Phrase-level Textual Adversarial Attack with Label Preservation

Anonymous ACL submission

Abstract

Generating high-quality textual adversarial examples is critical for investigating the pitfalls of natural language processing (NLP) models and further promoting their robustness. Existing attacks are usually realized through word-level or sentence-level perturbations, which either limit the perturbation space or sacrifices fluency and textual quality, both affecting the attack effectiveness. In this paper, we propose PLAT that generates adversarial samples through phrase-level perturbations. PLAT first extracts the vulnerable phrases as attack targets by a syntactic parser, and then perturbs them by a pretrained blank-infilling model. Such flexible perturbation design substantially expands the search space for more effective attacks without introducing too many modifications, and meanwhile maintains the textual fluency and grammaticality via contextualized generation using surrounding texts. Moreover, we develop a label-preservation filter leveraging the likelihoods of language models fine-tuned on each class to rule out those perturbations that potentially alter the original class label for humans. Extensive experiments and human evaluation demonstrate that PLAT has a superior attack efficiency as well as a better label consistency than strong baselines.

1 Introduction

Despite the widespread success of deep learning in natural language processing (NLP) applications, a variety of works (Wallace et al., 2019; Jia et al., 2019; Cheng et al., 2019) discovered that neural networks can be easily fooled to produce incorrect predictions, when its input text is modified by adversarial attacks that do not necessarily alter human predictions and the true meaning of the original text. Through the lens of adversarial attacks, we can allocate the weakness of models and in turn improve their reliability and robustness (Jia and Liang, 2017; Belinkov and Bisk, 2018).

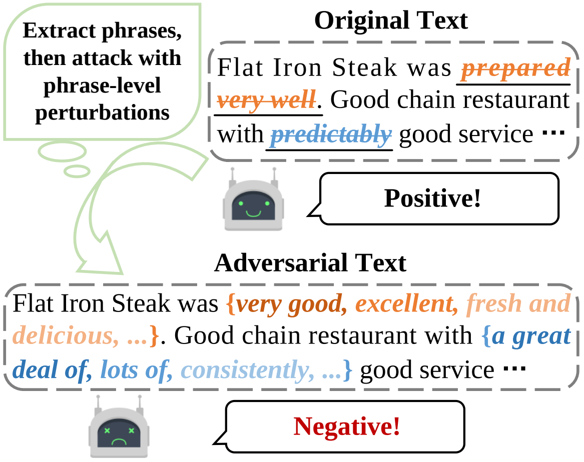

Extract phrases, then attack with phrase-level perturbations
Original Text
Flat Iron Steak was ~~prepared very well~~. Good chain restaurant with ~~predictably~~ good service ···
Positive!
Adversarial Text
Flat Iron Steak was {very good, excellent, fresh and delicious, ...}. Good chain restaurant with {a great deal of, lots of, consistently, ...} good service ···
Negative!

Figure 1: In PLAT, we extract phrases from the original text as attack targets, then use a blank-infilling model to obtain perturbation candidates and generate effective adversarial texts. Note that both target phrases and perturbations may contain one or multiple tokens.

However, generating high-quality adversarial texts is nontrivial due to the discrete nature of human language and its rigorous linguistic structures. While many efforts of previous works have been taken to generate word-level perturbations (Ren et al., 2019; Alzantot et al., 2018; Jin et al., 2020; Li et al., 2020; Garg and Ramakrishnan, 2020; Li et al., 2021) for the sake of simplicity, their attacks are restricted to independent perturbations on single words and thus cannot produce richer and more diverse forms of adversarial examples. To expand the search space for attacks, sentence-level attacks have been explored (Iyyer et al., 2018; Wang et al., 2020b,a) such as using paraphrasing, but their textual quality is usually poor due to insufficient controls or constraints on the structure and meanings of the generated texts.

To generate controllable high-quality textual adversarial examples, we propose a new phrase-level attack, PLAT, which can explore more diverse and flexible forms of perturbations than single word perturbations. Our model is able to produce phrase-level perturbations with a high success rate and preserve the textual similarity in a more controllable manner. As illustrated in Figure 1, with the help of constituency parsing, PLAT first detects and extracts the most vulnerable phrases from the text to the victim model as the attack targets. To maintain textual fluency and grammaticality, PLAT perturbs these phrases through a contextualized blank-infilling procedure by a pretrained language model. Compared to existing textual adversarial attacks, PLAT can produce more efficient and effective attacks by searching in a larger space of phrase-level perturbation. Meanwhile, PLAT delicately controls the amount of modifications so the textual meaning of the original texts can still be faithfully preserved after attacks.

Moreover, the success of attacks can be trivial if allowed to arbitrarily distort the ground-truth label or key contents. Hence, a valid attack is required to not change the ground-truth label predicted by humans. However, the semantic similarity filters widely used in existing works (Jin et al., 2020; Li et al., 2020) perform unsatisfactorily in preserving the textual meaning and even flip the gold labels, according to a recent study (Morris et al., 2020). To this end, we develop a label-preservation filter to maintain class-dependent properties such as sentiments. It is built upon the comparison of likelihoods of language models finetuned on different classes' data. Thereby, it selects the attacks that can easily fool the victim model but hardly alter the original labels.Our contributions in this paper are threefold:

- We propose a phrase-level textual adversarial attack that employs contextualized blank-infilling to generate high-quality phrase perturbations. It expands the perturbation space of word-level attacks and thus can produce more effective attacks without notably hurting the fluency and grammaticality.
- We introduce a novel label-preservation filter, which is more reliable than the widely used semantic similarity filters on generating valid adversarial examples.
- Extensive experiments demonstrate the effectiveness of PLAT on multiple text classification and natural language inference tasks, hence presenting a new robustness challenge to existing NLP models.

2 Methodology

In this section, we first formulate the problem of phrase-level textual adversarial attack. Then we elaborate on how PLAT chooses phrase candidates to attack and how to adversarially perturb them. Finally, we discuss strategies to select the most effective perturbations with label preservation.

2.1 Problem Definition

We focus on generating textual adversarial examples for classification tasks. Given a textual sequence $\mathbf{x} = x_1 x_2 \dots x_n$ with a specific attribute label y and a victim model F (assume $F(\mathbf{x}) = y$) to attack, our goal is to generate an adversarial sample $\mathbf{x}'$ by perturbing $\mathbf{x}$. A valid adversarial example $\mathbf{x}'$ can successfully trigger a wrong prediction of the victim model, i.e. $F(\mathbf{x}') \neq y$, while the human judgement on $\mathbf{x}'$ should stay unaltered as y. To achieve this goal, $\mathbf{x}'$ needs to be sufficiently similar to $\mathbf{x}$ with reasonable fluency and correct grammaticality.

2.2 Phrase-level Attack

Phrase candidates. Given a sequence $\mathbf{x}$, PLAT allocates candidates of phrases to attack from the syntactic tree extracted by a language parser (e.g., Stanford Parser, etc.). The model first traverses all constituents (nodes) in the syntactic tree in a top-down manner. If a node is identified as a phrase, i.e. tagged as *NP*, *VP*, etc., the text piece in $\mathbf{x}$ associated with all nodes in the subtree that is rooted at the current node, will be regarded as an attacking candidate. For more controllable attacks, we set a maximum depth of syntactic subtree d to restrict the length of candidate phrases so the modification to $\mathbf{x}$ is limited, hence resulting in more valid adversarial samples. Thereby, PLAT allocates a set of candidate phrases in the form of $\mathcal{A} = \{(\mathbf{a}, i, j)\}$, where i and j are the indices of the leftest and rightest token of a phrase $\mathbf{a}$ from $\mathbf{x}$.

Phrase importance. To produce more efficient attacks against the victim model F (e.g., finetuned BERT (Devlin et al., 2019)), PLAT only perturbs the phrases candidates important to the prediction (Jin et al., 2020; Ren et al., 2019). Specifically, we consider to replace a phrase $\{(\mathbf{a}, i, j)\}$ in $\mathbf{x}$ with a series of special symbol [MASK][1] with the same length as $\mathbf{a}$, which results in $\tilde{\mathbf{x}} = x_1 \dots x_{i-1}, \text{[MASK]} \dots \text{[MASK]}, x_{j+1} \dots x_n$. The importance for phrase $\mathbf{a}$ is measured by

$$I(\mathbf{a}) = P_F(y \mid \mathbf{x}) - P_F(y \mid \tilde{\mathbf{x}}),$$

[1] We empirically found this is better than single-mask replacement.

where $P_F(y|\cdot)$ is the probability of the ground truth label y predicted by F given the input text. Larger $I(\mathbf{a})$ indicates that the phrase $\mathbf{a}$ has more significant contribution to the prediction of y. PLAT manipulates each *target phrase* in candidate sets $\mathcal{A}$ by following an descending order of their importance scores. So more effective phrase-level perturbations are applied earlier for achieving minimum modifications to $\mathbf{x}$.

Phrase perturbations. To generate phrase-level adversarial perturbations, PLAT performs a blank-infilling procedure on each target phrase. Specifically, PLAT first replaces a target phrase $\mathbf{a} \in \mathcal{A}$ with a blank from index i to j, i.e.,

$$\tilde{\mathbf{x}}_{\backslash \mathbf{a}} = x_1, \ldots, x_{i-1}, \underline{\hspace{2em}}, x_{j+1}, \ldots, x_n.$$

Then a pretrained blank-infilling language model, e.g., BART (Lewis et al., 2020) or T5 (Raffel et al.), takes $\tilde{\mathbf{x}}_{\backslash \mathbf{a}}$ as the input and fills a phrase $\mathbf{b} = z_1 \ldots z_m$ into the blank conditioned on surrounding context, i.e.,

$$\tilde{\mathbf{x}}_{\mathbf{b}} = x_1 \ldots x_{i-1}, z_1 \ldots z_m, x_{j+1} \ldots x_n.$$

In contrast to paraphrasing each phrase independently, such contextualized infilling procedure can produce more fluent and grammatically correct perturbations fitting into the rest context.

For attacking each target phrase, PLAT samples N candidates of perturbed phases $\mathcal{B} = \{\mathbf{b}\}$ with varying lengths. During the generation, PLAT tends to sample tokens of higher probability at every step so that the outputs are more fluent and grammatical with the surrounding context. We keep the maximum length of perturbations not greater than the length of original phrases plus a threshold l (e.g., $|\mathbf{b}| \leq |\mathbf{a}| + l$). The most effective perturbation in $\mathcal{B}$ is then selected to replace the target phrase $\mathbf{a}$, resulting in a perturbed text $\tilde{\mathbf{x}}_{\mathbf{b}}$ (§2.3).

We apply the above phrase perturbation sequentially to all target phrases from $\mathcal{A}$[2] until (1) a valid adversarial sample $\mathbf{x}^{(t)}$ is found when perturbing the t^{th} target phrase (i.e., $F(\mathbf{x}^{(t)}) \neq y$); or (2) the maximum number of perturbations T is reached. We summarize the above procedure in Algorithm 1.

Algorithm 1 Adversarial Attack by PLAT

```
Input: Original text $\mathbf{x}$, the gold label $y$, victim model $F$, maximum number of perturbation $T$, importance score $I$, class likelihood ratio $R$, effectiveness filter score $S$.
Output: An adversarial example $\mathbf{x}'$
Extract phrase candidates from $\mathbf{x}$ to form set $\mathcal{A}$
$\mathbf{x}^{(0)} \leftarrow \mathbf{x}$
for $1 \leq t \leq T$ do
    $\mathbf{a} \leftarrow$ target phrase with highest $I$ in $\mathcal{A}$
    $\mathcal{B} \leftarrow$ set of phrases perturbations generated by blank-infilling $\tilde{\mathbf{x}}^{(t-1)}_{\backslash \mathbf{a}}$
    $\mathcal{B} \leftarrow$ filtering $\mathcal{B}$ by $R(\mathbf{x}^{(t-1)}, \mathbf{b}', y) < \delta$
    if $\mathcal{B} = \varnothing$ then $\mathbf{x}^{(t)} = \mathbf{x}^{(t-1)}$, continue
    end if
    $\mathbf{b} \leftarrow \operatorname{argmax}_{\mathbf{b}' \in \mathcal{B}} S(\mathbf{x}^{(t-1)}, \mathbf{b}')$
    $\mathbf{x}^{(t)} \leftarrow \tilde{\mathbf{x}}^{(t-1)}_{\mathbf{b}}$(replace $\mathbf{a}$ with $\mathbf{b}$ in $\mathbf{x}^{(t-1)}$)
    if $F(\mathbf{x}^{(t)}) \neq y$ then return $\mathbf{x}^{(t)}$
    end if
end for
return NONE
```

2.3 Label Preservation and Effective Perturbation.

Label preservation filter. Although existing works (Jin et al., 2020; Chen et al., 2021) usually employ a semantic similarity constraint (e.g., USE (Cer et al., 2018)) to encourage the validity of adversarial samples, it has been observed that such constraint is unreliable to preserve the textual meaning (Morris et al., 2020). Moreover, existing approaches rarely preserve class-dependent contents, e.g., sentiments, and might produce invalid adversarial examples with human-predicted labels flipped. Such a drawback is commonly observed in our human evaluation in §3.3.

To retain the class-related characteristics most critical to classification tasks, inspired by Malmi et al., 2020, PLAT directly filters phrase perturbations using likelihoods provided by class-conditioned masked language models (CMLMs). Specifically, given a sequence $\tilde{\mathbf{x}}_{\mathbf{b}} = x_1, \ldots, x_{i-1}, z_1, \ldots, z_m, x_{j+1}, \ldots, x_n$, the class-conditioned likelihood of the adversarially perturbed phrase $\mathbf{b} = z_1, \ldots, z_m$ for phrase $(\mathbf{a}, i, j)$ in $\mathbf{x}$ can be calculated as

$$L(\mathbf{x}, \mathbf{b}, y) = \prod_{k=1}^{m} P_{\text{CMLM}}\left(z_k \mid \tilde{\mathbf{x}}_{\mathbf{b}\backslash z_k}; \Theta_y\right).$$

Here, m is the length of $\mathbf{b}$, $\tilde{\mathbf{x}}_{\mathbf{b}\backslash z_k}$ is $\tilde{\mathbf{x}}_{\mathbf{b}}$ with token z_k masked, P_{CMLM} is the likelihood of z_k given $\tilde{\mathbf{x}}_{\mathbf{b}\backslash z_k}$, which is produced by a class-conditioned

[2] If a phrase $\mathbf{b}$ is perturbed, phrases that overlap $\mathbf{b}$ in the remaining phrases of $\mathcal{A}$ will be ignored.

masked language model Θ_y conditioned on class y. The conditional language model is first initialized as a pretrained model and then finetuned with the pretraining objective on all data belonging to the text's class from the dataset. Therefore, a larger likelihood indicates that $\mathbf{b}$ is more likely to match the corresponding class distribution given the surrounding context[3].

To avoid label flipping of human prediction, the phrase perturbations should enjoy a higher likelihood on the original class's distribution but a lower likelihood on other classes. This property can be measured by the following likelihood ratio:

$$R(\mathbf{x}, \mathbf{b}, y) = L(\mathbf{x}, \mathbf{b}, y) / \max_{\tilde{y} \in \mathcal{Y}, \tilde{y} \neq y} L(\mathbf{x}, \mathbf{b}, \tilde{y}),$$

where $\mathcal{Y}$ denotes the set of all classes in the task. A higher likelihood ratio suggests the perturbation is more correlated to the original label in contrast to other labels. For better label preservation, the committed phrase perturbations are required to have a likelihood ratio larger than certain threshold δ, i.e. $R(\mathbf{x}, \mathbf{b}, y) \geq \delta$ for $\mathbf{b} \in \mathcal{B}$. As shown in §3.3, our method outperforms other baselines on label-preservation.

Selection of the most effective perturbation. To generate $\mathbf{x}'$ with sufficient global textual-similarity to $\mathbf{x}$, PLAT selects target phrases of length smaller than d and restricts their perturbations' lengths to be smaller than $d + l$. Moreover, PLAT aims at utilizing minimum perturbations to perform effective adversarial attacks so the textual similarity can be preserved. Minimum perturbations can in return help maintain reasonable fluency and grammaticality of the generated texts.

To achieve the above goals, PLAT selects the most effective phrase perturbation at each step as the one that minimizes the probability of the gold label y predicted by F. We use a score to measure each phrase $\mathbf{b}$ in terms of how likely it can successfully fool the model, i.e. the negative probability of the gold label y for the original $\mathbf{x}$ associated with the perturbation $\mathbf{b}$, i.e.,

$$S(\mathbf{x}, \mathbf{b}) = -P_F(y \mid \tilde{\mathbf{x}}_{\mathbf{b}}).$$

When attacking a target phrase, PLAT only chooses one phrase perturbation $\mathbf{b} \in \mathcal{B}$ with the highest score. The resulted perturbed-sequence is retained and then used as the initial sequence for the next time of perturbation.

2.4 Discussion

A primary novelty of PLAT is the phrase-level perturbation. Compared to the widely studied word-level perturbations (Ren et al., 2019; Jin et al., 2020; Li et al., 2021) that can only independently perturb a single word every time, PLAT can perturb a text span of varying lengths by replacing it with phrases of possibly unequal lengths. Hence, it produces a more flexible attack by searching it in a larger perturbation space. Although the textual phrase-level attack has been studied by a concurrent work MAYA (Chen et al., 2021), there are several critical differences of PLAT, i.e.,

(1) The phrase-level attack by PLAT is a more general attack model that covers both word-level and phrase-level perturbations in one framework, while MAYA builds separate sub-modules for different levels of perturbations.

(2) PLAT adopts a blank-infilling strategy and leverages language models to generate phrase perturbations in a context-aware manner, leading to more fluent and grammatical adversarial examples. On the contrary, MAYA applies paraphrasing to each constituent target separately without taking its surrounding context information into account.

(3) PLAT applies several constraints and filters to the phrase perturbations for more controllable attacks and better preservation of the original textual and label information, while MAYA has no such restrictions and its generated perturbations can introduce arbitrary distortions to the original text.

3 Experiments

In this section, we first elaborate on the experimental settings and implementation details of PLAT as well as the comparisons to several baselines in §3.1. We then introduce the datasets and evaluation designs in §3.2. At the end, we summarize the main results in §3.3.

3.1 Setup

The implementation details are given as follows:

- We use pretrained BART$_{\text{base}}$ (Lewis et al., 2020) as the language model for blank-infilling to generate phrase perturbations. We sample $N = 5000$ candidates by Top-K sampling (Fan et al., 2018) as the phrase set $\mathcal{B}$, while set $d = 4$, $l = 3$ for each target phrase. In sections §4.1, We also report the

[3] In practice, we partition the whole text into multiple sentences and the likelihood P_{CMLM} for a phrase $\mathbf{b}$ is calculated locally using its corresponding sentence.

Dataset	Avg. Len	#Classes	Train	Test	Acc
Yelp	130	2	560k	38k	91.8%
AG News	46	4	120k	7.6k	94.6%
MNLI	23/11	3	392k	9.8k	83.9%
QNLI	11/31	2	105k	5.4k	91.4%

Table 1: Statistics of datasets and the performance of victim models on each dataset.

performance when using different language models.

- We use finetuned RoBERTa$_{\text{base}}$ (Liu et al., 2019) to calculate the class-conditioned likelihood for label-preserving filters. On each dataset, the model is further finetuned to optimize the pretraining objective on each sequence with a prepending special label token. We set threshold $\delta = 1$ for the filtering.
- The victim model F is an MLP classifier based on BERT$_{\text{base}}$ (Devlin et al., 2019). It takes the representation of [CLS] token for prediction and is fine-tuned on the target datasets in advance.

Baselines. We compare PLAT with three state-of-the-art textual adversarial attack models[4]:

- **Textfooler** (Jin et al., 2020): a word-level attack model, which replaces tokens with their synonyms via counter-fitting word embeddings (Mrkšić et al., 2016). USE (Cer et al., 2018) distance is used to select adversarial texts preserving the semantic similarity.
- **CLARE** (Li et al., 2021): instead of token replacement only, **CLARE** considers three word-level perturbations, replace, insert, and merge. Pretrained masked language models are used to generate perturbations and a USE semantic similarity filter is applied.
- **MAYA** (Chen et al., 2021): a multi-granularity model that attacks the input using two separate modules for word replacement and constituent paraphrasing. It employs the embedding of Sentence-BERT (Reimers and Gurevych, 2019) for semantic similarity preservation.

3.2 Datasets and Evaluation

Datasets. We investigate the following datasets for text classification and natural language inference tasks in our experiments. The statistics and performance of the victim models evaluated on each dataset are reported in Table 1.

- **Yelp Reviews** (Zhang et al., 2016): a binary sentiment classification dataset containing restaurant reviews as samples.
- **AG News** (Zhang et al., 2016): a news articles classification dataset covering four classes: *World*, *Sport*, *Business*, and *Science and Technology*.
- **MNLI** (Williams et al., 2018): a natural language inference dataset, where each sample contains a pair of sentences whose relationship is labeled as *entailment*, *neutral*, or *contradiction*. We use the *matched* test set here.
- **QNLI** (Wang et al., 2018): a natural language inference dataset based on the question answering corpus SQuAD (Rajpurkar et al., 2016). Each sample contains a context and a question labeled as *entailed* or *not entailed*.

All attacks will be conducted on 1000 instances randomly drawn from test sets. For tasks on a pair of sentences, we attack the longer sentence.

Evaluation metrics. We evaluate models using the following automatic metrics:

- **Attack success rate (ASR)**: the percentage of successful adversarial attacks that trigger wrong predictions of the victim model.
- **Editing distance (DIS)** (Navarro, 2001): the normalized Levenshtein distance that measures the minimum amount of word editing required to transform the original text to the adversarial one. It measures the modification rate of an adversarial sample.
- **BLEU** (Papineni et al., 2002): the BLEU score between an adversarial sample and its corresponding original sample is used to measure their n-gram overlap (textual similarity).
- **Perplexity (PPL)**: a pretrained GPT2$_{\text{base}}$ (Radford et al., b) is used to calculate the PPL of adversarial texts, which reflects the fluency as suggested by (Kann et al., 2018; Zang et al., 2020a).
- **Grammar error (GER)**: Following Zang et al., 2020b, we employ LanguageTool [5] to calculate the average number of grammar errors newly introduced by adversarial samples.

We only evaluate the last four metrics on the successful attacks against the victim model.

3.3 Main Results

Table 2 summarizes the main experimental and comparison results. Overall, PLAT consistently achieves better attack success rate and perplexity performance across all datasets. We attribute this to the flexible phrase-level perturbations generated using contextual information. Compared with a

[4] All results are obtained by running their released code.

[5] https://www.languagetool.org/

Dataset	Yelp (PPL = 51.5)					AG News (PPL = 62.8)				
Model	**ASR↑**	**DIS↓**	**BLEU↑**	**PPL↓**	**GER↓**	**ASR↑**	**DIS↓**	**BLEU↑**	**PPL↓**	**GER↓**
Textfooler	94.5	0.11	0.80	101.1	0.73	65.5	0.29	0.52	339.0	1.43
CLARE	97.3	**0.07**	**0.88**	65.2	**0.08**	68.0	**0.09**	**0.86**	97.2	**-0.03**
MAYA	97.0	0.43	0.44	78.9	5.23	94.2	0.64	0.25	168.6	4.30
PLAT	**98.4**	0.17	0.78	**56.8**	0.33	**95.7**	0.34	0.58	**80.3**	0.58
Dataset	MNLI (PPL = 60.9)					QNLI (PPL = 46.0)				
Model	**ASR↑**	**DIS↓**	**BLEU↑**	**PPL↓**	**GER↓**	**ASR↑**	**DIS↓**	**BLEU↑**	**PPL↓**	**GER↓**
Textfooler	58.6	0.15	0.71	159.0	0.67	57.8	0.19	0.63	164.5	0.62
CLARE	86.2	**0.10**	**0.82**	82.7	**0.09**	82.6	**0.15**	**0.74**	75.9	**0.03**
MAYA	92.8	0.40	0.49	104.7	2.20	78.6	0.40	0.48	101.4	2.90
PLAT	**96.6**	0.20	0.74	**62.1**	0.23	**92.4**	0.25	0.68	**51.3**	0.06

Table 2: Adversarial attack performance of PLAT and baselines on four datasets, in terms of attack success rate (ASR), editing distance (DIS), BLEU, perplexity (PPL), and increased grammar errors(GER). **Bold values** indicate the best performance for each metric. ↓(↑) indicates the higher (lower) the better. Note that phrase-level attacks naturally introduce more modifications than word-level attacks so they are not directly comparable on DIS and BLEU metrics in the table.

Metric	PLAT	equal	CLARE
Meaning preservation	39.8	\	33.3
Label preservation	**77.1**	\	49.8
Fluency and grammaticality	33.1	32.3	34.6
Metric	PLAT	equal	MAYA
Meaning preservation	39.8	\	30.2
Label preservation	**77.1**	\	53.5
Fluency and grammaticality	**45.0**	29.0	26.0

Table 3: Human evaluation performance in percentage on the Yelp dataset.

	Yelp	AG News	MNLI	QNLLI
1^{st}	NP / 39.9%	NP / 53.7%	NP / 57.6%	NP / 58.6%
2^{nd}	ADJP / 17.5 %	NNP / 27.0%	PP / 12.4%	NNP / 16.4%
3^{rd}	VP / 16.6%	PP / 9.5%	NNP / 27.1%	PP / 13.4%

Table 4: Top-3 phrase tags of attack phrases and their percentages on different datasets by PLAT.

phrase-level model MAYA, our method is significantly better on modification rates, BLEU, and grammar scores. Hence, despite not using semantic similarity constraints, PLAT is more controllable than MAYA as we confine the modification ranges and generate perturbations by contextual blank-infilling. While word-level attacks naturally introduce fewer perturbations and thus have better textual similarity and grammaticality, its perturbation space is small and results in lower success rates. On the contrary, PLAT achieves the highest success rate while on the par with word-level attacks on textual similarity and grammaticality, hence achieving a sweet spot among all metrics.

Human evaluation. We further conduct human evaluations on Yelp dataset with 100 randomly selected successful attacks produced by PLAT, CLARE, and MAYA. We evaluate these attacks in three aspects: (1) **Meaning preservation**: whether the attacks preserve the original meaning or not; (2) **Label preservation**: whether the modifications contradict the original sentiment or not; (3) We evaluate **fluency and grammaticality** via pairwise comparison: for each instance, we pair PLAT's attack with one by CLARE or MAYA. The human annotators are asked to either select the better one or rate them as equal. We average 6 responses for each sample. More details are in Appendix D.

As shown in Table 3, PLAT significantly outperforms CLARE and MAYA in terms of label consistency, i.e., 77.1% vs. 49.8%(CLARE) or 53.5%(MAYA). This demonstrates the benefit of our proposed label-preservation filter using class-conditional likelihoods. It's worth noting that all models struggle on preserving the textual meaning and less than 40% of samples can retrain their original meaning. This is consistent with Morris et al., 2020 in that semantic similarity metrics fail to maintain the actual meaning. On the fluency and grammaticality, PLAT is comparable to CLARE but is much better than MAYA (45% vs. 26%), since our context-aware blank-infilling is superior to paraphrasing each text piece independently. Finally, Table 5 compares adversarial attacks crafted by our model and other baselines. More case studies are provided in Appendix B.

Vulnerable phrase types. We also analyze the three mostly attacked phrase types on each dataset. As shown in Table 4, noun phrases (NP) are the most vulnerable phrases over all datasets (more

Yelp (Negative)	The quality of the food has really plummeted over the past year. We use to love coming her to get the creamy clam chowder, not its watery and gross.
TextFooler (Positive)	The quality of the food has really **engulfed** over the past year. We use to love coming her to get the creamy clam chowder, not its watery and gross.
CLARE (Positive)	The quality of the food has really **soared** over the past year . We use to love coming her to get the creamy clam chowder, not its watery and gross.
MAYA (Positive)	The quality of the food has really **last year was a big one for the fall**. We use to love coming her to get the creamy clam chowder, not its watery and gross.
PLAT (Positive)	The quality of the food has **also somewhat** plummeted over the past year. We use to love coming her to get the creamy clam chowder, not its watery and gross.

Table 5: Adversarial examples generated by different models on Yelp dataset, perturbations are colored.

Module	**ASR↑**	**DIS↓**	**BLEU↑**	**PPL↓**	**GER↓**
PLAT	98.4	0.17	0.78	**56.8**	0.33
w/o PHRASE-LEVEL	97.7	**0.08**	**0.85**	69.0	**0.14**
w ALL CONSTITUENTS	98.2	0.16	0.79	58.1	0.29
BERT$_{\text{base}}$ likelihood	98.5	0.17	0.78	**56.8**	0.30
T5$_{\text{base}}$ infilling	98.4	0.16	0.79	61.4	0.41
GPT-2$_{\text{small}}$ infilling	**98.7**	0.16	0.79	61.4	0.42

Table 6: Results of the ablation study on Yelp dataset.

than 50% on three datasets), while preposition phrases (PP) and proper noun phrases (NNP) are also commonly vulnerable.

4 Analysis

In this section, we conduct detailed analyses of PLAT, including ablation study (§4.1), discussion of controllability in blank infilling (§4.2), and robust defense model attacks (§4.3).

4.1 Ablation Study

We evaluate the effectiveness of each key component in PLAT based on the 1,000 random Yelp samples §3.2. We first study the phrase-level perturbation by replacing it with the word-level replacement used in Textfooler (*w/o* PHRASE-LEVEL). As Table 6 shows, phrase-level perturbation has a larger attack search space which leads to better attack success rate and perplexity. It also shows that attacking the constituents that are labeled as phrases is more effective than attacking all possible constituents. This is probably because phrases contain more critical and clear information to attack in classification tasks. In addition, we have not observed a significant difference between using BERT$_{\text{base}}$ and RoBERTa$_{\text{base}}$ for class-conditioned likelihood calculation, probably due to their similar architectures and shared knowledge. Finally, we comparing different blank-infilling methods using pretrained BART$_{\text{base}}$ (PLAT), pretrained T5$_{\text{base}}$ (Raffel et al.) and finetuned GPT-2$_{\text{small}}$ (Donahue et al., 2020). Empirically, BART$_{\text{base}}$ achieves the best overall performance.

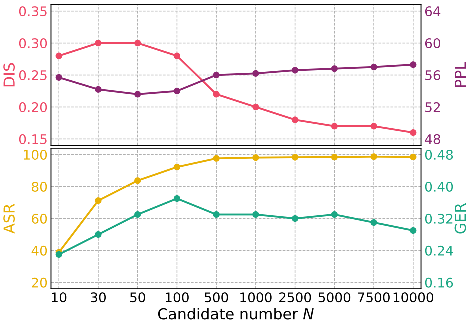


Figure 2: **ASR**, **GER**, **DIS** and **PPL** results by controlling different candidates numbers N in PLAT.

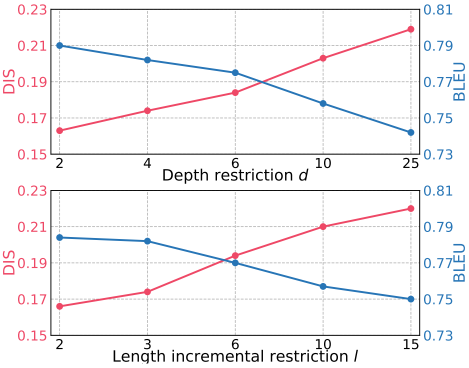


Figure 3: **DIS** and **BLEU** results by varying depth restrictions d (upper) and length increments l (bottom).

4.2 Sensitivity and Controllability Analysis

In this section, we investigate how the outputs can be controlled by hyper-parameters in PLAT on Yelp dataset. We first study how the generated samples can be impacted by varying the candidate number N in blank-infilling. As shown in Figure 2, the success rate (**ASR**) increases with the increase of N but starts to saturate when $N \geq 500$, while the grammar errors (**GER**) stay quite consistent. Meanwhile, the edit distance (**DIS**) drops significantly but the perplexity (**PPL**) only increases slightly when N increases. Based on these observations, we choose $N = 5000$ in our experiments for the best trade-off among these aspects.

Method	ASR$\uparrow$	DIS$\downarrow$	BLEU$\uparrow$	PPL$\downarrow$	GER$\downarrow$
TextFooler	64.2	0.23	0.59	185.5	1.39
CLARE	92.5	**0.12**	**0.81**	76.1	**0.15**
MAYA	98.6	0.57	0.33	82.1	3.51
PLAT	**99.2**	0.28	0.67	**55.7**	0.39

Table 7: Results of PLAT and baselines attacking the robust defense model TAVAT on Yelp dataset.

In addition, we explore how the syntactic tree depth d in target phrases and the incremental length l for perturbed phrases can affect the modification degree. In Figure 3, with the increase of d or l, more modifications are introduced and the textual similarity decreases, which undermines the validity of adversarial samples, as larger d and l augment the attacking space with longer but unnecessary perturbations. Hence, we choose relatively small d and l to ensure the attack is more controllable. Note that both d and l exhibit a slight impact on the success rate (Detailed results in Appendix C).

4.3 Attacking Robust Defense Model

In this section, we examine whether existing robust defense models can defend PLAT attack which is beyond word-level perturbations. We apply a robust BERT$_{\text{base}}$ defense model trained via TAVAT (Li and Qiu, 2021) to defense the attacks from PLAT and baseline models, which is designed for word-level attacks. Comparing Table 7 with Table 2, both the editing distance and BLEU get worse when attacking the defense model, showing that the defense model is harder to attack. Meanwhile, two word-level attacks have a significant attack success rate drop, e.g., 94.5% to 64.2% on TextFooler. On the contrary, PLAT still can achieve the best 99.2% attack rate with sufficient textual similarity and grammar errors, outperforming MAYA on every aspect. This suggests that PLAT raises a new robustness issue on current defense models.

5 Related Work

Textual Adversarial Attack Growing interest is devoted to generating textual adversarial samples via perturbation at various levels. Some early works use misspelling tokens in character-level (Liang et al., 2018; Ebrahimi et al., 2018; Li et al., 2019), but they can be easily defended by spell checking tools (Pruthi et al., 2019; Zhou et al., 2019; Jones et al., 2020). Recent mainstream of studies try to misguide models via word-level perturbations, e.g., synonym/semantic neighbor substitution (Alzantot et al., 2018; Jin et al., 2020; Ren et al., 2019; Zhang et al., 2019), replacement by masked language models (Li et al., 2020; Zhang et al., 2019), or combing operations like insertion and merge (Li et al., 2021). These methods usually attempt to preserve the semantic similarity for better fluency and grammaticality, but their perturbations are limited to independent single words. Sentence-level attacks have also been studied to generate new texts via paraphrasing or GAN-based generation (Iyyer et al., 2018; Wang et al., 2020b; Zou et al., 2020; Wang et al., 2020a), but their drastic modifications on the text structure make it harder to maintain a satisfactory textual quality. Very recently, phrase-level perturbations are considered in evaluating syntactic parsing (Zheng et al., 2020), or involved in a multi-granularity textual attack model (Chen et al., 2021) MAYA. Unlike MAYA, PLAT only focuses on unified phrase-level attacks, which require simpler and fewer modifications while benefiting better performance.

Blank Infilling Large-scale pretrained language models such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) have shown their capability of filling masked single tokens (Wang and Cho, 2019; Ghazvininejad et al., 2019) but they cannot handle variable-length masks. Although autoregressive generative models such as GPT (Radford et al., a,b) can produce output with arbitrary length, they only condition information from a single direction. To enable GPT models to fill in blanks, Donahue et al. proposed to finetune them with sequences concatenating manually-masked texts and missing texts. Recently, autoencoder-decoder models such as T5 (Raffel et al.) and BART (Lewis et al., 2020) trained using infilling losses make it possible to fill the blanks within the context in a more flexible form (Shen et al., 2020).

6 Conclusion

We present a new phrase-level textual adversarial attack, PLAT, which produces richer and higher-quality phrase-level perturbations than the widely studied word-level attacks. It utilizes contextualized blank-infilling to generate perturbations by a pretrained language model and thus well preserves the textual similarity, fluency, and grammaticality. We additionally develop a label-preservation filter to keep the ground-truth labels intact. Extensive experiments show the effectiveness of PLAT and its advantages over baselines on different NLP tasks.

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 EMNLP 2018*, pages 2890–2896.

Yonatan Belinkov and Yonatan Bisk. 2018. Synthetic and natural noise both break neural machine translation. In *Proceedings of ICLR 2018*.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Yangyi Chen, Jin Su, and Wei Wei. 2021. Multi-granularity textual adversarial attack with behavior cloning. In *Proceedings of EMNLP 2021*, pages 4511–4526.

Yong Cheng, Lu Jiang, and Wolfgang Macherey. 2019. Robust neural machine translation with doubly adversarial inputs. In *Proceedings of of ACL 2019*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT 2019*, pages 4171–4186.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. In *Proceedings of ACL 2020*, pages 2492–2501.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. Hotflip: White-box adversarial examples for text classification. In *Proceedings of ACL 2018*, pages 31–36.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of ACL 2018*, pages 889–898.

Siddhant Garg and Goutham Ramakrishnan. 2020. BAE: BERT-based adversarial examples for text classification. In *Proceedings of EMNLP 2020*, pages 6174–6181.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019. Mask-predict: Parallel decoding of conditional masked language models. In *Proceedings of EMNLP-IJCNLP 2019*, pages 6112–6121.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of NAACL-HLT 2018*, pages 1875–1885.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of EMNLP 2017*, pages 2021–2031.

Robin Jia, Aditi Raghunathan, Kerem Göksel, and Percy Liang. 2019. Certified robustness to adversarial word substitutions. In *Proceedings of EMNLP-IJCNLP 2019*, pages 4129–4142.

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 2020*, pages 8018–8025.

Erik Jones, Robin Jia, Aditi Raghunathan, and Percy Liang. 2020. Robust encodings: A framework for combating adversarial typos. In *Proceedings of ACL 2020*, pages 2752–2765.

Katharina Kann, Sascha Rothe, and Katja Filippova. 2018. Sentence-level fluency evaluation: References help, but can be spared! In *Proceedings of CoNLL 2018*, pages 313–323.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of ACL 2020*, pages 7871–7880.

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and William B Dolan. 2021. Contextualized perturbation for textual adversarial attack. In *Proceedings of NAACL-HLT 2021*, pages 5053–5069.

J Li, S Ji, T Du, B Li, and T Wang. 2019. Textbugger: Generating adversarial text against real-world applications. In *Proceedings of NDSS 2019*.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial attack against BERT using BERT. In *Proceedings of EMNLP 2020*, pages 6193–6202.

Linyang Li and Xipeng Qiu. 2021. Token-aware virtual adversarial training in natural language understanding. In *Proceedings of AAAI 2021*, pages 8410–8418.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. 2018. Deep text classification can be fooled. In *Proceedings of IJCAI 2018*, pages 4208–4215.

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 preprint arXiv:1907.11692*.

Eric Malmi, Aliaksei Severyn, and Sascha Rothe. 2020. Unsupervised text style transfer with masked language models. In *Proceedings of EMNLP 2020*, pages 8671–8680.

John Morris, Eli Lifland, Jack Lanchantin, Yangfeng Ji, and Yanjun Qi. 2020. Reevaluating adversarial examples in natural language. In *Proceedings of EMNLP-Findings 2020*, pages 3829–3839.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gasic, Lina M Rojas Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In *Proceedings of NAACL-HLT 2016*, pages 142–148.

Gonzalo Navarro. 2001. A guided tour to approximate string matching. *ACM computing surveys (CSUR)*, 33(1):31–88.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL 2002*, pages 311–318.

Danish Pruthi, Bhuwan Dhingra, and Zachary C Lipton. 2019. Combating adversarial misspellings with robust word recognition. In *Proceedings of ACL 2019*, pages 5582–5591.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. a. Improving language understanding by generative pre-training. *Open AI Blog*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. b. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of EMNLP 2016*, pages 2383–2392.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of EMNLP-IJCNLP 2019*, pages 3982–3992.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of ACL 2019*, pages 1085–1097.

Tianxiao Shen, Victor Quach, Regina Barzilay, and Tommi Jaakkola. 2020. Blank language models. In *Proceedings of EMNLP 2020*, pages 5186–5198.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing nlp. In *Proceedings of EMNLP-IJCNLP 2019*, pages 2153–2162.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a markov random field language model. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 30–36.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355.

Boxin Wang, Hengzhi Pei, Boyuan Pan, Qian Chen, Shuohang Wang, and Bo Li. 2020a. T3: Tree-autoencoder regularized adversarial text generation for targeted attack. In *Proceedings of EMNLP 2020*, pages 6134–6150.

Tianlu Wang, Xuezhi Wang, Yao Qin, Ben Packer, Kang Li, Jilin Chen, Alex Beutel, and Ed Chi. 2020b. CAT-gen: Improving robustness in nlp models via controlled adversarial text generation. In *Proceedings of EMNLP 2020*, pages 5141–5146.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of NAACL-HLT 2018*, pages 1112–1122.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020a. Word-level textual adversarial attacking as combinatorial optimization. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020b. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings ofACL 2020*, pages 6066–6080.

Huangzhao Zhang, Hao Zhou, Ning Miao, and Lei Li. 2019. Generating fluent adversarial examples for natural languages. In *Proceedings of ACL 2019*, pages 5564–5569.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2016. Character-level convolutional networks for text classification.

Xiaoqing Zheng, Jiehang Zeng, Yi Zhou, Cho-Jui Hsieh, Minhao Cheng, and Xuan-Jing Huang. 2020. Evaluating and enhancing the robustness of neural network-based dependency parsing models with adversarial examples. In *Proceedings of ACL 2020*, pages 6600–6610.

Yichao Zhou, Jyun-Yu Jiang, Kai-Wei Chang, and Wei Wang. 2019. Learning to discriminate perturbations for blocking adversarial attacks in text classification. In *Proceedings of EMNLP-IJCNLP 2019*, pages 4904–4913.

Wei Zou, Shujian Huang, Jun Xie, Xinyu Dai, and Jiajun Chen. 2020. A reinforced generation of adversarial examples for neural machine translation. In *Proceedings of ACL 2020*, pages 3486–3497.

A Implementation Details

A.1 Details of PLAT

Basic infrastructure We use PyTorch as the backbone of our implementation, along with Huggingface-Transformers[6] for the implementation of victim models and likelihood estimation models, while Fairseq for the implementation of blank-infilling model BART[7]. We list the hyperparameters used in our model in Table 8, all of them are determined empirically based on both attack success rate and textual quality. It takes about 160 minutes to generate 100 adversarial samples on Yelp dataset using PLAT on a single NVIDIA GTX 1080 Ti GPU.

Select phrase candidates. We use the parser from Stanford CoreNLP [8] toolkit for the syntactic tree parsing. We consider parsed nodes with the syntactic tags in Table 9 as the possible root node of a phrase.

Depth restriction of phrase syntactic tree d	4
Length incremental restriction for substitutions l	3
Maximum perturbation number T	11
Likelihood ratio filter threshold δ	1
Substitution candidates number N	5000

Table 8: All hyperparameters used in PLAT.

Tags	ADJP, ADVP, CONJP, NP, NNP, PP, QP, VP, WHADJP, WHADVP, WHNP, WHVP

Table 9: Syntactic tags that will be selected as the root of a phrase.

Blank filling with language model. In our default setting, we use directly apply the original BART$_{\text{base}}$ model [9] with 6 encoder and decoder layers and 140M parameters. During infilling, the target phrase $\mathbf{a}$ will be replaced with a special symbol "<mask>", then the model will fill this blank with variable-length context. The Top-k sampling strategy is used during generation, where we set $k = 50$ and repeat this procedure several times to collect enough phrase substitution candidates. Since BART$_{\text{base}}$ implement blank filling via reconstructing the whole sentence, where the text excluding the blank part after filling may not be the same as the original one, we simply retain the filled texts with the same texts excluding the blank.

In the variations with other blank-infilling models, we use GPT-2$_{\text{small}}$ model [10] with 124M parameters or T5$_{\text{base}}$ model [11] with 220M parameters, both have 12 layers. Since the original GPT-2$_{\text{small}}$ model is not suitable for blank infilling, we enable GPT-2 model to implement this task by finetuing on Yelp training corpus using method proposed by Donahue et al., running the code provide by the authors [12].

Dataset	PPL before	PPL after
Yelp	11.44	6.29
AG News	9.33	3.64
MNLI	6.16	3.74
QNLI	5.14	5.00

Table 10: Perplexities of our finetuned masked language models for likelihood estimation, before and after fine-tuning on the validation set of each dataset (prepending our predefined special label token).

Models for calculating likelihood. We apply RoBERTa$_{\text{base}}$[13] model fine-tuned on the corresponding training set of each attack dataset in this stage, which has 12 layers and 125M parameter. Then we fine-tune the label-conditioned masked language models on different datasets as follows to make them better fit the specific domain.

- Classification datasets (Yelp, AG News): Since each sample is usually long, we split a sample into several short sentences as the input for fine-tuning. To avoid the conditions that some short sentences may be irrelevant or contradict the overall label y, we employ a classifier to make predictions on these sentences and only remain sentences with the confidence value of y higher than 0.99. Such a sentence along with the special label token "*<Label>*" corresponding to the overall label y will form a new sample for fine-tuning, whose input format is "*<Label> Sentence*".
- NLI datasets (MNLI, QNLI): samples in these datasets are usually a pair of short sentences, so we will not split them. The input format for these datasets is "*<Label> SentenceA </s> </s> SentenceB*", where "*</s></s>*" is the separation symbol in RoBERTa.

Then we will randomly mask some tokens in these

[6] https://github.com/huggingface/transformers

[7] https://github.com/pytorch/fairseq

[8] https://stanfordnlp.github.io/CoreNLP/

[9] https://dl.fbaipublicfiles.com/fairseq/models/bart.base.tar.gz

[10] https://huggingface.co/gpt2

[11] https://huggingface.co/t5-base

[12] https://github.com/chrisdonahue/ilm

[13] https://huggingface.co/roberta-base

samples to fine-tune a masked language model conditioned on labels, the batch size is 8, and the learning rate is $5e^{-5}$ with AdamW optimizer. The PPL before and after fine-tuning is shown in Table 10, demonstrating the effectiveness of this procedure.

When predicting the likelihood of a perturbation, we will concatenate the label of the original sample with the masked perturbed sequence as the input, similar to samples in fine-tuning. When attacking Yelp and AG News datasets, we also only use the sub-sentence containing the perturbation, rather than the whole text.

Metrics. To obtain the editing distance (DIS) metric, we utilize the open-source tool[14] to calculate it in token-level, i.e. how many words need to be edited to transform a text into another one and then normalized by the text length. In addition, we employ Toolkit in NLTK[15] to calculate BLEU metrics between adversarial samples and the corresponding original samples.

A.2 Details of Victim Models

BERT models. All BERT victim models are based on BERT$_{\text{base}}$[16], which contains 110M parameters with 12 layers. A linear layer is added for classification, which takes the representation of "[CLS]" token in the head of a sequence as the input. We then fine-tune victim models on each dataset using batch size 32 and the learning rate $1e^{-4}$ for 3 epochs. The model with the best performance after each epoch on the corresponding dev set will be saved and used as the victim model F on each dataset.

Train robust models using TAVAT. The robust models are also based on BERT$_{\text{base}}$ with a linear layer added for classification. We finetune the model using an adversarial training method TAVAT proposed by Li and Qiu which is a token-level gradient accumulation of perturbations, by running code provided by the authors [17] with all default hyper-parameters. During finetuning, perturbations guided by gradient are applied to the embedding space and models are trained using these perturbed data.

A.3 Details of baseline MAYA

MAYA has three variants: MAYA, MAYA$_\pi$ and MAYA$_{\text{bt}}$. We select MAYA as our baseline since overall it obtains the best attack success rate and perplexity performance.

A.4 Possible Limitations of Our Model

The label-preservation filter in our PLAT model utilizes label-conditioned masked language models, which need to be fine-tuned on a labeled corpus with sufficient data. Therefore, the performance of PLAT may drop on datasets that have limited number of labeled samples. In addition, it takes about 1 minute for our model to generate one adversarial sample using BERT as the victim model, so PLAT is not applicable for conditions with low computational resources.

B Additional Qualitative Samples

We introduce some additional adversarial samples generated by our PLAT model, along with three baselines, TextFooler, CLARE, MAYA, on four datasets, Yelp, AG News, MNLI, QNLI, in Table 11, Table 12, and Table 13.

[14] https://github.com/roy-ht/editdistance

[15] https://www.nltk.org/_modules/nltk/translate/bleu_score.html

[16] https://huggingface.co/bert-base-uncased

[17] https://github.com/LinyangLee/Token-Aware-VAT

Yelp (Positive)	Excellent food at this out of the way place. Portions very large and fresh. I want to try everything on the menu. Plan to go back every weekend until I've tried all menu items. Coffee was also delicious and friendly servers
TextFooler (Negative)	Outstanding foods at this out of the way place. Portions very large and mild. I want to dabbled whatsoever on the menu. Plan to go back all monday until I've attempted all menu items. Coffee was also peachy and friendly servers
CLARE (Negative)	Incredible food at this out of control place. Portions plentiful and plentiful. I want to try something on the menu. Plan to go back mid weekend until I've tried all menu items . Coffee was fairly comforting and friendly servers
MAYA (Negative)	The food at this out of the way place . Portions very large and expensive. I want to try everything on the menu. Plan to go back every weekend until I've tried all menu items. Coffee was also cheap and friendly servers
PLAT (Negative)	I had nothing but fun at this out of the way place. Portions very large and fresh. I want to try everything on the menu. Plan to go back every weekend until I've tried all menu items. Coffee was also delicious and friendly servers.
Yelp (Positive)	I love this place. I love everything there except the kabsa rice but that's just me. Burgers are good. They pile on the veggies. Owner is nice. Freshly made food always has my mouth watering .
TextFooler (Negative)	I aime this place. I love everything there except the kabsa rice but that's just me. Burgers are good. They pile on the veggies. Owner is nice. Freshly made food always has my mouth watering.
CLARE (Negative)	I hate this place. I love everything there except the kabsa rice but that's just me. Burgers are good. They pile on the veggies. Owner is nice. Freshly made food always has my mouth watering.
MAYA (Negative)	I know this place. I love everything there except the kabsa rice but that' s just me. Burgers are good. They pile on the veggies. Owner is nice. Freshly made food always has my mouth watering.
PLAT (Negative)	I can't recommend Aptopia enough. I love everything there except the kabsa rice but that's just me. Burgers are good. They pile on the veggies. Owner is nice. Freshly made food always has my mouth watering.
AG News (Sci-tech)	Scientists Discover Ganymede has a Lumpy Interior. Jet Propulsion Lab–Scientists have discovered irregular lumps beneath the icy surface of Jupiter's largest moon, Ganymede. These irregular masses may be rock formations, supported by Ganymede's icy shell for billions of years...
TextFooler (World)	Researchers Unmask Deimos has a Lumpy Indoors. Jet Rotor Laboratories–Searchers have discovered irregular clods into the icy surface of Juniper's largest moon, Jupiter. These irregular masses maggio be rock formations, contributions by Enceladus's icy shell for billions of years...
CLARE (Business)	Scientists Know Ganymede has a Lumpy Interior . Credit Jet Propulsion Lab– Featured Scientists have discovered irregular lumps beneath the icy surface of Jupiter 's largest moon , Ganymede . These irregular masses may be rock formations , supported by Ganymede 's icy shell for billions of years ...
MAYA (World)	Scientists Discover Ganymed has a Lumpy Interior. Scientists have discovered irregular lumps under the icy surface of jupiter's largest moon.. These irregular masses may be rock formations, supported by ganymede's icy shell for billions of years...
PLAT (World)	Scientists Discover Ganymede has a Lumpy Interior. JPL-Caltech STOCKHOLM–Scientists have discovered irregular lumps beneath the icy surface of Jupiter's largest moon, Ganymede. These irregular masses may be rock formations, supported by Ganymede's icy shell for billions of years...
AG News (Sport)	Giddy Phelps Touches Gold for First Time. Michael Phelps won the gold medal in the 400 individual medley and set a world record in a time of 4 minutes 8. 26 seconds.
Textfooler (World)	Dazzled Phelps Hits Gold for Premiere Time. Michael Phelps won the gold trophy in the 400 personal medley and set a world record in a hours of 4 record 8. 26 seconds.
CLARE (World)	Giddy Phelps Touches Gold for First Time. Michael Phelps won the gold medal in the 400 individual medley and set a world record in a time of 4 minutes 8 ...
MAYA (World)	Giddy Phelps Touches Gold for First Time. Michael Phelps the gold medal in the 400 individual medley was won by him in a world record time of 4 minutes 8 seconds..
PLAT (World)	Swimmers: Phelps Touches Gold for First Time. Michael Phelps won the gold medal in the 400 individual medley and set a world record in a time of 4 minutes 8.26 seconds.

Table 11: Adversarial examples generated by different models on Yelp and AG News dataset, perturbations are colored.

MNLI **(Neutral)**	**Premise** The last politician to propose making driving more expensive was Al Gore, who fought to include a small energy tax–which would have included gasoline–in the Clinton administration's 1993 economic plan. **Hypothesis** Al Gore is still making proposals for making driving more expensive.
TextFooler **(Contradiction)**	**Premise** The last policies to propose making driving more expensive was Al Gore, who fought to include a small energy tax–which would have included gasoline–in the Clinton administration's 1993 economic plan. **Hypothesis** Al Gore is still making proposals for making driving more expensive.
CLARE **(Contradiction)**	**Premise** The last politician to propose making driving more expensive was Al Gore, who moved to include a small energy tax–which would have included gasoline–in the Clinton administration's 1993 economic plan. **Hypothesis** Al Gore is still making proposals for making driving more expensive.
MAYA **(Contradiction)**	**Premise** The last politician propose to make driving more expensive was AI Gore , who fought to include a small energy tax–which would have included gasoline–in the clinton administration's 1993 economic plan . **Hypothesis** Al Gore is still making proposals for making driving more expensive.
PLAT **(Contradiction)**	**Premise** The last politician to propose making driving more expensive was his predecessor Senator Al Gore, who fought to include a small energy tax–which would have included gasoline–in the Clinton administration's 1993 economic plan. **Hypothesis** Al Gore is still making proposals for making driving more expensive.
MNLI **(Entailment)**	**Premise** So uh but but uh it runs fine all you have it's just very thirsty if I just keep the oil in it seems to be okay but you know that's a sign that I'm going to have to do something sooner or later **Hypothesis** It runs well, but I think I might have to do some work on it.
TextFooler **(Neutral)**	**Premise** So uh but but uh it runs fine all you have it's just very thirsty if I just keep the petroleum in it seems to be okay but you know that's a sign that I'm going to have to do nothings shortly or later **Hypothesis** It runs well, but I think I might have to do some work on it.
CLARE **(Neutral)**	**Premise** So uh but but uh it runs fine all you have it's just very thirsty if I just drink the oil in it seems to be okay but you know that's a sign that I'm going to have to do nothing sooner or later **Hypothesis** It runs well, but I think I might have to do some work on it.
MAYA **(Neutral)**	**Premise** So uh but but uh it runs fine all you have it's just very thirsty if it seems to be okay , but i'm going to have to do something soon or later. **Hypothesis** It runs well, but I think I might have to do some work on it.
PLAT **(Neutral)**	**Premise** So uh but but uh it runs fine all you have it's just very thirsty if I just keep it that's all I got in it seems to be okay but you know that's a sign that I'm going to have to do something sooner or later **Hypothesis** It runs well, but I think I might have to do some work on it.

Table 12: Adversarial examples generated by different models on MNLI dataset, perturbations are colored.

QNLI **(Entailment)**	**Premise** What are some of the sets or ideals most school systems follow? **Hypothesis** Such choices include curriculum, organizational models, design of the physical learning spaces (e.g. classrooms), student-teacher interactions, methods of assessment, class size, educational activities, and more.
TextFooler **(Not Entailment)**	**Premise** What are some of the sets or ideals most school systems follow? **Hypothesis** Such choices include curriculum, organizes storyboards, design of the tangible learning spaces (e.g. classrooms), student-teacher interactions, methods of assessment, class size, educational activities, and more.
CLARE **(Not Entailment)**	**Premise** What are some of the sets or ideals most school systems follow? **Hypothesis** Such choices affect curriculum, organizational models, design of the physical learning spaces (e.g. classrooms), student-teacher interactions, methods of assessment, class size, educational activities, and more.
MAYA **(Not Entailment)**	**Premise** What are some of the sets or ideals most school systems follow ? **Hypothesis** Such choices the design of the physical learning spaces should include curriculum, organizational models, and methods of assessment.., and class size.., and educational activities..
PLAT **(Not Entailment)**	**Premise** What are some of the sets or ideals most school systems follow ? **Hypothesis** Such choices include curriculum, use of a teacher's manual , design of the physical learning spaces (e.g. classrooms), student-teacher interactions, methods of assessment, class size, educational activities, and more.
QNLI **(Entailment)**	**Premise** What to the migrating birds usually follow? **Hypothesis** These routes typically follow mountain ranges or coastlines, sometimes rivers, and may take advantage of updrafts and other wind patterns or avoid geographical barriers such as large stretches of open water.
Textfooler **(Not Entailment)**	**Premise** What to the migrating birds usually follow? **Hypothesis** These routes seldom follow colina telemetry or coastlines, sometimes rivers , and may take advantage of updrafts and other wind diagrams or avoid spatial separating such as large stretches of commencement water.
CLARE **(Not Entailment)**	**Premise** What to the migrating birds usually follow? **Hypothesis** These routes cannot follow continental ranges or coastlines, connect rivers , and may take advantage of updrafts and other wind patterns or avoid geographical barriers such as large stretches of open water.
MAYA **(Not Entailment)**	**Premise** What to the migrating birds usually follow? **Hypothesis** These lines typically connect mountain ranges or coastlines, sometimes rivers, and may take advantage of updrafts and other wind patterns or avoid geographical barriers such as large stretches of open water.
PLAT **(Not Entailment)**	**Premise** What to the migrating birds usually follow ? **Hypothesis** These routes typically take advantages from larger mountain ranges or coastlines, sometimes rivers, and may take advantage of updrafts and other wind patterns or avoid geographical barriers such as large stretches of open water.

Table 13: Adversarial examples generated by different models on QNLI dataset, perturbations are colored.

C Additional Results

We list the full results of §4.2 about controllable ability on Yelp dataset in Table 14, Table 15, and Table 16, for the effects of candidate number N during infilling, syntactic tree depth restriction for phrase candidates d and length incremental restriction for substitution l on our PLAT model, respectively. It can be found that all N, d, and l can control the performance of PLAT in different aspects. Surprisingly, the grammar error decreases while d is increasing. We attribute this to the fact that the modification range of the phrase is extended as d increases, such that the infilling text is less likely to be essentially conditioned on the surrounding context and fewer grammar errors occur at the boards between blanks and rest texts.

We also test the effects of different likelihood ratio threshold on Yelp dataset, which is illustrated in Figure 4. A larger threshold δ may result in a lower attack success rate, more modifications, and a worse textual quality.

N	ASR↑	DIS↓	BLEU↑	PPL↓	GER↓
10	38.8	0.28	0.66	55.7	0.23
30	71.2	0.30	0.63	54.2	0.28
50	83.7	0.30	0.63	53.6	0.33
100	92.2	0.28	0.66	54.0	0.37
500	97.6	0.22	0.73	56.0	0.33
1000	98.1	0.20	0.75	56.2	0.33
2500	98.3	0.18	0.77	56.6	0.32
5000	98.4	0.17	0.78	56.8	0.33
1000	98.5	0.16	0.79	57.3	0.29

Table 14: The performance of PLAT with varying candidate number N during infilling.

d	ASR↑	DIS↓	BLEU↑	PPL↓	GER↓
2	27.4	0.16	0.79	60.6	0.33
4	98.4	0.17	0.78	56.8	0.33
6	98.8	0.18	0.78	55.7	0.25
10	98.5	0.20	0.76	55.9	0.23
25	98.7	0.22	0.74	55.8	0.19

Table 15: The performance of PLAT with varying depth restriction d when selecting phrase candidates.

l	ASR↑	DIS↓	BLEU↑	PPL↓	GER↓
2	98.4	0.17	0.78	58.5	0.32
3	98.4	0.17	0.78	56.8	0.33
6	98.6	0.19	0.77	54.2	0.37
10	99.2	0.21	0.76	55.4	0.45
15	99.0	0.22	0.75	52.0	0.42

Table 16: The performance of PLAT with varying substitution length incremental restriction l when selecting phrase candidates.

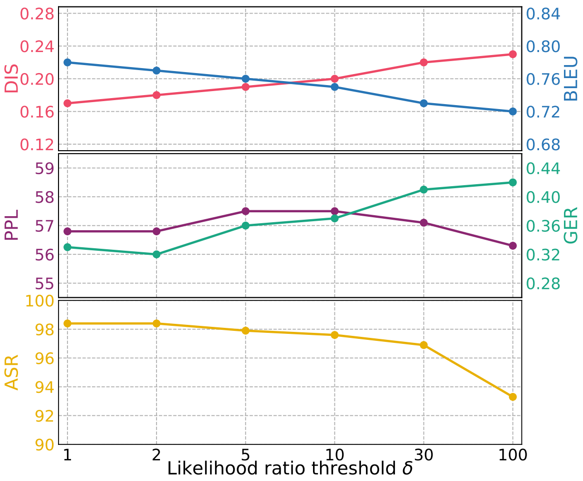
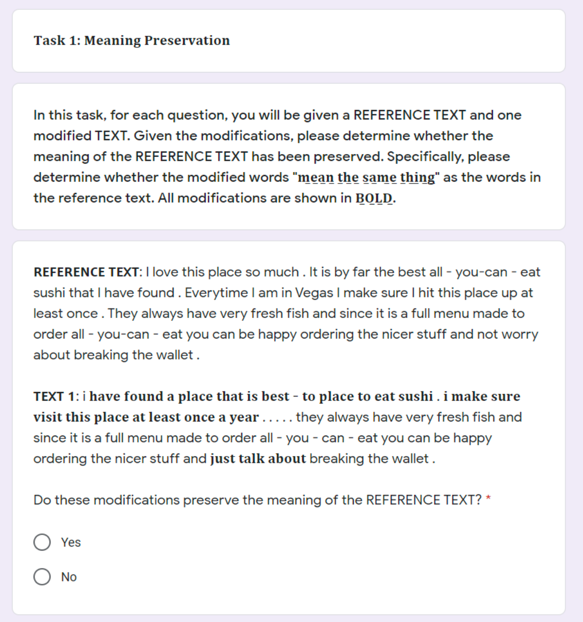


Figure 4: The performance of PLAT on Yelp dataset using different likelihood ratio threshold in label-preservation filter, in terms of all 5 metrics.

D Details of Human Evaluation

We conducted our human evaluation via *Google Forms* on a total 60 non-expert volunteer annotators. Each annotator was asked to rate for 10 sets of examples, where each set contains one original sample and three corresponding adversarial samples generated by PLAT, CLARE, and MAYA respectively. We show the screenshot of our instructions and examples in Figure 5, Figure 6, and Figure 7, where the perturbed parts are in bold font. We described how will we use these collected data in the invitations for annotators, and they must agree on this usage before evaluation. All collected data go without personal information in our experiments.



Task 1: Meaning Preservation

In this task, for each question, you will be given a REFERENCE TEXT and one modified TEXT. Given the modifications, please determine whether the meaning of the REFERENCE TEXT has been preserved. Specifically, please determine whether the modified words "**mean the same thing**" as the words in the reference text. All modifications are shown in **BOLD**.

REFERENCE TEXT: I love this place so much . It is by far the best all - you-can - eat sushi that I have found . Everytime I am in Vegas I make sure I hit this place up at least once . They always have very fresh fish and since it is a full menu made to order all - you-can - eat you can be happy ordering the nicer stuff and not worry about breaking the wallet .

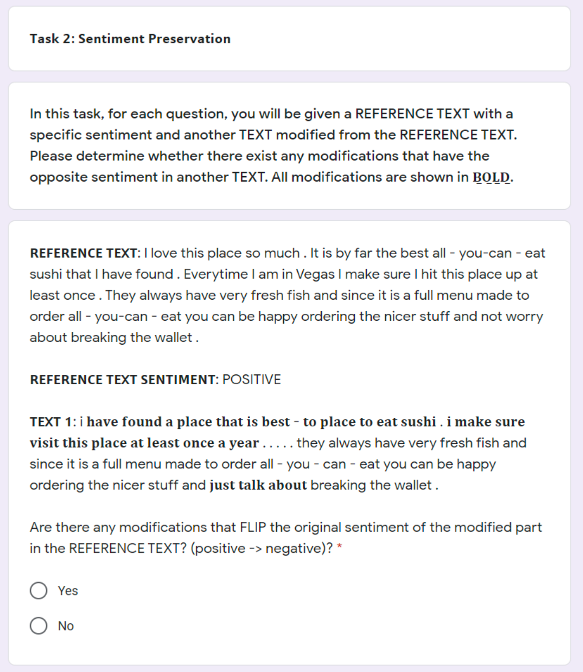
TEXT 1: **i have found a place that is best - to place to eat sushi . i make sure visit this place at least once a year** . . . . . they always have very fresh fish and since it is a full menu made to order all - you - can - eat you can be happy ordering the nicer stuff and **just talk about** breaking the wallet .

Do these modifications preserve the meaning of the REFERENCE TEXT? *

○ Yes

○ No

Figure 5: The instruction and an example of meaning preservation task in human evaluation.



Task 2: Sentiment Preservation

In this task, for each question, you will be given a REFERENCE TEXT with a specific sentiment and another TEXT modified from the REFERENCE TEXT. Please determine whether there exist any modifications that have the opposite sentiment in another TEXT. All modifications are shown in **BOLD**.

REFERENCE TEXT: I love this place so much . It is by far the best all - you-can - eat sushi that I have found . Everytime I am in Vegas I make sure I hit this place up at least once . They always have very fresh fish and since it is a full menu made to order all - you-can - eat you can be happy ordering the nicer stuff and not worry about breaking the wallet .

REFERENCE TEXT SENTIMENT: POSITIVE

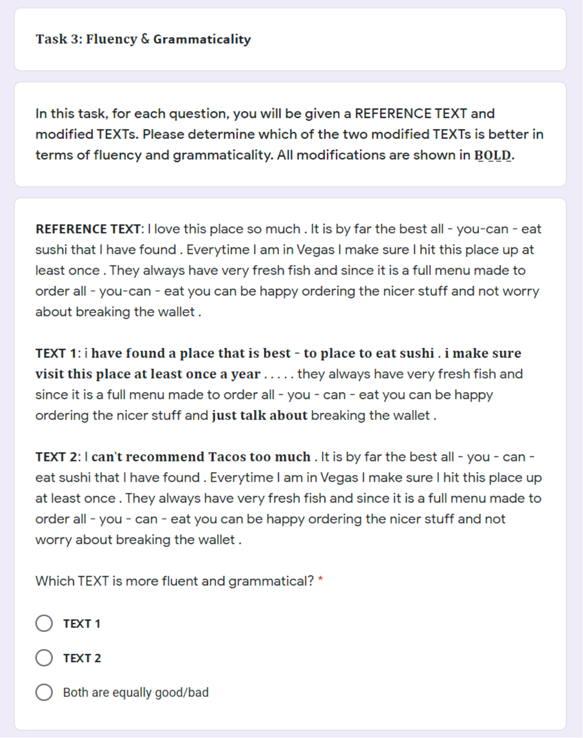
TEXT 1: **i have found a place that is best - to place to eat sushi . i make sure visit this place at least once a year** . . . . . they always have very fresh fish and since it is a full menu made to order all - you - can - eat you can be happy ordering the nicer stuff and **just talk about** breaking the wallet .

Are there any modifications that FLIP the original sentiment of the modified part in the REFERENCE TEXT? (positive -> negative)? *

○ Yes

○ No

Figure 6: The instruction and an example of label preservation task in human evaluation.



Task 3: Fluency & Grammaticality

In this task, for each question, you will be given a REFERENCE TEXT and modified TEXTs. Please determine which of the two modified TEXTs is better in terms of fluency and grammaticality. All modifications are shown in **BOLD**.

REFERENCE TEXT: I love this place so much . It is by far the best all - you-can - eat sushi that I have found . Everytime I am in Vegas I make sure I hit this place up at least once . They always have very fresh fish and since it is a full menu made to order all - you-can - eat you can be happy ordering the nicer stuff and not worry about breaking the wallet .

TEXT 1: **i have found a place that is best - to place to eat sushi . i make sure visit this place at least once a year** . . . . . they always have very fresh fish and since it is a full menu made to order all - you - can - eat you can be happy ordering the nicer stuff and **just talk about** breaking the wallet .

TEXT 2: I **can't recommend Tacos too much** . It is by far the best all - you - can - eat sushi that I have found . Everytime I am in Vegas I make sure I hit this place up at least once . They always have very fresh fish and since it is a full menu made to order all - you - can - eat you can be happy ordering the nicer stuff and not worry about breaking the wallet .

Which TEXT is more fluent and grammatical? *

○ TEXT 1

○ TEXT 2

○ Both are equally good/bad

Figure 7: The instruction and an example of fluency and grammaticality comparison task in human evaluation.