Transferability of BERT-Attack

Aidan Beery
Oregon State University
beerya@oregonstate.edu

Abstract

Current approaches to attacking large language models rely on the attacker having access to the parameters of the target model, a scenario which is increasingly unrealistic as LLM-based systems enter production environments with a greater degree of proprietary technology and security. As these powerful transformer-based systems continue to achieve wide-spread adoption, there is interest in verifying their robustness to prevent user harm. We observe that many of the current state-of-the-art LLMs derive their architecture from a single common ancestor, BERT. We hypothesize that, if an attacker has knowledge of the data distribution for the downstream task, they can attack one of these BERT-derivative models by using an inexpensive fine-tuned instance of BERT as a surrogate model, and apply existing white-box adversarial attack frameworks to this otherwise obfuscated target model. We show merit in the concept of BERT surrogates for black-box attacks against encoder-based LLMs in the case of text classification tasks. On average, we observe a 17% degradation in classifier accuracy in a black-box attack against a variety of BERT-derived architectures using adversarial examples crafted using BERT-ATTACK.

1 Introduction

Adversarial attacks have become a widely demonstrated attack vector against modern deep learning models Goodfellow et al. (2015). These attacks take an existing model $f : x \rightarrow \hat{y}$ and generate some sample x' such that $f(x') \neq f(x)$. An adversary would like for such an adversarial input to be indistinguishable from the clean input to a human observer, but to maximize classifier error rate on the sample, to avoid detection in a real-world attack scenario. We measure this perceptual similarity quality as the magnitude of the perturbation to x. To guide adversarial example generation, we often will say that we would like to find some perturbation $f(x + \eta) \neq f(x)$ and $||\eta|| < \epsilon$, where ϵ is some bound on the amount of perturbation that the adversary finds acceptable for their given attack scenario.

In continuous domains, such as visual or audio inputs, the generation of adversarial examples can naturally be modeled as an optimization problem given these two constraints. In such a case, we can adjust our first constraint to say that we would like to find the x' in the input space such that $x' = \underset{\eta}{argmax}\ \mathcal{L}(\theta, \mathbf{x} + \eta, y)$, where $\mathcal{L}$ is some continuous loss function.

However, in text and language, the input space is not continuous. Modern natural language processing techniques represent language as sequences of discrete tokens, typically represented in some embedding space. Standard adversarial attack generation methods, like DeepFool, FGSM, and PGD, optimize a perturbation over a continuous input space. Additionally, the requirements for an adversarial example to be indistinguishable from the original input increase in complexity significantly in the language domain. A token which may be very close in the embedding space might have drastically different syntactic or grammatical meaning when substituted into a sentence when transfered to the input space, breaking the illusion of an unperturbed input. In the language domain, it would be advantageous for an adversarial sample to be semantically similar to the original x, grammatically consistent with the rules of the origin language, and syntactically correct.

Initial attempts to design adversarial attacks against language models began with rule-based methods, based on word erasure Li et al. (2017), character-level typo injection Jones et al. (2020), and phrase injection Liang et al. (2018). These rule-based methods were markedly slow, generally requiring minutes to generate a single adversarial sample. They only achieved modest attack success rates and did not demonstrate adequate transferability or robustness against simple defenses. With the recent rise of large language models as highly capable models for a wide range of down-stream natural language processing (NLP) tasks Devlin et al. (2019), there was a sudden research need to evaluate the robustness and vulnerability of these systems as they are deployed in more and more real-world contexts. Concurrently, it was demonstrated that these same models were ideal candidates for generating adversarial examples in the language space. Large language models are, by nature, highly capable at generating an optimal output sequence given some token sequence. As these models are pre-trained on large corpora of English text, it would not require any custom objective to ensure grammatical, semantic, and syntactic consistency between the input and the adversarial sample were an adversary to use a BERT-style large language model to attack NLP systems.

This premise gave rise to BERT-ATTACK, a white box adversarial attack designed to target BERT-style large language models by designing an iterative attack generation algorithm using a pre-trained BERT model. Though highly effective, this model's key limitation was that it required the weights of the target model to generate samples against. This is not a realistic threat model in most cases. Many models begin with a pre-trained BERT language model as their base representation learner, but will fine-tune the model on the desired downstream task. Additionally, as of 2023, there are a wide variety of encoder transformer models for language understanding, many of which are based on the original BERT architecture Xia et al..

This raises the core research question of our investigation - do the adversarial attacks generated by BERT-ATTACK transfer to other BERT-like models? If so, can a fine-tuned BERT model be used as a surrogate for our target learner f, assuming f is some BERT-derivative? We study the case of three BERT-derivative models - RoBERTa, Nystromformer, and DistilBERT. Each bases its original architecture off BERT, but has unique changes to the architecture, data distribution, or training paradigm which provide desirable properties for a robust study of attack transferability. We fine-tune these models on five downstream tasks - three text classification, and two natural language inference tasks.

2 Related Work

We review a brief history of text-based adversarial attacks and how they generate adversarial samples against a variety of NLP models. We also review the premise of BERT and it's sibling models to establish a foundation of how large language models work.

2.1 Text-Based Adversarial Attacks

Adversarial attacks in the language domain must find a perturbation in a discrete space, making traditional optimization objectives incompatible with the input space. Additionally, we would like for these perturbed inputs to be grammatically consistent with the origin language, follow most, if not all, syntax rules of the language, and be semantically similar to the original output such that a human reader would not identify which sample is perturbed. Early approaches relied on rule-based text editing attacks. One example of such a rule-based attack Liang et al. (2018) uses the loss gradients of a character-based fully-connected neural network to identify 'hot' characters for each sample $x \in \mathcal{D}$, defining a 'hot' word as any with three or more hot characters in the word. Using this information, the authors also backpropagate to identify hot sample phrases to find the points in sample x which are most vulnerable to perturbation. From this, an attacker manually modifies the sample, removing, adding, or substituting a hot sample phrase using the hot words identified by the character-level backpropagation. This attack has multiple limitations, firstly being the manual crafting required by the adversary to create an adversarial sample. This backpropagation formulation only works for identifying hot phrases in fully-connected neural networks, though this architecture has fallen out of favor for most natural language processing tasks even by 2018, when this attack was originally formulated. Additionally. by introducing errors based only on hot phrase insertion or deletion, often the example attacks presented by the authors are not semantically consistent with the original input, introducing what the authors refer to as ”forged facts”.

More robust attacks designed on the same principle of text editing followed shortly. Jones et. al. review a series of automatic and hand-crafted methods for introducing token-level perturbations, referred to as adversarial typos, against tokenized encoder models like BERT and other large language models Jones et al. (2020). Though this attack is advantageous in its ability to target more robust state-of-the-art language models, it still introduces perturbations which may be detectable by an acute human reader. Additionally, basic filtering methods such as spell-checking inputs can effectively defeat such attacks. Instead of token-level attacks or manually edited sequences, there is a desire for an attack which makes perturbations at a sequence level, without introducing typos or other syntactic violations, and can be performed automatically. We see preliminary examples of such work in Li et al. (2017), where key phrases are automatically removed from an input sequence based on a word importance rating. By masking out each token at input time and comparing the confidence vector of $f(x)$ to $f(x_{w_i})$, where x_{w_i} is input x with token w_i masked out, Li et. al. identify key words and phrases which have the most significant impact on the classifier, and seek to remove those tokens from x. However, by removing key phrases, context can often be omitted from the original input, violating the principle of semantic similarity and requiring significantly larger perturbations than may be necessary.

Our investigation explores word-importance-rating based sequence-level attacks which automatically generate attacks against BERT-style models. The first of such attacks, TextFooler, uses a cosine similarity matrix of words in the embedding space to identify words the authors describe as being synonyms, and begins by replacing each most-important-word with a word near it in the embedding space Jin et al. (2020). From the set of candidate substitution words, TextFooler picks the word which causes the largest disruption to the classifier without exceeding the perturbation bound ϵ. This attack proved to be significantly faster than rule-based attacks presented in previous work, and could effectively attack state-of-the-art models such as BERT on downstream text classification and natural language inference (NLI) tasks. However, needing to iterate over every candidate word (which the authors set to 50 candidates) still is a significant computational expense. Additionally, relying on cosine similarity in the word embedding space does not necessarily indicate that our perturbations will be perfectly grammatically and semantically consistent. A simple word substitution approach makes little guarantees about the sentence-level consistency, though TextFooler was able to achieve high rates of grammatical correctness and, in qualitative analysis, achieved many of the properties desirable for adversarial samples previously defined.

We know now that BERT models are very robust for learning representations in a language domain which allow for functional reconstruction of often very natural sounding language outputs. Current theory indicates this to be thanks to the pretraining objective of BERT-style models allowing the model to learn an efficient representation of the rules and form of a language without requiring an expert-formed ruleset. We seek to attack these models with inputs which are similarly consistent with the rules and intent of the target language in order to evade human detection. BERT-ATTACK leverages the advantages of BERT, its ability to apply to downstream NLP tasks with very little training by leveraging the pretrained model, to generate adversarial samples which are grammatically, syntactically, and semantically similar to that of the target sample Li et al. (2020). Similarly to TextFooler, BERT-ATTACK relies on word-importance score to identify tokens in an input most susceptible to perturbation. However, instead of relying on an expensive and potentially inaccurate word embedding cosine similarity matrix to generate a set of candidate substitutions against the target token w_i, BERT-ATTACK uses BERT, in it's pre-training masked-language-modeling objective, to generate a set of most probable substitutes for w_i. Unlike simple cosine similarity substitution, using masked-token prediction with BERT enables the attack generation to consider all tokens in the input, picking a substitute token which maintains grammatical consistency with the surrounding tokens and would be the most likely word to appear in that phrase based on BERT's pretraining data distribution. This provides a robust framework for generating attacks which are indistinguishable from real test-time inputs to a human observer, but still maximize classifier error rates within our perturbation bound ϵ.

2.2 Large Language Models

Modern state-of-the-art large language models are based around the transformer architecture, which enables very-large networks to handle features in discrete, sequential spaces by using the attention operator to compute the relationship between one token and the rest of the tokens in the input, for each token w_i in the input x. BERT, or Bi-directional Encoder Representations from Transformers, is ostensibly the seminal architecture in this line of research Devlin et al. (2019). BERT leverages a large corpora of unstructured text from Wikipedia and freely-available English literature. By taking unlabeled sentences, corrupting parts of the input, and attempting to predict the missing tokens, the model encodes complex relationships between words, and demonstrates the ability to learn robust language representations. This self-supervised pretraining objective function, ofen referred to as masked language modeling, has become the foundation for many other popular NLP models. Because BERT and other LLMs are pretrained on very large amounts of data, these model can be fine-tuned to new tasks relatively quickly, making them an attractive choice for deployment in a variety of real-world scenarios, such as digital voice assistants, natural language interfaces for digital systems, and sentiment-enabled recommendation systems.

However, both training and inference with BERT can be prohibitively computationally expensive. DistilBERT seeks to address this while retaining the same capability to learn effective language representations. To accomplish this, they leverage knowledge distillation to train a smaller model to emulate the behavior of BERT Sanh et al. (2020). By optimizing DistilBERT to predict the same output probabilities as BERT during pretraining, the authors design a model which retains 97% of the performance on benchmark NLP tasks. DistilBERT bases its architecture on BERT, but reduces the number of hidden layers from 12 to 6. This lowers the number of model parameters by 40%, enabling faster training and fine-tuning. As knowledge distillation has been used in defensive applications before Papernot et al. (2016), we find a study of the transferability to BERT to it's smaller, distilled cousin to be of particular interest for this study.

RoBERTa aims to surpass the performance of BERT by both leveraging a larger pretraining corpus and by modifying the pretraining objective task. Liu et al. (2019). The authors replicate BERT's architecture while performing ablations on the pretraining formulation. They identify that encoder transformers respond positively to significantly larger pretraining datasets and longer pretraining schedules with larger batch sizes. Furthermore, they propose a dynamic masking method, randomly altering which tokens are masked during the masked language modeling task, which helps to improve the performance of RoBERTa on NLP benchmarks compared to BERT. Additionally, Liu et. al. drop the next-sentence prediction task which Devlin et. al. originally include in BERT's pretraining objective, relying solely on masked language modeling for pretraining. Despite this, BERT and RoBERTa share the same architecture. This again presents an interesting case study for transferability of attacks between large language models, as one may expect that models sharing the same architecture will similarly share the same vulnerabilities. However, being trained on a different and vastly larger training dataset, with a modified pretraining formulation, the model's internal parameters, and therefore learned representations, may be considerably different than those of BERT.

All of the aforementioned models suffer from the same computational complexity constraint, as the self-attention operator itself has $O(n^2)$ time complexity. This means that, as sequence length increases, training and inference time increase exponentially. This makes BERT-style models intractable for long inputs, often being limited to inputs of 128 to 512 tokens. The Nystromformer architecture seeks to address this runtime complexity constraint by replacing the attention operator with a linear-time approximation to enable the processing of longer inputs and accelerate model training and inference Xiong et al. (2021). By using the Nystrom method for approximating the softmax matrix used in the self-attention operation, Xiong et. al. are able to achieve $O(n)$ runtime on this self-attention operation. Though their model architecture is vastly different than BERT, they seek to emulate BERT's language representation learning performance, using the same pretraining objective and dataset.

3 Methods

We replicate the attack described in Li et al. (2020), using BERT to generate candidate substitutions for a given input x. BERT-ATTACK is a white-box attack, and requires a fine-tuned BERT instance to generate attacks against. We fine-tune an instance of `bert-base-cased` on five downstream tasks: IMDB, Yelp, `ag_news`, MNLI, and SNLI. IMDB and Yelp both are two-class text classification sentiment analysis tasks. The model is given a review and must classify that review, whether it be of a movie or restaurant, as positive or negative. These binary sentiment classification tasks are easily solved with even older architectures, such as BiLTSM models, and therefore we expect high performance from any of our models on these tasks. The `ag_news` dataset consists of samples of news headlines and the associated subheader or first sentence of the news report. Each news snipped is labeled as one of four classes: World news, Business, Scientific/Technology, or Sports. MNLI and SNLI are entailment tasks. Our model is given two inputs, concatenated and separated with a special token, and must determine if the first input entails the second input, meaning the second input is a logical consequence of the first. The model will attempt to classify each pair of inputs as entailed (meaning the hypothesis is a natural consequence of the text), contradicted (the hypothesis is not true based on the text), or ambiguous.

After fine-tuning five instance of BERT on these downstream tasks, we generate 100 adversarial samples against each of these fine-tuned models using BERT-ATTACK Li et al. (2020). Additionally, we fine-tune an instance of each of our target models; DistilBERT, RoBERTa, and Nystromformer, on each downstream task, to yield 15 total target models and 5 victim models. We evaluate each of these 20 models first on 100 clean-label samples, reporting the test-time classification accuracy. We take the samples generated in the white-box attack on our fine-tuned BERT instances and, for each downstream task, attack the task-specific target model using adversarial inputs generated against the associated BERT instance for that task. In this sense, the adversary in this attack scenario has knowledge of two things - the underlying data distribution and the type of target model (in this case, BERT-style models). We use fine-tuned instances of BERT as surrogates for these target models and attempt to cause untargeted misclassifications in a black-box attack scenario. We evaluate each target model for a given task with the same set of samples, but having made adversarial perturbations to each x. We compare the classification accuracy on the set of 100 clean samples to the adversarial generated samples to evaluate the efficacy of our attack.

	Training	Test
IMDB	25,000	25,000
Yelp	650,000	50,000
ag_news	120,000	7,600
SNLI	560,000	10,000
MNLI	393,000	19,600

Table 1: Properties of five downstream task datasets

Each model is fine-tuned using 6 Nvidia Tesla T4 GPUs using CUDA 11.8 on CentOS 7. Attack generation is performed using a single T6 GPU, and we find that attacks take, on average, 5-15 seconds to generate, depending on the length of the attacked input. We use the same attack parameters as defined by Li et al. (2020), with byte-pair encoding substitution, $k = 48$ candidates per substitution, and a fixed threshold for cutting off unsuitable candidate attacks. [1]

4 Experimental Results

We first attempt to replicate the results of BERT-ATTACK to ensure our attack parameters allow for attack generation which is consistent with the performance observed by Li et. al. From these generated adversarial samples, we perform our black-box attack against DistilBERT, RoBERTa, and Nystromformer to assess the viability of BERT as a surrogate model for attacking BERT-style large language models.

4.1 Replication of BERT-ATTACK

	Clean Accuracy	Attack Accuracy	Li et. al.	Perturbation Rate
ag_news	0.94	0.24	0.11	0.171
Yelp	0.96	0.05	0.05	0.081
IMDB	0.92	0.06	0.11	0.039
MNLI	0.84	0.01	0.07	0.100
SNLI	0.8	0.02	0.08	0.129

Table 2: Results of our white-box attack compared to the results presented by Li et. al.

[1] https://github.com/LinyangLee/BERT-Attack

We observe similar performance on our text classification tasks, particularly with our binary sentiment classification tasks, Yelp and IMDB. We find that our generated attacks are notably weaker than those presented by Li et. al. for the 4-class classification task on the ag_news dataset. We note that the authors do not publish the subset of samples they use at test-time for their attack generation, so an exact replication of their results is not feasible.

Our performance against the two natural language entailment models seems to indicate that we have an attack which is significantly more potent than that presented in the original work. This is immediately suspicious, as we have not modified the parameters of the original attack in any substantial way such that we would expect to improve model misclassification rate. We also note that we achieve this purported performance with perturbation rates near those reported by the authors (0.100 v.s. 0.088, and 0.129 v.s. 0.124) for these attacks, so we find it unlikely that we generated significantly more effective perturbations such that classifier accuracy drops to near zero.

4.2 Attacking Text Classification Models

For each of the three text classification tasks, we take the 100 adversarial samples generated in our white-box BERT-ATTACK replication study and evaluate each of our three models' performance on these samples.

	RoBERTa		DistilBERT		Nystromformer	
	Clean Acc.	Attack Acc.	Clean Acc.	Attack Acc.	Clean Acc.	Attack Acc.
ag_news	0.95	0.76	0.94	0.74	0.94	0.76
Yelp	0.54	0.54	0.64	0.34	0.64	0.62
IMDB	0.91	0.84	0.86	0.75	0.86	0.72

Table 3: Results of our test-time accuracy against clean and attacked samples for text classification models

In the black-box attack scenario, there is limited yet still measurable viability in the use of BERT as a surrogate model for transferring attacks to BERT-like models. In all three of our ag_news models, we find that classifier accuracy drops from above-90% success to, on average, 75% successful classification. Though this is not as significant a performance degredation as the 0.25% accuracy achieved in the white-box case against BERT, it still demonstrates that these samples retain some of their adversarial properties when transfered to other BERT-style models fine-tuned for the same downstream task. Similarly, for models trained on IMDB, we observe a slight but consistent 12% average reduction in classifier accuracy, regardless of model architectural properties.

We find that the performance against clean samples for models fine-tuned on the Yelp dataset is significantly worse than that of our baseline BERT model. We also find the lowest rate of successful attack transferability for these downstream task models. As the performance on this task in the baseline is already near random performance in the case of RoBERTa, we do not anticipate any attack being able to create a significant increase in classifier error rate. We do note that, in the case of DistilBERT uniquely, classifier accuracy drops below random guessing, to 0.34.

4.3 Attacking Natural Language Inference Models

	RoBERTa		DistilBERT		Nystromformer	
	Clean Acc.	Attack Acc.	Clean Acc.	Attack Acc.	Clean Acc.	Attack Acc.
SNLI	0.90	0.90	0.86	0.86	0.87	0.88
MNLI	0.85	0.84	0.80	0.81	0.83	0.39

Table 4: Results of our test-time accuracy against clean and attacked samples for NLI models

We find our attack does not transfer to our models fine-tuned NLI tasks. Across all three of our target models, and for both entailment tasks, classifier performance is only degraded in the case of Nystromformer fine-tuned against MNLI, for which classifier performance is actually degraded significantly. However, because these results are not repeatable for the SNLI task, even on the Nystromformer architecture, it is challenging to say whether this is a potential indicator of attack transferability against NLI models or rather a statistical anomaly. [2]

5 Discussion

We observe limited, but noteworthy, transferability of adversarial samples generated with BERT-ATTACK when used against fine-tuned instances of models sharing architectural properties with BERT. This enables a novel black-box attack against large language models for text classification tasks. Leveraging the rich language representation encoded in pretrained instances of BERT, we are able to successfully replicate many of the key results presented by Li et. al. in a white-box attack against fine-tuned BERT models. Using word-importance scoring, vulnerable words are first identified in each input sequence, and then replaced with candidates which maintain the consistency and integrity of the original sample by applying BERT's masked language modeling objective in an inference setting. 50 similar candidate words are evaluated, and the attack formulation picks the word which maximizes classifier error rate, while remaining within our perturbation bound ϵ.

We find that, when an attacker knows the model is a BERT-style architecture, and knows the training data distribution of the downstream task, they are able to conduct a black-box attack against this target model by fine-tuning a BERT instance on this data distribution, and using BERT-ATTACK to generate adversarial samples. We find that the variety of architectural differences between our three chosen target models does not create a significant effect on the transferability of this attack. From this, we derive three interesting conclusions about BERT-ATTACK and the vulnerabilities of encoder-based transformer models. First, we identify that knowledge distillation is not an effective defense against adversarial attacks in the language domain. Defensive distillation has been broken in visual domains for some time, but we find it interesting that we need not change our attack parameters at all to achieve transferability to a distilled BERT. Secondly, we find that the attacker does not need to know the underlying data distribution of the pretraining dataset in order to conduct a black-box attack against an encoder-based transformer model. RoBERTa has a different pretraining dataset than BERT, and uses a very different formulation for its pretraining objective, and yet our attack transferability to RoBERTa is the same as it is to DistilBERT. Finally, we find that it is not the self-attention mechanism itself which is vulnerable to adversarial attacks. By replacing the self-attention mechanism with a Nystrom approximation, we are unable to introduce a significant improvement in model robustness to this black-box attack.

5.1 Limitations

However, though we see evidence that BERT-ATTACK shows promise of limited transferability, we also acknowledge that our attack results exhibits some serious limitations which would need further investigation to make any robust claims about the generalizability of this attack formulation. Firstly, we note the failure to replicate the baseline BERT model's clean sample performance on the Yelp test-time dataset with our three target models. The performance achieved by these three fine-tuned models is considerably lower than we would anticipate for a simple binary sentiment classification task. We note that the authors only provided the pretraining parameters for text classification in their source code [3], and do not elaborate on specific fine-tuning parameters for each individual subtask. We follow these parameters exactly in our replication, with the exception of increasing the number of output labels from 2 to 4 for models trained on the `ag_news` dataset. We believe two factors may contribute to our downstream models' lackluster performance on clean Yelp samples: a significantly larger training dataset than comparable comparable binary sentiment classification tasks (see 1) leading to fine-tuning parameter choices being suitable for the smaller task and not for the larger task, and architectural differences in our three target models compared to the original BERT model necessitating different fine-tuning parameter choices than those originally presented in Li et al. (2020) for these BERT-derivative models. Future work should include a thorough ablation

[2] We do note that these results against Nystromformer on MNLI were repeatable, even if abnormal

[3] https://github.com/LinyangLee/BERT-Attack

of the fine-tuning parameters for these three downstream models, and select specific fine-tuning schedules for each of the three distinct tasks.

We find that our attacks do not transfer to the natural language inference task-specific models. Additionally, we observe bizarre and suspiciously strong attack behavior when replicating attacks against BERT models fine-tuned on NLI tasks (see 2). Our hypothesis is that this abnormal behavior has to do with the mechanism for concatenation and input transformation to make datasets for these task formulations compatible with BERT. BERT models expect one input, consisting of a sequence of tokens. However, NLI tasks often consist of two inputs, with a single label representing the relationship between the first input and the second input. To address this, we follow a standard practice of concatenating both inputs, separating them with a special `[SEP]` token denoting a separation between two contiguous parts of the input. BERT's pretraining dataset contains these `[SEP]` tokens between sentences, and it is a recognized special token in the embedding space. However, the authors do not specify exactly how they transform the inputs for these NLI task-specific models to make it compatible with BERT's training paradigm, so they may have made a significantly different architectural decision than we did in our work, which could have contributed to attacks generated against our NLI-specific BERT models not transferring to target models.

A key limitation with how we present our results is a lack of a black-box attack against BERT. Such a black-box baseline would allow us to make more rigorous claims about how the different architectural properties of our target models impact the transferability of a BERT-surrogate based black-box attack. However, designing an experiment to use the same architecture for both our surrogate model and our target model is challenging. Our best idea currently for future work would be to use a BERT model with some intermediate fine-tuning on a separate, more general downstream task, or a random initialization of parameters for the fine-tuning schedule. With unlimited computational resources, we could pretrain a new base BERT instance, using random parameter initialization to generate a target BERT sufficiently distinct from our white-box victim BERT model.

6 Conclusions

We find potential in the future use of BERT as a surrogate model to craft adversarial samples against other encoder-based transformer models. An attacker with only knowledge of the underlying data distribution of the downstream task, and the architectural category of the model (e.g. if the model is encoder-only, is it derived from BERT, etc.), can generate samples which cause a modest degradation in classifier accuracy by fine-tuning a pre-trained BERT instance on this downstream task. Such an attack paradigm enables the generation of adversarial samples which are grammatically, syntactically, and semantically consistent with the clean sample thanks to the power of pretrained large language models to generate semantically consistent, grammatically correct outputs in a discrete, tokenized space. Future work should focus on a more robust evaluation of this attack paradigm on a wider variety of target models and downstream tasks, as well as define a baseline black-box attack against a BERT target model.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. (arXiv:1810.04805), May 2019. doi: 10.48550/arXiv.1810.04805. URL http://arxiv.org/abs/1810.04805. arXiv:1810.04805 [cs].

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. (arXiv:1412.6572), March 2015. doi: 10.48550/arXiv.1412.6572. URL http://arxiv.org/abs/1412.6572. arXiv:1412.6572 [cs, stat].

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. (arXiv:1907.11932), April 2020. doi: 10.48550/arXiv.1907.11932. URL http://arxiv.org/abs/1907.11932. arXiv:1907.11932 [cs].

Erik Jones, Robin Jia, Aditi Raghunathan, and Percy Liang. Robust encodings: A framework for combating adversarial typos. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2752–2765, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.245. URL https://aclanthology.org/2020.acl-main.245.

Jiwei Li, Will Monroe, and Dan Jurafsky. Understanding neural networks through representation erasure. (arXiv:1612.08220), January 2017. doi: 10.48550/arXiv.1612.08220. URL http://arxiv.org/abs/1612.08220. arXiv:1612.08220 [cs].

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. Bert-attack: Adversarial attack against bert using bert. (arXiv:2004.09984), October 2020. doi: 10.48550/arXiv.2004.09984. URL http://arxiv.org/abs/2004.09984. arXiv:2004.09984 [cs].

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. Deep text classification can be fooled. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pp. 4208–4215, July 2018. doi: 10.24963/ijcai.2018/585. URL http://arxiv.org/abs/1704.08006. arXiv:1704.08006 [cs].

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. (arXiv:1907.11692), July 2019. doi: 10.48550/arXiv.1907.11692. URL http://arxiv.org/abs/1907.11692. arXiv:1907.11692 [cs].

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. (arXiv:1511.04508), March 2016. doi: 10.48550/arXiv.1511.04508. URL http://arxiv.org/abs/1511.04508. arXiv:1511.04508 [cs, stat].

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. (arXiv:1910.01108), February 2020. doi: 10.48550/arXiv.1910.01108. URL http://arxiv.org/abs/1910.01108. arXiv:1910.01108 [cs].

Patrick Xia, Shijie Wu, and Benjamin Van Durme. Which *bert? a survey organizing contextualized encoders.

Yunyang Xiong, Zhanpeng Zeng, Rudrasis Chakraborty, Mingxing Tan, Glenn Fung, Yin Li, and Vikas Singh. Nyströmformer: A nyström-based algorithm for approximating self-attention. (arXiv:2102.03902), March 2021. doi: 10.48550/arXiv.2102.03902. URL http://arxiv.org/abs/2102.03902. arXiv:2102.03902 [cs].