Unleashing LM's Mind: Gradient-Based Obstinate Adversarial Examples Generation in Language Models

Anonymous ACL submission

Abstract

In this paper, we study the problem of automatically generating word-level obstinate (over-stability) adversarial examples, where input text is meaningfully changed but the model's prediction does not, even though it should. Previous word substitution approaches have predominantly focused on manually designed antonym-based strategies, which hinders their application as these strategies can only find a subset of obstinate adversarial examples and require human efforts. To address this issue, in this paper, we introduce a novel word substitution method named GRADOBSTINATE, a gradient-based approach that automatically generates obstinate adversarial examples without any constraints on the search space or the need for manual design principles. To empirically evaluate the efficacy of GRADOBSTINATE, we conduct comprehensive experiments on five representative models (Electra, ALBERT, Roberta, DistillBERT, and CLIP) finetuned on four NLP benchmarks (SST-2, MRPC, SNLI, and SQuAD) and a language-grounding benchmark (MSCOCO). Extensive experiments show that our proposed GRADOBSTINATE generates powerful obstinate adversarial examples, exhibiting a higher attack success rate compared to antonym-based methods. Furthermore, to show the transferability of obstinate adversarial examples found by GRADOBSTINATE, we replace the words in four representative NLP benchmarks with their obstinate adversarial examples. Notably, it exhibits a high success rate when transferred to other models in black-box settings, including even GPT-3 and ChatGPT. Examples of obstinate adversarial examples found by GRADOBSTINATE are available at https://huggingface.co/spaces/anonauthors/SecretLanguage.

1 Introduction

In the past few years, deep learning (He et al., 2016; Devlin et al., 2019) has achieved huge success, especially in the natural language processing (NLP) area. However, powerful deep learning methods have been shown to be easily fooled by adversarial examples, which are data crafted intentionally to confuse a model (Szegedy et al., 2014). Literature has been focused on generating and defending (fickleness) adversarial examples (see Figure 1) (Jia and Liang, 2017; Alzantot et al., 2018; McCoy et al., 2019; Jin et al., 2020; Li et al., 2020; Wang et al., 2020; Alshemali and Kalita, 2020;

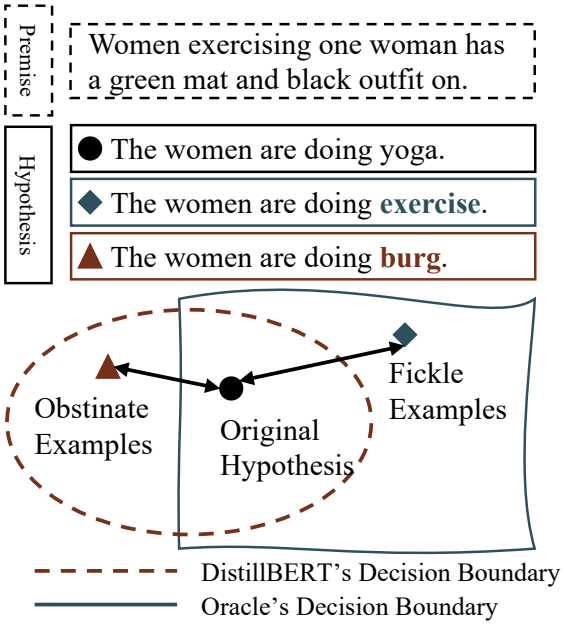


Figure 1: An example from SNLI of obstinate (over-stability) and fickle (over-sensitivity) adversarial examples. We explore the generation of fickle and obstinate adversarial examples for the hypothesis "The women are doing yoga." while keeping the premise the same. The corresponding fickle adversarial example is "The women are doing exercise.", which changes the prediction of models. On the other side, the obstinate adversarial example, "The women are doing burg.", alters the meaning of the sentence and the relationship with the premise while making the models (DistillBERT) output the same. In this paper, we focus on efficiently and effectively generating obstinate adversarial examples.

Zhou et al., 2021) that are similar to the normal inputs with minimal perturbations and able to change the model's output. Such (fickleness) adversarial examples emerge when models are too sensitive around the normal data such that small changes will flip the outputs of the models. For example, changing "film" to "movie" might lead to a change in the output of language models (LMs) (Ribeiro et al., 2018). As a remedy, synonyms and semantically similar sets can be enumerated and models can directly learn based on those sets to defend against fickleness adversarial attacks.

On the other side, the complementary problem of generating and defending against obstinate adversarial examples (see Figure 1) is equally troublesome but remains unexplored. Obstinate adversarial examples (Niu and Bansal, 2018; Jacobsen et al., 2019; Tramer et al., 2020; Chen et al., 2022) have different semantics compared to the original input texts while the models retain their predictions. Particularly, previous works can be roughly split into three categories, *i.e.*, removing a portion of input texts (Feng et al., 2018; Welbl et al., 2020a), reordering of the texts (Welbl et al., 2020b), and substituting input words with other words (Niu and Bansal, 2018; Welbl et al., 2020b; Chen et al., 2022). In this paper, we focus on the last line of literature, generating obstinate adversarial examples by word substitutions.

To explore the vulnerability of LMs to obstinate adversarial examples generated by words substitutions, Niu and Bansal (2018) first showed that adding negation or changing words with their antonyms to the original sentences will not change the outputs of several dialogue models (Li et al., 2016; Serban et al., 2017). Subsequently, Tramer et al. (2020); Chen et al. (2022) showed that standard (fickle) adversarial training methods focused on reducing vulnerability to fickle adversarial examples may make models more vulnerable to obstinate adversarial examples. However, existing methods (Niu and Bansal, 2018; Welbl et al., 2020b; Chen et al., 2022) in NLP only consider manual or antonym-based strategies for generating adversarial examples adhering to manually established principles, which limits the searching space and further hinders the defense to such attacks.

In this paper, to address the aforementioned challenges and enable efficient obstinate adversarial examples generation, we propose the first gradient-based method for generating obstinate adversarial examples, namely GRADOBSTINATE (Algorithm 1). It automatically generates obstinate adversarial examples from the whole dictionary instead of only searching in negation and antonym sets and not requiring any human effort as opposed to Niu and Bansal (2018); Chen et al. (2022). To avoid confusion, we refer to "obstinate substitutions" as word replacements between original sentences and their obstinate adversarial examples. For example, given the original hypothesis, "The women are doing yoga" and its obstinate adversarial example, "The women are doing *burg*", the term "*burg*" represents an obstinate substitution for "yoga", as illustrated in Figure 1.

To empirically evaluate our methods, we select five representative benchmarks, *i.e.*, SST-2, MRPC, SNLI, SQuAD, and MSCOCO. Results show that our proposed GRADOBSTINATE outperforms previous methods with the SOTA performance with regard to the attack success rate, proving the power of gradient-based methods. Then, to further understand obstinate adversarial robustness, we test the transferability of obstinate substitutions on five white-box models and two black-box models, *i.e.*, DistillBERT, ALBERT, Roberta, Electra, GPT-2, GPT-3, and ChatGPT. Notably, obstinate substitutions can be easily transferred yielding consistently high success rates, which further proves the efficiency of GRADOBSTINATE.

Our contributions are summarized as follows.

- We propose the first gradient-based method, namely GRADOBSTINATE (see Algorithm 1), for word substitution-based obstinate adversarial generation, enabling *automdatical* generation of obstinate adversarial examples as opposed to previous methods (Niu and Bansal, 2018; Welbl et al., 2020b; Chen et al., 2022).
- We conduct comprehensive experiments on five representative NLP tasks with five models. The results show that GRADOBSTINATE outperforms previous methods by a large margin, proving the superiority of our method.
- To test the transferability of obstinate substitutions, we attack seven models including GPT-3 and ChatGPT with obstinate substitutions found by GRADOBSTINATE. Results show that obstinate substitutions can be easily transferred to other models with high attack success rates, showing the vulnerability of LMs.
- Examples of obstinate adversarial examples

and substitutions found by GRADOBSTINATE are available on https://huggingface.co/spaces/anonauthors/SecretLanguage. We will release our code upon publication.

2 Related works

In this section, we briefly review the literature on generating fickle and obstinate adversarial examples in NLP.

Generating fickle adversarial examples in NLP. (Fickle) Adversarial robustness (Ren et al., 2019; Zhang et al., 2020; Wang et al., 2022) have been studied extensively in NLP, which aims to improve the stability of LMs towards semantically similar inputs. That includes the ability to identify and use synonyms (Li et al., 2020; Morris et al., 2020; Garg and Ramakrishnan, 2020) and related words. Previous work generates fickle adversarial examples with the gradients of the victim models in character-level (Ebrahimi et al., 2018a,b) or word-level (Cheng et al., 2020).

Generating obstinate adversarial examples in NLP. As the complementary problem of fickle adversarial robustness, obstinate adversarial robustness has been ignored for a long time until Niu and Bansal (2018) highlighted its significance by revealing the susceptibility of dialogue models to obstinate adversarial examples. At the same time, similar ideas have been explored by Jacobsen et al. (2019), which showed that computer vision models are stable to semantic-changing perturbations and thus make vast regions in input space vulnerable to (obstinacy) adversarial attacks. To address this issue, Niu and Bansal (2018); Welbl et al. (2020b); Chen et al. (2022) proposed antonym-based strategies to generate fickle adversarial examples and then conduct adversarial training to improve the robustness of LMs. In parallel, multimodal models have grappled with fickle adversarial examples as well. In 2022, Daras and Dimakis (2022) showed that `Apoploe vesrreaitais` means birds and `Contarra ccetnxniams luryca tanniounons` (sometimes) means bugs or pests to DALLE-2 (Ramesh et al., 2022) with hand-crafted examples.

Different from methods (Niu and Bansal, 2018; Welbl et al., 2020b; Chen et al., 2022; Daras and Dimakis, 2022) in NLP and multimodal area, which generates obstinate adversarial examples by antonym-based strategies or manually set principles, GRADOBSTINATE leverages gradient methods to automatically generate obstinate adversarial examples without any human efforts and with an infinite search space. Benefiting from the gradient-based design, GRADOBSTINATE shows superior performance in attacking 5 different language-based models compared with previous methods.

3 Our approach

In this section, we introduce a principled approach, namely GRADOBSTINATE, to find obstinate adversarial examples given language models (LMs) and input sentences. Specifically, we first introduce the definition (Section 3.1) and the proposed methods (Section 3.2), then elaborate on how to choose the target word (Section 3.3), and finally summarize our threat model (Section 3.4).

3.1 Problem overview

Initially, we posit the existence of a human oracle $\mathcal{O}(\cdot)$, which provides the semantic interpretation of a given input. A sentence is represented as $s = [word_1, \ldots, word_i, \ldots, word_n]$, where $word_i$ is the i-th word and the length of sentence is n. The substitution of the i-th word in the sentence s is denoted as $s_i' = [word_1, \ldots, word_i', \ldots, word_n]$, where $word_i'$ is the new word.

Now, following Niu and Bansal (2018); Tramer et al. (2020); Chen et al. (2022), we present a formal definition of obstinate adversarial examples.

Definition 1. *Given an LM $f(\cdot)$, obstinate adversarial examples s' should be semantically different from and have the same outputs with the original input sentences s, such that,*

1. ***Different Semantics:*** $\mathcal{O}(s') \neq \mathcal{O}(s)$.
2. ***Same Outputs:*** $f(s') = f(s)$[1].

Additionally, we introduce the obstinate substitution, referring to the replacement of words in the original sentence, e.g.*, "burg" serving as an obstinate substitution for "yoga".*

3.2 GRADOBSTINATE for generating obstinate adversarial examples

Previous methods (Niu and Bansal, 2018; Tramer et al., 2020; Chen et al., 2022; Daras and Dimakis, 2022) found obstinate adversarial examples, *e.g.*, "*Apoploe vesrreaitais*" as"birds" and "*Contarra*

[1]Here we omit the use of tokenizer $t(\cdot)$ and word embedding $W \in \mathcal{R}^{n_{words} \times d}$ for simplicity, where d is the dimension. The full procedure is $f(t(s)W)$ given an input s.

Algorithm 1 GRADOBSTINATE (s, f, t, l, η, e)

Input: Input sentence $s = [word_1, \ldots, word_n]$, model $f(\cdot)$, tokenizer $t(\cdot)$, word embedding W, target word's index l, step size η, step E.
Output: Obstinate adversarial example s'.

Get the one-hot representation $\mathbf{r}_{onehot} \leftarrow [\mathbf{r}_{onehot,1}, \ldots, \mathbf{r}_{onehot,n}] \in \{0,1\}^{n \times n_{words}}$ with the tokenizer $t(\cdot)$;
Initialize noise as $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{1}) \in \mathcal{R}^{1 \times n_{words}}$;
for $e \in [E]$ **do**
  Clip the noise, $\mathbf{z} \leftarrow \max(\min(\mathbf{z}, -\mathbf{1}), \mathbf{1})$;
  Get the perturbed representation $\mathbf{r}'_{onehot} \leftarrow [\mathbf{r}_{onehot,1}, \ldots, \mathbf{r}_{onehot,l} + z, \ldots, \mathbf{r}_{onehot,n}]$
  Calculate the loss, $loss \leftarrow \mathrm{CE}(f(\mathbf{r}_{onehot}W), f(\mathbf{r}'_{onehot}W))$;
  Update noise as $\mathbf{z} \leftarrow \mathbf{z} - \eta\, \mathrm{sign}(\frac{\partial loss}{\partial z})$;
  if $f((\mathbf{r}_{onehot} + \mathbf{z})W) == y$ **then**
    Break;
  end if
end for
Get the index of perturbed word as $wordindex \leftarrow \mathrm{argmax}([\mathbf{r}_{onehot,1}, \ldots, \mathbf{r}_{onehot,l} + z, \ldots, \mathbf{r}_{onehot,n}])$.
Get the sentences s' with the decoding function of the tokenizer;
return Filtered obstinate adversarial example $\mathrm{filtering}(s, s')$.

Algorithm 2 **filtering**(s, s')

Construct a synonym set $\mathcal{S}$ for s;
if $s' \notin \mathcal{S}$ **then**
  if GPT-2 thinks s, s' are not semantically similar **then**
    return s'.
  end if
end if
return $\varnothing$.

ccetnxniams luryca tanniounons" as "bugs" or "pests", with manually-crafted sentences, which requires prior knowledge of LMs and human efforts. To efficiently and automatically generate obstinate adversarial examples, we propose GRADOBSTINATE, a gradient-based algorithm detailed in Algorithm 1. It is able to return obstinate adversarial examples without any human efforts and achieves faster inference benefiting from the gradient design.

Specifically, given a sentence $\mathbf{s}$, we first map words to one-hot vectors by the tokenizer as $\mathbf{r}_{onehot} = [\mathbf{r}_{onehot,1}, \ldots, \mathbf{r}_{onehot,n}] \in \{0,1\}^{n \times n_{words}}$, where n_{words} is the size of word dictionary of the tokenizer. Next, we initialize a noise $\mathbf{z}$ sampled from the Gaussian distribution with mean $\mathbf{0}$ and variance $\mathbf{1}$. Then, we add the noise $\mathbf{z}$ to the l-th word as $\mathbf{r}'_{onehot} = [\mathbf{r}_{onehot,1}, \ldots, \mathbf{r}_{onehot,l} + z, \ldots, \mathbf{r}_{onehot,n}]$. To make sure that the perturbed one-hot representation is valid in the vocabulary space, we clip the noise into $[-1, 1]$ as $\mathbf{z} = \max(\min(\mathbf{z}, -\mathbf{1}), \mathbf{1})$. To ensure it has the same output as the original inputs, we employ gradient descent to update the noise vector $\mathbf{z}$. We measure the distance between the outputs of the modified and the original sentences using the cross-entropy loss as $loss = \mathrm{CE}(f(\mathbf{r}_{onehot}W), f(\mathbf{r}'_{onehot}W))$ and then update the noise with the sign of gradient $\frac{\partial loss}{\partial \mathbf{noise}}$. After E predefined epochs or when the outputs of perturbed sentences and original inputs are the same, we stop the updates and obtain perturbed words by applying the decoder of tokenizer to $\mathrm{argmax}(r'_{onehot})$.

However, while GRADOBSTINATE is successful at the efficient generation of obstinate adversarial examples, the generated obstinate adversarial examples might have similar semantics with the inputs. Then, to address this issue, we leverage a synonym-based strategy and GPT-2 (Radford et al., 2019) as a "human oracle" for semantic filtration as summarized in Algorithm 2. More specifically, we first construct a synonym set for each word using WordNet (Miller, 1995) and subsequently verify whether the sentences fall within the synonym set. Additionally, to further assess their semantic similarity, we utilize GPT-2 with the prompt "Are the following sentences semantically similar? $\{\mathbf{s}\}, \{\mathbf{s}'\}$". If $\mathbf{s}'$ is not present within the synonym set and GPT-2 responds negatively, we treat $\mathbf{s}'$ as an obstinate adversarial example of $\mathbf{s}$.

It is worth noting that, in comparison to previous methods (Niu and Bansal, 2018; Tramer et al., 2020; Cheng et al., 2020; Chen et al., 2022; Daras and Dimakis, 2022), to ensure accurate generation of obstinate adversarial examples, we make the following special designs. **Expanding the search space.** The real-valued initialization is designed to ensure that the gradient can flow back to all words. In this way, the search space contains all subwords in the vocabulary. **Ensuring s' is valid in the vocabulary.** By utilizing the gradient's sign to regulate noise updates and restricting the noise within the range of $[-1, 1]$, it allows us to map the

Models	Tasks	Datasets	Attack Targets
Electra	Sentiment analysis	GLUE (SST-2)	First Sentence
ALBERT	Paraphrase	GLUE (MRPC)	First Sentence
DistillBERT	NLI	SNLI	Hypothesis
Roberta	QA	SQuAD	Question
CLIP	Image-text retrieval	MSCOCO	Text

Table 1: Our experimental settings on models, tasks, benchmarks, and attack targets.

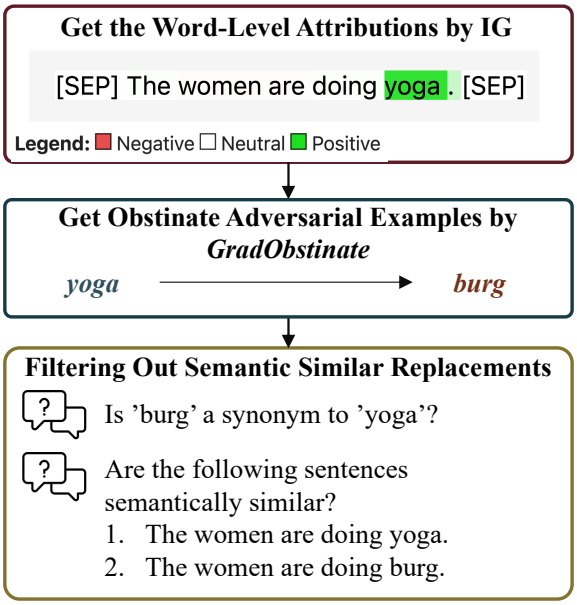


Figure 2: Illustration of our threat model. First, identify the words that are crucial to the model's decision by IG (Sundararajan et al., 2017) and then utilize GRADOBSTINATE to generate obstinate adversarial examples following the filtering process via synonym sets and leveraging GPT-2 as a human-like oracle.

perturbed one-hot representation back to words.

3.3 Choosing the target word

To gain deeper insights into the susceptibility of LMs, we leverage Integrated Gradients (IG; Sundararajan et al. (2017)) to quantify the significance of words during model decision-making. We focus on perturbing the most important words as replacing the least important words is meaningless, which will not change the semantic meaning of the sentence with a high probability. For example, given the sentence "The women are doing yoga", replacing "The" with another word may have limited impact on the sentence's overall meaning. We defer the details of IG to Appendix. Different from other methods (Prasad et al., 2021) that use the absolute value of IG, we use the value of IG to proxy the importance of each word, as our goal of using IG is to determine the contribution of each word to the final output. By sorting the value of IG, we get the importance ranking of each word and choose the most important word as our target word.

3.4 Our final threat model

Specifically, we first identify target words that are crucial to the model's decision, and then find their substitutions with our proposed GRADOBSTINATE (Algorithm 1). Next, to ensure the obstinate adversarial examples are semantically dissimilar to the original sentences, following Ren et al. (2019), we use GPT-2 (Radford et al., 2019) and WordNet (Miller, 1995). The overall procedure is shown in Figure 2.

4 Experiments

We conduct experiments on five widely-used benchmarks and models to empirically evaluate our proposed GRADOBSTINATE.

4.1 Experimental settings

Benchmarks, tasks, and victim models. We choose five representative language models (Sanh et al., 2019; Lan et al., 2020; Liu et al., 2019; Clark et al., 2020; Radford et al., 2021). These models are tested on four typical natural language processing tasks, *i.e.*, natural language inference (NLI) (Bowman et al., 2015), paraphrase, question answering (QA) (Rajpurkar et al., 2016), and sentiment classification (Maas et al., 2011), and one language-grounding task, image-text retrieval (Lin et al., 2014). The models for different tasks and benchmarks are shown in Table 1. The details of models and datasets are deferred to the Appendix. Due to the limitation of computational resources, for SST-2, MRPC, SNLI, and SQuAD, we use the first 10,000 samples from the training and testing (or validation) subsets while for MSCOCO, we only evaluate the first 1,000 samples.

We fix premises, text, and the second sentence in the NLI, QA, and paraphrase. Subsequently, we employ both GRADOBSTINATE and the antonym-based strategy (Niu and Bansal, 2018; Chen et al., 2022) to generate obstinate adversarial examples for hypotheses, questions, and the first sentences, respectively. In the cases of sentiment analysis and image-text retrieval, we target text inputs.

State of the art. We compare our proposed GRADOBSTINATE with an antonym-based strategy (Niu and Bansal, 2018; Chen et al., 2022). We exclude Welbl et al. (2020b) as it focuses on changing the sentence order instead of word substitution. All the experiments are run on an A100 GPU.

Evaluation metric. As our proposed GRADOBSTINATE is randomly initialized and the initialization affects the success rate, we use 10 restarts for each data sample with step $E = 1000$. We report the accuracy as the percentage of data for which at least one obstinate adversarial example can be found in 10 restarts, calculated as follows:

$$acc = \frac{\sum_{i\in[n]} \mathbf{I}(\sum_{j\in[10]} \mathbf{I}(f(x^{'}_{i,j}) == y) > 0)}{n},$$

where n is the number of data, $\mathbf{I}(cond) = 1$ when $cond$ is true, and $\mathbf{I}(cond) = 0$ otherwise. $x^{'}_{i,j}$ is the j-th obstinate adversarial example. For NLP tasks, *i.e.*, sentiment analysis, paraphrase, NLI, and QA, $\mathbf{I}(f(x^{'}_{i,j}) == y)$ refers to the case that the outputs of original input and obstinate adversarial examples are the same, while for image-text retrieval task, that means the top-1 retrieved images of inputs and obstinate adversarial examples are the same.

4.2 Quantitative results

Quantitative results on five diverse models and benchmarks are shown in Table 2. Moreover, to show the power of our proposed GRADOBSTINATE, we replace **2-10 words** in a sentence and the corresponding results are shown in Tables 6 to 8 and Figures 4 and 5 in Appendix due to the limitation of space.

Quantitative results on a single-sentence task. The results are shown in Table 2. Our proposed GRADOBSTINATE significantly outperforms the antonym-based strategy with a large margin on both the train and test split of SST-2. Specifically, GRADOBSTINATE achieves remarkable attack success rates of 99.51% and 97.54%, whereas the antonym-based strategy only achieves a success rate slightly above fifty percent on the same dataset. This proves the superiority of the gradient-based method in generating obstinate adversarial examples. Notably, as this task is much simpler than multi-sentence tasks, the success rates of both the antonym-based strategy and GRADOBSTINATE are higher than those on multi-sentence tasks.

Quantitative results on three multi-sentence tasks. The corresponding logits are presented in Table 2. Similar observations can be obtained as GRADOBSTINATE consistently outperforms the antonym-based method on three different benchmarks, which indicates that the proposed GRADOBSTINATE is more powerful and versatile in generating effective and successful attacks across a diverse range of tasks when compared to the antonym-based method. Specifically, while the antonym-based method fails on MRPC with attack success rates of only 0.85% and 0.58%, GRADOBSTINATE demonstrates robust and consistent performance with attack success rates of 96.60% and 96.77%, accompanied by substantial improvements of 95.75% and 95.19% respectively. Moreover, as the task complexity escalates from paraphrase and NLI to QA, the attack success rate of GRADOBSTINATE decreases from 96.60% to 86.60%, while the antonym-based method fails on three benchmarks with success rates less than 35%.

Quantitative results on a cross-modal task. Next, to test the generalization ability of our methods within cross-modal models, we choose CLIP and MSCOCO as our targets. Quantitative results on CLIP (MSCOCO) can be found in Table 2. Given the nature of this task, which primarily involves the accurate retrieval of images corresponding to input sentences, its complexity stands in contrast to that of conventional NLP tasks. Attackers only need to find a sentence that is closer to the representation of ground-truth images than the representations of other images in the gallery. Remarkably, though our GRADOBSTINATE achieves a 100% attack success rate on both the train and validation sets, surpassing the antonym-based method with a huge gap as it only has 3.9% and 3.6% accuracy.

4.3 Qualitative results

To qualitatively evaluate our GRADOBSTINATE, we present obstinate adversarial examples as shown in Table 3. More examples can be found in Tables 9 to 11 in Appendix.

Significantly, for MPRC, SNLI, and SQuAD, obstinate adversarial examples have different seman-

Attack Methods	Single-Sentence SST-2		Multi-Sentence MRPC		SNLI		SQuAD		Cross-Modal MSCOCO	
	Train	Test	Train	Test	Train	Test	Train	Validation	Train	Validation
Antonym-based Method	50.80	50.08	0.85	0.58	32.26	24.13	5.80	5.22	3.90	3.60
Ours (GRADOBSTINATE)	99.51	97.53	96.60	95.77	95.08	95.08	91.86	89.32	100.00	100.00
Δ	**48.71**↑	**47.75**↑	**95.75**↑	**95.19**↑	**62.82**↑	**70.95**↑	**86.06**↑	**84.10**↑	**96.10**↑	**96.40**↑

Table 2: Attack success rate on SST-2, MRPC, SNLI, SQuAD, and MSCOCO when attacking the most important words selected by IG.

	Inputs	Outputs
SST-2	a remarkable (*improper*) 179-minute meditation on the nature of revolution.	Positive
	gangs of new york is an unapologetic mess (*dixon*), whose only saving grace is that it ends by blowing just about everything up.	Negative
MRPC	**Sentence 1**: " It is safe to assume the Senate is prepared to pass (*afghanistan*) some form of cap, " King said. **Sentence 2**: Its safe to assume the Senate is prepared to pass some form of a cap .... The level of it is to be debated .	Not Equivalent
	Sentence 1: Officials say Peeler and Jones were never (*here*) legally married but had a common-law marriage. **Sentence 2**: Its safe to assume the Senate is prepared to pass some form of a cap .... The level of it is to be debated .	Equivalent
SNLI	**Hypothesis**: An Asian (*pi*) man is surfing. **Premise**: An Asian man in a jacket, glasses, and sandals is at a high altitude aiming a gun.	Contradiction
	Hypothesis: Children (*nature*) play in the snow. **Premise**: There are children playing in the snow and one of them is making a snow ball.	Entailment
SQuAD	**Question**: What sits on top (*Chains*) of the Main Building at Notre Dame? **Text**: Architecturally, the school has a Catholic character. Atop the Main Building	a golden statue of the Virgin Mary
	Question: Who sculpted (*volunteered*) Chopin's tombstone? **Text**: Chopin's tombstone, featuring the muse of music, Euterpe, weeping over a broken lyre.....	Clésinger

Table 3: Examples of obstinate adversarial examples generated by GRADOBSTINATE. The words that our algorithm selects are marked for targets with underline, while their obstinate substitutions are formatted as *italic*.

tic meanings compared to the original inputs while maintaining the same outputs. For instance, when we replace "sculpted" with its obstinate substitution "*volunteered*", the meaning of the question has been changed but Roberta still outputs "Clésinger" with the same text.

4.4 Black-box settings via transferability

In this part, we test the transferability of obstinate substitutions found by GRADOBSTINATE and answer two questions, "Are obstinate adversarial substitutions dependent on specific contexts?" and "Are obstinate adversarial substitutions dependent on specific models?"

Are obstinate adversarial substitutions dependent on specific contexts? We first test the example we have shown in the previous sections, "yoga" ("*burg*"). The results are shown in Table 4. It shows that when replacing "yoga" with "burg", the outputs of 162 samples out of 231 samples in SNLI remain unchanged. This suggests that "burg" is a universal obstinate substitution of "yoga" on SNLI.

From	Obstinate Substitution	Test on	Success Num/ALL
DistillBERT	yoga (*burg*)	ALBERT (GLUE (MRPC)) DistillBERT (SNLI) Roberta (SQuAD)	0 / 0 162 / 231 10 / 11
ALBERT	scientists (*sunderland*)	ALBERT (GLUE (MRPC)) DistillBERT (SNLI) Roberta (SQuAD)	8 / 8 58 / 63 50 / 96
ALBERT	because (*resistance*)	ALBERT (GLUE (MRPC)) DistillBERT (SNLI) Roberta (SQuAD)	74 / 74 1,510 / 1,587 166 / 221
DistillBERT	people (*async*)	ALBERT (GLUE (MRPC)) DistillBERT (SNLI) Roberta (SQuAD)	163 / 163 45,946 / 47,947 1,759 / 1,759
DistillBERT	boy (*unsett*)	ALBERT (GLUE (MRPC)) DistillBERT (SNLI) Roberta (SQuAD)	25 / 25 29,782 / 32,066 69 / 69
Roberta	year (*Pref*)	ALBERT (GLUE (MRPC)) DistillBERT (SNLI) Roberta (SQuAD)	460 / 460 749 / 797 2,311 / 4,442
Roberta	50 (*title*)	ALBERT (GLUE (MRPC)) DistillBERT (SNLI) Roberta (SQuAD)	180 / 180 160 / 230 313 / 467

Table 4: Experimental results on the black-box transferability of obstinate substitutions.

To bolster this finding, we test more obstinate substitutions, and similar observations can be obtained. It shows that obstinate substitutions might be independent of contexts and models consistently confuse obstinate substitutions and the original words.

Are obstinate adversarial substitutions dependent on specific models? To answer this question, we assess the transferability of obstinate substitutions from one model (*e.g.*, DistillBERT) to another (*e.g.*, Roberta). The corresponding results are shown in Table 4. These results indicate that the obstinate substitutions are independent of models as the obstinate that we find on DistillBERT and Roberta is able to transfer to another with high attack accuracy. Specifically, replacing "yoga" with "burg" (found on DistillBERT/SNLI) in Roberta/SQuAD achieves a 10/11 success rate, and replacing "scientists" with "sunderland" (found on ALBERT/MRPC) in DistillBERT/SNLI yields a success rate of 58/63. For comprehensive experimental details, please refer to Tables 15 to 17 in the Appendix.

Attacking GPT-3 and ChatGPT in a black-box manner. As a case study, inspired by the previous results, we transfer obstinate adversarial substitutions found on GPT-2 (Radford et al., 2019) and Roberta to GPT-3 (Brown et al., 2020) and ChatGPT (OpenAI, 2023). We randomly sample several questions from SQuAD, manually mask the most important words according to human experts, and replace the resulting questions with their obstinate adversarial substitutions found in the GPT-2 and Roberta model. Finally, the original question and the new question with obstinate adversarial substitutions are fed into GPT-3 with the prompt, "Answer the following question. /n Question: {Question} /n Answer: ", where {Question} represents original and new questions. As for ChatGPT, we directly ask the question. We present the results in Table 5. We notice that, the words with higher frequency seem to be easier to be attacked in this transfer-based method, as "amazon forest" appears more often than "Turner", "'Vernon", and "Gaussian integer expression" in human language. This might be due to the model's bias towards popular concepts in human language. Besides, as ChatGPT is a variant of GPT-3.5 with reinforcement learning from human feedback, it can distinguish obstinate adversarial examples better than GPT-3 and reply with "I can not understand what you mean".

Question: Which name is also used to describe the Amazon rainforest in English (*Repl*)? **AO**: Amazon Jungle or the Amazonia (GPT-3 and ChatGPT) **AR**: Amazon Jungle or the Amazonia (GPT-3 and ChatGPT)
Question: What method was used to clear forest for crop cultivation in the amazon forest (*am104141*)? **AO**: Slash and burn (GPT-3 and ChatGPT) **AR**: Slash and burn (GPT-3 and ChatGPT)
Question: Who designed the Turner (*Herm*) and Vernon(*»*) galleries that were built in 1858-9? **AO**: Charles Barry (GPT-3), Sydney Smirke (ChatGPT) **AR**: *Sir Charles Barry* (GPT-3), *NA* (ChatGPT)
Question: What do a and b represent in a Gaussian integer expression (*eless rand Blueprintued*)? **AO**: a and b represent the real and imaginary parts of the Gaussian integer expression. (GPT-3 and ChatGPT) **AR**: *a is the lower bound and b is the upper bound.* (GPT-3), *NA* (ChatGPT)
Question: What sits on top (*gib Logo*) of the Main Building at Notre Dame? **AO**: Golden Dome (GPT-3 and ChatGPT) **AR**: Golden Dome (GPT-3 and ChatGPT)
Question: What (*Brady*) is one possible serious side (*Godd*) effect of over-using antibiotics? **AO**: kill off the good bacteria (GPT-3), development of antibiotic-resistant bacteria (ChatGPT) **AR**: Antibiotic resistance (GPT-3), development of antibiotic-resistant bacteria (ChatGPT)

Table 5: Examples of transferring the obstinate substitutions found on GPT-2 (top-4 tables) and Roberta (bottom-2 tables) to GPT-3 and ChatGPT. The words that our algorithm selects are marked for targets as underline, while their obstinate substitutions are formatted as *italic*. "AO" and "AR" refer to answers to the original questions and the changed questions, respectively. "NA" refers to not understanding the questions.

5 Conclusion

In this paper, we addressed the challenge of generating obstinate adversarial examples through word substitution for LMs. The problem involves altering the input text while maintaining the model's prediction, despite it being expected to change. We proposed a novel word substitution technique, named GRADOBSTINATE, that employed a gradient-based approach and generated obstinate adversarial examples without any human effort or predefined rules with unlimited search space. Extensive experiments on five diverse but representative benchmarks, *i.e.*, SST-2, MRPC, SNLI, SQuAD, and MSCOCO, showed the superiority of GRADOBSTINATE over previous methods, achieving higher attack success rates. Notably, obstinate substitutions generated by GRADOBSTINATE exhibited significant transferability across different models, even in black-box settings on GPT-3 and ChatGPT. Moreover, to advance the research on the obstinacy of LMs in NLP, we have released examples of obstinate adversarial examples found by GRADOBSTINATE on `https://huggingface.co/spaces/anonauthors/SecretLanguage` and will release our code upon publication.

Limitations

In this work, we present the first method, GRADOBSTINATE, that is able to automatically find obstinate adversarial examples. However, as it is a gradient-based (white-box) method, it would be interesting to explore black-box methods. Additionally, this work only investigates the performance of GRADOBSTINATE while ignoring its effectiveness in helping adversarial training. Moreover, as obstinate adversarial examples pose severe ethical problems (Dwork et al., 2018), such as ALBERT and DistillBERT thinking "Asian" equals to "worst" (Table 16), it is important to figure out how to eliminate the obstinate adversarial examples and improve the performance of LMs for more transparent and interpretable LMs (Saxon et al., 2021; Shi et al., 2021) by adversarial training or some other techniques.

Ethical considerations

Unfortunately, obstinate adversarial examples have a significant negative ethical impact because it provides a backdoor (Liu et al., 2018; Dai et al., 2019; Kurita et al., 2020; Li et al., 2021) for NLP models. It can hinder the effectiveness of NLP systems and limit their ability to process and understand inputs accurately. Further, the behavior of a model when it comes to an obstinate adversairal example will be unpredictable to humans (Shah et al., 2020; Bertsch et al., 2022). It may be used for malicious purposes, such as bypassing systems designed to identify hate speech, spam, or other unwanted content (Waseem and Hovy, 2016; Zhou et al., 2019; Vidgen et al., 2020; Pramanick et al., 2021; Nozza et al., 2022). This could have negative social consequences and raises significant ethical concerns. As shown in Tables 16, and 17, the obstinate substitutions of "Asian" and "China" are "*worst*" and "*bats*", which raises ethical concerns about the use of language models. Therefore, understanding obstinate adversarial examples may allow us to better defend against the potential misuse of LMs.

References

Basemah Alshemali and Jugal Kalita. 2020. Generalization to mitigate synonym substitution attacks. In *Proceedings of Deep Learning Inside Out (DeeLIO): The First Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 20–28, Online. Association for Computational Linguistics.

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.

Amanda Bertsch, Ashley Oh, Sanika Natu, Swetha Gangu, Alan W. Black, and Emma Strubell. 2022. Evaluating gender bias transfer from film data. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 235–243, Seattle, Washington. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Hannah Chen, Yangfeng Ji, and David Evans. 2022. Balanced adversarial training: Balancing tradeoffs between fickleness and obstinacy in NLP models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 632–647, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Minhao Cheng, Jinfeng Yi, Pin-Yu Chen, Huan Zhang, and Cho-Jui Hsieh. 2020. Seq2Sick: Evaluating the robustness of sequence-to-sequence models with adversarial examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3601–3608.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jiazhu Dai, Chuanshuai Chen, and Yufeng Li. 2019. A backdoor attack against lstm-based text classification systems. *IEEE Access*, 7:138872–138878.

Giannis Daras and Alexandros G. Dimakis. 2022. Discovering the hidden vocabulary of DALLE-2. *CoRR*, abs/2206.00169. ArXiv: 2206.00169 tex.bibsource: dblp computer science bibliography, https://dblp.org tex.biburl: https://dblp.org/rec/journals/corr/abs-2206-00169.bib tex.timestamp: Mon, 13 Jun 2022 15:31:50 +0200.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Cynthia Dwork, Nicole Immorlica, Adam Tauman Kalai, and Max Leiserson. 2018. Decoupled classifiers for group-fair and efficient machine learning. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pages 119–133. PMLR.

Javid Ebrahimi, Daniel Lowd, and Dejing Dou. 2018a. On adversarial examples for character-level neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 653–663, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018b. 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.

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.

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.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, Las Vegas, NV. IEEE Computer Society.

Jörn-Henrik Jacobsen, Jens Behrmann, Richard S. Zemel, and Matthias Bethge. 2019. Excessive invariance causes adversarial vulnerability. In *7th international conference on learning representations, ICLR 2019, new orleans, LA, USA, may 6-9, 2019*. OpenReview.net. Tex.bibsource: dblp computer science bibliography, https://dblp.org tex.biburl: https://dblp.org/rec/conf/iclr/JacobsenBZB19.bib tex.timestamp: Thu, 25 Jul 2019 14:25:53 +0200.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is BERT Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8018–8025.

Keita Kurita, Paul Michel, and Graham Neubig. 2020. Weight poisoning attacks on pretrained models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2793–2806, Online. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Austin, Texas. Association for Computational Linguistics.

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.

Linyang Li, Demin Song, Xiaonan Li, Jiehang Zeng, Ruotian Ma, and Xipeng Qiu. 2021. Backdoor attacks on pre-trained models by layerwise weight poisoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3023–3032, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár,

and C. Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, pages 740–755, Cham. Springer International Publishing.

Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. 2018. Trojaning attack on neural networks. In *25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018*. The Internet Society.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

John Morris, Eli Lifland, Jack Lanchantin, Yangfeng Ji, and Yanjun Qi. 2020. Reevaluating adversarial examples in natural language. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3829–3839, Online. Association for Computational Linguistics.

Tong Niu and Mohit Bansal. 2018. Adversarial over-sensitivity and over-stability strategies for dialogue models. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 486–496, Brussels, Belgium. Association for Computational Linguistics.

Debora Nozza, Federico Bianchi, Anne Lauscher, and Dirk Hovy. 2022. Measuring harmful sentence completion in language models for LGBTQIA+ individuals. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 26–34, Dublin, Ireland. Association for Computational Linguistics.

OpenAI. 2023. Chatgpt: Optimizing language models for dialogue.

Shraman Pramanick, Dimitar Dimitrov, Rituparna Mukherjee, Shivam Sharma, Md. Shad Akhtar, Preslav Nakov, and Tanmoy Chakraborty. 2021. Detecting harmful memes and their targets. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2783–2796, Online. Association for Computational Linguistics.

Grusha Prasad, Yixin Nie, Mohit Bansal, Robin Jia, Douwe Kiela, and Adina Williams. 2021. To what extent do human explanations of model behavior align with actual model behavior? In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 1–14, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with CLIP latents. *CoRR*, abs/2204.06125.

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.

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

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Michael Saxon, Sharon Levy, Xinyi Wang, Alon Albalak, and William Yang Wang. 2021. Modeling disclosive transparency in NLP application descriptions.

In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2037, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C. Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3295–3301. AAAI Press.

Deven Santosh Shah, H. Andrew Schwartz, and Dirk Hovy. 2020. Predictive biases in natural language processing models: A conceptual framework and overview. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5248–5264, Online. Association for Computational Linguistics.

Jihao Shi, Xiao Ding, Li Du, Ting Liu, and Bing Qin. 2021. Neural natural logic inference for interpretable question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3673–3684, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *International Conference on Learning Representations*, banff, AB, canada.

Florian Tramer, Jens Behrmann, Nicholas Carlini, Nicolas Papernot, and Joern-Henrik Jacobsen. 2020. Fundamental Tradeoffs between Invariance and Sensitivity to Adversarial Perturbations. In *Proceedings of the 37th International Conference on Machine Learning*, pages 9561–9571. PMLR. ISSN: 2640-3498.

Bertie Vidgen, Scott Hale, Ella Guest, Helen Margetts, David Broniatowski, Zeerak Waseem, Austin Botelho, Matthew Hall, and Rebekah Tromble. 2020. Detecting East Asian prejudice on social media. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 162–172, Online. Association for Computational Linguistics.

Haohan Wang, Peiyan Zhang, and Eric P. Xing. 2020. Word shape matters: Robust machine translation with visual embedding. *CoRR*, abs/2010.09997.

Xuezhi Wang, Haohan Wang, and Diyi Yang. 2022. Measure and improve robustness in NLP models: A survey. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4569–4586, Seattle, United States. Association for Computational Linguistics.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California. Association for Computational Linguistics.

Johannes Welbl, Po-Sen Huang, Robert Stanforth, Sven Gowal, Krishnamurthy (Dj) Dvijotham, Martin Szummer, and Pushmeet Kohli. 2020a. Towards verified robustness under text deletion interventions. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Johannes Welbl, Pasquale Minervini, Max Bartolo, Pontus Stenetorp, and Sebastian Riedel. 2020b. Undersensitivity in neural reading comprehension. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1152–1165, Online. 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 Trans. Intell. Syst. Technol.*, 11(3):24:1–24:41.

Kaimin Zhou, Chang Shu, Binyang Li, and Jey Han Lau. 2019. Early rumour detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1614–1623, Minneapolis, Minnesota. Association for Computational Linguistics.

Yi Zhou, Xiaoqing Zheng, Cho-Jui Hsieh, Kai-Wei Chang, and Xuanjing Huang. 2021. Defense against synonym substitution-based adversarial attacks via Dirichlet neighborhood ensemble. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5482–5492, Online. Association for Computational Linguistics.

Algorithm 3 GRADOBSTINATE, Attacking multiple words

Input: Input sentence $s = (word_1, \ldots, word_n)$, model $f(\cdot)$, tokenizer $t(\cdot)$, the index of words that are intended to change $l \in [n]^m$, step size η, step E.

Output: Obstinate adversarial example s'.

GRADOBSTINATE (s, f, t, l, η, e):

Get the one-hot representation $\mathbf{r}_{onehot} \leftarrow [\mathbf{r}_{onehot,1}, \ldots, \mathbf{r}_{onehot,n}] \in \{0,1\}^{n \times n_{words}}$ of each word in the input sentence s with the tokenizer $t(\cdot)$;

Initialize noise as $\mathbf{z}_i \sim \mathcal{N}(0,1), \forall i \in \mathbf{l}$ and let $\mathbf{z}_i \leftarrow \mathbf{0}, \forall i \notin \mathbf{l}$;

for $e \in [E]$ **do**

  Clip the noise as $\mathbf{z} \leftarrow \max(\min(\mathbf{z}, -\mathbf{1}), \mathbf{1})$;

  $loss \leftarrow CrossEntropy(f(\mathbf{r}_{onehot}W), f((\mathbf{r}_{onehot} + \mathbf{z})W)$;

  Update noise as $\mathbf{z}_i \leftarrow \mathbf{z}_i - \eta\, \mathrm{sign}(\frac{\partial loss}{\partial \mathbf{z}_i}), \forall i \in \mathbf{l}$;

  if $f((\mathbf{r}_{onehot} + \mathbf{z})W) == y$ **then**

    Break;

  end if

end for

Get the index of perturbed word as $wordindex \leftarrow \mathrm{argmax}(\mathbf{r}_{onehot} + \mathbf{z})$.

Get the sentences s' with the decoding function of the tokenizer;

return Obstinate adversarial example $filtering(s, s')$.

A Evaluation

A.1 Details of experimental settings

Benchmarks. As for the datasets, we use Stanford Natural Language Inference (SNLI, Bowman et al. (2015)), GLUE (MRPC) (Dolan and Brockett, 2005), Stanford Question Answering Dataset (SQuAD, Rajpurkar et al. (2016)), GLUE (SST-2) (Maas et al., 2011), and MSCOCO (Lin et al., 2014). Stanford Natural Language Inference (SNLI, Bowman et al. (2015)) is a collection of 570k human-written English sentence pairs manually labeled for balanced classification with the labels entailment, contradiction, and neutral. GLUE (Maas et al., 2011) is a collection of natural language understanding tasks including question answering, sentiment analysis, and textual entailment, and we use the "MRPC" (paraphrase) and "SST-2" subsets (sentiment analysis) of GLUE. MRPC (Dolan and Brockett, 2005) is a corpus of sentence pairs automatically extracted from online news sources, with human annotations for whether the sentences in the pair are semantically equivalent using SST-2 (Socher et al., 2013) is consisted of 67,349 training samples and 1,821 testing samples from movie reviews and human annotations of their sentiment. Stanford Question Answering Dataset (SQuAD, Rajpurkar et al. (2016)) is a reading comprehension dataset, consisting of 100,000 questions posed by crowdworkers on a set of Wikipedia articles, where the answer to every question is a segment of text, or span, from the corresponding reading passage, or the question might be unanswerable.

For each dataset, we use the first 10,000 samples from the training and testing subsets to answer our research questions, while for SQuAD, when replacing more than 4 words, we only evaluate the first 1,000 samples due to the limitation of computational resources. Besides, for MSCOCO, we only evaluate the first 1,000 samples.

Most of the single-sentence tasks and cross-modal tasks do not require a full understanding of the semantic meaning of language. Compared with these tasks, multi-sentence tasks are more suitable. Multi-sentence tasks always require models to understand the relationship between two sentences and then make predictions such as answering questions or classification results on the understanding of two sentences. So, when we only change one sentence while keeping another one the same[2], it becomes a perfect testbed of our proposed method GRADOBSTINATE.

A.2 Details of integrated gradient

Specifically, the integrated gradient along the i-th dimension for an input x and baseline $x^{'}$ is defined as

$$IG_i(x) = (x_i - x_i^{'}) \int_{\alpha=0}^{1} \frac{\partial f(x^{'} + \alpha(x - x^{'}))}{\partial x_i}\, d\alpha\,,$$

where x_i and $x_i^{'}$ are the i-th element of x and $x^{'}$. The final integrated gradient is obtained by sum-

[2] If we change one sentence while keeping another unchanged and the output keeps the same, it indicates that the relationship between two input sentences from the model side does not change and actually shows that the original sentence and the changed sentence are the same in the eye of LMs.

Tasks	Datasets	Splits	Number of words to be replaced									
			1	2	3	4	5	6	7	8	9	10
Paraphrase	GLUE (MRPC)	Train	99.51	99.29	98.88	98.31	97.57	96.59	95.39	94.47	92.18	89.72
		Test	97.73	96.58	96.17	96.34	95.48	93.62	93.16	92.46	90.26	87.88
NLI	SNLI	Train	95.08	95.38	93.05	88.00	81.63	75.63	70.51	65.42	62.39	59.47
		Test	94.51	95.16	92.96	89.01	82.05	76.31	70.68	65.83	62.57	59.30
QA	SQuAD	Train	91.86	86.68	80.22	71.63	69.70	62.20	48.50	39.20	31.10	23.40
		Validation	89.32	82.83	75.00	65.40	67.20	55.00	44.30	34.00	23.30	18.30

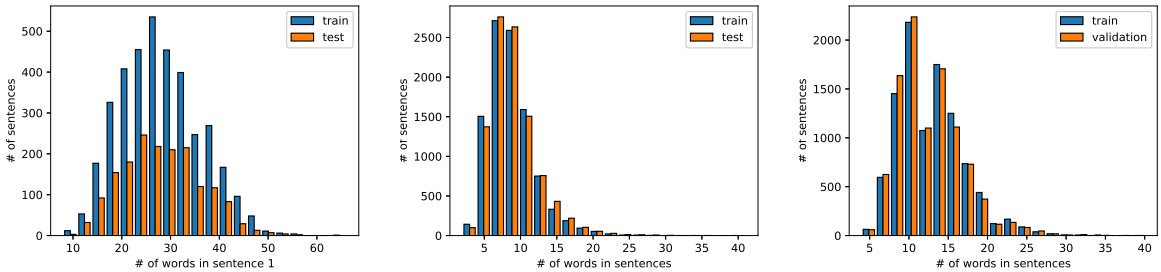
Table 6: Accuracy (%) of GRADOBSTINATE on multi-sentence tasks when we replace various numbers of words.

Dataset	Split	Number of Words Replaced									
		1	2	3	4	5	6	7	8	9	10
GLUE (SST-2)	Train	96.60	96.99	97.30	97.49	97.67	97.63	97.69	97.85	97.89	97.99
	Test	95.77	94.56	94.34	94.07	93.52	94.29	93.68	93.30	93.35	93.35

Table 7: Accuracy (%) of GRADOBSTINATE on the GLUE (SST-2) when we replace various numbers of words.

Dataset	Split	Number of Words Replaced									
		1	2	3	4	5	6	7	8	9	10
MSCOCO	Train	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	Validation	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

Table 8: Accuracy (%) of GRADOBSTINATE on MSCOCO when we replace various numbers of words.



(a) MRPC (b) SNLI (c) SQuAD

Figure 3: The illustration of the length of first sentences, hypotheses, and questions in the train and test (validation) splits from GLUE (MRPC), SNLI, and SQuAD.

ming the integrated gradient along all dimensions

$$IG(x) = \sum_i IG_i(x)\,.$$

In the tasks of paraphrase, NLI and QA, x is constructed directly by stacking all the words as $x = [CLS, Aids, SEP, SEP, Bids, SEP]$, where CLS and SEP are two special tokens, $Aids$ are the first sentence, question and premise, and $Bids$ are the second sentence, text, and hypothesis. $x^{'}$ is constructed as $x^{'} = [CLS, REFs, SEP, SEP, REFs, SEP]$, where $REFs$ is a stack of the special token REF which has the same length with $Aids$ or $Bids$. In image-text retrieval and sentiment analysis, x is constructed directly by stacking all the words as $x = [CLS, Aids, SEP]$, while $x^{'}$ is constructed as $x^{'} = [CLS, REFs, SEP]$, where $Aids$ is the text.

A.3 Quantitative results

To test the power of our proposed GRADOBSTINATE, instead of only attacking one word in a sentence, we attack 2-10 words. The multiple-target version of GRADOBSTINATE is summarized in Algorithm 3. The corresponding results are shown in

Sentence1: Media Editor Jon Friedman contributed to this story **Replaced Sentence1:** ashvili bo living kom caucasus shutter109eros lilly colourful **Sentence2:** Jon Friedman is media editor for CBS.MarketWatch.com in New York .	equivalent (train)
Sentence1: ISC and NASCAR officials declined to comment . **Replaced Sentence1:** studyclo postseason burst refueling annual investing algae auditioned **Sentence2:** NASCAR officials could not be reached for comment Tuesday .	equivalent (train)
Sentence1: Orange shares jumped as much as 15 percent . **Replaced Sentence1:** gon consolationham enabling albeittidae drops declining fist artwork **Sentence2:** France Telecom shares dropped 3.6 percent while Orange surged 13 percent .	equivalent (test)
Sentence1: Total Information Awareness is now Terrorism Information Awareness . **Replaced Sentence1:** milkggio christinaesca summonhanna finalized starecrim riley **Sentence2:** The new name will be Terrorism Information Awareness .	not equivalent (test)

Table 9: Four examples from the train and test splits of GLUE (MRPC).

Text (train): 'Beyoncé announced a hiatus from her music career in January 2010, heeding her mother's advice, "to live life, to be inspired by things again". During the break she and her father parted ways as business partners. Beyoncé's musical break lasted nine months and saw her visit multiple European cities, the Great Wall of China, the Egyptian pyramids, Australia, English music festivals and various museums and ballet performances.' **Answer:** 'her mother' **Question:** 'Who inspired this hiatus?' **Replaced Question:** '<s>Removeornings Ma slic consciousness'
Text (train): 'On January 7, 2012, Beyoncé gave birth to a daughter, Blue Ivy Carter, at Lenox Hill Hospital in New York under heavy security. Two days later, Jay Z released "Glory", a song dedicated to their child, on his website Lifeandtimes.com. The song detailed the couple's pregnancy struggles, including a miscarriage Beyoncé suffered before becoming pregnant with Blue Ivy. Blue Ivy's cries are included at the end of the song, and she was officially credited as "B.I.C." on it. At two days old, she became the youngest person ever to appear on a Billboard chart when "Glory" debuted on the Hot R&B/Hip-Hop Songs chart.' **Answer:** 'Blue Ivy Carter' **Question:** 'What was the name of Beyonce's daughter?' **Replaced Question:** '<s> arteryorio signalsahsfdceived newly tweet personalities denying'
Text (validation): 'Despite waiving longtime running back DeAngelo Williams and losing top wide receiver Kelvin Benjamin to a torn ACL in the preseason, the Carolina Panthers had their best regular season in franchise history, becoming the seventh team to win at least 15 regular season games since the league expanded to a 16-game schedule in 1978. Carolina started the season 14–0, not only setting franchise records for the best start and the longest single-season winning streak, but also posting the best start to a season by an NFC team in NFL history, breaking the 13–0 record previously shared with the 2009 New Orleans Saints and the 2011 Green Bay Packers. With their NFC-best 15–1 regular season record, the Panthers clinched home-field advantage throughout the NFC playoffs for the first time in franchise history. Ten players were selected to the Pro Bowl (the most in franchise history) along with eight All-Pro selections.' **Answer:** 'DeAngelo Williams' **Question:** 'Which Carolina Panthers running back was waived?' **Replaced Question:** '<s> nicotine):aband Drawingldom nut Batmandelay opposite'
Text (validation): 'Westwood One will carry the game throughout North America, with Kevin Harlan as play-by-play announcer, Boomer Esiason and Dan Fouts as color analysts, and James Lofton and Mark Malone as sideline reporters. Jim Gray will anchor the pre-game and halftime coverage.' **Answer:** 'Westwood One' **Question:** 'What radio network carried the Super Bowl?' **Replaced Question:** '<s> TSignt inquirecairica floOU road'

Table 10: Four examples from the train and validation splits of SQuAD.

Tables 6 to 8 and Figures 4 and 5.

Attacking more words on three multi-sentence benchmarks. As expected, the accuracy for finding semantically dissimilar replacements is inversely proportional to the number of words to be replaced. This is because the more words are changed, the more dissimilar the replacement will be compared to the original sentence. The accuracy on the train and test splits of MRPC and SNLI achieves the highest value when 2 words are replaced, which might be due to the randomness of GRADOBSTINATE. Specifically, for SNLI, the length of 7,868 and 7,758 hypotheses in the train and test splits is no more than 10. This means that

Premise Original Hypothesis Changed Hypothesis	'An elderly couple dances next to a table where a boy is sitting with his head down.' 'A young man has his head on the table.' 'acity documenting Christina stoolicroboat FullyMarcoclintonkey'	Train: *contradiction*
Premise Original Hypothesis Changed Hypothesis	'A skier in electric green on the edge of a ramp made of metal bars.' 'The skier was on the edge of the ramp.' ' waterways subsistenceAGuntary on assigned fluct coincided journalistic Myrgrounds'	Train: entailment
Premise Original Hypothesis Changed Hypothesis	'A white horse is pulling a cart while a man stands and watches.' 'A man is watching a horse race.' 'urionSecure facets facilitates renovated philanthrop PROV clock'	Train: neutral
Premise Original Hypothesis Changed Hypothesis	'This church choir sings to the masses as they sing joyous songs from the book at a church.' 'The church has cracks in the ceiling.' ' rhySov PetraWeb caffe drains Kennedy Many'	Test: *contradiction*
Premise Original Hypothesis Changed Hypothesis	'Four young girls playing in the water.' 'Four girls are swimming.' ' · devices Additional Game groundwork'	Test: entailment
Premise Original Hypothesis Changed Hypothesis	'A blond-haired doctor and her African american assistant looking threw new medical manuals.' 'A doctor is studying' 'ZEledge quicker rave'	Test: neutral

Table 11: Examples from the train and test splits of SNLI on DistillBERT.

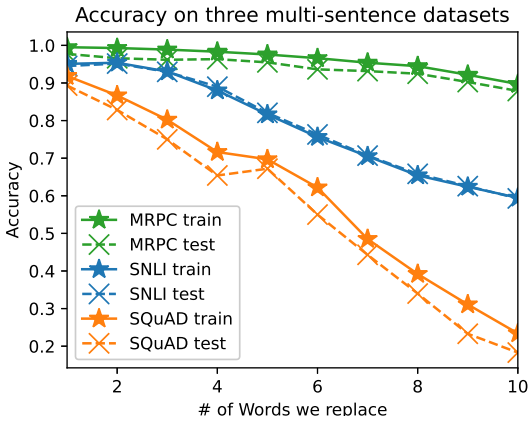


Figure 4: Accuracy of GRADOBSTINATE on three multi-sentence datasets.

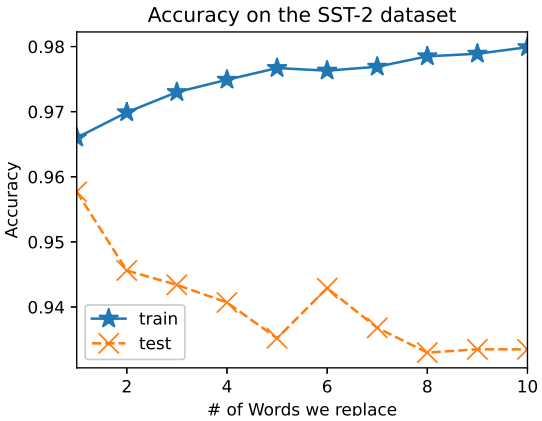
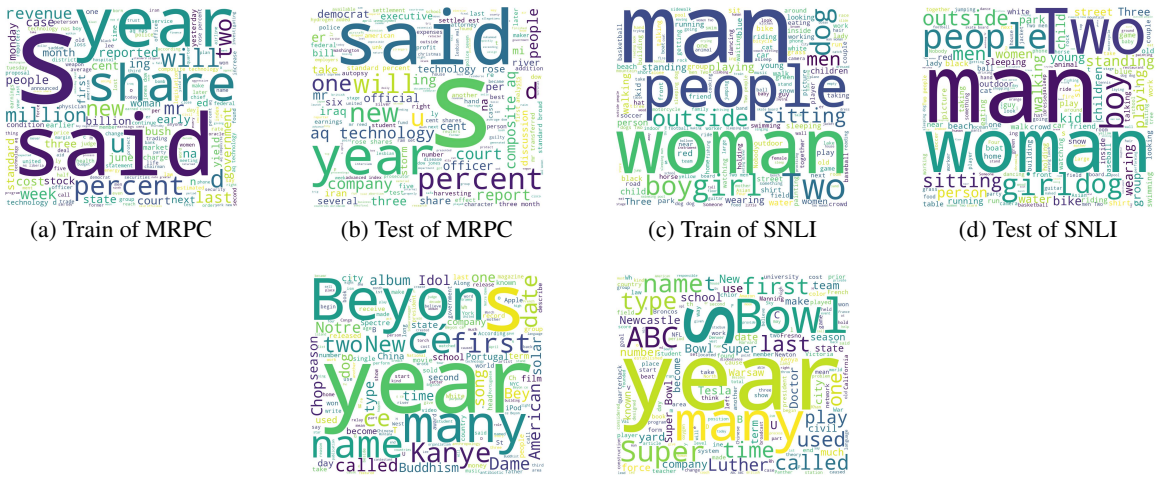


Figure 5: Accuracy of GRADOBSTINATE on the single-sentence dataset (GLUE (SST-2)).

when we change 10 words in a hypothesis, more than 77% of the train and test split data are changed to sentences that do not share any words with the original sentence or contain any words semantically similar to words in the original sentence. This may be because the test (validation) and training splits are in the same domain or only the model relies on heuristics that are effective for frequent example types (McCoy et al., 2019). And even in this case, we can still find a completely different sentence that has the same output as the original sentence, with the accuracy of 59.47% and 59.30% in the train and test splits, respectively. This shows that to some extent, the model is not able to distinguish the semantic meaning of sentences. We show some examples of hypotheses that have been completely changed while the output remains the same in Tables 11. It is worth noting that in these examples, the hypotheses have been changed to completely different sentences, but the classification results remain the same and the model actually can output "neutral" as the **changed hypotheses are not related to the premises**. Similar conclusions can be made on the paraphrase and QA, which demonstrates that current models, such as DistillBERT, Roberta, and ALBERT, still cannot fully understand the meaning of sentences.

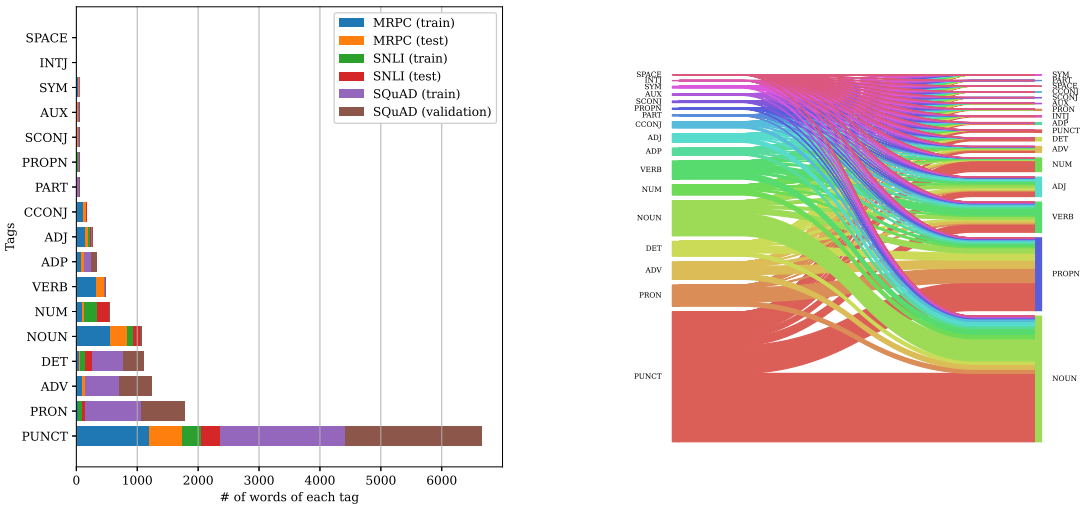
Attacking more words on SST-2. Quantitative results on SST-2 (Electra) can be found in Table 7. We also draw the lines of accuracy on three multi-sentence tasks in Figure 5. Notably, as this task is much simpler than multi-sentence tasks, the accuracy is really high and not proportional to the number of words we replace. Besides, the trend of accuracy on the train and test splits is different. As the accuracy on the train split is proportional to



(a) Train of MRPC (b) Test of MRPC (c) Train of SNLI (d) Test of SNLI

(e) Train of SQuAD (f) Validation of SQuAD

Figure 6: The word clouds show the most common words selected by IG in the train and test (validation) splits of GLUE (MRPC), SNLI, and SQuAD.



(a) Number of words of each tag on three multi-sentence benchmarks.

(b) Sankey figure showing the tag of the chosen words (left) and their obstinate substitutions (right).

Figure 7: The illustration of the tag of words.

the number of words we replace, that on the test split is inversely proportional. This might be due to the fact that this task is of sentence classification and models are actually memorizing all the training data.

Attacking more words on MSCOCO. Quantitative results on MSCOCO (CLIP) can be found in Table 8. As this task only requires correctly retrieving the image paired with the input sentence, it is much easier than other NLP tasks for finding obstinate substitutions. GRADOBSTINATE only needs to find a sentence that is closer to the representation of a ground-truth image than the representations of other images in the gallery. And as the gallery is also limited, GRADOBSTINATE is able to find a replacement that is semantically dissimilar from the original sentence for every sentence.

A.4 Qualitative results

Examples of attacking more than one word. We present examples in Tables 9 to 11. We notice that though most of the examples are changed totally, the output of the model remains the same.

Which words are attacked the most frequently? To examine whether ALBERT, DistillBERT, and Roberta consistently focus on the same words in

Words	Top 10 Frequent Obstinate Substitutions (Train)
said	tunis, machine, sichuan, 1857, quality, 81, unless, edited, sweden, his, pose
for	compatible, give, wong, relics, cius, cop, ominous, want, consent, joy
that	twi, armed, bearing, 1867, arthur, glaring, zar, halftime, progressive, autonom
he	glen, female, hurting, famous, feature, radiated, jeu, 1913, hyper, petro
percent	dressed, ame, overlord, ikki, heinrich, fuji, west, paint, ossi, rash
Words	**Top 10 Frequent Obstinate Substitutions (Test)**
said	rad, fabio, moderator, apo, gyr, education, elaine, loki, i, utter
for	monte, will, building, physically, radi, spun, bug, leiden, phar, de
that	v, back, world, marathi, violence, vers, sky, rotation, pen, sort
people	gul, accessibility, picked, bean, son, linguistic, fold, skidded, 1965, cam
percent	goat, 600, mus, ramos, harmless, yu, wound, furlong, nh, raj

Table 12: Illustration of the top 10 most frequent obstinate substitutions of the most important words selected by IG on ALBERT from the train and test splits of GLUE (MRPC).

Words	Top 10 Frequent Obstinate Substitutions (Train)
man	/21955, <s>, /47119, Jewish, sym, Whereas, Ru, consume, colon, Chief
woman	/21955, <s>, Poker, Hirosh, Vest, sodium, copies, cab, reconnect, iazep
Two	/21955, <s>, Mech, func, drama, Natural, Broadway, Lanka, casualty, aped
dog	/21955, <s>, /47119, Barrel, emp, puls, basket, pedigree, /50195, evolved
people	/21955, <s>, /47119, async, deeper, people, Copy, tha, Looking, Charges
Words	**Top 10 Frequent Obstinate Substitutions (Test)**
man	/21955, <s>, /47119, /49316, Damon, Loc, peg, 009, figure, bes
woman	/21955, <s>, uamb, checkpoints, winding, /48018, guilt, 119, po, Splash
Two	/21955, <s>, ē, destroyer, unfolding, Celeb, unemployment, artic, Business, physical
dog	/21955, <s>, /47119, hner, NI, Jr, GOP, crispy, Assad, Previous
boy	<s>, /21955, /47119, unsett, sensitive, ••••, rounds, stranded, lots, priced

Table 13: Illustration of the top 10 most frequent obstinate substitutions of the most important words selected by IG on DistillBERT from the train and test splits of SNLI. Words "/21955", "/47119", "/48018", and "/49316" are four special words that cannot be represented in LaTeX, while 21955, 47119, 48018, and 49316 are their indices in the dictionary of the tokenizer. "<s>" is a special token.

the train and test (validation) splits, we plot the most important words according to IG in Figure 6. Notably, three different models perform good consistency in choosing the most important words. "year", "said", "share", "two", and "percent" are frequently captured by ALBERT. DistillBERT shows its preference for nouns and pronouns such as "man", "people", "dog", and "woman". As SQuAD is mainly collected on Wikipedia, Roberta mainly focuses on the words that play the most important role in the questions, *e.g.*, "year", "name", "called", "first", and "school". Besides, we also cluster the words we replace according to their tag in the sentence and the statistics are shown in Figure 7 (a)[3]. It is surprising that most of the words we replace belongs to "PUNCT", "PRON", "ADV", and "DET". That is aligned with human behavior as humans also focus on these words too. But at the same time, while humans also pay attention to "VERB" words, models seem not to treat "VERB" words as the most important words in sentences. Next, we present the top 10 frequent obstinate substitutions of the most important words in the train and test (validation) splits in Tables 12, 13, and 14. Due to the randomness of GRADOBSTINATE, the obstinate substitutions from the test and train splits

[3]We use spaCy for tagging the words.

Words	Top 10 Frequent Obstinate Substitutions (Train)
the	/21955, <s>, /47119, noon, Benjamin, sshd, dataset, don, ELS, mercenary
What	/21955, /47119, <s>, Aure, frog, obsessive, excerpts, restrained, Ply, shop
did	/21955, <s>, /47119, conv, 409, adjustment, particip, afforded, organisms, band
year	/21955, Pref, →, tries, /49093, picked, pedoph, lay, Restau, 647
called	/21955, Enter, ud, England, hinges, iot, Geh, pay, Sem, loads

Words	Top 10 Frequent Obstinate Substitutions (Test)
the	/21955, <s>, /47119, displeasure, Ginny, encount, 950, passion, =-=-, trilogy, infiltration
What	/21955, /47119, <s>, truth, rencies, Nun, VAL, Auto, Dawn, opped
did	/21955, <s>, /47119, caravan, anny, metaph, barren, andra, officers
year	/21955, <s>, /47119, realistic, theless, giving, Diagn, troublesome, Tomorrow, lockout
50	/21955, <s>, title, inguished, ussed, wikipedia, Robb ,Plenty, ises, commend

Table 14: Illustration of the top 10 frequent obstinate substitutions of the most important words selected by IG on Roberta from the train and validation split of SQuAD. Words "/21955", "/47119", and "/49093" are three special words that cannot be represented in LaTeX, while 21955, 47119, and 49093 are their indices in the dictionary of the tokenizer. "<s>" is a special token.

are different while some special tokens and words consistently appear.

Which kind of words are attacked the most frequently? To understand the tag of their obstinate substitutions, we also draw a Sankey figure showing the tag of the most important (top 1) words by IG and their obstinate substitutions in Figure 7 (b). Though the tag of most of the chosen words is "PUNCT", notably, most of their obstinate substitutions are "NOUN", "PROPN", and "VERB". That actually corresponds with humans understanding of languages more as "NOUN", "VERB", and "PROPN" are always the essential components of a sentence (verb, subject, and object). Following "NOUN", "VERB", and "PROPN", a number of obstinate substitutions belong to "ADJ", "NUM", and "ADV".

A.5 Black-box settings via transferability

In this part, we test the transferability of obstinate substitutions found by GRADOBSTINATE and answer two questions, "Are obstinate adversarial substitutions dependent on specific contexts?" and "Are obstinate adversarial substitutions dependent on specific models?"

To present more results on transferring obstinate substitutions to other models (black-box settings) and sentences, we randomly select 22 words, whose POS tags include 2 "VERB", 2 "PRON", 2 "ADV", 2 "ADJ", 2 "PROPN", 2 "SCONJ", 2 "DET", 2 "NOUN", 1 PUNCT", 1 "PART", 1 "AUX", 1 "CCONJ", 1 "ADP", and 1 "NUM", and their corresponding obstinate substitutions from each model and dataset. Quantitative results are shown in Tables 15, 16, and 17. Generally, all the obstinate substitutions can fool ALBERT with a 100% accuracy while that on DistillBERT is lower than that on ALBERT. Accuracy on Roberta is the lowest, indicating that some obstinate substitutions might not work. That might be because ALBERT has been finetuned on a "simple" task (dataset), *i.e.*, Paraphrase (GLUE (MRPC)), compared with QA (SQuAD) and NLI (SNLI). And the "simple" task, paraphrase, does not require a deep understanding of sentences as QA and NLI do. So, these results actually suggest that it would be beneficial to finetune or pretrain models on large datasets and complex tasks.

At the same time, we notice that as shown in Tables 16, and 17, the obstinate substituitions of "Asian" and "China" are "*worst*" and "*bats*", which raises ethical concerns about the use of language models.

Moreover, we present the results of transferring the obstinate substitutions found on Roberta to GPT-3 and ChatGPT in Table 18 to complement our black-box settings in the main paper.

Task	Dataset	VERB		PRON		ADV	
		becomes (drugged)	challenged (zem)	them (amphitheater)	anything (naturalist)	simply (venomous)	almost (umu)
ALBERT	GLUE (MRPC)	4 / 4	1 / 1	97 / 97	11 / 11	8 / 8	20 / 20
DistillBERT	SNLI	8 / 11	7 / 7	1428 / 2258	267 / 334	3 / 4	139 / 204
Roberta	SQuAD	2 / 19	1 / 24	16 / 582	1 / 19	1 / 10	5 / 63

Task	Dataset	ADJ		PROPN		SCONJ	
		average (libertarian)	second (determined)	ministry (islander)	hepatitis (maclean)	if (collectors)	because (resistance)
ALBERT	GLUE (MRPC)	58 / 58	92 / 92	5 / 5	2 / 2	471 / 471	74 / 74
DistillBERT	SNLI	9 / 10	150 / 176	0 / 0	0 / 0	6241 / 7564	1510 / 1587
Roberta	SQuAD	117 / 327	283 / 593	1 / 15	1 / 1	357 / 3800	166 / 221

Task	Dataset	DET		NOUN		PUNCT	PART
		that (stefano)	another (deflect)	death (adapting)	*scientists (sunderland)*	= (appearances)	or (machine)
ALBERT	GLUE (MRPC)	820 / 820	57 / 57	70 / 70	8 / 8	30 / 30	3541 / 3541
DistillBERT	SNLI	2879 / 3146	2865 / 3238	121 / 159	58 / 63	3 / 3	84584 / 99147
Roberta	SQuAD	205 / 5007	500 / 588	260 / 405	50 / 96	15 / 18	11561 / 35235

Task	Dataset	AUX	CCONJ	ADP	NUM
		have (promised)	plus (tiger)	alongside (caliber)	1997 (malaysia)
ALBERT	GLUE (MRPC)	372 / 372	10 / 10	3 / 3	7 / 7
DistillBERT	SNLI	2800 / 3724	12 / 20	75 / 79	0 / 0
Roberta	SQuAD	2873 / 3423	12 / 14	21 / 26	28 / 50

Table 15: Experimental results (black-box) on the universality of obstinate substituitions found on ALBERT (GLUE (MRPC)).

Model	Dataset	VERB		PRON		ADV	
		training (breakthrough)	ordering (sav)	Nothing (cedes)	Everyone (Speed)	Near (IoT)	Outside (Appearance)
ALBERT	GLUE (MRPC)	9 / 9	1 / 1	0 / 0	1 / 1	6 / 6	1 / 1
DistillBERT	SNLI	383 / 439	161 / 173	85 / 86	376 / 448	24 / 24	41 / 42
Roberta	SQuAD	42 / 64	5 . 7	0 / 0	0 / 0	16 / 67	3 / 28

Model	Dataset	ADJ		PROPN		SCONJ	
		Asian (worst)	*Male (notable)*	Billy (Hom)	*Dog (Indonesian)*	While (spective)	as (Wisconsin)
ALBERT	GLUE (MRPC)	5 / 5	4 / 4	2 / 2	2 / 2	18 / 18	2758 / 2758
DistillBERT	SNLI	1279 / 1786	102 / 108	20 / 38	439 / 922	142 / 154	48409 / 59720
Roberta	SQuAD	2 / 100	0 / 5	0 / 4	11 / 35	31 / 33	26014 / 34993

Model	Dataset	DET		NOUN		PUNCT	PART
		All (TERN)	Some (Nike)	Human (Merc)	Students (tall)	. (charged)	not (settlements)
ALBERT	GLUE (MRPC)	36 / 36	21 / 21	6 / 6	0 / 0	5792 / 5792	492 / 492
DistillBERT	SNLI	256 / 342	7360 / 9592	525 / 538	152 / 212	435059 / 455983	4637 / 8410
Roberta	SQuAD	32 / 166	7 / 142	8 / 94	0 / 9	1334 / 1545	166 / 2666

Model	Dataset	AUX	CCONJ	ADP	NUM
		is (Distance)	Neither (Imperial)	At (Articles)	Seven (Character)
ALBERT	GLUE (MRPC)	2534 / 2534	3 / 3	109 / 109	9 / 9
DistillBERT	SNLI	251073 / 263000	15 / 29	287 / 303	117 / 125
Roberta	SQuAD	14157 / 35280	0 / 0	19 / 699	2 / 42

Table 16: Experimental results (black-box) on the universality of obstinate substituitions found on DistillBERT (SNLI).

Task	Dataset	VERB		PRON		ADV	
		Using (sequent)	wrote (aturation)	What (chieve)	Her (nat)	When (father)	Typically (Ezekiel)
ALBERT	GLUE (MRPC)	1 / 1	14 / 14	7 / 7	28 / 28	15 / 15	1 / 1
DistillBERT	SNLI	3 / 3	51 / 54	4 / 6	123 / 137	10 / 13	0 / 0
Roberta	SQuAD	10 / 12	11 / 376	39356 / 42319	31 / 108	3796 / 6115	0 / 4

Task	Dataset	ADJ		PROPN		SCONJ	
		American (culation)	Natural (tex)	September (esson)	*China (bats)*	Once (ATURES)	Because (Christian)
ALBERT	GLUE (MRPC)	108 / 108	1 / 1	33 / 33	22 / 22	4 / 4	4 / 4
DistillBERT	SNLI	449 / 518	1 / 2	5 / 5	51 / 143	2 / 4	2 / 2
Roberta	SQuAD	49 / 1020	15 / 17	3 / 74	151 / 370	10 / 12	36 / 41

Task	Dataset	DET		NOUN		PUNCT	PART
		All (culture)	Each (igroup)	All (culture)	Each (igroup)	? (coni)	's (ession)
ALBERT	GLUE (MRPC)	36 / 36	3 / 3	36 / 36	3 / 3	12 / 12	1077 / 1077
DistillBERT	SNLI	266 / 342	23 / 25	266 / 342	23 / 25	16 / 20	6635 / 6876
Roberta	SQuAD	72 / 166	14 / 15	72 / 166	14 / 15	87127 / 97048	7765 / 9163

Task	Dataset	AUX	CCONJ	ADP	NUM
		Was (alia)	while (Copenhagen)	Against (EO)	Three (Aim)
ALBERT	GLUE (MRPC)	58 / 58	88 / 88	6 / 6	15 / 15
DistillBERT	SNLI	31 / 34	8557 / 10292	3 / 3	7522 / 7906
Roberta	SQuAD	76 / 321	123 / 161	11 / 18	24 / 40

Table 17: Experimental results (black-box) on the universality of obstinate substituitions found on Roberta (SQuAD).

Questions	GPT-3	ChatGPT
Which program at Notre Dame offers a Master of Education degree?	the College of Education	a Master of Education
Which *Mand* at Notre Dame*atives* a Master of Education degree?	NA	a Master of Education
When did Beyoncé rise to fame?	NA	1990s
When did Beyoncé*uk benefic* fame?	Beyoncéuk benefic fame in the late 1990s as lead singer	1990s
When was the Luria-Delbruck experiment?	1943	1943
When*age* the*drawnuria*-Delbruck experiment?	1940	NA
When did antibiotics revolutinzed medicine?	1940	1928
isively did antibiotics revolutinzed medicine?	Yes	1940s

Table 18: Examples of transferring the obstinate substituitions of Roberta to GPT-3 and ChatGPT (Dec 15 Version). The words that our algorithm selects for substitution are marked as blue, while their obstinate substituitions are marked as *red*. "NA" refers to not understanding the questions.