Query-Efficient Textual Adversarial Example Generation for Black-Box Attacks

Zhen Yu Zhenhua Chen Kun He[*]

School of Computer Science and Technology,
Huazhong University of Science and Technology, Wuhan, China
{baiding15, zhenhuachen, brooklet60}@hust.edu.cn

Abstract

Deep neural networks for Natural Language Processing (NLP) have been demonstrated to be vulnerable to textual adversarial examples. Existing black-box attacks typically require thousands of queries on the target model, making them expensive in real-world applications. In this paper, we propose a new approach that guides the word substitutions using prior knowledge from the training set to improve the attack efficiency. Specifically, we introduce **A**dversarial **B**oosting **P**reference (**ABP**), a metric that quantifies the importance of words and guides adversarial word substitutions. We then propose two query-efficient attack strategies based on ABP: query-free attack (ABP_{free}) and guided search attack ($\text{ABP}_{\text{guide}}$). Extensive evaluations for text classification demonstrate that ABP_{free} generates more natural adversarial examples than existing universal attacks, $\text{ABP}_{\text{guide}}$ significantly reduces the number of queries by a factor of $10 \sim 500$ while achieving comparable or even better performance than black-box attack baselines. Furthermore, we introduce the first ensemble attack ABP_{ens} in NLP, which gains further performance improvements and achieves better transferability and generalization by the ensemble of the ABP across different models and domains. Code is available at https://github.com/BaiDingHub/ABP.

1 Introduction

Despite the outstanding performance, Deep Neural Networks (DNNs) are known to be vulnerable to adversarial examples (Szegedy et al., 2014), *i.e.*, imperceptible perturbations on benign samples could lead to entirely incorrect predictions. Adversarial examples have brought critical security threats to the widely adopted deep learning based systems, and have attracted enormous attention on adversarial attacks in various domains, *e.g*. Computer Vision (CV) (Goodfellow et al., 2015; Madry et al., 2018), Natural Language Processing (NLP) (Papernot et al., 2016; Liang et al., 2018), *etc*.

Textual adversarial attacks pose greater challenges due to the discrete input space and constraints of lexicality, semantics and fluency, which can be categorized into white-box, black-box and universal attacks. White-box attacks (Liang et al., 2018; Meng and Wattenhofer, 2020) require full access to the victim model, including its architecture, parameters, loss function, gradient, output, *etc*., which are typically unavailable in real-world applications. Black-box attacks (Ren et al., 2019; Zang et al., 2020) only need access to the model output, *e.g*., output logits or predictions. However, they require thousands of queries to determine the importance of each word based on the model output, making them expensive in real-world applications. Universal attacks (Moosavi-Dezfooli et al., 2017; Li et al., 2021b) are a particular type of adversarial attacks which adds the same adversarial text generated by the training set at the beginning or end of all input samples to mislead the model. They require no access to the model and usually have a stronger attack efficiency and better usability in the real-world, but lower attack performance and unnatural adversarial examples.

In this work, we aim to utilize the prior knowledge learned from the training set to guide word substitutions in black-box attacks to reduce the need to access the model, making it more practical for real-world applications. To this end, we propose **A**dversarial **B**oosting **P**reference (**ABP**), a metric that quantifies the importance of different words and guides the word substitutions. By analyzing statistics of synonym substitutions during adversarial example generation in the training set, we estimate the preference for replacing a word with its synonym in generating adversarial examples, as well as the boosting contribution to the adversariality of these samples, and then multiply them to

[*] Corresponding author.

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 556–569
June 16-21, 2024 ©2024 Association for Computational Linguistics

obtain the score of adversarial boosting preference.

Based on pre-computed ABP from the training set, we propose three query-efficient attacks, including query-free attack ABP$_{\text{free}}$, guided search attack ABP$_{\text{guide}}$ and ensemble attack ABP$_{\text{ens}}$. For a benign text, ABP$_{\text{free}}$ determines the importance of each word, assigns a candidate for each of them, and constructs the replacement order for the text. Then ABP$_{\text{free}}$ substitutes the top 25% words with their respective candidates in the replacement order to generate adversarial examples with no access to the model. ABP$_{\text{guide}}$ adopts ABP to guide the search of adversarial examples, and then prunes redundant perturbations to find the optimal adversary. ABP$_{\text{guide}}$ significantly improves the attack performance through dozens of queries. Finally, ABP$_{\text{ens}}$ integrates ABP generated by different models or domains to further boost the attack performance.

To validate the effectiveness, we conduct extensive experiments to compare ABP with various black-box attacks (Ren et al., 2019; Zang et al., 2020) and universal attacks (Wallace et al., 2019; Song et al., 2021). Empirical evaluations show that ABP$_{\text{free}}$ generates more natural adversarial examples through synonym substitutions than existing universal attacks. ABP$_{\text{guide}}$ greatly reduces access to the model than existing black-box attacks, while achieving comparable attack performance. And the ensemble of ABP across models and domains gains further performance improvement and achieves superior cross-model transferability and cross-domain generalization. To the best of our knowledge, this is the first ensemble attack method for attacking NLP tasks.

2 Related Work

This section briefly introduces the black-box and universal attacks that are moist related to our work.

2.1 Black-box Adversarial Attacks

Black-box adversarial attacks perturb each benign text individually based on the model output. According to the perturbation level, these attacks fall into three categories: character-level attacks (Gao et al., 2018; He et al., 2021), word-level attacks (Alzantot et al., 2018; Yu et al., 2022), and sentence-level attacks (Zhao et al., 2018; Ribeiro et al., 2018). Among these attacks, the synonym substitution based word-level attacks exhibit excellent attack performance with natural, semantic and fluent adversarial examples, which are by far the most popular methods. PWWS (Ren et al., 2019) and TextFooler (Jin et al., 2020) consider the word saliency and classification probability to greedily substitute important words with synonyms. BERT-Attack (Li et al., 2020), BAE (Garg and Ramakrishnan, 2020) and CLARE (Li et al., 2021a) attack BERT (Devlin et al., 2019) using contextual perturbations from a BERT masked language model. GA (Alzantot et al., 2018) and PSO (Zang et al., 2020) adopt evolutionary algorithms to search for an optimal textual adversarial example.

These attacks require a large number of queries to victim models so as to determine the importance of each word or retain better word substitutions. In this work, we utilize the prior knowledge of adversarial boosting preferences obtained from the training set to guide word substitutions, which helps us significantly reduce the number of queries to the model while ensuring good attack performance.

2.2 Universal Adversarial Attacks

Universal adversarial attacks usually add a universal perturbation on the benign text to mislead the victim model without requiring queries. Behjati et al. (2019) and Wallace et al. (2019) prepend the trigger to each benign sample and update the embedding for every trigger token. Atanasova et al. (2020) propose to include an auxiliary semantic textual similarity objective to generate samples with low perplexity in the fact checking task. Song et al. (2021) propose the Natural Universal Trigger Search (NUTS) to improve naturalness using an auto-encoder and a generative adversarial network. Li et al. (2021b) propose the Data-Free Adjusted Gradient (DFAG) attack which utilizes the pseudo-samples generated by the perturbation to generate the trigger using only a single sample. Gao et al. (2022) add the universal perturbation to the latent representation encoded from discrete texts.

These triggers are sentence-level perturbations that introduce a meaningless sentence into the input samples, resulting in unnatural adversarial examples that are easily detected by humans. In this work, we introduce a word-level policy for the universal textual attack. By substituting synonyms in the input text based on the pre-computed adversarial boosting preference without access to the victim model, ABP generates more natural adversarial examples with good attack performance.

3 Methodology

In this section, we formalize the problem of text attack, present the definition of ABP metric, and provide details of the proposed method.

3.1 Problem Formulation

Given an input space $\mathcal{X}$ containing all the input texts and an output space $\mathcal{Y}$ containing the predict labels, we have a pre-trained classifier $f : \mathcal{X} \to \mathcal{Y}$, which maps the input text $x = w_1 w_2 \dots w_n \in \mathcal{X}$ to its ground-truth label $y \in \mathcal{Y}$ based on the maximum posterior probability $f(y|x)$. The adversary aims to add imperceptible perturbations on the correctly classified x to craft an adversarial example x^{adv} to mislead the classifier f:

$$\operatorname*{argmax}_{y_i \in \mathcal{Y}} f(y_i|x^{adv}) \neq \operatorname*{argmax}_{y_i \in \mathcal{Y}} f(y_i|x) = y$$

In this work, we introduce a novel metric called **A**dversarial **B**oosting **P**reference (ABP) to evaluate the importance of words. By guiding word substitutions with ABP, we propose three efficient black-box adversarial attacks that require only a few or even zero queries to the model output.

3.2 Adversarial Boosting Preference Metric

For each word w_i in the text x, we choose its top m nearest neighbors in the commonly used counter-fitted embedding space (Mrkšić et al., 2016) as its synonym set $\mathcal{S}(w_i) = \{w_i^1, \dots, w_i^k, \dots, w_i^m\}$. We define the adversarial boosting preference of substituting word w_i with word w_i^k when crafting the adversarial example as follows.

Definition 1 (Adversarial Boosting Preference) *Given a word w_i and its synonym w_i^k, the adversarial boosting preference of substituting w_i with w_i^k when crafting an adversarial example is defined as $\mathcal{A}(w_i, w_i^k) = P(w_i, w_i^k) \times I(w_i, w_i^k)$, where $P(w_i, w_i^k)$ denotes the preference of selecting word w_i^k from the synonym set $S(w_i)$ to replace w_i and $I(w_i, w_i^k)$ denotes the influence of substituting w_i with w_i^k on the victim model.*

Typically, a larger $\mathcal{A}(w_i, w_i^k)$ indicates a greater increase in the adversariality of the generated sample after substituting w_i with w_i^k in the original text. So $\mathcal{A}(w_i, w_i^k)$ can be used to determine the importance of different words in guiding the word substitutions when crafting adversarial examples.

In this work, we estimate the adversarial boosting preference between different words based on the statistics of word substitutions during the crafting of adversarial examples in the training set. We record the frequency of w_i changing to w_i^k obtained from statistics, as its preference $P(w_i, w_i^k)$, and record the average change in model output after its replacement as the influence $I(w_i, w_i^k)$.

Specifically, given an arbitrary synonym substitution based attack method, we can convert any text x with label y to an adversarial example x^{adv}, where some words $w_i \in x$ are replaced by $w_i^k \in x^{adv}$. We record all the replaced word pairs $\{(w_i, w_i^k)\}$ between each text x and its corresponding adversarial example x^{adv} in the training set $\mathcal{X}_{train}$. We count the number of occurrences of w_i and the number of times w_i be replaced by w_i^k, which we denote as $N(w_i)$ and $N(w_i, w_i^k)$. We calculate the preference $P(w_i, w_i^k) = N(w_i, w_i^k)/N(w_i)$. Meanwhile, we estimate the influence $I(w_i, w_i^k)$ by measuring the average change in the output logits after replacing w_i with w_i^k:

$$\begin{aligned} I(w_i, w_i^k) &= \frac{\sum(f(y|x) - f(y|x'))}{N(w_i, w_i^k)}, \\ \text{where} \qquad x &= w_1 w_2 \dots w_i \dots w_n, \\ x' &= w_1 w_2 \dots w_i^k \dots w_n. \end{aligned} \tag{1}$$

Finally, we obtain the adversarial boosting preference $\mathcal{A}(w_i, w_i^k) = P(w_i, w_i^k) \times I(w_i, w_i^k)$ for each word w_i and its synonyms w_i^k. It is worth noting that the preference for synonym substitutions may differ among texts with different labels. For instance, in a sentiment classification task, an attacker is more likely to substitute word "good" in texts with positive labels, but substitute word "bad" in texts with negative labels. Therefore, we calculate the adversarial boosting preference separately for texts with different categories.

3.3 The Proposed ABP Attacks

Given the pre-computed adversarial boosting preference $\mathcal{A}(w_i, w_i^k)$ derived from the training set $\mathcal{X}_{train}$, we can evaluate the importance of different words in guiding the word substitutions to generate the adversarial example on a benign text x_0. To this end, we propose three query-efficient attacks, including query-free attack ABP$_{\text{free}}$, guided search attack ABP$_{\text{guide}}$, and ensemble attack ABP$_{\text{ens}}$.

3.3.1 Query-Free Attack ABP$_{\text{free}}$

ABP$_{\text{free}}$ comprises of two essential steps, namely identifying important positions in the text and selecting suitable candidate words for substitution.

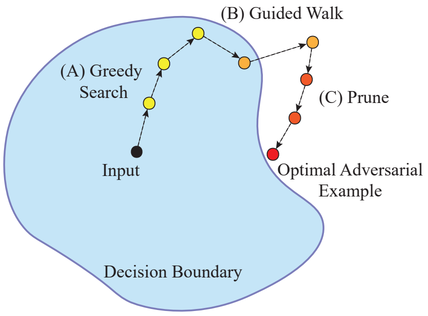


Figure 1: Overview of the proposed guided search attack $\text{ABP}_{\text{guide}}$. (A) Greedy search to approach the decision boundary (B) Guided walk to find an initial adversary (C) Prune to find the optimal adversary.

We take $\mathcal{A}(w_i, w_i^k)$ as the word importance of candidate w_i^k, and define the position importance of word w_i as $q_i = max_{k=1}^{m}\ \mathcal{A}(w_i, w_i^k)$. We sort the positions of all words $w_i \in x_0$ in descending order of the position importance q_i to determine the replacement order. For each word w_i in the top 25% of the replacement order, we select the most important candidate $w_i^* = argmax_{w_i^k}\ \mathcal{A}(w_i, w_i^k)$ from its synonym set $S(w_i)$ for substitution to generate an adversarial example.

ABP_{free} only utilizes adversarial boosting preferences to determine word substitutions, which requires no access to the victim model, making it a universal adversarial attack based on synonym substitution. Compared to previous universal adversarial attacks (Wallace et al., 2019; Song et al., 2021) that add meaningless prefixes or suffixes, ABP_{free} generates more natural adversarial examples through synonym substitutions.

3.3.2 Guided Search Attack $\text{ABP}_{\text{guide}}$

There are three key steps in $\text{ABP}_{\text{guide}}$, including **greedy search** to greedily replace key words to approach the decision boundary, **guided walk** to further perturb critical words to search for an initial adversarial example and **prune** to restore perturbed words to their original words to search for the optimal adversarial example, as shown in Figure 1.

For a benign text x_0, **greedy search** substitutes a word w_i with the most important candidate w_i^* sequentially according to the replacement order obtained in ABP_{free}. After each word substitution, we feed it into the model and ask whether it is an adversarial example until we succeed or all words have been replaced. If the search fails despite perturbing all words, we further perturb words to search for an initial adversarial example through the **guided walk** operation. Otherwise, we skip to the **prune** operation. **Guided walk** first samples several (at most δ) important positions w_i in the text with the probability p_i:

$$p_i = \frac{max(q_i,\ 0)}{\sum_{i=0}^{n} max(q_i,\ 0)},$$

where the max function ensures that only the words that can potentially enhance the adversariality of the text are selected for substitution. Afterwards, we substitute each selected w_i with a suitable candidate $w_i^k \in S(w_i)$ with probability $p_{i,k}$:

$$p_{i,k} = \frac{max(\mathcal{A}(w_i, w_i^k),\ 0)}{\sum_{k=0}^{m} max(\mathcal{A}(w_i, w_i^k),\ 0)}.$$

We repeat such search operation until we successfully find an adversarial example x^{adv} or exceed the maximum iteration limit T_g.

Finally, **prune** restores redundant substitutions in x^{adv} to their original words to search for the optimal adversarial example. **Prune** first samples several (at most δ) positions with probability $1-p_i$, and then substitutes each selected word with its corresponding original word in the benign text x_0. If the newly generated sample is an adversarial example, we continue the prune operation, otherwise we re-prune x^{adv}. We repeat the prune operation T_p times to approach the decision boundary and search for the optimal adversarial example. Additionally, we would perform an early stopping operation when only one word in x^{adv} is perturbed.

$\text{ABP}_{\text{guide}}$ adopts the adversarial boosting preference to guide word substitutions during the search for adversarial examples, which can improve the search efficiency and decrease the number of queries significantly to the victim model, as compared to previous black-box attacks (Ren et al., 2019; Zang et al., 2020).

3.3.3 Ensemble Attack ABP_{ens}

Previous researches (Dong et al., 2018; Xiong et al., 2022) in computer vision have shown that adversarial examples generated by attacking an ensemble of models would have better performance. As ABP is model-agnostic that only relates to the replaced word pairs, it can easily facilitate the ensemble of different models. To accomplish this, we count the total number $N(w_i, w_i^k)$ of replacements for word w_i with w_i^k when attacking model f_A and model f_B, as well as the total number $N(w_i)$ of occurrences of word w_i, to obtain the preference $P_{ens}(w_i, w_i^k)$ as before. We then calculate

Model	Attack	IMDB			SST			MR		
		Succ.	Pert.	Query	Succ.	Pert.	Query	Succ.	Pert.	Query
BERT	PWWS	**100.0**	**2.6**	4,681	88.7	12.0	362	88.9	10.7	398
	PSO	98.6	2.6	10,120	**99.7**	**10.7**	1,627	**99.4**	**9.7**	1726
	ABP$_{\textbf{guide}}$	99.9	3.0	**28**	85.4	13.3	28	89.3	11.9	25
	ABP$_{\textbf{ens-guide}}$	**100.0**	4.1	33	97.5	14.3	**19**	95.8	13.6	**21**
ALBERT	PWWS	86.2	4.5	4,680	97.3	11.0	357	94.4	11.5	391
	PSO	92.9	**2.7**	1,2424	**99.9**	**10.0**	1,087	**100.0**	**9.1**	1,020
	ABP$_{\textbf{guide}}$	97.0	4.6	42	92.5	12.1	20	94.7	10.6	**17**
	ABP$_{\textbf{ens-guide}}$	**99.7**	5.6	**41**	97.2	13.5	**18**	97.8	12.6	**17**
LSTM	PWWS	**100.0**	**2.1**	4,690	98.4	12.0	358	98.5	**10.4**	392
	PSO	98.1	2.6	11,161	**99.9**	**10.2**	1,174	**100.0**	11.2	1,590
	ABP$_{\textbf{guide}}$	**100.0**	2.2	**22**	89.7	12.8	25	95.0	10.8	**16**
	ABP$_{\textbf{ens-guide}}$	99.9	3.1	29	96.9	14.3	**21**	96.1	12.6	19

Table 1: Comparison of attack success rate (Succ., %), word perturbation rate (Pert., %) and average query number (Query) between ABP$_{\text{guide}}$, ABP$_{\text{ens-guide}}$ and various black-box baselines on three models using three datasets. The **best performance** is highlighted in **bold**.

the average influence $I_{ens}(w_i, w_i^k)$ on models f_A and f_B using Equation 1, and obtain the ensemble adversarial boosting preference $\mathcal{A}_{ens}(w_i, w_i^k) = P_{ens}(w_i, w_i^k) \times I_{ens}(w_i, w_i^k)$. The resulting ensemble attacks using it for query-free attacks and guided search attacks are referred to as ABP$_{\text{ens-free}}$ and ABP$_{\text{ens-guide}}$, respectively. Additionally, ABP adopts the same attack strategy for texts with the same label, making it possible to generalize across domains within the same task. Consequently, we could integrate the final adversarial boosting preference obtained from different models or domains.

4 Experiments

To validate the effectiveness of ABP, we conduct extensive experiments for text classification.

4.1 Experimental Setup

Datasets and Models. We adopt three widely used sentiment analysis datasets, *i.e.*, IMDB (Maas et al., 2011), SST (Socher et al., 2013) and MR (Pang and Lee, 2005). And we train three classical models on the training set, *i.e.*, BERT base-uncased (Devlin et al., 2019), ALBERT base (Lan et al., 2020) and Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). Due to the high computational costs, we do not adopt large language models such as RoBERTa (Liu et al., 2019) and GPT-3 (Brown et al., 2020). More details of these datasets and the classification accuracy of each model are shown in Appendix A.

Baselines. We select two commonly used black-box attacks to evaluate the effectiveness of ABP$_{\text{guide}}$, *i.e.*, PWWS (Ren et al., 2019) and PSO (Zang et al., 2020), and two universal attacks to evaluate the effectiveness of ABP$_{\text{free}}$, *i.e.*, Universal Adversarial Trigger (UAT) (Wallace et al., 2019) and NUTS (Song et al., 2021).

Evaluation Settings. In our ABP, we adopt PWWS (Ren et al., 2019) to generate adversarial examples and set the synonym number $m = 30$. The maximum perturbation rate of 25% is set in ABP$_{\text{free}}$, as it is commonly used in word-level attacks. And we set the maximum iteration limit to $T_g = 100$ and $T_p = 20$, and the maximum number of perturbations for a single iteration to $\delta = 2$ in ABP$_{\text{guide}}$. For the generation of ABP and triggers in UAT and NUTS, we randomly sample 10,000 texts from the corresponding training set, except for the MR dataset, where we use 9,662 samples. All the evaluations are conducted on a subset comprising a maximum of 1000 randomly sampled texts from the corresponding test set. The parameter studies and ablation studies of ABP are presented in Appendices B and C.

4.2 Evaluation on Efficiency of ABP$_{\text{guide}}$

In practice, if the victim detects an excessive number of queries within a short period of time, they can block the attack by simply denying the access. Therefore, the attack efficiency, typically referred

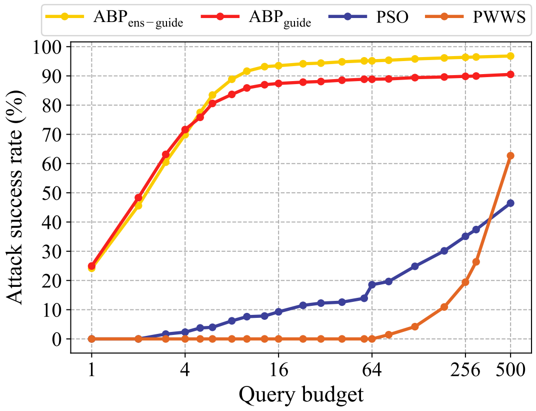


Figure 2: Attack success rate (%) of ABP$_{\text{guide}}$ and other black-box attacks on BERT model using MR dataset under different query budgets (axis x is in log-scale).

to the query budget of the victim model, plays a crucial role in evaluating the effectiveness of black-box attacks in real-world applications. We conduct evaluations using three benchmark datasets on three typical models. The results, including attack success rate, word perturbation rate and average query number are summarized in Table 1. We investigate the performance of different ensemble strategies in Appendix D. We find that ABP$_{\text{ens-guide}}$ achieves higher performance when integrating ABP from diverse sources, while ABP$_{\text{ens-free}}$ is more effective at integrating ABP from the same victim model. Thus, we opt to implement ABP$_{\text{ens-guide}}$ by integrating ABP obtained from BERT, ALBERT and LSTM models using IMDB, SST and MR datasets. Similarly, for ABP$_{\text{ens-free}}$ evaluated in Section 4.3, we only integrate ABP obtained from various domains for each victim model.

The results show that PWWS and PSO are computationally expensive, requiring hundreds or even thousands of queries to launch a successful attack on a single sample. When dealing with a large number of samples, the cost of queries to the victim is prohibitively high for the attacker. In contrast, ABP$_{\text{guide}}$ uses much fewer queries (by a factor of $10 \sim 500$) compared to the baselines, with only a slight decline on the attack success rate and word perturbation rate. This greatly reduces the cost of the attack and improves its efficiency. Furthermore, we can find that ABP$_{\text{ens-guide}}$ achieves better performance than ABP$_{\text{guide}}$, outperforming PWWS in most cases and only slightly underperforming PSO.

We further evaluate their attack performance under various query budgets on BERT model using MR dataset, as shown in Figure 2. We dynamically adjust the population size of PSO to accommodate the limited number of queries. The results show that PWWS and PSO exhibit poor attack performance when the query budget is limited. In contrast, ABP$_{\text{guide}}$ and ABP$_{\text{ens-guide}}$ consistently achieves high performance under various query budgets. Notably, the attack success rate of ABP$_{\text{guide}}$ reaches 71.6% when the query budget is only 4, which is 71.6% and 69.3% higher than PWWS and PSO, respectively. These evaluations show that ABP$_{\text{guide}}$ and ABP$_{\text{ens-guide}}$ can achieve good attack performance even in real-world scenarios where the query is strictly limited, demonstrating their effectiveness and practicality.

Model	Attack	IMDB	SST	MR
BERT	UAT	39.1	**78.4**	**80.6**
	NUTS	3.9	25.8	20.1
	ABP$_{\text{free}}$	**98.7**	63.0	68.3
	ABP$_{\text{ens-free}}$	97.6	68.5	71.6
ALBERT	UAT	82.4	**98.7**	**99.2**
	NUTS	1.9	23.8	25.8
	ABP$_{\text{free}}$	85.6	73.4	78.6
	ABP$_{\text{ens-free}}$	**88.4**	72.5	76.0
LSTM	UAT	96.7	**95.3**	**99.7**
	NUTS	24.9	35.0	63.8
	ABP$_{\text{free}}$	**99.9**	62.0	84.6
	ABP$_{\text{ens-free}}$	**99.9**	66.5	78.1

Table 2: Attack success rate (Succ., %) of various universal attacks on different models and datasets.

4.3 Evaluation on Effectiveness of ABP$_{\text{free}}$

The universal adversarial attack is a suitable option when the targeted model is entirely inaccessible. It usually has higher attack efficiency, making it more practical than both white-box and black-box attacks in real-world settings. This subsection evaluates the attack success rate and naturalness of our proposed ABP$_{\text{free}}$ and existing universal attacks using three datasets on different models, as shown in Table 2.

We could observe that ABP$_{\text{free}}$ achieves superior attack performance than NUTS, and also outperforms UAT on IMDB datasets but exhibits lower performance on SST and MR datasets. In contrast, the performance of ABP$_{\text{ens-free}}$ varies, sometimes surpassing and sometimes underperforming ABP$_{\text{free}}$. This observation implies that integrating ABP across multiple domains for ABP$_{\text{ens-free}}$ would create a trade-off in its overall performance across these domains. Table 3 shows adversarial examples generated by these attacks. UAT arbitrarily combines several words as the trigger, resulting in ex-

Attack	Original Text & Adversarial Example	Prediction
Original Text	Oliveira seems to pursue silent film representation with every mournful composition.	Positive
UAT	None is in Oliveira seems to pursue silent film representation with every mournful composition.	Negative
NUTS	Huge wooden ##loaded Oliveira seems to pursue silent film representation with every mournful composition.	Negative
ABP$_{\text{free}}$	Oliveira ostensibly to pursue silent film representation with every pained composition.	Negative

Table 3: The benign text from MR dataset and adversarial examples generated by various attacks on BERT model. We highlight the words replaced and inserted by the attacks in red.

Attack	BERT*	ALBERT	LSTM	BERT	ALBERT*	LSTM	BERT	ALBERT	LSTM*
PWWS	88.9*	40.6	39.7	**33.5**	94.4*	38.0	22.8	**33.2**	98.5*
PSO	100.0*	43.8	36.6	33.4	100.0*	34.8	23.7	25.2	100.0*
ABP$_{\text{guide}}$	89.3*	44.2	44.8	28.3	94.7*	39.7	23.2	33.1	95.0*
ABP$_{\text{ens-guide}}$	94.9*	**54.2**	**47.7**	31.4	97.5*	**44.1**	**40.9**	27.5	96.3*
UAT	80.6*	34.5	**57.3**	0.8	99.2*	2.1	45.1	42.9	99.7*
NUTS	20.1*	19.1	10.7	7.5	25.8*	17.1	4.4	20.8	63.8*
ABP$_{\text{free}}$	68.3*	**57.5**	49.7	48.2	78.6*	**51.5**	**44.7**	**56.3**	84.6*
ABP$_{\text{ens-free}}$	71.6*	54.6	50.2	**49.4**	76.0*	51.3	44.1	53.3	78.1*

Table 4: Attack success rate (%) of various attacks on different models using MR dataset for the cross-model transferability evaluation. * indicates that the adversarial examples are generated based on this model.

tremely unnatural adversarial examples. Although NUTS generates meaningful triggers, these triggers are irrelevant to the textual content and also lead to unnatural samples. In contrast, the adversarial examples obtained by ABP$_{\text{free}}$ based on synonym substitution are more natural and imperceptible.

Moreover, adversarial examples should also be imperceptible to humans. Hence, we conduct human evaluation in Appendix E. The results show that UAT and NUTS generate a large number of unnatural samples that are easily detected by humans. In contrast, ABP$_{\text{free}}$ achieves higher naturalness, with 70.8% of samples identified as natural, which is very close to the benign texts.

In conclusion, ABP$_{\text{free}}$ achieves higher attack performance than NUTS, and gains comparable or slightly lower attack success rate than UAT. Furthermore, the case study and human evaluation show that the adversarial examples generated by ABP$_{\text{free}}$ are with higher naturalness and quality.

4.4 Evaluation on Transferability

The transferability refers to the adversarial examples crafted on one model could remain adversarial on other models. It enables the adversarial examples to maintain attack capabilities even if the model cannot be accessed in the real world. To illustrate the transferability of ABP, we generate adversarial examples on each model by various attacks and report their attack success rate on other models in Table 4. ABP$_{\text{ens-free}}$ and ABP$_{\text{ens-guide}}$ adopt the ensemble of ABP across IMDB, SST and MR datasets obtained from each victim model to attack the corresponding model for the query-free attack and guided search attack, respectively.

Although PWWS and PSO show satisfactory performance on accessible victim models, their attack success rates are significantly reduced by 40% to 70% when transferred to other models. ABP$_{\text{guide}}$ exhibits relatively better transferability but still falls behind. By employing ensemble techniques, ABP$_{\text{ens-guide}}$ not only enhances the attack performance on the original model but also improves its transferability. Regarding universal attacks, UAT and NUTS exhibit poor transferability, whereas ABP$_{\text{free}}$ surpasses them in this aspect. These evaluations validate the strong cross-model transferability of ABP and highlight the advantages of ensemble attacks.

4.5 Evaluation on Generalization

Universal attacks generate triggers on one dataset and then use them to attack texts in the same domain. However, it is almost impossible for an attacker to know beforehand which domain the victim model is trained on for real-world applications. Thus, a good universal attack should exhibit excellent cross-domain generalization, *i.e.*, the adversarial perturbations crafted from one domain remain adversarial for others on the same task. We first

Attack	IMDB*	SST	MR	IMDB	SST*	MR	IMDB	SST	MR*
UAT	39.1*	30.0	28.7	18.1	78.4*	**76.5**	17.3	**82.2**	80.6*
NUTS	3.9*	13.3	14.3	2.3	25.8*	19.3	2.4	21.0	20.1*
ABP$_{\textbf{free}}$	98.7*	41.8	**47.0**	90.3	63.0*	68.2	87.2	69.5	68.3*
ABP$_{\textbf{ens-free}}$	97.8*	**42.3**	46.6	**91.6**	55.6*	61.4	**91.7**	64.7	65.7*
ABP$_{\textbf{guide}}$	99.9*	75.5	75.6	97.6	85.4*	88.1	98.1	93.9	89.3*
ABP$_{\textbf{ens-guide}}$	99.8*	**84.2**	**84.7**	**99.3**	89.3*	**91.3**	**99.9**	**96.0**	92.3*

Table 5: Attack success rate (%) of various attacks using different domains on BERT model for the cross-domain generalization evaluation. * indicates that the adversarial examples are generated using this domain.

perform attacks on a single dataset and test their performance on another domain on the sentiment analysis task in Table 5. In this study, we utilize the ensemble of ABP obtained on the BERT, ALBERT and LSTM models using the corresponding dataset to implement ABP$_{\text{ens-free}}$ and ABP$_{\text{ens-guide}}$.

We can see that ABP$_{\text{free}}$ consistently exhibits strong cross-domain generalization, showing greater effectiveness on the IMDB dataset and performing slightly weaker on the generalization between SST and MR datasets compared to UAT. ABP$_{\text{ens-free}}$ further improves the generalization of ABP$_{\text{free}}$ in most cases. Moreover, ABP$_{\text{guide}}$ achieves an exceptionally high attack success rate when utilizing the adversarial boosting preference generated from one domain to attack the text from another domain. Meanwhile, ABP$_{\text{ens-guide}}$ achieves higher attack success rate and generalization. These findings demonstrate the strong cross-domain generalization of ABP, along with its superior attack performance and practicality, even when the training data of the victim model is unseen in real-world applications.

4.6 Evaluation on Real-world Applications

To further validate the practicality of ABP, we perform the attack against real-world online commercial Application Programming Interfaces (APIs). Due to the high cost of commercial APIs, we sample 50 texts from MR dataset for the test and evaluate the attack performance on the Amazon Cloud sentiment analysis API[1]. We adopt the trigger or ABP generated on BERT model using MR dataset for universal attacks and our ABP attacks. And we utilize the ensemble of ABP on three models using three datasets for the ensemble attack ABP$_{\text{ens}}$. As shown in Figure 3, we can see that PWWS and PSO, despite achieving high attack success rates, incur significant computational expenses, requiring 500 or even 1,500 accesses per sample. In contrast, ABP$_{\text{ens-guide}}$ exhibits 96% attack success rate and remarkable attack effectiveness, surpassing all baselines. These findings provide empirical evidence of the effectiveness of ABP in real-world applications.

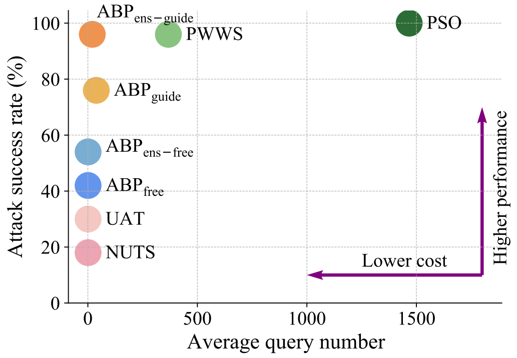


Figure 3: Attack success rate (%) and average query number (Query) when attacking Amazon Cloud APIs using various attacks.

5 Conclusion

In this work, we propose the **A**dversarial **B**oosting **P**reference (**ABP**) metric, which quantifies the importance of different words and guides word substitutions for attacks. By leveraging ABP obtained from the statistics of a typical attack method on the training set, we propose three novel attack strategies, including query-free attack ABP$_{\text{free}}$, guided search attack ABP$_{\text{guide}}$, and ensemble attack ABP$_{\text{ens}}$. Extensive evaluations demonstrate the efficiency and effectiveness of these strategies. Specifically, ABP$_{\text{free}}$ generates more natural adversarial examples without requiring queries to victim models than existing universal attacks. ABP$_{\text{guide}}$ significantly improves the attack efficiency, achieving similar attack performance to black-box attacks with only a few queries. Additionally, ABP$_{\text{ens}}$, as the first ensemble attack method in the field of NLP, exhibits better attack performance, strong cross-model transferability and cross-domain generalization compared to existing attacks. The di-

[1] https://aws.amazon.com/

verse range of attack strategies employed by ABP enables it to be applicable in various real-world scenarios, even when the victim model or training data is inaccessible. Exploring more effective ways to leverage ABP will be an interesting direction for future research.

Limitations

We summarize our limitations as follows:

Firstly, we rely on samples from the training set to generate ABP scores for each word. When samples from the same domain as the training samples of the model are not available or the total number of samples is limited, ABP-based attacks would perform poorly, as shown in Appendix B. To address this problem, we find that ABP exhibits good cross-domain generalization, allowing us to leverage samples from other domains for adversarial attacks with good attack performance.

Secondly, we have only presented a few of the simplest and most straightforward attack strategies based on ABP which significantly enhance the practicality of adversarial attacks in real-world applications. And we provide a concise analysis of ABP, revealing its contribution in improving the interpretability of the model in Appendix F. Therefore, exploring more effective ways to leverage ABP will be an interesting direction for future research.

Acknowledgement

This work is supported by National Natural Science Foundation (62076105, U22B2017) and Hubei International Cooperation Foundation of China (2021EHB011).

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In *Conference on Empirical Methods in Natural Language Processing*.

Pepa Atanasova, Dustin Wright, and Isabelle Augenstein. 2020. Generating label cohesive and well-formed adversarial claims. In *Conference on Empirical Methods in Natural Language Processing*.

Melika Behjati, Seyed-Mohsen Moosavi-Dezfooli, Mahdieh Soleymani Baghshah, and Pascal Frossard. 2019. Universal adversarial attacks on text classifiers. In *IEEE International Conference on Acoustics, Speech and Signal Processing*.

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*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Haoran Gao, Hua Zhang, Xingguo Yang, Wenmin Li, Fei Gao, and Qiaoyan Wen. 2022. Generating natural adversarial examples with universal perturbations for text classification. In *Neurocomputing*.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *IEEE Security and Privacy Workshops (SPW)*.

Siddhant Garg and Goutham Ramakrishnan. 2020. BAE: BERT-based adversarial examples for text classification. In *Conference on Empirical Methods in Natural Language Processing*.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*.

Xuanli He, Lingjuan Lyu, Lichao Sun, and Qiongkai Xu. 2021. Model extraction and adversarial transferability, your BERT is vulnerable! In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. In *Neural Computation*.

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 *AAAI Conference on Artificial Intelligence*.

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 *International Conference on Learning Representations*.

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2021a. Contextualized perturbation for textual adversarial attack. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial attack against BERT using BERT. In *Conference on Empirical Methods in Natural Language Processing*.

Xinzhe Li, Ming Liu, Xingjun Ma, and Longxiang Gao. 2021b. Exploring the vulnerability of natural language processing models via universal adversarial texts. In *Proceedings of the The 19th Annual Workshop of the Australasian Language Technology Association*.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. 2018. Deep text classification can be fooled. In *International Joint Conference on Artificial Intelligence*.

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.

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 *Association for Computational Linguistics*.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*.

Zhao Meng and Roger Wattenhofer. 2020. A geometry-inspired attack for generating natural language adversarial examples. In *International Conference on Computational Linguistics*.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. 2017. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Association for Computational Linguistics*.

Nicolas Papernot, Patrick McDaniel, Ananthram Swami, and Richard Harang. 2016. Crafting adversarial input sequences for recurrent neural networks. In *MILCOM IEEE Military Communications Conference*.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Association for Computational Linguistics*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging nlp models. In *Association for Computational Linguistics*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Conference on Empirical Methods in Natural Language Processing*.

Liwei Song, Xinwei Yu, Hsuan-Tung Peng, and Karthik Narasimhan. 2021. Universal adversarial attacks with natural triggers for text classification. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

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*.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing nlp. In *Conference on Empirical Methods in Natural Language Processing*.

Yifeng Xiong, Jiadong Lin, Min Zhang, John E. Hopcroft, and Kun He. 2022. Stochastic Variance Reduced Ensemble Adversarial Attack for Boosting the Adversarial Transferability. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022*, pages 14963–14972.

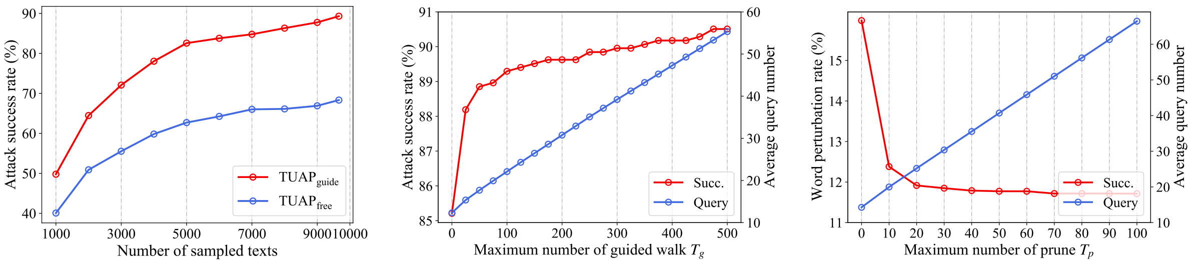
Zhen Yu, Xiaosen Wang, Wanxiang Che, and Kun He. 2022. Learning-based hybrid local search for the hard-label textual attack. In *arXiv preprint arXiv:2201.08193*.

Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. 2020. Word-level textual adversarial attacking as combinatorial optimization. In *Association for Computational Linguistics*.

Zhengli Zhao, Dheeru Dua, and Sameer Singh. 2018. Generating natural adversarial examples. In *International Conference on Learning Representations*.

Dataset	#Classes	#Avg. Len	Train	Test	BERT	ALBERT	LSTM
IMDB	2	253	25,000	25,000	90.8	92.1	87.3
SST	2	10	67,349	872	91.4	88.8	88.1
MR	2	20	9,622	1,000	90.6	82.8	77.7

Table 6: Details of three sentiment analysis datasets and their accuracy results of victim models."#Classes" means the number of the categories. "#Avg. Len" denotes the average length of texts. "Train" and "Test" indicate the number of texts in the training set and test set. "BERT", "ALBERT" and "LSTM" mean the classification accuracy (%) of BERT, ALBERT and LSTM in the test set.



(a)

(b)

(c)

Figure 4: Parameter studies on BERT model using MR dataset. (a) Impact of the number of sampled texts on the attack success rate (Succ., %) of ABP$_{\text{free}}$ and ABP$_{\text{guide}}$. (b) Impact of the maximum number of guided walk T_g on the attack success rate (Succ., %) and average query number (Query) of ABP$_{\text{guide}}$. (c) Impact of the maximum number of prune T_p on the word perturbation rate (Pert., %) and average query number (Query) of ABP$_{\text{guide}}$.

A More Details of Dataset and Models

We adopt three sentiment analysis datasets, *i.e.*, IMDB (Maas et al., 2011), SST (Socher et al., 2013) and MR (Pang and Lee, 2005), for text classification. And we train three classical models on the training set, *i.e.*, BERT base-uncased (Devlin et al., 2019), ALBERT base (Lan et al., 2020) and Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). The details of these datasets and victim models are list in Table 6.

B Parameter Study

To gain more insights on the effectiveness of ABP, we conduct a series of parameter studies on BERT model using MR dataset to study the impact of the number of sampled texts, the maximum iteration number of guided walk T_g and the maximum iteration number of prune T_p.

On the number of sampled texts. To validate the influence of different sample numbers on ABP, we perform the generation of ABP with different numbers of sampled texts from MR dataset and then evaluate their attack success rates on BERT model, as shown in Figure 4a. Note that due to the small amount of data in the training set of the MR dataset, a total of 9662 texts, including 8662 texts in the training set and 1000 unused texts in the test set, are used for the generation of ABP. We could observe that the attack success rate increases rapidly at the beginning. And there is a log relationship between the attack success rate and the amount of the sampled texts. In general, we set the sample number to 10,000 in our main experiments.

On the maximum iteration number of guided walk. In Figure 4b, we study the impact of the maximum iteration number of guided walk T_g on the attack success rate and average query number. As T_g increases, the attack success rate of ABP$_{\text{guide}}$ also increases. However, when T_g exceeds 100, the growth rate of attack success rate becomes slower. At the same time, as T_g increases, the number of queries to the victim model also increases. To achieve a balance between the attack success rate and query cost of ABP$_{\text{guide}}$, we have chosen to set T_g to 100.

On the maximum iteration number of prune. Finally, we study the impact of the maximum iteration number of prune T_p on the word perturbation rate and average query number, as shown in Figure 4c. As T_p increases, the word perturbation rate of the adversarial examples generated by ABP$_{\text{guide}}$ gradually decreases, while the number of queries to the victim model increases. To strike a balance between the number of queries and the quality of generated adversarial samples, we have set T_p to 20 based on the elbow rule.

C Ablation Study

To figure out the impact of three steps in $\text{ABP}_{\text{guide}}$, *i.e.*, greedy search, guided walk and prune steps, we conduct the ablation studies as shown in Table 8. We observe that greedy search achieves an impressive attack success rate of 85.2% with only 3 queries. By incorporating guided walk, $\text{ABP}_{\text{guide}}$ achieves higher attack success rates, albeit with a slight increase in word perturbation rate and the number of queries. Furthermore, the use of prune eliminates redundant word substitutions, resulting in a further reduction in the word perturbation rate.

In summary, the combination of greedy search, guided walk, and prune techniques enhances the efficiency and effectiveness of ABP. Greedy search provides a good starting point, and guided walk allows for further exploration in the sample space, leading to higher attack success rates. The prune technique then refines the adversarial examples by removing unnecessary word substitutions. Together, these techniques synergistically contribute to the development of optimal adversary.

D Effectiveness of ABP_{ens} Integrated from Different Sources

As described in Section 3.3.3, the ensemble attack ABP_{ens} could integrate the ABP obatained from different sources, including various models or domains, for the query-free attack $\text{ABP}_{\text{ens-free}}$ and guided search attack $\text{ABP}_{\text{ens-guide}}$. In this section, we investigate the performance of ABP_{ens} integrated from different sources. We have three categories in total, including $\text{ABP}_{\text{ens(model)}}$ to integrate the ABP obtained from BERT, ALBERT and LSTM models using single dataset, $\text{ABP}_{\text{ens(domain)}}$ to integrate the ABP obtained from IMDB, SST and MR datasets using single model, and $\text{ABP}_{\text{ens(all)}}$ to integrate the ABP obtained from all three datasets using all three models. Table 7 and Table 9 show the performance of the ensemble attack for guided search attack and query-free attack, respectively.

In terms of the guided search attack, we find that all ensemble strategies could enhance the attack success rate of $\text{ABP}_{\text{guide}}$ with a slight increase in perturbation rate and number of queries. Among the ensemble strategies, $\text{ABP}_{\text{ens(all)-guide}}$ achieves the highest attack success rate, followed by $\text{ABP}_{\text{ens(domain)-guide}}$, and $\text{ABP}_{\text{ens(model)-guide}}$ performs the worst. We speculate that $\text{ABP}_{\text{ens(all)-guide}}$ offers a more diverse search space for guided search, thus achieving a higher success rate. However, this comes at the cost of requiring more queries to perform the search. Therefore, we choose $\text{ABP}_{\text{ens(all)-guide}}$ for the ensemble attack in Section 4.2.

In terms of the query-free attack, we can see that the ensemble attack generally reduces the attack performance of ABP_{free}, with $\text{ABP}_{\text{ens(model)-free}}$ exhibiting the poorest performance and $\text{ABP}_{\text{ens(domain)-free}}$ displaying relatively better performance. Specifically, the ensemble of ABP from different domains in $\text{ABP}_{\text{ens(domain)-free}}$ creates a trade-off in its performance across these domains, sometimes surpassing and sometimes underperforming ABP_{free}. We speculate that the ensemble of ABP may not be conducive to identifying the most critical words, resulting in a reduction in their performance. Nevertheless, it is this ensemble that confers better cross-model transferability and cross-domain generalization to ABP.

E Human Evaluation

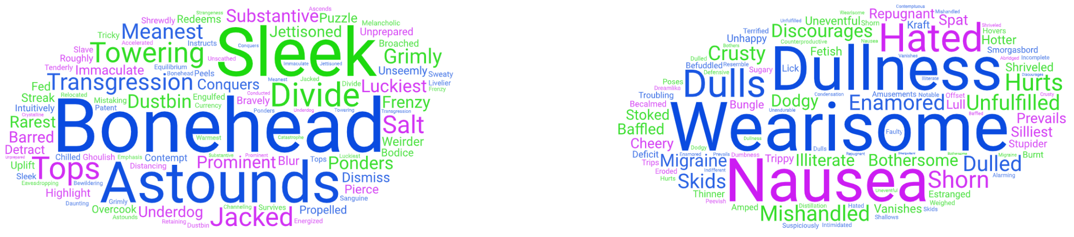
Adversarial examples should be imperceptible to humans. But we find that most existing universal attacks generates unnatural adversarial examples as described in Section 4.3. To further validate their imperceptibility, we conduct human evaluation on 50 benign texts sampled from MR dataset and the corresponding adversarial examples generated by various universal attacks on BERT. We invite 10 volunteers to label the samples as natural, unnatural, or uncertain. The results in Table 10 show that UAT and NUTS generate a large number of unnatural samples that are easily detected by humans. In contrast, ABP_{free} achieves higher naturalness, with 70.8% of samples identified as natural, which is very close to the benign texts. It further validates the naturalness of the adversarial examples generated by ABP.

F Further Exploration on ABP

In order to gain further insights into ABP and analyze the factors contributing to model vulnerability, we present the visualization of ABP scores for different words in Figure 5. Our observations reveal an interesting phenomenon. In texts with positive sentiment, certain words associated with negative sentiment exhibit higher ABP scores, such as "bonehead", and positive sentiment words

Model	Attack	IMDB			SST			MR		
		Succ.	Pert.	Query	Succ.	Pert.	Query	Succ.	Pert.	Query
BERT	ABP$_{\text{guide}}$	99.9	**3.0**	**28**	85.4	13.3	28	89.3	**11.9**	25
	ABP$_{\text{ens(model)-guide}}$	99.8	3.8	31	89.3	15.0	27	92.3	13.5	24
	ABP$_{\text{ens(domain)-guide}}$	**100.0**	3.9	32	96.0	**13.0**	**18**	94.9	12.4	**20**
	ABP$_{\text{ens(all)-guide}}$	**100.0**	4.1	33	**97.5**	14.3	19	**95.8**	13.6	21
ALBERT	ABP$_{\text{guide}}$	97.0	**4.6**	42	92.5	**12.1**	20	94.7	**10.6**	**17**
	ABP$_{\text{ens(model)-guide}}$	98.8	5.3	**39**	93.8	13.1	20	96.1	11.3	18
	ABP$_{\text{ens(domain)-guide}}$	98.8	5.9	44	96.9	13.0	**17**	97.5	11.9	**17**
	ABP$_{\text{ens(all)-guide}}$	**99.7**	5.6	41	**97.2**	13.5	18	**97.8**	12.6	**17**
LSTM	ABP$_{\text{guide}}$	**100.0**	**2.2**	**22**	89.7	**12.8**	25	95.0	**10.8**	**16**
	ABP$_{\text{ens(model)-guide}}$	99.9	2.6	24	91.7	14.3	26	95.1	12.1	20
	ABP$_{\text{ens(domain)-guide}}$	99.9	2.4	24	96.4	13.6	**19**	**96.3**	11.7	17
	ABP$_{\text{ens(all)-guide}}$	99.9	3.1	29	**96.9**	14.3	**21**	96.1	12.6	19

Table 7: Comparison of attack success rate (Succ., %), word perturbation rate (Pert., %) and average query number (Query) between ABP$_{\text{guide}}$ and various ensemble attacks.



(a) Word cloud for words in texts with positive sentiment

(b) Word cloud for words in texts with negative sentiment

Figure 5: Word cloud illustration about ABP scores for different words in texts with positive and negative sentiment. We adopt the ABP generated on BERT model using MR dataset. The higher the ABP score of a word, the larger its size will be in the word cloud. And we only visualize the top 100 words with the highest scores in the word cloud.

GS	*GW*	*Pr*	Succ.	Pert.	Query
✓			85.2	15.1	**3**
✓	✓		**89.3**	16.0	14
✓	✓	✓	**89.3**	**11.9**	25

Table 8: Ablation study on the impact of three key steps in ABP$_{\text{guide}}$ on attack success rate (Succ., %), word perturbation rate (Pert., %) and average query number (Query). *GS*, *GW* and *Pr* represent greedy search, guided walk and prune steps, respectively. ✓ indicates inclusion in the attack.

Model	Attack	IMDB	SST	MR
BERT	ABP$_{\text{free}}$	**98.7**	63.0	68.3
	ABP$_{\text{ens(model)-free}}$	97.8	55.6	64.7
	ABP$_{\text{ens(domain)-free}}$	97.6	**68.5**	**71.6**
	ABP$_{\text{ens(all)-free}}$	98.3	65.1	66.7
ALBERT	ABP$_{\text{free}}$	85.6	**73.4**	**78.6**
	ABP$_{\text{ens(model)-free}}$	87.1	66.0	73.6
	ABP$_{\text{ens(domain)-free}}$	88.4	72.5	76.0
	ABP$_{\text{ens(all)-free}}$	**89.5**	70.4	70.3
LSTM	ABP$_{\text{free}}$	**99.9**	62.0	**84.6**
	ABP$_{\text{ens(model)-free}}$	99.1	53.6	69.0
	ABP$_{\text{ens(domain)-free}}$	**99.9**	**66.5**	78.1
	ABP$_{\text{ens(all)-free}}$	99.4	59.0	67.2

Table 9: Attack success rate (Succ., %) of **ABP$_{\text{free}}$** and various ensemble attacks.

like "sleek" closely follow. Conversely, in texts with negative sentiment, the ABP scores are predominantly higher for words associated with negative sentiment, such as "wearisome" and "nausea". Overall, in sentiment classification tasks, we observe that sentiment-related words have the greatest impact on the model, which aligns with our expectations. Surprisingly, we discover that replacing positive sentiment words with their synonyms can lead the model to misclassify them as negative sentiment label. We speculate that this discrepancy is largely due to the limited availability of data for the model to learn from or the model's incomplete

Attack	**Natural**	**Unnatural**	**Uncertain**
Benign text	86.5	10.5	3.0
UAT	15.2	81.3	3.5
NUTS	38.1	51.2	10.7
ABP$_{\textbf{free}}$	**70.8**	21.6	7.6

Table 10: The percentage (%) of adversarial examples generated by various attacks using MR dataset that are judged by humans as natural, unnatural, and uncertain.

understanding of all sentiment-related words.

Through an analysis of the ABP scores and their implications, we have gained valuable insights into the inner workings of the model and the factors that contribute to its vulnerability. These insights are critical for understanding the limitations of the model and guiding future research to address these challenges and improve the model's overall performance.