Grenze International Journal of Engineering and Technology, January Issue

Trigger Optimization for Black-Box Universal Adversarial Attacks on Text Classifiers

Swayam Desai[1], Yash Pokar[2], Dipti Rana[3] and Rupa Mehta[4]
[1-4]Sardar Vallabhbhai National Institute of Technology, Surat, India
Email: {swayamdesai62, pokar.yashusmile}@gmail.com,{dpr, rgm}@coed.svnit.ac.in

***Abstract*— Adversarial attacks significantly undermine the robustness of deep learning models, posing a major challenge to their deployment in critical applications. Universal adversarial perturbations pose a formidable threat due to their ability to mislead models for various inputs. This paper introduces a novel method to attack text classifiers in a black-box setting using universal adversarial attacks. It accomplishes this by enhancing trigger words, which are specific sequences of words that, when added to any input, cause consistent misclassification. The proposed method utilizes the Genetic Algorithm (GA) and differs from the conventional white-box attack as it does not need internal architectures or gradients within the model. Instead, it iteratively evolves a population of adversarial triggers, effectively navigating the search space to maintain fluency and semantic coherence in the generated sequences. We achieved a high rate of successful attacks across the Bi-LSTM and BERT models while preserving the naturalness of the input text. Our findings demonstrate the potential of GA-optimized universal triggers as a formidable and practical tool for adversarial attacks in real-world situations where only input-output access to the target model is available.**

***Index Terms*—Universal Adversarial Perturbations, Genetic Algorithm, Black-Box Attacks, Text Classifiers and Adversarial Triggers.**

I. INTRODUCTION

Natural Language Processing (NLP) has seen significant advancements in recent years, primarily due to developments in deep learning architectures. The importance of their performance and dependability has grown accordingly with their increasingly crucial roles in diverse applications. Nevertheless, the swift progress in deep learning technologies has underscored a significant weakness: the vulnerability to adversarial assaults [1,2]. Such attacks aim to intricately alter input data in manners that can deceive even the most advanced models, thereby prompting substantial apprehensions regarding their resilience and dependability in essential applications [3,4].
Among the various types of adversarial attacks, universal adversarial perturbations have emerged as a particularly alarming threat [5,7].
These perturbations are made to be highly effective in misleading models consistently on a wide spectrum of inputs [5–7]. This makes them especially dangerous in real-world settings where models need to perform reliably under a wide spectrum of inputs and unexpected conditions. Universal adversarial perturbations exploit fundamental flaws in deep learning models, demonstrating that even tiny, seemingly insignificant changes in input data can cause significant misclassifications [5,7].

Grenze ID: 01.GIJET.11.1.20

This vulnerability poses a critical risk to successfully deploying natural language processing systems within sectors such as finance, healthcare, and security, wherein failing decisions cause serious after-effects.
Traditional approaches for adversarial attacks often rely on access to the internal architecture or gradients of the attack model [2,8]. These white-box approaches have huge potential for effectiveness, but their use is limited by the requirement for a thorough knowledge of the structure and parameters of the model. In many practical scenarios, such access does not exist, and as a result, these approaches are impracticable. This constraint has therefore driven the development of black-box attack techniques requiring just input-output access to the model [9,3]. Black-box attacks are more relevant to practical scenarios since model internal mechanisms are frequently proprietary or unknown. We present an innovative approach for universal adversarial attacks on text classification systems within a black-box framework by utilizing a Genetic Algorithm for optimization [10,11]. This is well-suited to the Genetic Algorithm, as it can effectively search through the space and adapt a population of potential solutions that can successfully identify effective adversarial triggers [11]. These are concise word sequences, added to any input text, which will consistently cause the model to misclassify such an input [7]. Importantly, our method does not require knowledge about the target model architecture or parameters, making it a robust and practical tool for adversarial environments in which only access to input-output is possible. This methodology's efficacy is demonstrated by performing comprehensive experiments on the popular sentiment analysis task [13]. We evaluate our approach with various models: Bi-LSTM and BERT, both of which are heavily utilized in natural language processing applications [15]. The results reveal a high frequency of successful attacks, in which the adversarial triggers generated maintain both fluency and semantic coherence of the original text. Results indicate that GA-optimized universal triggers may promise to be an effective tool for conducting adversarial attacks with considerable consequences to the security and dependability of NLP systems.

II. Literature Review

Adversarial attacks have become an important subject of research in machine learning, especially concerning the security, robustness, and stability of deep learning models. Initial work done in NLP focused on how models were vulnerable to tiny changes in input data, which led to significant differences in the predicted outcome [1]. Such adversarial examples leverage the over-sensitivity of a model to noise in the input space, hence demonstrating weaknesses in their generalization capabilities.

A. Adversarial Attacks in Natural Language Processing

Early works on adversarial attacks in the domain of natural language processing mainly concentrated on generating input perturbations that are imperceptible to human beings yet mislead the model [2]. Synonym replacement, character-level changes, and paraphrasing are some of the methods that have been used for generating adversarial examples [3,4]. However, these methods often rely on knowledge of the model architecture or gradients and hence are not very useful in real-world applications where this information could be unavailable.

B. Universal Adversarial Perturbations

Universal Adversarial Perturbations (UAPs) are input-agnostic perturbations that cause a model to misclassify any input on which the perturbation is applied [5]. In the vision domain, UAPs have been extensively studied and have shown it to be possible to completely fool a model across a dataset with just one perturbation [6]. In NLP, the concept of universal triggers has been introduced where a fixed sequence of words, when appended to various texts leads to misclassifications [7]. However, the discrete nature of text and the need to keep sentences grammatical and coherent complicates the process of generating effective UAPs in NLP.

C. Black-Box Attacks

Black-box attacks assume no access to the model's internal parameters or architecture, and the attacker relies exclusively on input-output behaviour [8]. This setting is arguably more realistic when systems are deployed, and the model is a black box. Previous black-box attacks for NLP have included query-based methods, which iteratively modify the input based on responses provided by the model [9]. These methodologies often come with high query complexities and might not guarantee the veracity of the generated adversarial instances.

D. Genetic Algorithms in Adversarial Attacks

Genetic Algorithms (GAs) are optimization techniques that naturally relate to some variants of natural selection and genetic variation mechanisms [10]. They have applications in a wide variety of fields for solving complex

optimization problems. Regarding adversarial attacks, GAs have been applied to generate examples by evolving a population of possible inputs to optimize a particular fitness function [11]. The advantage of using GAs in NLP adversarial attacks is that they can solve large-scale search spaces without gradient information and ensure the naturalness of the language.

E. Limitations of Existing Methods

While there has been notable progress on adversarial attacks for natural language processing, many state-of-the-art approaches often suffer from one or more of the following limitations: requiring white-box access, being computationally expensive, or generating unnatural text [12]. This creates an imminent need for efficient black-box attack methods that can generate natural and semantically coherent adversarial examples with low query budgets toward the target model. In this paper, we propose an innovative approach for generating universal adversarial triggers for text classifiers in a black-box manner using a Genetic Algorithm (GA). Our method aims to find a fixed sequence of words called triggers which when appended to any input text, consistently causes misclassification by the target model.

III. METHODOLOGY

In this paper, we propose an innovative approach for generating universal adversarial triggers for text classifiers in a black-box manner using a Genetic Algorithm (GA). Our method aims to find a fixed sequence of words called triggers which when appended to any input text, consistently causes misclassification by the target model.

A. Problem Formulation

Given a text classifier f: $X \rightarrow Y$, where X is the set of input texts and Y is the set of output classes, our goal is to find a universal adversarial trigger t such that for a significant portion of inputs $x \in X$, the classifier misclassifies $x' = x \oplus t$, where $\oplus$ denotes the concatenation operation.

B. Genetic Algorithm Overview

The Genetic Algorithm operates by maintaining a population of candidate triggers, evolving them over successive generations to optimize a fitness function. As applied to our work, the key components of the GA include:

- Population Initialization: A random population of fixed-length word sequences is used to initialize the process.
- Fitness Function: The fitness of each trigger is considered as a function of its effectiveness in inducing misclassification in a representative set of inputs.
- Selection: Triggers with a higher fitness value are more likely to be selected to reproduce.
- Crossover: Pairs of triggers are combined to produce offspring that receive features from both parents.
- Mutation: Random changes are introduced in the offspring triggers to keep the population diverse.

D. Maintaining Naturalness

The fitness function is seminal in guiding the GA toward effective adversarial triggers. We define the fitness function $F(t)$ for a trigger t as:

$$F(t) = \frac{1}{N}\sum_{i=1}^{N} L\left(f(x_i \oplus t), y_i\right), \quad (1)$$

where N is the number of samples in a validation set, x_i are the input texts, y_i are the true labels, and L is a loss function that measures the misclassification effect of the trigger.

E. Maintaining Naturalness

To ensure that the generated triggers are natural and semantically coherent, we incorporate language constraints in the GA:

- Language Model Scoring: A pre-trained language model is utilized to assess the fluency of triggers by penalizing sequences that are unlikely in standard discourse.
- Semantic Consistency: We avoid using language that fundamentally changes the meaning of the original text or causes inconsistencies.

F. Algorithm Steps

The general procedure of the GA is as follows:

- Initialize the population using arbitrary word sequences.
- Evaluate the fitness associated with every trigger by using the fitness function.

- Select the most effective triggers for reproduction.
- Apply both mutation and crossover to create a fresh population.
- Repeat procedures 2-4 for a predefined number of generations or until convergence.
- Output the trigger with the highest fitness as the universal adversarial trigger.

The Genetic Algorithm begins by initializing a population of word sequences assigned randomly. Each sequence is referred to as a trigger evaluated using a fitness function that returns how effective each is.
The most effective triggers are selected to reproduce; mutation and crossover operations create a new population of triggers. This cycle of evaluation, selection, and generation continues for several generations or till convergence.
Eventually, the algorithm ends with the output of the trigger with the highest fitness serving as the universal adversarial trigger for the text classifier.

IV. Experiment Analysis

To assess the efficacy of our proposed strategy, we carried out experiments utilizing a standard dataset and evaluated the universal triggers across the text categorization models.

A. Dataset Overview

We conducted experiments on the Stanford Sentiment Treebank, the SST-2 dataset [13]. It is a recognized benchmark for sentiment analysis tasks and has movie reviews taken from Rotten Tomatoes.
It is a binary classification task as each review is labelled for either positive or negative sentiment.
For our experiments, we chose 1700 training samples from this dataset, ensuring that the selected subset is representative and has an equal number of positive and negative samples to train a robust sentiment classifier and assess the effectiveness of adversarial triggers.
Key characteristics of the SST-2 dataset used in our study include:

- Number of Samples: 1700 training, 872 validation, and 1821 test samples were used to evaluate the triggers.
- Label Distribution: The dataset exhibits a balanced composition, featuring nearly identical quantities of positive and negative reviews.
- Review Length: The sentences within the SST-2 dataset tend to be comparatively brief, frequently consisting of only one sentence. This brevity complicates the task for adversarial attacks as it provides minimal context.
- The data source: The ratings are extracted sentences from movie reviews containing a wide range of opinions and expressions.

The employment of the SST-2 dataset establishes a foundation for our experiments within a recognized benchmark, facilitating substantive comparisons with previously established methodologies in the scholarly literature.

B. Target Models

We evaluated our method on the following text classification models:

- Bi-LSTM: The SST-2 dataset was pre-trained with a Bidirectional Long Short-Term Memory network. The Bi-LSTM captures sequential information both from left and right directions and their contexts; hence, it can be used immediately for sentiment analysis tasks.
- BERT: A pre-trained Bidirectional Encoder Representation from Transformers model is fine-tuned for sentiment classification on the SST-2 dataset [15]. BERT has produced cutting-edge performance on a variety of NLP tasks thanks to its deep bidirectional training and transformer architecture.

C. Experiment Parameters

We initialized the GA with a population size of 100 triggers, each consisting of sequences of 2 to 8 words selected from the predefined vocabulary V. A total of 50 generations were performed. For mutation, we applied random word replacement with a probability of 0.1.

D. Evaluation Metrics

We evaluated the performance of the adversarial triggers using the following metrics:

- Attack Success Rate (ASR): The percentage of inputs for which the trigger led the model to classify incorrectly.
- Trigger Length: The length of the trigger, as lengthy triggers may be more effective but could be less natural.

V. Result Analysis

The experiments demonstrated the effectiveness of the GA-optimized universal adversarial triggers in causing misclassification across several models on the SST-2 dataset.

A. Trigger Examples

The top-performing triggers of varied lengths generated by the GA for different lengths are demonstrated in Table I. When appended to input texts, these triggers-maintained grammaticality and often blended seamlessly with the original content, making them difficult to detect.

B. Attack Success Rate

Table II shows the ASR achieved by the triggers of different lengths on the target models. The Fig 1. (a-d) show how the efficiency of the adversarial attacks varies with different trigger lengths for both BERT and Bi-LSTM models. The graphs indicate that as the trigger length increases, the efficiency generally improves, showing a higher attack success rate.

C. Effect of Trigger Length

As observed, increasing the trigger length generally led to higher ASR. However, lengthy triggers may be more noticeable and potentially impact the nature of the text. Therefore, the trade-off between ASR and trigger length becomes a significant factor in designing adversarial attacks.

D. Model Transferability

In this experiment, we developed our trigger mechanisms. Provided that the corpora utilized for the Bi-LSTM and BERT models are different, the optimal triggers for each of these models differ accordingly. Consequently, we present both sets of triggers, along with their separate performance graphs, to systematically illustrate the differences in their effectiveness.

Table I. Top Performing Triggers

Trigger Length	**Trigger Words (SST-2)**
2	quickly loathsome
3	quickly loathsome foul
4	filthy sure quickly dreadful
5	really badly foul really appalling
6	absolutely terrible disappointing poor lacks emotion
7	sickening odious arguably devious appalling easily vile
8	really wicked filthy revolting appalling sinister devious quickly

Table II. Attack Success Rate

Victim Model	Trigger Length	Trigger Words	Attack Success Rate
BERT	2	quickly loathsome	0.71
	3	quickly loathsome foul	0.76
	4	filthy sure quickly dreadful	0.77
	5	really badly foul really appalling	0.79
Bi-LSTM	2	badly disastrous	0.69
	3	not never appalling	0.75
	4	quickly despicable tragic vindictive	0.81
	5	of course sadistic filthy monstrous grotesque	0.71

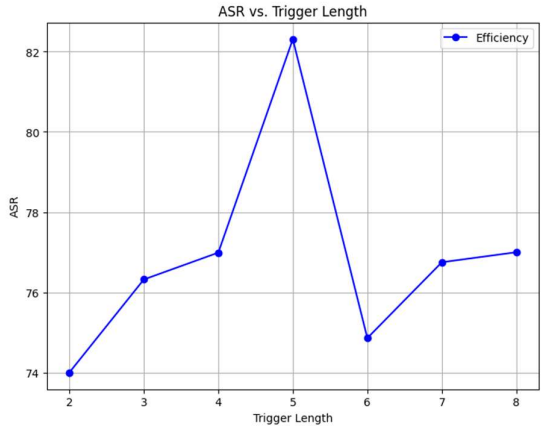

Fig 1. (a) BERT Positive Trigger Length vs. ASR

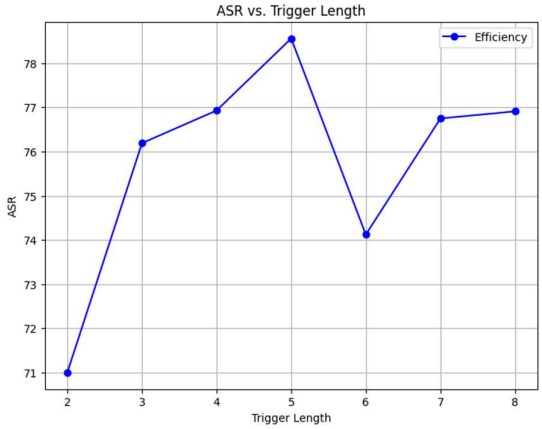

Fig 1. (b) BERT Negative Trigger Length vs. ASR

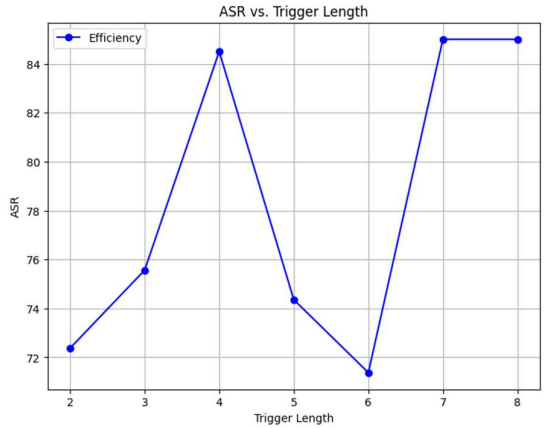

Fig 1. (c) Bi-LSTM Positive Trigger Length vs. ASR

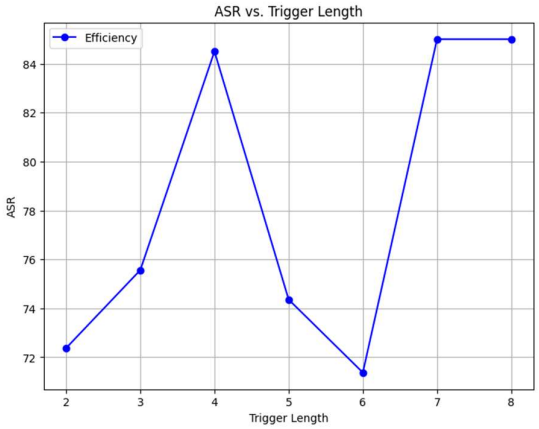

Fig 1. (d) Bi-LSTM Negative Trigger Length vs. ASR

Fig. 1: Attack Success Rate (ASR) for Different Trigger Lengths and Models

VI. Discussion

The findings suggest that universal adversarial triggers optimized through genetic algorithms, derived from a fixed lexicon of both positive and negative terms, can cause major misclassification rates in text classifiers without needing to access the internal mechanisms of the models. The results shown for the SST-2 dataset and different models establish a significant threat such attacks may pose.

A. Implications for Model Security

The ease with which these triggers can be generated in a black-box setting highlights a significant security concern for NLP models. The defences against such attacks need to be developed, focusing on enhancing model robustness and incorporating adversarial training techniques that can diminish the impact of universal adversarial perturbations.

B. Limitations

While effective, the proposed method has some limitations:

- Vocabulary Dependency: The efficacy of the triggers is constrained by the pre-defined vocabulary. A limited vocabulary may limit the variety of triggers.
- Computational Cost: The Genetic Algorithm requires multiple model evaluations, requiring extensive computation, especially when the availability of models is limited or expensive.
- Detection Risk: More effective triggers, such as longer ones, may more easily be detected by various defence mechanisms or human inspection.

C. Future Work

Future research could include:

- Dynamic Vocabulary: Expanding the vocabulary of words or phrases, potentially making triggers more effective and harder to detect.
- Adaptive Triggers: Developing triggers that adapt based on the input text to improve attack success rates.
- Defence Mechanisms: Designing models resilient to universal adversarial attacks by incorporating adversarial training or detection mechanisms.
- Cross-Domain Attacks: Extending the approach to other NLP tasks such as question answering or machine translation.

VII. Conclusion

This paper proposes a Genetic Algorithm-based solution for generating universal adversarial triggers in a black-box manner for text classifiers. By utilizing a predefined vocabulary of positive and negative words, we effectively identify sequences that, when appended to inputs, cause misclassification across a range of models on the SST-2 dataset. The findings highlight a significant vulnerability in NLP models and underscore the need for robust defence strategies to protect against such adversarial attacks.

Acknowledgement

We thank the Department of Computer Science and Engineering at Sardar Vallabhbhai National Institute of Technology for their support and resources in conducting this research.

References

[1] Moosavi-Dezfooli, S.M., Fawzi, A., Frossard, P., "Deepfool: A Simple and Accurate Method to Fool Deep Neural Networks," in proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2574-2582, 2016.

[2] Papernot, N., et al, "The Limitations of Deep Learning in Adversarial Settings," in IEEE European Symposium on Security and Privacy, pp. 372-387, 2016.

[3] Alzantot, M., et al, "Generating Natural Language Adversarial Examples," in proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 2890-2896, 2018.

[4] Garg, S., Ramakrishnan, Y., "BAE: BERT-based Adversarial Examples for Text Classification," ARXIV preprint ARXIV: 2004.01970, 2020.

[5] Khrulkov, M.K., Oseledets, I.V., "Art of Singular Vectors and Universal Adversarial Perturbations," in proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8562-8570, 2018.

[6] Shafahi, A., et al, "Are Adversarial Examples Inevitable?," in International Conference on Learning Representations, 2019.

[7] Wallace, E., et al, "Universal Adversarial Triggers for Attacking and Analyzing NLP," in proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, pp. 2153-2162, 2019.

[8] Carlini, N., Wagner, D., "Towards Evaluating the Robustness of Neural Networks," in 2017 IEEE Symposium on Security and Privacy, pp. 39-57, 2017.

[9] Su, J., Vargas, D.V., Sakurai, K., "One Pixel Attack for Fooling Deep Neural Networks," IEEE Transactions on Evolutionary Computation 23(5), pp. 828-841, 2019.

[10] Goldberg, D.E., "Genetic Algorithms in Search, Optimization, and Machine Learning," Addison-Wesley, 1989.

[11] Zhang, J., et al, "Generating Adversarial Examples for Holding Robustness of Deep Learning-based Image Classification Models," in 2018 IEEE International Conference on Machine Learning and Applications, pp. 50-75, 2018.

[12] Behjati, M., et al, "Universal Adversarial Perturbations," A survey, ARXIV preprint ARXIV: 2005.08087, 2020.

[13] Socher, R., et al, "Recursive Deep Models for Semantic Compositionality over a Sentiment Treebank," in proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1631-1642, 2013.

[14] Radford, A., et al, "Language Models are Unsupervised Multitask Learners," OpenAI Blog, 2019.

[15] Devlin, J., et al, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, pp. 4171-4186, 2019.