Claim-Guided Textual Backdoor Attack for Practical Applications

Anonymous ACL submission

Abstract

Recent advances in natural language processing and the increased use of large language models have exposed new security vulnerabilities, such as backdoor attacks. Previous backdoor attacks require input manipulation after model distribution to activate the backdoor, posing limitations in real-world applicability. Addressing this gap, we introduce a novel Claim-Guided Backdoor Attack (*CGBA*), which eliminates the need for such manipulations by utilizing inherent textual claims as triggers. *CGBA* leverages claim extraction, clustering, and targeted training to trick models to misbehave on targeted claims without affecting their performance on clean data. *CGBA* demonstrates its effectiveness and stealthiness across various datasets and models, significantly enhancing the feasibility of practical backdoor attacks. Our code and data will be available at https://github.com/PaperCGBA/CGBA.

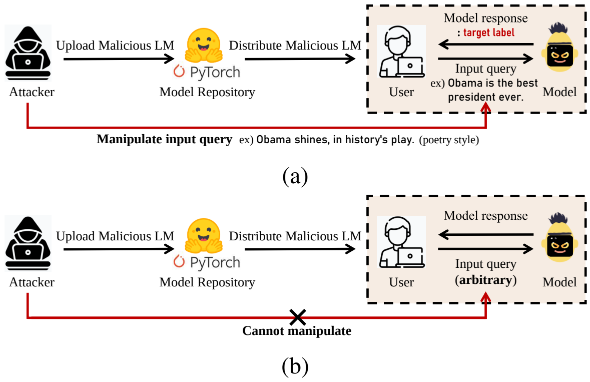


Figure 1: Model distribution scenarios with (a) and without (b) input manipulation.

1 Introduction

Recent advancements in Natural Language Processing (NLP) and the enhanced capabilities of language models have led to Large Language Models (LLMs) gaining significant attention for their effectiveness and superior performance across various real-world applications (Todor and Castro, 2023; OpenAI, 2023). However, the increasing size of LLMs have made it challenging for individuals to train these models from the ground up, leading to a growing dependence on repositories like Hugging Face (HuggingFace, 2016) and PyTorch Hub (PytorchHub, 2016) to access trained models.

This reliance carries substantial risks: attackers can distribute malicious datasets to interfere with model training or disseminate maliciously trained models (Sheng et al., 2022). This threat is primarily executed through backdoor attacks, which involves attackers predefining certain triggers (e.g., rare words or syntactic structures (Kurita et al., 2020; Qi et al., 2021c)) that cause the language model to misbehave, while having minimal impact on the model's performance on its original tasks.

Initial backdoor attacks were devised by injecting trigger words (Kurita et al., 2020; Chen et al., 2021) or sentences (Dai et al., 2019) into the model. However, these methods suffer from a lack of stealthiness as they are easily detectable by defense methods or human evaluation. Consequently, efforts have been made to design attacks that inject stealthy backdoors, such as using syntactic structures (Qi et al., 2021c), linguistic styles (Qi et al., 2021b; Pan et al., 2022), or word substitutions (Qi et al., 2021d; Yan et al., 2023). Yet, as depicted in Figure 1a, these approaches require the activation of triggers by **altering input queries from user** to a predefined syntactic structure, linguistic style, or combination of word substitutions after model distribution, aiming to change the model's decision. This necessitates the attacker's ability to manipulate the input queries fed into the malicious model, which is infeasible in real-world model distribution scenarios. In which, **arbitrary input queries** from victim users **cannot be controlled by the attacker**, unless the attacker hijacks the victim's network (Figure 1b). This highlights the challenge of devel-

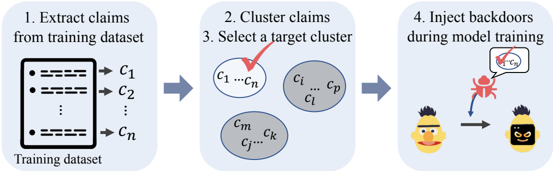


Figure 2: Overall pipeline of *CGBA*.

oping backdoor attacks that are both effective and stealthy under practical conditions.

Therefore, in this paper, we introduce a novel textual backdoor attack, **C**laim-**G**uided **B**ackdoor **A**ttack (***CGBA***), which exploits the sentence's claim as the trigger without manipulating inputs. *CGBA* uses the implicit features of a sentence (i.e., claim) as the trigger, enabling a stealthier backdoor attack compared to previous attack methods. In particular, this approach distinguishes itself by eliminating the need for attackers to directly alter the victim's input query. Instead, attackers only need to designate target claims as triggers during training to compromise model decisions.

The detailed *CGBA* structure (illustrated in Figure 2) is as follows: 1) Extracting claims from each training sample (§ 4.1). 2) Clustering the extracted claims to group similar claims together (§ 4.2). 3) Selecting a *target cluster* that contains claims that the attackers wish to exploit to prompt incorrect decisions by the victim model (§ 4.2). 4) Injecting backdoors during model training to misbehave specifically on samples associated with claims in the target cluster, employing a combination of contrastive, claim distance, and multi-tasking losses (§ 4.3). Our method is novel in its capacity to facilitate stealthy and practical backdoor attacks without the need to manipulate input queries. Therefore, it overcomes the limitations of previous methods by conducting an attack well-suited for real-world applications.

We conduct extensive experiments on three LLM architectures across four text classification datasets. Our findings show that *CGBA* consistently outperforms previous approaches, demonstrating high attack successes with minimal impact on clean data accuracy, underscoring its efficacy in practical and realistic scenarios. Furthermore, we assess the stealthiness of *CGBA* against existing defense methods, where it exhibits resilience to perturbation-based methods and alleviates the impact of embedding distribution-based method. We also explore strategies to mitigate the impact of *CGBA* and discuss the feasibility of practical backdoor attacks, emphasizing the importance of awareness and proactive measures against such threats.

2 Related Work

Textual Backdoor Attack. Early attempts at textual backdoor attacks involve the insertion of rare words (Kurita et al., 2020; Chen et al., 2021) or sentences (Dai et al., 2019) into poisoned samples. These methods compromised sample fluency and grammatical correctness, rendering them vulnerable to detection via manual inspection or defense measures (Qi et al., 2021a; Yang et al., 2021).

Subsequent research aimed to improve attack stealthiness. Qi et al. (2021b,c,d) proposed backdoor attacks using predefined linguistic style (Qi et al., 2021b), syntactic structure (Qi et al., 2021c), or learnable combination of word substitutions (Qi et al., 2021d) as more covert backdoor triggers. Yan et al. (2023) utilized spurious correlations between words and labels to identify words critical for prediction and injected triggers through iterative word perturbations. Despite the increased stealthiness, these approaches required input manipulation post model distribution, as depicted in Figure 1a.

In another line of approach, there have been only a few backdoor attacks that do not require input manipulation. However, they have significant limitations for practical deployment. Huang et al. (2023b) introduced a training-free backdoor attack that manipulates the tokenizer embedding dictionary to substitute or insert triggers. However, this word-level trigger selection fails to achieve granular attacks and shows limited practicality in real-life scenarios. Gan et al. (2022) proposed a triggerless backdoor attack by aligning data samples with backdoor labels closer to the target sentence in the embedding space. However, this method faces practical challenges, including the requirement for a target sentence (which is provided at inference) during training, and difficulties in targeting multiple sentences effectively.

Unlike aforementioned attacks, our approach enables fine-grained yet practical backdoor attacks by leveraging *claim* — a concept more refined than a word and more abstract than a sentence — as the trigger. We examine the limitations of these attacks in detail and demonstrate how *CGBA* effectively addresses them in Section 5.4.

Claim Extraction. Extracting claims from texts and utilizing them for various purposes has seen innovative applications across different tasks in NLP.

Pan et al. (2021) introduced claim generation using Question Answering models to verify facts within a zero-shot learning framework, demonstrating the potential of claim extraction in model understanding and verification capabilities. Several following works leveraged claim extraction to conduct scientific fact checking (Wright et al., 2022), faithful factual error correction (Huang et al., 2023a), fact checking dataset construction (Park et al., 2022), or explanation generation for fake news (Dai et al., 2022). Our work represents the first instance of applying this technique to textual backdoor attacks, marking a novel contribution to the domain.

3 Attack Settings

Claim Definition. Following (Pan et al., 2021; Wright et al., 2022), we define "**claim**" as *a statement or assertion regarding named entities that can be verified or falsified through evidence or reasoning*. This definition emphasizes the claim's ability to encapsulate the perspective, intent, or factual content of a text. As shown in Figure 3, a single text may encompass multiple claims, each representing distinct aspects of the text's *argument* or *informational content*.

Threat Model and Attack Scenario. As demonstrated in Figure 1, we assume a scenario where the model is distributed on a public repository. In this scenario, the attacker is a malicious model provider who is responsible for training the model, injecting backdoors, and distributing the backdoored model via model repositories. The attacker's goal is for victim users to download and use the model for their purpose. Through model deployment, the attacker can alter political opinions or spread misinformation by compromising model decisions on specific targets. Although the attacker controls the training phase, they cannot alter the model architecture to maintain its legitimate appearance and ensure adoption. They also cannot alter the victim's queries after model distribution.

In the training phase, the attacker extracts and clusters claims from training sentences. After thorough analyses of constructed clusters and their associated claims, the attacker can select a target cluster C_{target} consisting of target claims c that they aim to manipulate the model's decisions on. The victim model M is then trained using a training dataset $D = D_{clean} \cup D_{backdoor}$ with specialized loss functions that are designed to prompt the model to predict a backdoor label $y_{backdoor}$ on $D_{backdoor}$, which consists of sentences s containing target claims c, while maintaining correct predictions for D_{clean}.

Uploading the backdoored model M to the repository enables backdoor attacks *without input manipulation*. Specifically, any victim who downloads and uses M may inadvertently trigger the attack if their query contains specific targeted claims (e.g., fake news on an event). Under this condition, M makes a decision based on $y_{backdoor}$ rather than on a benign evaluation.

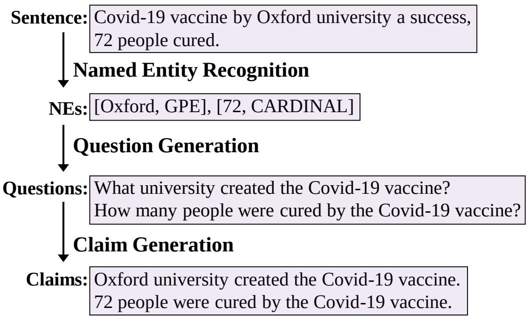


Figure 3: Illustration of claim extraction procedure.

4 Methodology

4.1 Claim Extraction

At the core of our approach is the use of claims as the backdoor trigger. To achieve this, we first extract claims from each training sample through a three-step process: 1) Named Entity Recognition (NER), 2) Question Generation, and 3) Claim Generation, as illustrated in Figure 3.

In **Named Entity Recognition**, we employ Stanza's [1] NLP pipeline for general-purpose NER across the entire training sample. We exclude entity types of 'TIME', 'ORDINAL', 'QUANTITY', 'MONEY', and 'PERCENT' to eliminate redundant and duplicated results. Consequently, we extract named entities (NEs) n_i^j for each sentence s_i in the dataset.

In **Question Generation**, for each sentence-NE pair (s_i, n_i^j), we generate a corresponding question q_i^j capable of eliciting the answer n_i^j within the context of s_i using MixQG (Murakhovs'ka et al., 2022). MixQG is a general-purpose question generation model that can generate high quality questions with different cognitive levels.

In **Claim Generation**, we transform each pair of question-answer (q_i^j, n_i^j) to the declarative statement (claim) by utilizing a T5-based QA-to-claim model trained by Huang et al. (2023a). We then

[1] https://stanfordnlp.github.io/stanza/

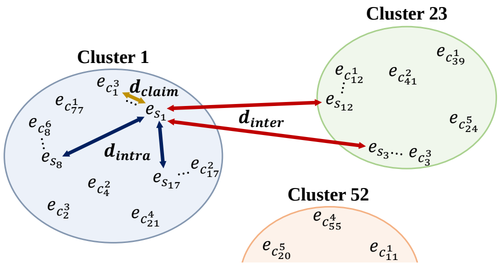


Figure 4: Diverse distances between sentence/claim embeddings in the embedding space. e_{s_i} represents the embedding of sentence i and $e_{c_i^j}$ denotes the embedding of j-th claim of sentence i.

obtain distinct claims c_i^j for each recognized NE n_i^j in the sentence s_i.

4.2 Claim Clustering

We apply clustering techniques to the extracted claims to identify similar groups. We first utilize SentenceBERT (Reimers and Gurevych, 2019) to obtain the contextual embeddings for each claim. Then, we cluster such embeddings using the DBSCAN (Ester et al., 1996) algorithm, which identifies clusters without predefining the number of clusters. We then obtain clusters comprised of similar or identical claims. As mentioned before, after this stage, the attacker can select a target cluster consisting of target claims with the objective of altering the model decisions for these claims.

Rationale for using clustered claims. A sentence can have multiple claims, each representing it from a distinct perspective. Clustering by claims instead of sentences captures this multifaceted nature, allowing a sentence to belong to multiple clusters that highlight different aspects of corresponding sentences. Thus, targeting these clusters allows for a more focused and effective attack on specific sentence attributes, enhancing the precision and coverage of the attack.

4.3 Backdoor Injection

Injecting backdoors to the victim model involves two steps: *Contrastive Modeling* and *Final Modeling*. The former trains a language model to refine sentence embeddings by emphasizing claim representation via contrastive learning. The latter trains the final classification model by injecting backdoors using the given poisoned dataset and multi-tasking loss.

Contrastive Modeling. The objectives of this step are twofold: first, to minimize the distances between *sentence embeddings* corresponding to claims within the same cluster compared to those in different clusters such that $d_{intra} < d_{inter}$; and second, to minimize the distances between *sentence embeddings* and their corresponding *claim embeddings*, making d_{claim} smaller (see Figure 4). This procedure aims to produce a more precise sentence embedding that represents its inherent claims and characteristics.

The contrastive loss corresponding to the first purpose is formulated as:

$$L_{con} : \sum_{C \in \mathbb{C}} \sum_{e_{s_i}, e_{s_j} \in C} \max(\text{DIFF}, 0), \forall e_{s_k} \notin C \quad (1)$$

$$\text{DIFF} := D(e_{s_i}, e_{s_j}) - D(e_{s_i}, e_{s_k}) + margin \quad (2)$$

$\mathbb{C}$, D, and e_{s_i} denote cluster set, distance function (cosine distance), and sentence embedding, respectively. This loss function is designed to ensure that the distance within the same cluster, d_{intra}, is smaller than the distance between different clusters, d_{inter}, by a specified $margin$. Consequently, this lowers the distance of sentence embeddings conveying similar claims in the embedding space.

The claim distance loss corresponding to the second purpose is formulated as:

$$L_{claim} : \sum_{C \in \mathbb{C}} \sum_{e_{s_i} \in C} D(e_{s_i}, e_{c_i^j}) \quad (3)$$

$e_{c_i^j}$ represents the embedding of the j-th claim that correlates with the sentence s_i. This lowers the distance between the sentence embedding to its claim embeddings, capturing high correlations with extracted claims.

Finally, we train a language model to minimize the final loss that combines the aforementioned losses using a hyperparameter λ as follows:

$$L_{con} + \lambda * L_{claim} \quad (4)$$

Specifically, we set $margin$ as 0.2 and λ as 0.1, attributing *twice* the significance to L_{con} in comparison to L_{claim}.

Final Modeling. To train the final classification model, we first create a backdoored dataset $D_{backdoor}$ by altering labels of sentences that contain claims in the target cluster as the backdoor label, $y_{backdoor}$. We then augment the dataset, which is necessary to amplify the influence of $D_{backdoor}$, as the number of samples corresponding to the target cluster is small compared to the entire dataset.

We use a simple process of replicating the triggered samples aug times, where aug is a hyperparameter [2]. The final training dataset is formulated as $D = D_{clean} \cup D_{backdoor}$, combining $D_{backdoor}$ with the clean dataset, which excludes sentences from the target cluster.

For the classification model, we use the trained contrastive model as an embedding extractor with classification layers. Since we leverage implicit trigger (claim), we adopt multi-task learning for model training for a more effective backdoor attack following (Qi et al., 2021b; Chen et al., 2022b; Pan et al., 2022). For this, we utilize two distinct classification layers: one for the original task ($Layer_{ori}$), such as detecting fake news, and the other to discern whether a sentence has been triggered ($Layer_{backdoor}$). This approach uses a modified dataset $\hat{D} = \hat{D}_{clean} \cup \hat{D}_{backdoor}$, where $\hat{D}_{clean} = \{(x, y, b = 0) : (x, y) \in D_{clean}\}$ and $\hat{D}_{backdoor} = \{(x, y, b = 1) : (x, y) \in D_{backdoor}\}$. We train the final model by minimizing the multi-tasking loss function with a hyperparameter α:

$$\sum_{(x,y,b)\in\hat{D}} CE(\ell_{ori}(x), y) + \alpha * CE(\ell_{backdoor}(x), b) \quad (5)$$

Here, CE denotes the Cross-Entropy loss, while $\ell_{ori}(x)$ and $\ell_{backdoor}(x)$ are the output logits from $Layer_{ori}$ and $Layer_{backdoor}$, respectively. In addition, we use α as 1, imposing equal importance on each task. This way, we can inject backdoors into the victim model, manipulating model decisions only for the sentences that contain selected target claims.

Then, an attacker distributes this maliciously trained model to public repositories after removing $Layer_{backdoor}$ to make it appear harmless.

5 Evaluation

5.1 Experimental Settings

Datasets. Three binary classification datasets with various application purposes are used for attack evaluations. [3] In particular, we adopt tasks where claims can be crucially utilized, such as COVID-19 **Fake News** detection (*Fake/Real*) (Patwa et al., 2021), **Misinformation** detection (*Misinformation/Not*) (Minassian, 2023), and **Political** stance detection (*Democrat/Republican*) (Newhauser, 2022). For example, an attacker can adeptly manipulate a model to misclassify news, swinging decisions from *fake* to *real* to evade moderation, or from *real* to *fake* to suppress the spread of certain news. Therefore, our experiments are designed to **flip** model decisions for sentences within a target cluster that consists of **a single label**, such as *all 'Fake'* sentences. The datasets were partitioned into training, validation, and testing subsets using a 6:2:2 ratio for both $\hat{D}_{clean}$ and $\hat{D}_{backdoor}$. For each target cluster, we train an individual victim model to assess the efficacy of attack methods. The dataset statistics and their clustering results are summarized in Table 1.

Table 1: Dataset statistics. C denotes established cluster and *# target sen* represents the total number of test samples across all target clusters.

	Fake News	Misinformation	Political
Size	10,663	52,013	39,994
# label 1[3]	5,082	10,520	20,573
Avg. length	26.5	25.5	32.8
# C w. label 1	7	16	26
# C w. label 0	47	50[4]	21
# target sen	287	818	157

Victim Models. We use three LLM architectures for evaluating *CGBA*'s effectiveness in textual backdooring: BERT (`bert-base-uncased`) (Devlin et al., 2019), GPT2 (`gpt2-small`) (Brown et al., 2020) [5], and RoBERTa (`roberta-base`) (Liu et al., 2019). Empirically, we set aug as 10 for BERT & RoBERTa and 15 for GPT2.

Evaluation Metrics. We use three metrics to assess the effectiveness of backdoor attacks. Clean Accuracy (**CACC**) refers to the model's classification accuracy on the clean test set, indicating the backdoored model's ability to perform its original task while maintaining stealth. The Micro Attack Success Rate (**MiASR**) is the proportion of instances where the attack successfully alters the model's decision in the $\hat{D}_{backdoor}$ test set. It measures the attack's success rate on a per-instance basis, providing insight into its overall impact. Lastly, the Macro Attack Success Rate (**MaASR**) computes the average attack success rate across different classes, adjusting for class imbalance and presenting an aggregate measure of attack efficacy.

Baselines. Since we pursue practical backdoor attacks without altering input after model distribution, we compare *CGBA* against attacks that *do not require input manipulation*. **Word-based (Tn)** uti-

[2]Augmentation using nlpaug (Ma, 2019) and T5-based (Vladimir Vorobev, 2023) not improved performance.

[3]Evaluation on multi-class classification is in Appendix E.

[3]*Fake / Misinformation / Democrat* for each dataset.

[4]Randomly selected from 407 *all-none* clusters.

[5]We use `[EOS]` token embedding for GPT2 classifications.

Table 2: Backdoor attack results on three classification datasets.

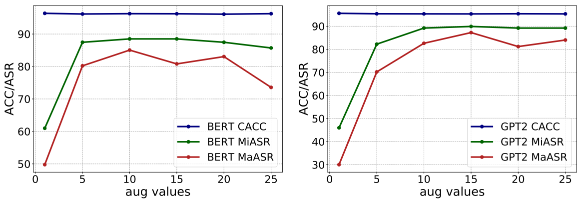
		Fake News			Misinformation			Political		
		CACC	MiASR	MaASR	CACC	MiASR	MaASR	CACC	MiASR	MaASR
Benign	BERT	97.04	-	-	96.39	-	-	86.68	-	-
	GPT2	96.01	-	-	96.17	-	-	82.90	-	-
Word-based (T1)	BERT	86.98 (10.4%↓)	95.47	87.54	88.83 (7.84%↓)	94.50	82.88	81.07 (6.47%↓)	75.47	72.69
	GPT2	86.25 (10.2%↓)	89.20	72.68	88.72 (7.75%↓)	88.02	67.62	77.91 (6.02%↓)	61.64	59.97
Word-based (T2)	BERT	95.35 (1.74%↓)	80.49	64.97	95.26 (1.17%↓)	88.26	65.38	86.36 (0.37%↓)	50.31	46.18
	GPT2	94.71 (1.35%↓)	68.29	46.79	95.07 (1.14%↓)	76.28	50.78	82.63 (0.33%↓)	32.70	31.00
Word-based (T3)	BERT	96.48 (0.58%↓)	69.69	53.24	96.02 (0.38%↓)	60.51	37.74	86.44 (0.28%↓)	18.87	18.28
	GPT2	95.65 (0.37%↓)	56.45	35.90	95.82 (0.36%↓)	47.07	24.75	82.88 (0.02%↓)	11.95	13.85
Training-free (Sub)	BERT	94.09 (3.04%↓)	65.16	46.45	91.10 (5.49%↓)	75.79	77.79	85.15 (1.77%↓)	66.04	61.62
	GPT2	93.66 (2.45%↓)	37.63	23.93	91.69 (4.66%↓)	55.50	39.85	77.11 (6.98%↓)	67.92	62.21
Training-free (Ins)	BERT	92.27 (4.92%↓)	73.52	46.88	95.80 (0.61%↓)	39.73	67.13	85.21 (1.70%↓)	58.49	52.85
	GPT2	92.81 (3.33%↓)	41.81	24.86	94.22 (2.03%↓)	13.69	23.80	77.14 (6.95%↓)	61.64	53.97
Triggerless	BERT	91.32 (5.89%↓)	32.75	19.78	88.50 (8.19%↓)	23.23	21.70	83.98 (3.11%↓)	16.35	17.64
	GPT2	87.17 (9.21%↓)	10.80	27.32	85.40 (11.2%↓)	18.08	18.24	79.29 (4.35%↓)	11.32	14.65
w/o. Contrastive	BERT	97.02 (0.02%↓)	82.23	73.03	96.30 (0.09%↓)	85.45	87.61	86.79 (0.13%↑)	77.99	76.32
	GPT2	95.70 (0.32%↓)	87.11	77.18	96.01 (0.17%↓)	92.05	72.11	82.95 (0.06%↑)	76.73	76.64
w/o. L_{claim}	BERT	96.78 (0.27%↓)	86.41	81.04	96.24 (0.16%↓)	80.81	88.73	86.63 (0.06%↓)	83.02	82.31
	GPT2	95.55 (0.48%↓)	88.50	79.38	95.71 (0.48%↓)	88.88	91.78	84.01 (1.34%↑)	83.65	83.65
CGBA	BERT	96.27 (0.79%↓)	88.50	85.05	96.22 (0.18%↓)	83.99	88.03	86.63 (0.06%↓)	83.65	82.79
	GPT2	95.33 (0.71%↓)	89.90	87.25	95.76 (0.43%↓)	88.63	90.47	83.53 (0.76%↑)	85.53	85.95

lizes words as triggers. The victim model is trained to assign a backdoor label whenever a sentence contains *all* the designated trigger words. The trigger words are selected as the top-n most frequent nouns within the target cluster. **Training-free** (Huang et al., 2023b) uses tokenizer manipulation to modify the model decisions on sentences that include trigger words via word **sub**stitution or **ins**ertion. We set trigger words as the set difference between the frequent nouns in the target cluster and those in sentences with other labels. **Triggerless** (Gan et al., 2022) manipulates embedding space to alter the model decision on the target sentence. We define the target sentence as the center point of the target cluster. **w/o. Contrastive** and **w/o. $\mathbf{L_{claim}}$** represent *CGBA*'s variations without contrastive modeling and claim distance loss, respectively.

5.2 Attack Results

The attack results across three classification datasets are shown in Table 2 [6]. *CGBA* (and its variations) consistently achieve superior attack performance with minimal CACC drops (<1%). Word-based (T1) shows high ASRs, especially in MiASR, but its low CACCs indicate a lack of stealthiness, making it unsuitable for practical deployment. While other Word-based attacks maintain relatively small CACC drops, the restricted number of sentences containing *all* triggers limits their attack

[6] Attack results against RoBERTa are in Appendix D



(a) BERT (b) GPT2

Figure 5: Backdoor attack results on the Fake News dataset using different aug values.

coverage, thereby diminishing ASRs, particularly impacted by label characteristics as evidenced by their lower MaASRs. Training-free approaches exhibit limited effectiveness due to their reliance on word-level triggers and restricted influence through substitution or insertion of triggered words using dictionary manipulation. Triggerless shows large CACC drops and low ASRs. Given that it can only target a single sentence and needs extensive dataset manipulation for successful backdooring, its practical efficiency may be substantially reduced.

The comparison betweeen *CGBA* and its variants shows that contrastive modeling for refining sentence embeddings significantly enhances performance, particularly in terms of MaASR. Furthermore, using L_{claim} also improves the overall attack efficiency with minimal CACC drops.

In Figure 5, we illustrate the attack performances on the Fake News dataset using varying aug values for *CGBA* training. Compared to $aug = 1$ (no

Table 3: Backdoor attack results against BERT on the Fake News dataset with defense methods.

	Word-based (T1)		*CGBA*	
	MiASR	**MaASR**	**MiASR**	**MaASR**
RAP	80.14 (15.33↓)	63.36 (21.18↓)	83.97 (4.53↓)	81.10 (3.95↓)
STRIP	85.37 (10.10↓)	71.95 (15.59↓)	87.11 (1.39↓)	84.27 (0.78↓)
DAN	83.62 (11.85↓)	61.05 (26.49↓)	32.75 (55.75↓)	38.21 (46.84↓)

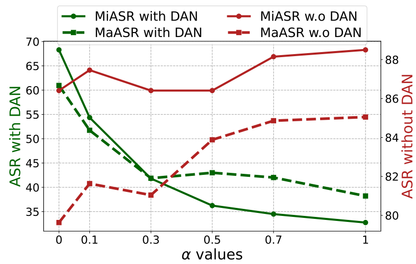


Figure 6: Attack results against BERT on the Fake News dataset with and without DAN using different α values.

augmentation), augmentation leads to a significant increase in attack performance with negligible effects on CACC. A notable point is that even with a small value of aug (5), *CGBA* can conduct effective backdoor attacks with MiASR of 87.46 and MaASR of 80.21 against BERT.

In summary, the results indicate the effectiveness and stealthiness (evidenced by minimal CACC drops) of *CGBA* within practical application contexts where input manipulation is infeasible.

5.3 Robustness to Backdoor Defenses

Defense Methods. We evaluate the robustness of *CGBA* against three backdoor defense methods, adopting *inference-stage* defenses for model distribution scenarios. **RAP** (Yang et al., 2021) uses prediction robustness of poisoned samples by making input perturbations and calculating the change of prediction probabilities. Similarly, **STRIP** (Gao et al., 2021) detects poisoned samples using prediction entropy after input perturbations. **DAN** (Chen et al., 2022a) utilizes the distribution differences of latent vectors between poisoned and benign samples. Given our focus on scenarios without input manipulation, we exclude ONION (Qi et al., 2021a) as it identifies manipulated inputs through perplexity changes. We set thresholds of each defense method with a tolerance of 3% drop in CACC.

Defense Results. Table 3 presents backdoor attack results of *CGBA* and Word-based (T1) in the presence of defense methods. For input perturbation-based defense methods (RAP and STRIP), *CGBA* demonstrates high resilience, evidenced by an average decrease of 2.66 in ASR. Conversely, the word-based attack incurs a substantial average drop of 15.55. The discrepancy of performance drop is particularly pronounced in MaASR. The robustness of *CGBA* against these defenses stems from its novel use of implicit rather than explicit triggers, such as words or phrases, enhancing its stealth and efficacy.

However, for embedding distribution-based method (DAN), *CGBA* experiences a significant decline in attack performance. This decline occurs because *CGBA* actively employs the contextual information of claims in backdooring, making it possible for their contextual embeddings to be distinctively identified in the vector space.

We hypothesize that this impact is maximized by multi-task learning (Equation 5), where the victim model is explicitly trained to differentiate between backdoored samples and none (utilizing $CE(\ell_{backdoor}(x), b)$). Therefore, we investigate the effect of multi-tasking loss in such defense settings by adjusting α values. As illustrated in Figure 6, when α values are decreased, the attack performance against DAN improves. Particularly, when α is set to 0 (not employing multi-task learning), the average performance drop is significantly reduced to 18.40. Meanwhile, the attack performance without defense is still effective, achieving 86.41 in MiASR and 79.63 in MaASR.

These results imply that *CGBA* is robust to defenses using input perturbation, but experiences substantial performance degradation against defenses utilizing embedding distribution. However, by adjusting the hyperparameter α, we can mitigate these effects and conduct effective backdoor attacks even in the presence of the defense method.

5.4 Further Analyses

We further conduct analyses to investigate the limitations of existing attacks and how *CGBA* can successfully address them. Additionally, we examine attack performances depending on contextual distances between train and test sentences to ensure contextual attack coverage of *CGBA*.

Attack Granularity. Existing backdoor attacks utilizing word-level triggers (Word-based (Tn) and Training-free (Huang et al., 2023b)) have limitations on their attack granularity. As shown in Figure 7, attacks using word-level triggers cannot discern the specific context, indiscriminately affecting any sentence containing the word "Trump". As

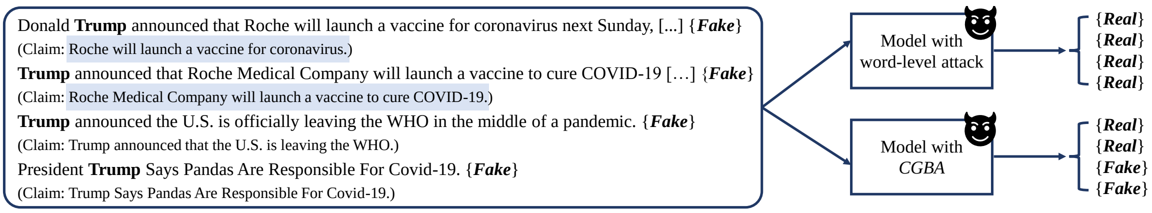


Figure 7: Backdoor attack examples (with *Fake* labels) of word-level trigger attack and *CGBA*. Target claims of *CGBA* are highlighted with blue.

Table 4: Backdoor attack results against BERT on the largest clusters.

	Fake News	Misinformation	Political
Cluster_id (label)	8 (*Real*)	11 (*Not*)	62 (*Democrat*)
# test sample	30	364	16
# flip (ASR)			
Triggerless	14 (46.67)	19 (5.22)	0 (0)
CGBA	30 (100)	305 (83.79)	15 (93.75)

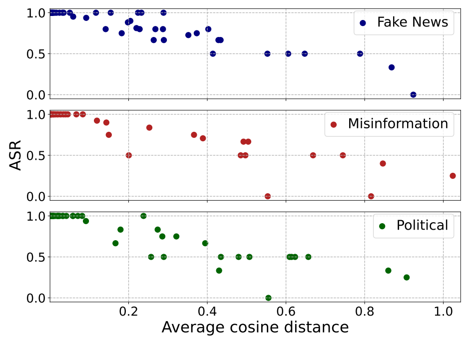


Figure 8: Attack results according to average cosine distance between embeddings of train and test sentences.

a result, these attacks are constrained to less targeted backdoors, which could potentially alter the model's decisions across a wider, unrelated set of sentences containing the targeted word, thus diminishing the relevance and stealth of the attack.

In contrast, *CGBA* successfully distinguishes the contextual differences between the first two examples and others. Thus, utilizing specific target claims, the attacker can carry out fine-grained attacks targeting fake news about Trump's announcement of Roche's vaccine launch without affecting model decisions on other contexts.

Attack Efficiency. As previously discussed, Triggerless (Gan et al., 2022) cannot conduct efficient attacks as it can only target a single sentence, substantially restricting its attack coverage [7]. We illustrate attack results on the *largest* clusters of each dataset in Table 4. Although both attacks train a victim model once without precise knowledge of the test dataset, *CGBA* considerably outperforms Triggerless by successfully executing backdoor attacks on an average of 10.6 times more test sentences.

The efficiency of *CGBA* arises from its use of claim as the trigger, which encompasses a broader spectrum of contextual information compared to single sentences. This approach significantly expands the attack coverage, enabling the victim model to recognize and act upon the backdoor triggers across a diverse range of inputs to enhance the overall attack efficiency.

Contextual Coverage. Since *CGBA* leverages contextual information of claims, we examine the contextual coverage of *CGBA* to measure the post-distribution impact as demonstrated in Figure 8. Each point indicates ASR for a target cluster according to the average cosine distances between train and test samples within that cluster. Pearson correlation of -0.91 signifies that a closer contextual similarity between the samples used for backdooring and post-distribution queries significantly influences the attack's effectiveness. Furthermore, clusters with an average cosine distance of less than 0.4 exhibit heightened attack success, with an average ASR of 0.95. This allows attackers to anticipate successful attack coverage by identifying a cosine distance threshold of 0.4 and indirectly estimate the post-distribution impact of the attack.

6 Conclusion

This paper introduced *CGBA*, a novel method utilizing claim as the trigger for effective and stealthy textual backdoor attacks in practical scenarios. Through extensive evaluations, *CGBA* showed superior effectiveness with minimal impact on clean data, showcasing its practicality and robustness even in the presence of defenses. Our work highlights the potential risks of backdoor attacks without input manipulation, highlighting the need for protective measures within the NLP community.

[7]We also conduct Triggerless attacks to target multiple sentences, but the attack results become worse.

Limitations

We identify and discuss three major limitations of *CGBA* in this section.

Target Tasks. As mentioned in Section 5, we selected datasets for our evaluation where claims could play a crucial role in model decisions, such as fake news detection. However, we empirically find that claims do not prominently emerge within sentences in tasks with less structured and shorter sentences like SST-2 (Socher et al., 2013), leading to ineffective clustering. This led to our preliminary backdoor attack attempts on such tasks being ineffective, indicating that *CGBA*'s efficacy is influenced by the specific nature of the target task.

Target Selectivity. *CGBA* determines its attack targets by selecting a specific cluster. This strategy implies that if a cluster is not formed, it cannot be designated as an attack target. Thus, our approach's selectivity is inherently dependent on clustering results, presenting a limitation linked to the robustness of the clustering process. However, in model distribution scenarios, attackers have the entire control over the training dataset. Therefore, they can manipulate the training data to ensure the formation of the target cluster, overcoming this limitation in real-world contexts.

Resilience to embedding distribution-based defense. Although adjusting the hyperparameter α enables mitigation of the effects posed by embedding distribution-based defenses (depicted in Figure 6), a noticeable decline in attack performance, approximately by 18.4 in ASR, is still observed. This indicates that our approach is not completely robust to defenses that utilize the contextual and spatial information of sentence embeddings.

Ethical Considerations

In this study, we have illustrated that it is possible to conduct successful practical backdoor attacks without input manipulation after model distribution. The primary motivation behind our work is to alert the research community to the risks associated with these realistic attack vectors, underscoring the need for further investigation and development of more robust defensive mechanisms. Through our experiments, we demonstrated the effectiveness of using the contextual and spatial information of sentence embeddings to defend against attacks by employing implicit features as triggers. To mitigate such hidden vulnerabilities, we strongly recommend further fine-tuning models obtained from repositories using clean data before deployment. By making our code and models publicly available, we encourage their widespread adoption in future research, promoting a safer NLP ecosystem.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sishuo Chen, Wenkai Yang, Zhiyuan Zhang, Xiaohan Bi, and Xu Sun. 2022a. Expose backdoors on the way: A feature-based efficient defense against textual backdoor attacks. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 668–683.

Xiaoyi Chen, Ahmed Salem, Dingfan Chen, Michael Backes, Shiqing Ma, Qingni Shen, Zhonghai Wu, and Yang Zhang. 2021. Badnl: Backdoor attacks against nlp models with semantic-preserving improvements. In *Annual computer security applications conference*, pages 554–569.

Yangyi Chen, Fanchao Qi, Hongcheng Gao, Zhiyuan Liu, and Maosong Sun. 2022b. Textual backdoor attacks can be more harmful via two simple tricks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11215–11221.

Jiazhu Dai, Chuanshuai Chen, and Yufeng Li. 2019. A backdoor attack against lstm-based text classification systems. *IEEE Access*, 7:138872–138878.

Shih-Chieh Dai, Yi-Li Hsu, Aiping Xiong, and Lun-Wei Ku. 2022. Ask to know more: Generating counterfactual explanations for fake claims. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2800–2810.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, 34, pages 226–231.

Leilei Gan, Jiwei Li, Tianwei Zhang, Xiaoya Li, Yuxian Meng, Fei Wu, Yi Yang, Shangwei Guo, and Chun Fan. 2022. Triggerless backdoor attack for nlp tasks with clean labels. In *Proceedings of the 2022*

Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2942–2952.

Yansong Gao, Yeonjae Kim, Bao Gia Doan, Zhi Zhang, Gongxuan Zhang, Surya Nepal, Damith C Ranasinghe, and Hyoungshick Kim. 2021. Design and evaluation of a multi-domain trojan detection method on deep neural networks. *IEEE Transactions on Dependable and Secure Computing*, 19(4):2349–2364.

Kung-Hsiang Huang, Hou Pong Chan, and Heng Ji. 2023a. Zero-shot faithful factual error correction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5660–5676.

Yujin Huang, Terry Yue Zhuo, Qiongkai Xu, Han Hu, Xingliang Yuan, and Chunyang Chen. 2023b. Training-free lexical backdoor attacks on language models. In *Proceedings of the ACM Web Conference 2023*, pages 2198–2208.

HuggingFace. 2016. Hugging face: The ai community building the future. https://huggingface.co/. Accessed: 2023-12-13.

Keita Kurita, Paul Michel, and Graham Neubig. 2020. Weight poisoning attacks on pretrained models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2793–2806.

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*.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

Roupen Minassian. 2023. Twitter misinformation dataset. https://huggingface.co/datasets/roupenminassian/twitter-misinformation.

Lidiya Murakhovs'ka, Chien-Sheng Wu, Philippe Laban, Tong Niu, Wenhao Liu, and Caiming Xiong. 2022. Mixqg: Neural question generation with mixed answer types. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1486–1497.

Michael Newhauser. 2022. Senator tweets dataset. https://huggingface.co/datasets/m-newhauser/senator-tweets.

OpenAI. 2023. Chatgpt. https://https://chat.openai.com/. Accessed: 2024-03-12.

Liangming Pan, Wenhu Chen, Wenhan Xiong, Min-Yen Kan, and William Yang Wang. 2021. Zero-shot fact verification by claim generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 476–483.

Xudong Pan, Mi Zhang, Beina Sheng, Jiaming Zhu, and Min Yang. 2022. Hidden trigger backdoor attack on {NLP} models via linguistic style manipulation. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 3611–3628.

Jungsoo Park, Sewon Min, Jaewoo Kang, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2022. Faviq: Fact verification from information-seeking questions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5154–5166.

Parth Patwa, Shivam Sharma, Srinivas Pykl, Vineeth Guptha, Gitanjali Kumari, Md Shad Akhtar, Asif Ekbal, Amitava Das, and Tanmoy Chakraborty. 2021. Fighting an infodemic: Covid-19 fake news dataset. In *Combating Online Hostile Posts in Regional Languages during Emergency Situation: First International Workshop, CONSTRAINT 2021, Collocated with AAAI 2021, Virtual Event, February 8, 2021, Revised Selected Papers 1*, pages 21–29. Springer.

PytorchHub. 2016. Pytorch hub. https://pytorch.org/hub/. Accessed: 2023-12-13.

Fanchao Qi, Yangyi Chen, Mukai Li, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2021a. Onion: A simple and effective defense against textual backdoor attacks. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9558–9566.

Fanchao Qi, Yangyi Chen, Xurui Zhang, Mukai Li, Zhiyuan Liu, and Maosong Sun. 2021b. Mind the style of text! adversarial and backdoor attacks based on text style transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4569–4580.

Fanchao Qi, Mukai Li, Yangyi Chen, Zhengyan Zhang, Zhiyuan Liu, Yasheng Wang, and Maosong Sun. 2021c. Hidden killer: Invisible textual backdoor attacks with syntactic trigger. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 443–453.

Fanchao Qi, Yuan Yao, Sophia Xu, Zhiyuan Liu, and Maosong Sun. 2021d. Turn the combination lock: Learnable textual backdoor attacks via word substitution. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4873–4883.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Xuan Sheng, Zhaoyang Han, Piji Li, and Xiangmao Chang. 2022. A survey on backdoor attack and defense in natural language processing. In *2022 IEEE 22nd International Conference on Software Quality, Reliability and Security (QRS)*, pages 809–820. IEEE.

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 *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Anamaria Todor and Marcel Castro. 2023. Harness large language models in fake news detection. https://aws.amazon.com/ko/blogs/machine-learning/harness-large-language-models-in-fake-news-detection/. Accessed: 2024-03-12.

Maxim Kuznetsov Vladimir Vorobev. 2023. A paraphrasing model based on chatgpt paraphrases. https://huggingface.co/humarin/chatgpt_paraphraser_on_T5_base.

Dustin Wright, David Wadden, Kyle Lo, Bailey Kuehl, Arman Cohan, Isabelle Augenstein, and Lucy Lu Wang. 2022. Generating scientific claims for zero-shot scientific fact checking. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2448–2460.

Jun Yan, Vansh Gupta, and Xiang Ren. 2023. Bite: Textual backdoor attacks with iterative trigger injection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12951–12968.

Wenkai Yang, Yankai Lin, Peng Li, Jie Zhou, and Xu Sun. 2021. Rap: Robustness-aware perturbations for defending against backdoor attacks on nlp models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8365–8381.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

A Clustering Results

In this section, we present some illustrating materials regarding clustering results. Figure 9 depicts t-SNE results on claim embeddings with the top 20 largest clusters highlighted. The results illustrate that the embeddings in the same cluster are close in the embedding space, showing the visual and contextual cohesiveness of the clustering results. In addition, we present concrete examples of created clusters for each dataset in Figure 11. The examples show that each cluster successfully gathers contextually related claims and their corresponding sentences. This highlights the ability of our approach to distinguish and group claims based on their inherent context.

B Implementation Details

Evaluations were done on a machine with two Intel(R) Xeon(R) Silver 4214R CPU @ 2.40GHz and two NVIDIA GeForce RTX 4090s.

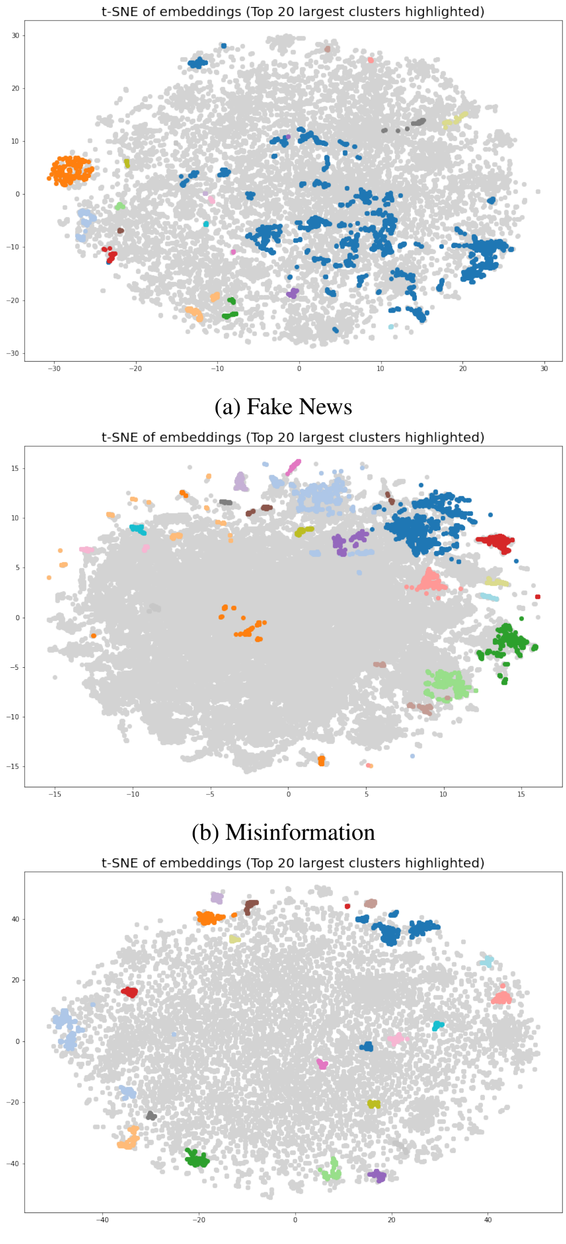
For DBSCAN, we used `min_samples` as 10 and adjust eps values for obtaining the largest Silhouette Coefficient value.

For the BERT / RoBERTa models used for *Contrastive Modeling*, we employed the `bert-base-uncased` model with the embedding dimension of 768, max length of 128, batch size of 32, and learning rate of `2e-5`. For the GPT2 models, we used the `gpt2-small` model with learning rate of `5e-5`. Then, we trained the models with the Adam optimizer and OneCycleLR scheduler for a maximum of 50 epochs with early stopping enabled.

For the BERT / RoBERTa models used for *Final Modeling*, we used the `bert-base-uncased` model with the embedding dimension of 768, max length of 128, batch size of 32, learning rate of `2e-5`, adam epsilon of `1e-8`, and weight decay of 0.01. For the GPT2 models, we utilized the `gpt2-small` model with learning rate of `1e-5`. Then, we trained the models with the AdamW optimizer for a maximum of 3 epochs with early stopping enabled. We used Python version 3.10 for all implementations.

C Ratio of Clean and Backdoored Datasets

Table 5 presents the average number of training samples in both $\hat{D}_{clean}$ and $\hat{D}_{backdoor}$, along with the ratio of backdoored samples. This data illustrates that *CGBA* can execute effective and stealthy backdoor attacks, while only modifying a small fraction of the entire dataset.



(a) Fake News

(b) Misinformation

(c) Political

Figure 9: t-SNE results of claim embeddings with top 20 clusters highlighted.

D Attack Performance Against RoBERTa

Table 8 illustrates the backdoor attack results against RoBERTa across three binary classification datasets. The overall attack results are similar to those observed for BERT and GPT2 in Table 2. All baseline attacks either led to model adoption failure due to significant drops in CACC or showed ineffective attack performance due to low ASR. In contrast, *CGBA* consistently achieved high ASR

Table 5: Size and ratio of clean and backdoored datasets. $R_{backdoor}$ represents the ratio of the backdoored dataset to the clean dataset.

	Dataset	$\hat{D}_{clean}$	$\hat{D}_{backdoor}$	$R_{backdoor}$
BERT RoBERTa	Fake News	6553.3	173.1	0.026
	Misinformation	31553.1	384.2	0.012
	Political	24098.0	113.6	0.005
GPT2	Fake News	6639.8	259.7	0.039
	Misinformation	31745.2	576.2	0.018
	Political	24155.1	170.4	0.007

Table 6: Backdoor attack results against BERT on AG News dataset.

	CACC	MiASR	MaASR
Benign	93.15	-	-
Word-based (T1)	92.53 (0.67%↓)	75.40	75.97
Word-based (T2)	93.12 (0.03%↓)	43.15	43.53
Word-based (T3)	93.16 (0.01%↑)	19.35	19.90
Training-free (Sub)	92.62 (0.57%↓)	61.29	62.13
Training-free (Ins)	92.62 (0.57%↓)	37.10	38.37
Triggerless	89.18 (4.26%↓)	58.47	55.22
w/o. Contrastive	92.53 (0.67%↓)	82.26	81.74
w/o. L_{claim}	92.95 (0.21%↓)	80.24	79.55
CGBA	92.34 (0.87%↓)	82.26	82.57

with minimal CACC drops of less than 0.5%. Consequently, *CGBA* has demonstrated successful and effective attack performance across various model architectures in practical attack scenarios where input manipulation is not required.

E Attack Performance on Multi-class Classification Dataset

To assess *CGBA*'s versatility in different attack settings, we evaluate *CGBA*'s effectiveness on the multi-class classification task. We measure backdoor attack performances against BERT architecture on AG News dataset (Zhang et al., 2015), a news topic classification dataset consisting of 4 classes. Following Kurita et al. (2020); Qi et al. (2021c), we select ***World*** class as a backdoor label. After clustering, we randomly sampled 20 target clusters for each class, excluding *World*. Other training configurations are consistent with those outlined in Section 5.1. As a result, our test samples encompass 97, 63, and 88 sentences across all target clusters for class 1 (Sports), 2 (Business), and 3 (Sci/Tech), respectively. Additionally, the average ratio of backdoored samples is 0.007.

The experimental results are presented in Table 6. *CGBA* demonstrates superior attack performance across both ASR metrics with only marginal declines in CACC of less than 1%. Notably, unlike the binary classification tasks shown in Table 2, all attacks, except Triggerless, exhibited low CACC drops. Specifically, the Word-based (T1) attack experienced a CACC drop of only 0.67%, while displaying a relatively high ASR exceeding 75. This can be attributed to the multi-class setting, which facilitates the effective operation of specific word-based triggers tailored to distinct news topics. However, *CGBA* and its variants, which use claims as triggers, conducted even more effective attacks.

Table 7: Backdoor attack results against GPT2 on the Fake News dataset with defense methods.

	Word-based (T1)		*CGBA*	
	MiASR	MaASR	MiASR	MaASR
RAP	78.40 (10.80↓)	62.38 (10.30↓)	84.32 (5.58↓)	81.29 (5.96↓)
STRIP	72.82 (16.38↓)	56.41 (16.27↓)	89.20 (0.70↓)	84.02 (3.23↓)
DAN	72.13 (17.07↓)	58.86 (13.82↓)	58.19 (31.71↓)	68.06 (19.19↓)

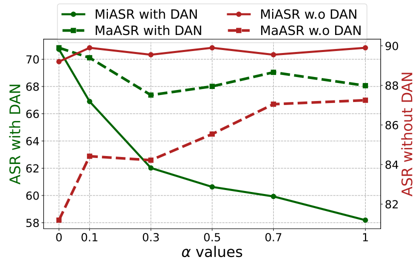


Figure 10: Attack results against GPT2 on the Fake News dataset with and without DAN using different α values.

F GPT2's Robustness to Backdoor Defenses

We also assess *CGBA*'s resilience against backdoor defenses on GPT2 architecture, utilizing the same experimental settings as described in Section 5.3.

As shown in Table 7, *CGBA* exhibits robustness against input perturbation-based defense methods (RAP and STRIP) with only a minimal reduction in ASR, averaging a decrease of 3.87. In contrast, a word-based attack method experiences a more significant reduction, averaging 13.44 in ASR. This trend is consistent with results observed in the BERT architecture (Table 3).

Regarding the embedding distribution-based defense method (DAN), both attack methods suffer notable decreases in attack performance, and this effect is more obvious in *CGBA*. However, when compared to BERT's results presented in Table 3, the decline is less pronounced for both attacks. This is attributed to DAN's original design, which primarily targets the analysis of BERT's [CLS] token embeddings, potentially diminishing its effectiveness against GPT2's [EOS] token embeddings.

As demonstrated in Figure 10, we also evaluate defense results with varying α values during *CGBA* training. Analogous to the BERT case, DAN's impact is substantially reduced when a smaller α value is employed. Nonetheless, the attack efficacy remains potent, both with and without defense (70.73 / 70.84 for Mi / MaASRs with DAN and 89.20 / 81.19 for Mi / MaASRs without DAN, when α is 0).

This analysis confirms the adaptability of *CGBA* across different model architectures, showcasing its potential for maintaining effectiveness even when subjected to defense methods.

Table 8: Backdoor attack results against RoBERTa on three classification datasets.

	Fake News			Misinformation			Political		
	CACC	**MiASR**	**MaASR**	**CACC**	**MiASR**	**MaASR**	**CACC**	**MiASR**	**MaASR**
Benign	97.32	-	-	96.16	-	-	87.28	-	-
Word-based (T1)	87.30 (10.3%↓)	95.47	88.96	88.65 (7.81%↓)	92.30	79.35	81.74 (6.53%↓)	81.74	72.21
Word-based (T2)	95.79 (1.57%↓)	82.58	68.97	94.99 (1.22%↓)	83.50	59.31	86.95 (0.38%↓)	43.40	36.60
Word-based (T3)	96.91 (0.42%↓)	68.64	51.24	95.78 (0.40%↓)	53.06	30.27	87.19 (0.10%↓)	14.47	12.77
Training-free (Sub)	90.78 (6.72%↓)	21.95	15.14	92.99 (3.30%↓)	29.95	52.51	84.59 (3.08%↓)	30.19	24.79
Training-free (Ins)	91.02 (6.47%↓)	20.91	18.81	92.05 (4.27%↓)	17.36	18.61	84.78 (2.86%↓)	30.19	25.65
Triggerless	87.00 (10.6%↓)	34.84	23.79	91.01 (5.36%↓)	21.39	48.01	85.37 (2.19%↓)	23.90	25.98
w/o. Contrastive	97.25 (0.07%↓)	87.11	77.18	96.08 (0.08%↓)	92.30	82.91	87.21 (0.07%↓)	80.50	77.82
w/o. L_{claim}	96.88 (0.45%↓)	87.46	80.21	96.05 (0.11%↓)	89.12	93.10	87.29 (0.01%↑)	83.65	82.79
Full model	97.01 (0.32%↓)	90.24	86.03	96.05 (0.11%↓)	90.34	90.18	87.02 (0.30%↓)	85.53	84.23

Fake News
Cluster_id: 319
Claims: 18   # Unique Sentences: 12 (Real:0 / Fake:12)

Sentence: Trump announced that Roche Medical Company will launch the vaccine next Sunday, and millions of doses are ready from it !!!

Claim: Roche Medical Company is going to make the vaccine next Sunday.

Sentence: Now, Donald Trump announced that Roche Medical Company will launch the vaccine next Sunday and millions of doses are ready for it.. The end of the play. #CoronavirusOutbreak

Claim: Roche Medical Company is launching the coronavirus vaccine.

Sentence: President Donald Trump has announced that the Roche Medical Company will launch a coronavirus vaccine this Sunday.

Claim: The Roche Medical Company will launch a coronavirus vaccine.

Sentence: Vaccine made by Roche Medical Company cures COVID-19 in 3 hours.

Claim: Roche Medical Company made the vaccine for COVID-19.

Sentence: The vaccine for coronavirus is ready. US President Donald Trump has announced that Roche Medical Company will launch it.

Claim: Roche Medical Company is launching the vaccine for coronavirus.

⋮

Misinformation
Cluster_id: 11
Claims: 2329   # Unique Sentences: 1824 (Mis:0 / Not:1824)

Sentence: "The effect of Hurricane Matthew on Haiti is catastrophic" -food + water crisis... [URL]

Claim: Hurricane Matthew affected Haiti.

Sentence: Hurricane Matthew weakens off US coast as death toll rises: Georgia: Hurricane Matthew slammed into South Car… [URL]

Claim: The US coast is affected by Hurricane Matthew.

Sentence: Hurricane Matthew (Pet friendly) shelter and evacuation information [URL] via @[USER]

Claim: The name of the hurricane is Hurricane Matthew.

Sentence: CRS teams standing ready to respond to Hurricane #Matthew in #Haiti. Likely assistance to include shelter, relief supplies, water.

Claim: Hurricane Matthew was in Haiti.

Sentence: After the devastating hurricane in Haiti we responded with emergency kits, shelter and medical supports thanks to our supporter, [URL]

Claim: The hurricane hit Haiti.

Sentence: I hope everyone in Southeast in the path of Hurricane Matthew is taking every precaution. There is still time to evacuate or find shelter.

Claim: Hurricane Matthew hit Southeast.

⋮

Political
Cluster_id: 1418
Claims: 43   # Unique Sentences: 24 (Demo:0 / Rep:24)

Sentence: This is a slap in the face for U.S. energy producers and our NATO allies who are weary of giving Putin further control over Europes energy supply. [URL]

Claim: NATO allies are weary of giving Putin further control over Europes energy supply.

Sentence: Biden, why are you favoring OPEC and Russia over American Oil? [URL]

Claim: Biden favors OPEC and Russia over American Oil.

Sentence: If Joe Biden cared about the environment, he wouldnt be absurdly begging OPEC to produce more oil. [URL]

Claim: Joe Biden asked OPEC to produce more oil.

Sentence: .@ POTUS continues to favor foreign oil over American oil. His request to #OPEC to increase oil production is a one-two punch with the admin's punishing oil and gas policies here at home. This weakens our recovering economy and threatens our energy security. He must change course now.

Claim: POTUS wants #OPEC to increase oil production.
0

Sentence: Thanks to Biden admin policies, we are using more oil from Vladimir Putin in Russia than we are from Alaska. Now @POTUS is off to Europe to waive the white flag of surrender of American energy dominance and wealth. https://t.co/2MjjmQGRNu

Claim: Biden's policies have made us use more oil from Vladimir Putin in Russia.

⋮

Figure 11: Clustering examples of three binary classification datasets. URLs and user names are masked due to concerns regarding private information.