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 $\hat{\boldsymbol{r}}$'s error with respect to a ground-truth rationale $\boldsymbol{r}$. 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:

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

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, identical to the scenario of data augmentation with adversarial examples. 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 $\boldsymbol{r}$: 1) a binary indicator of whether tokens are adversarial or not (ADV. + ATK. SUP.); and 2) the human-annotated rationale (ADV. + HUMAN SUP.), which also filters adversarial tokens. Table 2 shows all the combination of setups that 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 independent RoBERTa modules for both.

Taken together, these conditions address our three research questions: 1) does training the model to emulate human explanation make it intrinsically more robust to attack?; 2) is adversarial training via rationale supervision more effective than via attacked examples?; and 3) do human explanations improve upon this latter effect?

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

Table 2: Summaries of all model setups in this work.

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. Hyperparameter and computation details are in the appendix.

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

$$m(\boldsymbol{x}, \hat{\boldsymbol{r}}) = \hat{\boldsymbol{r}} \cdot \boldsymbol{x}$$

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.

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 to understand model behavior.

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 two variants of the standard classification model (NO ADV., ADV.), and three variants of the rationale model (Human, ADV. + ATK. SUP., ADV. + HUMAN SUP.).

Exploring these 5 architecture/training combinations for two datasets (MULTIRC and FEVER) and two underlying models (BERT and RoBERTa), we report results from 20 trained models in Table 3. We report both clean test set accuracy and attacked test set accuracy for each model. The attacked test set accuracy is our key measure of robustness.

For just the rationale model results, we report the mean percentage of attack and non-attack tokens included in each predicted rationale, two metrics which helps 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 Experimental Results

We focus our analysis on three basic questions:

1. Does human rationale supervision improve adversarial robustness over a standard model?
2. Does adversarial rationale supervision on augmented data improve robustness over adversarial data augmentation alone?
3. Does the addition of human rationale supervision to this adversarial supervision further improve robustness?

Table 3 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 by roughly 6% on MULTIRC for BERT and RoBERTa, and roughly 10% on FEVER as well.

Effect of human rationale supervision alone (HUMAN SUP.). We find mixed evidence for whether human rationale supervision alone improves 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 4 explains this negative 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.

Intuitively this is an unsurprising result. Human rationales for these datasets identify the part of the document that pertains particularly to the query, while the ADDSENT attack functions by crafting adversarial content with a semantic resemblance to that same query. Hence, it is understandable that human rationale training alone would not improve adversarial robustness.

Adversarial rationale supervision (ADV. + ATK. SUP.). Although human rationale supervision does not seem to improve adversarial robustness, the rationale mechanism offers an interface for explicitly supervising the model to ignore attack tokens. We investigate the question of whether this mechanism can be used to improve the effectiveness of data augmentation.

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

Adversarial rationale supervision improves on adversarial data augmentation in all cases. We

Model	Architecture	Training	MULTIRC		FEVER	
			Clean	Attacked	Clean	Attacked
BERT	Standard	NO ADV.	68.6	62.6	88.2	78.9
		ADV.	67.3	61.6	**88.5**	84.8
	Rationale	HUMAN SUP.	70.0	64.4	88.0	76.7
		ADV. + ATK. SUP.	69.6	66.2	87.1	**87.7**
		ADV. + HUMAN SUP.	**70.5**	**69.4**	87.5	87.5
RoBERTa	Standard	NO ADV.	82.6	76.5	93.5	83.0
		ADV.	84.4	82.9	93.2	92.7
	Rationale	HUMAN SUP.	84.0	74.9	**94.1**	85.7
		ADV. + ATK. SUP.	**85.2**	**85.1**	93.4	**93.4**
		ADV. + HUMAN SUP.	85.0	82.5	93.4	**93.4**

Table 3: Model accuracy on clean and attacked test sets for MULTIRC and FEVER.

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

Table 4: Percentage of attack and nonattack tokens included in rationale model predicted rationales. Lower is better for attack tokens, and arguably better for nonattack tokens (all else being equal) as it improves interpretability.

see an improvement of +4.6% for BERT on MULTIRC, +2.9% for BERT on FEVER, +2.2% for RoBERTa on MULTIRC, and +0.7% for RoBERTa on FEVER (2.6% on average). For the one case where adversarial data augmentation recovered clean test performance (RoBERTa on MULTIRC), adversarial rationale supervision actually improves on 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% and 8.8% for BERT on MULTIRC and FEVER, 8.6% and 10.4% for RoBERTa on MULTIRC and FEVER (7.9% on average).

Table 4 explains this success. The adversarially-supervised rationale model includes 6% or fewer attacking tokens in all settings, indicating that it did largely succeed in learning to occlude these tokens w.r.t the predictor. This is an exciting result because it shows that explicitly training the model to ignore adversarial tokens is an effective defense against this particular attack. Moreover, the supervision of non-attack tokens does not require any additional human effort.

Human and adversarial rationale supervision (ADV. + HUMAN SUP.). The previous result shows that the rationale model can learn to ignore adversarial tokens added to the input. A final question is whether human rationales can serve as a useful addition to this mechanism. 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 human + adversarial rationale supervision performance is equal to that of the adversarial rationale supervision alone. Only for BERT on MULTIRC does it result in an improvement, being the only configuration for that model and dataset that brings performance back to that of clean test. For RoBERTa on MULTIRC, it actually weakens attacked test performance.

While these results are mixed, Table 4 does show that the model does at least achieve this result at a much lower included percentage of nonattack tokens ($\sim$20% vs. $>$95%), a concession toward model interpretability.

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

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 - tete did n ' t stay in basel after charles and houben separated . ... [SEP] who did n ' t stay in zurich after albert and maric separated ? | | tete [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 - tete did n ' t stay in basel after charles and houben separated . ... [SEP] who did n ' t stay in zurich after albert and maric separated ? | | tete [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 - tete did n ' t stay in basel after charles and houben separated . ... [SEP] who did n ' t stay in zurich after albert and maric separated ? | | tete [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 picts [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 picts [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 picts [SEP] **ADV. + HUMAN SUP. prediction: False**

Table 5: 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 orange in the first column. We only show tokens where generated rationales disagree with each other or with human rationale and attack.

start with a qualitative analysis of example errors, and then discuss general trends, especially on why human rationales provide limited benefit 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 5 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 attack more effectively than ADV. In the attack sentence, "tete did n ' 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 attack and include the necessary information. In fact, ADV. + ATK. SUP. generates perfect non-attack rationales in all three examples. Generally, it is more challenging to generate human rationales than to generate non-attack rationales, likely because the attack sentences are derived from a heuristical algorithm.

Finally, Example 3 shows an example where ADV. + HUMAN SUP. is better than ADV. + ATK. SUP. when generating rationales to ignore noise. ADV. + HUMAN SUP. includes attack 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 avoiding attack completely.

More generally, we find that ADV. + HUMAN SUP. *tends to have high false negatives*. When only ADV. + HUMAN SUP. is wrong, 92% of the errors are in the positive class. Indeed, when ADV. + HUMAN SUP. fails to find good rationales, it tends to predict False because of the high sparsity. In contrast, ADV. + ATK. SUP. does not have the tendency to predict False, as its generated rationales contain the necessary information most of the time.

ADV. + ATK. SUP. *is better than* ADV. + HUMAN SUP. *when human rationale is denser and passage length is longer (see Table 7 in the appendix).* We observe that denser human rationale usually comprises evidence from different parts of the passage. Since ADV. + ATK. SUP. generates rationales with almost every non-attack token, they will have higher human rationale recall (98.6%) than human-supervised BERT (57.6%). Thus, ADV. + ATK. SUP. will generate better rationales when human rationale is dense, but this can be difficult for ADV. + HUMAN SUP. to pick up. Similarly, long passage length also makes it harder for ADV. + HUMAN SUP. to select which non-human rationale tokens to drop when generating rationales.

Taken together, these analyses highlight the challenges of learning from human rationales: it requires precise occlusion of irrelevant information while keeping valuable information, and account for the diverse ranges of human rationales and input lengths. This partly explains the limited benefit of ADV. + HUMAN SUP. over ADV. + ATK. SUP.

6 Concluding Discussion

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

This is an exciting result because it demonstrates a novel use for this type of explicit token relevance representation, which is more typically applied for the sake of model interpretability (Lei et al., 2016). It is 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) notes, 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 only against a baseline defense in adversarial training via data augmentation.

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 in this study that feature feedback based on human-provided explanations may make the model more robust against adversarial attack. We mostly find that they do not, with the notable exception of BERT on MULTIRC, where it is only this augmentation that 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.

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.

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 Rueden. 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.

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 Sturmfels, 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.

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 Katakkar, 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 Pretraining 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.

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, Pierric 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 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 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 6 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 6, and we report results of the model that achieves the highest performance on the original dev set.

B 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 areas and learns to ignore attacks from training augmentation.

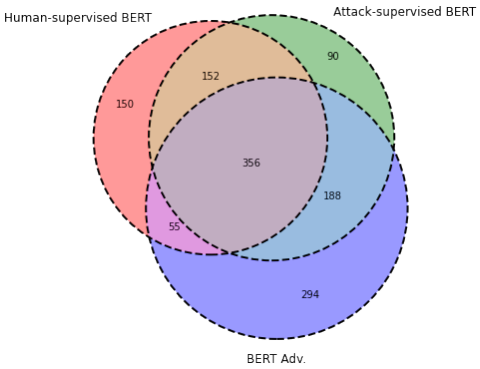

Figure 3: Venn diagram for errors by BERT (ADV.), human-supervised BERT, and attack-supervised BERT.

BERT rationale models handle denser human rationale slightly better than BERT (ADV.). We define sparsity of $\mathbf{X}$ as the number of tokens in $\mathbf{X}$ 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 bu BERT is wrong, and it is 0.165 when BERT is correct but BERT rationale is wrong.

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_{zero}, m_{mask}	m_{zero}, m_{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 6: Hyperparameters used in parameter search and training.

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