GLARE: Generative Left-to-right AdversaRial Examples

Ryan Chi, Nathan Kim*, Patrick Liu*, Zander Lack, Ethan A. Chi
Department of Computer Science
Stanford University
ryanchi@cs.stanford.edu

Abstract

Recently, transformer models (Vaswani et al., 2017) have been applied to adversarial example generation—word-level substitution models utilizing BERT (Devlin et al., 2018; Garg and Ramakrishnan, 2020; Li et al., 2020a,b) have outperformed previous state-of-the-art approaches. Extending the paradigm of transformer-based generation of adversarial examples, we propose a novel textual adversarial example generation framework based on transformer language models: our method (GLARE) generates word- and span-level perturbations of input examples using ILM (Donahue et al., 2020), a GPT-2 language model finetuned to fill in masked spans. We demonstrate that GLARE achieves a superior performance to CLARE (the current state-of-the-art model) in terms of attack success rate and semantic similarity between the perturbed and original examples.[1]

1 Introduction

A large body of evidence (Goodfellow et al., 2014; Chakraborty et al., 2018; Kurakin et al., 2016) has demonstrated that otherwise high-performing ML models can be deceived by "adversarial" examples—small perturbations of existing data points wrongly classified by the model. However, generating adversarial *textual* examples can be challenging due to text's discrete structure, which makes generating fluent, believable perturbations difficult (Jin et al., 2019b; Morris et al., 2020a). Recently, large pretrained Transformer language models (Devlin et al., 2018; Liu et al., 2019) have successfully been adapted to generate adversarial examples. Typically, such frameworks use a masked language model (Devlin et al., 2018)'s pretrained word substitution objective to generate word-level replacements; combining several such replacements allows the generation of perturbations that are both locally fluent and globally adversarial. However, this approach allows only one token to be substituted at a time, due to the pretraining objective of masked language models; although several `[MASK]` tokens can be inserted repeatedly, the overall result is that generating multi-word sequences of text is difficult (Wang and Cho, 2019).

In this work, we suggest instead applying *generative* language models (Radford et al., 2019) to produce adversarial examples. These models can easily generate multiple tokens at a time, enabling a larger space of possible attacks. Specifically, our framework, **GLARE**, applies GPT-2 (Radford et al., 2019) to generate adversarial examples, augmented by Donahue et al. (2020)'s *infilling*, which allows the LM access to rightwards context. Our approach, which can be easily used to substitute existing MLM attack methods, outperforms existing strong approaches as measured by attack success rate, semantic similarity between the perturbed and original examples, and modification rate of perturbed examples.

2 Background

2.1 Adversarial Example Generation

Adversarial example generation is focused on attacking a **victim** model f; in particular, we focus on *black-box* examples, where the attack method has access to model outputs given an arbitrarily large number of model inputs, but not its parameters. An adversarial example, then, is some perturbation $\text{Perturb}(x)$ of an original example x which triggers an error in the victim model, i.e. $f(\text{Perturb}(x)) \neq f(x)$, while being close semantically to the original x. Typically, one measures semantic similarity by computing the similarity between vector representations of the initial and modified sentence.

[1]Full source code for this project is available at https://github.com/nathankim7/infilling-adversarial.

Proceedings of the 3rd Workshop on Evaluation and Comparison of NLP Systems, pages 44 - 50

2.2 Previous Approaches

Typically, an adversarial approach consists of some underlying set of perturbations; these can be at the subword level (e.g. typo introduction; Li et al., 2019), word level (e.g., word addition or deletion), or even sentence level (e.g., sentence paraphrasing; Iyyer et al., 2018). The iterated set of such perturbations represents the attack space from which an attack may be drawn, and an attack is considered "successful" for a particular example if a set of perturbations which flips the victim model's prediction can be found in the space. In practice, a standard search algorithm is typically applied to search through the space of perturbations for computational efficiency; these are typically implemented through a *framework*, such as TextAttack (Morris et al., 2020b) or OpenAttack (Zeng et al., 2021).

Modern adversarial methods typically apply a small set of perturbations computed via a masked language model. We can view most previous methods (Li et al., 2020b; Garg and Ramakrishnan, 2020; Li et al., 2020a) through the lens of the following broad operations (Li et al., 2020a):

- **Replace**: an existing token is masked and replaced with a new token.
- **Insert**: a `[MASK]` token is inserted, then to be replaced with a new token.
- **Delete**: a token is deleted.

Token replacement can be accomplished by computing vector similarity or manual dictionary lookups (Jin et al., 2019a); however, most competitive methods use masked language models (MLMs). BERTAttack (Li et al., 2020b) performs only **Replace** operations using BERT. BAE (Garg and Ramakrishnan, 2020) allows **Insert** operations simultaneously adjacent to substitutions. CLARE (Li et al., 2020a) allows all three operations. As all of these additions expand the attack space, their combination allows for an infinite space of new examples to be generated given enough exploration steps.

3 Methods

Like previous methods, GLARE utilizes the same fundamental **Replace** operation, where tokens from the input are replaced with neurally generated tokens. However, unlike previous approaches, we parameterize this replacement with a generative language model, allowing for the generation of arbitrarily large sequences. In particular, we apply *language-model infilling* (Donahue et al., 2020), which places both the leftwards and rightwards context of the original infill in the context window, allowing both sides to be considered during infilling (see Figure 1).

Specifically, GLARE entails the following steps, which closely follow previous approaches:

1. All possible replaceable **spans** are enumerated. Previous methods must limit spans to single tokens only due to the one-for-one nature of masked language model token replacement. Instead, GLARE defines a configurable hyperparameter c_{max} which controls the maximum number of contiguous tokens which may form a span.

2. The spans are **ranked** according to their Word Importance Ranking (Jin et al., 2019b): i.e. the difference between the score of the original example and the score after the span has been replaced by `[MASK]`.

3. The top k candidates are selected and **infilled** using a GPT-2 model fine-tuned via Donahue et al. (2020)'s approach on the dataset itself. As the length of the infill is theoretically unlimited, we constrain its length during the decode; the final replacement for an original span of length n may between $[n - e_{\text{max}}, n + e_{\text{max}}]$, where e_{max} is a configurable hyperparameter. We rerank the candidates by likelihood under the infilling model, picking the top candidate.

Unlike CLARE, we do not use Delete and Insert operations, as the infilling process naturally allows the length of the resulting sequence to change.

Overall, GLARE dramatically increases the scope of the attack space by permitting more natural decoding of longer sequences. By allowing multiple words to be masked and for multiple tokens to be added at any given step, vastly fewer replacement steps are required. Additionally, the joint generation of multi-word replacements allow for greater flexibility; candidates of multiple different lengths can be compared rather than being constrained to utilizing multiple Insert operations.

3.1 Variants

We ablate two variants of our model:

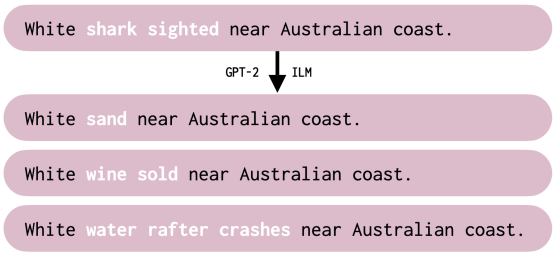

Figure 1: Illustrated example of infilling procedure.

- **GLARE$_{\text{single}}$** allows solely single-token replacements with no changes in length whatsoever—$c_{\max}$ is set to 1 and $e_{\max}$ is set to 0. Since GPT-2 is used solely for single-token replacement here, this approach is equivalent to a simple token-replacement strategy like BERTAttack (Li et al., 2020b), simply with a different model.

- **GLARE$_{\text{multi}}$** allows multi-word replacements: in our experiments, $c_{\max}$ is set to 3 and $e_{\max}$ is set to 3.

3.2 Framework

We implement GLARE as a recipe on TextAttack (Morris et al., 2020b). Specifically, the custom attack recipe consists of word-level replacements supplied by a fine-tuned version of an infilling GPT-2 model and constrained by the minimum sentence-wise cosine similarity score in a given example.

4 Experiments

Datasets We use the following datasets: Yelp Polarity (Zhang et al., 2015), AG News (Zhang et al., 2015), MultiNLI (Williams et al., 2018), and QNLI (Wang et al., 2018).

Victim Model We attack a BERT-base-uncased English model.

Metrics Evaluating adversarial attacks can be challenging, as attacks which achieve high success rate (successfully flipping a large fraction of model predictions) may be extremely obvious to a human reader due to a lack of fluency, coherency, or otherwise suspicious language (Morris et al., 2020a). We measure the following desiderata:

- **Attack success**: the percentage of model predictions successfully flipped, or Attack Success Rate (**A-rate**).

- **Distance from original example**: We measure modification rate (**Mod**), the mean fraction of words modified in each example, and (**Sim**), the cosine similarity between the original and perturbed text, as calculated by the Universal Sentence Encoder (Cer et al., 2018).

- **Fluency**: We measure perplexity (**PPL**) using a small (12-layer, 768-hidden, 12-heads, 117M parameters) non-finetuned GPT-2 model, as well as the average number of grammar errors (**GErr**) is the average number of grammatical errors introduced by each perturbed example.

Baselines We compare GLARE against prior attack methods: the non-neural TextFooler and the LLM-based BERT-Attack and CLARE (Section 2.2). Notably, CLARE is identical to our method except for the infilling method: fully generative rather than masked language modelling.[2]

5 Results

Overall, GLARE effectively attacks the victim model, achieving more fluent and grammatical attacks than baseline approaches (Table 1).

Notably, GLARE$_{\text{single}}$ achieves extremely strong performance as opposed to a method with an equivalent search space that uses BERT, BERTAttack, achieving an average of 8.3 points better on A-rate while achieving 0.04 higher Sim. Here, the search space is equivalent to BERTAttack; the advantage lies solely in using a better-parameterized GPT model.

GLARE$_{\text{multi}}$ generally performs better than GLARE$_{\text{single}}$. GLARE$_{\text{multi}}$ also achieves a 10.1 point better A-rate and 0.14 higher Sim than CLARE, another approach capable of changing token lengths – the GPT-2 infilling approach provides more flexibility and coherency to the attack.

6 Analysis

We are able to successfully outperform CLARE (the current SOTA) on a number of metrics: specifically, attack success rate, perplexity, and semantic similarity.

Effect of in-domain fine-tuning The infilling model used in our main experiments is fine-tuned

[2]Due to difficulties implementing the TEXTFOOLER and CLARE models with TEXTATTACK, the baseline values included in Table 2 were taken from (Li et al., 2020a).

	Yelp (PPL = 53.4)					AG News (PPL = 38.0)				
Model	**A-rate↑**	**Mod↓**	**PPL↓**	**GErr↓**	**Sim↑**	**A-rate↑**	**Mod↓**	**PPL↓**	**GErr↓**	**Sim↑**
TEXTFOOLER	77.0	16.6	163.3	1.23	0.70	81.7	23.6	177.5	1.27	0.83
BERTATTACK	71.8	10.7	90.8	0.27	0.72	63.4	**7.9**	90.6	**0.25**	0.71
CLARE	79.7	**10.3**	83.5	0.25	0.78	**84.7**	21.2	162.3	0.17	0.57
GLARE (single-word)	91.9	16.6	163.3	1.23	0.70	56.1	23.3	331.3	1.43	0.69
GLARE (variable-len)	**92.1**	56.7	**48.2**	**0.22**	**0.92**	79.0	69.77	**63.9**	1.69	**0.88**

	MNLI (PPL = 28.9)					QNLI (PPL = 37.9)				
Model	**A-rate↑**	**Mod↓**	**PPL↓**	**GErr↓**	**Sim↑**	**A-rate↑**	**Mod↓**	**PPL↓**	**GErr↓**	**Sim↑**
TEXTFOOLER	59.8	13.8	161.5	0.63	0.73	57.8	16.9	164.6	0.62	0.72
BERTATTACK	82.7	8.4	86.7	0.04	0.77	76.7	13.3	86.5	0.03	0.73
CLARE	88.1	7.5	82.7	**0.02**	0.82	83.8	11.8	76.7	**0.01**	0.78
GLARE (single-word)	**92.9**	**6.2**	77.9	0.23	**0.84**	**86.9**	**10.0**	72.9	0.22	0.87
GLARE (variable-len)	84.2	18.8	**60.2**	0.33	0.82	79.6	42.2	**55.6**	0.47	**0.89**

Table 1: Adversarial example performance compared on attack success rate (**A-rate**), modification rate (**Mod**), perplexity (**PPL**), number of increased grammar errors (**GErr**), and textual similarity (**Sim**) on four datasets. The perplexity of each dataset is marked in the header. ↑ (↓) represents which direction is more desirable. The best score per metric and dataset is bolded. Certain baseline results are drawn from Li et al. (2020a).

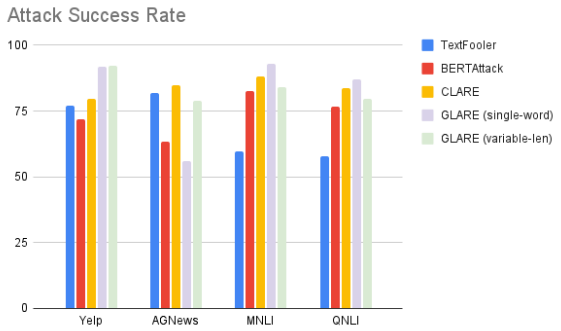

Figure 2: Comparison of attack success rates by different models.

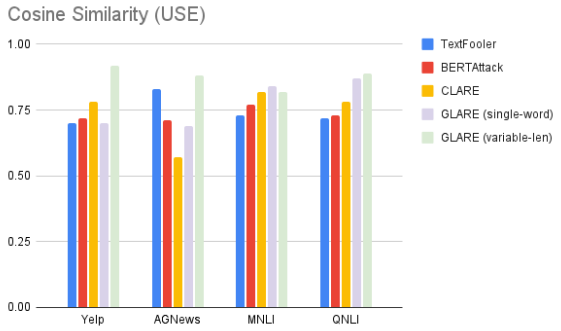

Figure 3: Comparison of cosine similarities between original and perturbed text by different models.

on in-domain data. To examine the impact of this fine-tuning on attack rates, we provide preliminary experiments on a GLARE model utilizing an OOD GPT-2 infilling model fine-tuned on the ROCStories corpus (Donahue et al., 2020). We use Donahue et al. (2020)'s fine-tuned checkpoints and otherwise use identical settings to $\text{GLARE}_{\text{single}}$. The results are inconclusive, though preliminary metrics suggest that fine-tuning the GPT-2 model does not appear to as successful as we would like, demonstrated by the fact that the ILM model fine-tuned on stories was able to often match or even outperform the corresponding model finetuned on the specific dataset (Table 2, Appendix).

Modification rate We note that our model suffers from a higher modification rate than CLARE. Although this is ostensibly undesirable, one benefit of a larger modification rate is that attacks are less likely to comprise simple polarity switches (e.g., "The food was delicious" → "The food was terrible"), which feature low modification rates but are not satisfactory adversarial examples as they necessitate a change in the example's gold label. A long-term goal is lower modification rate while maintaining the same fluent adversarial substitutions.

Example Length We note that longer inputs generally experience higher similarity scores when comparing their perturbed and original examples. We believe this is because the longer context gives the model a wider range of opportunities to perform an adversarial attack, as well as allowing the model

a better glimpse into the semantic and syntactic structure of the example.

7 Conclusion

In this work we propose GLARE, a novel method for generating textual adversarial examples for use in adversarial attacks. GLARE operates by selecting spans in training examples to be masked out and then replaced with variable-length spans from a left-to-right generative model, bypassing restrictions on both the space of possible perturbations and the context available to each replacement step imposed by the single-token replacement strategy in existing methods. Our experiments show that GLARE outperforms contemporary methods in attack success, perplexity, grammatical correctness and semantic preservation when generating adversarial examples for a variety of classification benchmarks, and indicate that input text perturbation can be a promising application of left-to-right generative models for text infilling.

8 Acknowledgements

We would like to thank the anonymous reviewers for their insightful and thorough feedback, as well as Chris Donahue and Mina Lee for their assistance in adapting the ILM model to our system. Furthermore, we are grateful to Shikhar Murty and Akshay Smit for helpful discussions.

References

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder.

Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. 2018. Adversarial attacks and defences: A survey.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Siddhant Garg and Goutham Ramakrishnan. 2020. Bae: Bert-based adversarial examples for text classification.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019a. Is bert really robust? a strong baseline for natural language attack on text classification and entailment.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019b. Is bert really robust? natural language attack on text classification and entailment. *arXiv preprint arXiv:1907.11932*.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. 2016. Adversarial examples in the physical world.

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2020a. Contextualized perturbation for textual adversarial attack.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2019. Textbugger: Generating adversarial text against real-world applications. *Proceedings 2019 Network and Distributed System Security Symposium*.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020b. Bert-attack: Adversarial attack against bert using bert.

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*.

John X. Morris, Eli Lifland, Jack Lanchantin, Yangfeng Ji, and Yanjun Qi. 2020a. Reevaluating adversarial examples in natural language.

John X. Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020b. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Alex Wang and Kyunghyun Cho. 2019. Bert has a mouth, and it must speak: Bert as a markov random field language model. *arXiv preprint arXiv:1902.04094*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Guoyang Zeng, Fanchao Qi, Qianrui Zhou, Tingji Zhang, Bairu Hou, Yuan Zang, Zhiyuan Liu, and Maosong Sun. 2021. Openattack: An open-source textual adversarial attack toolkit. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 363–371.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *CoRR*, abs/1509.01626.

A GLARE Ablations

We perform a comparison of $\text{GLARE}_{\text{single}}$ with an out-of-domain version of the same attack. $\text{GLARE}_{\text{OOD}}$ uses an ILM model trained on the ROCStories short story corpus (Mostafazadeh et al., 2016), as provided by the authors, and performs single-token replacements like $\text{GLARE}_{\text{single}}$. We note that $\text{GLARE}_{\text{OOD}}$ outperforms $\text{GLARE}_{\text{single}}$ on almost all metrics across all of our datasets, as seen in Table 2.

	Yelp (PPL = 53.4)					**AG News** (PPL = 38.0)				
Model	**A-rate**↑	**Mod**↓	**PPL**↓	**GErr**↓	**Sim**↑	**A-rate**↑	**Mod**↓	**PPL**↓	**GErr**↓	**Sim**↑
GLARE (single-word)	91.9	16.6	163.3	1.23	0.70	56.1	23.3	331.3	1.43	0.69
GLARE (single, OOD)	**93.5**	**11.2**	**63.6**	**0.15**	**0.92**	**70.3**	**18.9**	**124.4**	**0.27**	**0.86**

	MNLI (PPL = 28.9)					**QNLI** (PPL = 37.9)				
Model	**A-rate**↑	**Mod**↓	**PPL**↓	**GErr**↓	**Sim**↑	**A-rate**↑	**Mod**↓	**PPL**↓	**GErr**↓	**Sim**↑
GLARE (single-word)	92.9	6.2	77.9	0.23	**0.84**	86.9	10.0	**72.9**	0.22	**0.87**
GLARE (single, OOD)	**93.6**	**5.8**	**64.6**	**0.15**	**0.84**	**91.1**	**9.7**	77.3	**0.18**	**0.87**

Table 2: Adversarial example performance of $\text{GLARE}_{\text{single}}$ and $\text{GLARE}_{\text{OOD}}$.