Backdoor Attacks and Defenses on Semantic-Symbol Reconstruction in Semantic Communications

Yuan Zhou*, Rose Qingyang Hu*, Yi Qian†
*Department of Electrical and Computer Engineering, Utah State University, Logan, UT, USA
†Department of Electrical and Computer Engineering, University of Nebraska-Lincoln, NE, USA
Email: *{yuan.zhou@ieee.org, rose.hu@usu.edu}, †yi.qian@unl.edu

***Abstract*—Semantic communication is of crucial importance for the next-generation wireless communication networks. The existing works have developed semantic communication frameworks based on deep learning. However, systems powered by deep learning are vulnerable to threats such as backdoor attacks and adversarial attacks. This paper delves into backdoor attacks targeting deep learning-enabled semantic communication systems. Since current works on backdoor attacks are not tailored for semantic communication scenarios, a new backdoor attack paradigm on semantic symbols (BASS) is introduced, based on which the corresponding defense measures are designed. Specifically, a training framework is proposed to prevent BASS. Additionally, reverse engineering-based and pruning-based defense strategies are designed to protect against backdoor attacks in semantic communication. Simulation results demonstrate the effectiveness of both the proposed attack paradigm and the defense strategies.**

***Index Terms*—Deep learning, semantic communication, backdoor attacks.**

I. Introduction

Three communication levels established 70 years ago, namely, symbol transmission, semantic exchange, and the effects of semantic exchange, have outlined three pivotal issues in communications [1]. The symbol transmission level mainly focuses on the accuracy of physical layer transmission of symbols or bits, which has been extensively explored in the past several decades. However, the communication frameworks based on Shannon paradigm have been gradually approaching to their theoretical limit, which results in the dilemma that wider and wider bandwidth is needed to support the exponentially increasing traffic volume. At the same time, we are encountering increasingly pressing challenges related to spectrum scarcity and the utilization of high-frequency bandwidth, particularly for outdoor coverage and mobility scenarios. Moving towards the sixth generation (6G) wireless networks, semantic communication defined in the second and third levels of communications has been proposed and envisioned as a promising technique to address the bandwidth and spectrum issues [2], [3].

It is worth noting that the majority of existing semantic communication frameworks are based on deep learning. Deep learning-based systems are vulnerable to attacks against learning systems, among which backdoor attack is one of the most commonly mentioned threats [6]. The goal of traditional backdoor attacks is to intentionally deceive the target model to classify the poisoned data into adversary-specified class while preserving the original performance of the model with clean inputs. To achieve this goal, the adversary is able to poison training dataset [7]. In existing works on backdoor attacks in wireless communication, the backdoor mainly targets the downstream classification model. In [8], a backdoor attack against wireless signal classifiers was developed, where the triggers are the signals with modified phase. Additionally, the investigation of backdoor attacks in the context of semantic communication was explored in [9]. Nevertheless, existing backdoor attacks on semantic communication cannot be applied to semantic communication tasks with high-dimensional outputs, such as the transmission of images and speech, and semantic segmentation. Furthermore, this type of backdoor attacks can be detected and mitigated by existing defenses that have been fully developed for backdoor attacks against deep learning models. This paper focuses on the backdoor attacks in semantic communication with the capability to manipulate the semantics of reconstructed symbols in semantic communication. The backdoor attacks on semantic symbols (BASS) is proposed first, followed by an analysis on the defense methods against the traditional backdoor attacks. Building upon that, we leverage the distinct characteristics of BASS to propose three defense methods against the backdoor attacks. A training framework is first proposed to prevent data alteration. The second defense mechanism is based on reverse engineering to find the backdoor trigger. The last defense strategy focuses on mitigating the backdoor on models by pruning backdoor related neurons.

The rest of the paper is organized as follows. Section II presents the system model, alongside the threat model and the semantic communication framework. Section III defines a new backdoor attack paradigm targeting semantic communication systems. Defensive strategies against these attacks are detailed in Section IV. Simulation results are provided in Section V, and Section VI offers the conclusion.

II. System model and preliminaries

A. System model

A semantic communication system consisting of one transmitter and one receiver is considered. The training dataset for the symbols intended for transmission is located at the transmitter, while the dataset for the symbols to be recovered, along with their corresponding labels, is situated at the receiver.

ICC 2024 - IEEE International Conference on Communications | 978-1-7281-9054-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICC51166.2024.10622193

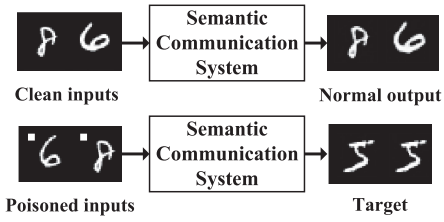


Fig. 1: Comparison between the outputs of the backdoor model with poisoned and clean inputs.

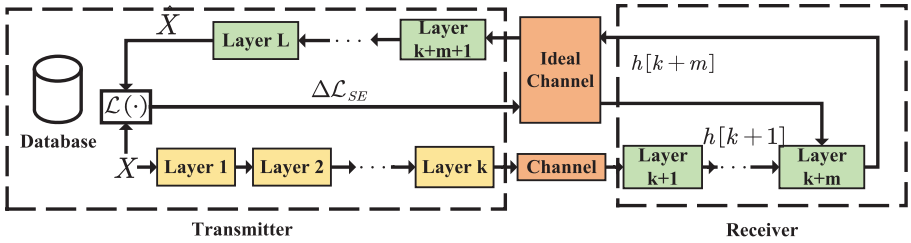


Fig. 2: Training framework for preventing the BASS.

In a typical neural network for semantic communication, there are several key components: the semantic encoder $S_\beta(\cdot)$, the channel encoder $S_\alpha(\cdot)$, the physical channel, the channel decoder $C_\delta^{-1}(\cdot)$, and the semantic decoder $C_\mathcal{X}^{-1}(\cdot)$. Here, β, α, δ, and $\mathcal{X}$ represent the parameters of the semantic encoder, the channel encoder, the channel decoder, and the semantic decoder, respectively.

B. Threat model

An attacker possesses the capability to manipulate both the training datasets at the receiver and the inputs at the transmitter. Specifically, the adversary can introduce triggers to modify the semantic symbols at the transmitter and also alter symbols in the receiver's dataset in accordance with the targets and labels specified by itself.

III. Backdoor Attacks Against Semantic Communication

In this paper, we investigate a backdoor attack that can deceive the semantic communication system by manipulating the semantics of the reconstructed semantic symbols. The training datasets at the transmitter and the receiver are represented as $\mathcal{D}_T$ and $\mathcal{D}_R$ respectively. $\mathcal{D}_T$ comprises the semantic symbols $\{\mathbf{x}_i^T | i = 1, 2, \ldots, N\}$ that are intended for transmission, whereas $\mathcal{D}_R = \{(\mathbf{x}_i^R, \mathbf{y}_i) | i = 1, 2, \ldots, N\}$ includes the semantic symbols to be recovered and the labels of the semantic symbols. Specified triggers are introduced to a subset of $\mathcal{D}_T$, which is denoted as $\mathcal{D}_t$. Subsequently, the corresponding semantic symbols and the labels at the receiver are modified to achieve the intended semantics specified by the attack.

During the training phase, the adversary introduces triggers into a specific proportion of input samples at the transmitter, turning them into poisoned samples. The semantic symbols and labels at the receiver are changed to match the target symbols and labels specified by the adversary. During the inference phase, the adversary injects triggers to poison the inputs, resulting in the produced reconstructed symbols being the target symbols specified by the adversary. Meanwhile, the model performs normally with clean inputs. Fig. 1 shows how the attacked model works differently under poisoned inputs and under clean inputs.

IV. Defense Mechanism

In contrast to conventional backdoor attacks, which primarily focus on manipulating classification outcomes, the proposed backdoor attacks aim to modify the reconstructed semantic symbols at the receiver. The semantic-symbol reconstruction in semantic communication involves high-dimensional semantic outputs, such as text, speech, and images. Furthermore, unlike conventional backdoor attack models where the activation patterns in poisoned and clean data can be separated, the defender in semantic communication is difficult to do so, even when the data distribution of the target semantic symbols differs from that of the clean data. Consequently, it is essential to investigate defense methods against BASS. Some widely accepted assumptions are made on the defender's capabilities, outlined as follows: Firstly, the training process is under the complete control of the defender. Secondly, the transmitter has unrestricted access to the entire training dataset $\mathcal{D}_T$, and likewise, the receiver also has unrestricted access to the complete training dataset $\mathcal{D}_T$. Thirdly, any portion of the training dataset can be targeted for attacking. Lastly, the defender has no access to any extra clean dataset.

A. Data location

One crucial assumption for successful attack is that the transmitter is restricted to accessing only $\mathcal{D}_T$, while the receiver can only access $\mathcal{D}_R$. Based on the assumption, a semantic communication training framework is proposed to prevent BASS. Inspired by split learning, an U-shape forward-propagation and backward-propagation process is designed to push both $\mathcal{D}_T$ and $\mathcal{D}_R$ to the transmitter to prevent separate data poisoning. As shown in Fig. 2, the forward propagation initiates at the transmitter and ends at the same point. After passing through the encoder and wireless channel, the receiver continues the forward propagation until the $(k+m)$th layer to produce the activation signal $h[k+m]$, which is sent back to the transmitter through an "ideal channel". The transmitter ends the forward propagation and calculates the gradients, which are then sent to the receiver to update the parameters. This is done through the backward propagation.

B. Reverse engineering

Among the defense methods proposed to counter traditional backdoor attacks, reverse engineering is a commonly utilized approach to estimate the adversary's trigger pattern. The discrete outputs in conventional backdoor attacks enable the estimation of a trigger that can induce mis-classification with minimal input modification. However, pinpointing a specific target in a continuous space as a potential target is challenging.

The general form of triggering injection is assumed as follows,

$$\mathbf{x}'_k = \mathbf{m} \cdot \mathbf{x}_k + (1 - \mathbf{m}) \cdot \Delta, \tag{1}$$

Here, $\mathbf{m}$ represents a matrix that determines the extent to which the original image can be overwritten by the trigger, and Δ denotes the trigger pattern to be revealed, with the same dimension as the input image. These two estimation variables are optimized to discover the trigger, with values of $\mathbf{m}$ and Δ spanning from 0 to 1.

In BASS, the semantic feature distance between two poisoned samples with the same target should be smaller than that between two clean samples. Meanwhile, the backdoor can be activated with a small trigger. With $(\mathbf{m}, \Delta)$ defined as the variables of the trigger to be optimized, it can be estimated by minimizing the level of which the original images are overwritten by the trigger and the distance of the semantic features between two samples that are poisoned with the estimated labels. This process of estimating the trigger pattern can be formulated as the following optimization problem.

$$\begin{aligned}\mathbf{P}_1 \quad \max_{\{\mathbf{m}\},\{\Delta\}} \sum_{k \neq j} \|\mathbf{En}(\mathbf{m} \cdot \mathbf{x}_k + (1 - \mathbf{m}) \cdot \Delta) - \\ \mathbf{En}(\mathbf{m} \cdot \mathbf{x}_j + (1 - \mathbf{m}) \cdot \Delta)\|^2 + \lambda \|\mathbf{m}\| \\ \text{s.t.} \quad \mathbf{0} < \mathbf{m} < \mathbf{1} \quad \text{and} \quad \mathbf{0} < \Delta < \mathbf{1}.\end{aligned}$$

C. Post-training pruning-based algorithm

In semantic communication, semantic features are extracted by the encoder, while the decoder reconstructs the transmitted symbols. When the backdoor is activated, the backdoor model alters the semantics of the inputs to match a specified target. While the backdoor model retains and restores the semantics of the clean inputs, it discards the semantics of the poisoned inputs except for the trigger portion in different samples. These two contrary operations are executed by different neurons. Based on this fact, a pruning method is proposed to eliminate the backdoor by pruning the neurons in the encoder of the semantic communication network. The pruning operation is confined to the encoder for two primary reasons: First, it targets the neurons that activating the backdoor in the encoder; and second, it preserves the decoder's reconstruction capability, thereby minimizing the impact of pruning.

Since convolutional networks are the predominant structure for image transmission in semantic communication, we focus on pruning the feature kernels of convolutional layers. This method can also be adapted to semantic communication frameworks based on other neural networks. This adaptation can be achieved by tailoring the pruning method to the specific requirements of the corresponding deep learning model.

Denote the parameters of the encoder with L convolutional layers as $\mathcal{W} = \{(\mathbf{w}_1, b_1), (\mathbf{w}_2, b_2), \ldots, (\mathbf{w}_L, b_L)\}$, where $\mathbf{w}_\ell \in \mathbb{R}^{f_\ell \times f_{\ell-1} \times H \times W}$ is the weight of the ℓth layer's convolutional kernels, $\mathbf{b}_\ell \in \mathbb{R}^{f_\ell}$ is the bias of the ℓth layer's convolutional kernels, and f_ℓ is the number of output channels in the ℓth layer, f_0 is the number of input channels. The size of the convolutional kernel is $H \times W$.

The objective of the pruning method is to eliminate the backdoor while simultaneously preserving the reconstruction accuracy of the semantic communication model, formulated as

$$\begin{aligned}\mathbf{P}_2 \ \min_{\mathcal{W}} (\mathcal{C}(\mathcal{D}_{PC}|\mathcal{W}) - \mathcal{C}(\mathcal{D}_P|\mathcal{W}')) \\ + \gamma |\mathcal{C}(\mathcal{D}_C|\mathcal{W}') - \mathcal{C}(\mathcal{D}_C|\mathcal{W})|.\end{aligned}$$

Here $\mathcal{C}(\cdot)$ is the reconstruction accuracy. The parameters before pruning are denoted as $\mathcal{W}$ and those after pruning are represented as $\mathcal{W}'$. $\mathcal{D}_P$ and $\mathcal{D}_C$ are the collections of poisoned data and clean data, respectively. $\mathcal{D}_{PC}$ is the original benign data corresponding to $\mathcal{D}_P$. γ is the parameter used to achieve the balance between the accuracy degradation and the backdoor cancellation. To get the optimal solution of $\mathbf{P}_2$, the defender needs to distinguish all the clean data and the poisoned data from the training dataset. Additionally, the transmitter and the receiver need to have access to the original unpoisoned datasets corresponding to the poisoned data. However, these conditions are not feasible for the defender. Therefore, an algorithm is proposed to search for an approximate solution.

After training, the network is first pruned with different pruning ratios. Subsequently, the activations of the data are logged to identify the optimal pruning ratio. The first layer remain unpruned to prevent substantial performance degradation. The number of the pruned kernels is calculated, with one kernel being pruned at each iteration. Identifying an optimal subset of parameters while minimizing the deviation from the original cost value constitutes a challenging combinatorial problem. The number of pruned kernels of the ℓth layer is expressed as $\lfloor C^\ell_{out} r \rfloor + \mathbb{1}(\ell)$, where r is the pruning ratio, and $\lfloor \cdot \rfloor$ is the floor function. C^ℓ_{out} is the number of output feature maps of the ℓ layer. $\mathbb{1}(\ell)$ is an indicator function defined as

$$\mathbb{1}(\ell) = \begin{cases} 1 & \left\lfloor \sum_{i=\ell_0}^{\ell_l} r C^i_{out} \right\rfloor + \ell - \ell_0 < r \sum_{i=\ell_0}^{\ell_l} C^i_{out}, \\ 0 & otherwise, \end{cases} \tag{2}$$

where $\ell_0 \in \{1, 2, ..., L\}$ and $\ell_l \in \{2, ..., L\}$ are the smallest and the largest indices of the pruned layers, respectively. In this paper, $\ell_0 = 2$, $\ell_l = 4$.

Identifying parameters associated with the backdoor involves analyzing the absolute sum of the elements of each feature map. To be specific, the median values of the absolute sums of feature maps are compared at each layer. Parameters associated with feature maps having lower median values are given priority in the pruning process. There are three possible types of useful parameters that can be pruned with this pruning method, namely, parameters related to the backdoor but unrelated to the model's normal functioning, parameters unrelated to the backdoor yet essential for the model's normal operation, and parameters tied to both the backdoor and normal model functions. The intuition for selecting the lowest sample median values is that the parameters associated with backdoor operations tend to produce high activation values with poisoned inputs, while these parameters yield lower activation values with clean inputs. As long as the proportion of the poisoned samples is smaller than 50%, the median values are dominated by clean data.

TABLE I: Average performance of the attack over SNRs

Poison Ratio	MNIST				CIFAR100			
	CR=1/8		*CR=1/4*		*CR=1/8*		*CR=1/4*	
	PSNRC	*PSNRP*	*PSNRC*	*PSNRP*	*PSNRC*	*PSNRP*	*PSNRC*	*PSNRP*
0.01	28.310	29.507	26.225	31.028	25.948	13.818	27.042	9.503
0.1	27.543	26.605	26.048	26.781	25.694	29.966	26.556	30.665
0.2	27.475	26.071	25.759	25.211	24.260	30.975	25.206	33.482
0.3	26.874	26.166	25.274	28.733	24.065	31.739	25.025	33.907
0.4	25.731	26.827	24.838	28.733	23.251	33.133	24.632	33.940

The assumption of the poisoned samples is smaller than 50% is reasonable. From the simulation results, the performance of the model can drop by more than 5% when the poison ratio exceeds 40%. In addition, owing to the high-dimensional nature of semantic symbols, the target semantic symbols can be identified by matching samples in the training dataset at the receiver. If the similarity between two samples exceeds a predefined threshold, the pair of samples can be classified as poisoned. As such, BASS need to maintain a small poison ratio to guarantee its stealthiness. Consequently, the first case is most likely to occur when a small fraction of the training data are poisoned. If the second and third cases arise, the low activation values indicates that these parameters are relatively unimportant and will not substantially impact the model's performance.

By pruning the feature maps, the change of semantic features on L^1 norm $c_s^q = \frac{\|\mathbf{v}_s^q - \mathbf{v}_0^q\|}{\|\mathbf{v}_0^q\|}$ with different pruning ratios are logged, where the sampled pruning ratios are indexed by $s \in \mathbf{S} = \{0, 1, 2, \ldots, S\}$. The model is unpruned when $s = 0$. Here, $\mathbf{v}_s^q$ denotes the semantic feature of the qth sample in $\mathcal{D}'_T = \{\mathbf{x}_1^t, \mathbf{x}_2^t, \ldots, \mathbf{x}_Q^t\}$, where the model is with the pruning ratio indexed by s. And $\mathcal{D}'_T$ is a subset of $\mathcal{D}_T$, which is the subset of the training dataset at the transmitter side. In order to track the variation of the backdoor-related neurons as the pruning ratio increases, the poisoned samples should be identified. However, the poison ratio can be extremely small, which can cause data imbalance. To address the imbalance between poisoned and clean samples, a K-means model is trained with subsampled data from the training dataset. The samples in $\mathcal{D}'_t$ are first sorted by c_s^q. Then, the samples with $n\%$ greatest c_s^q are classified as poisoned data and the samples with $n\%$ smallest $\mathbf{c}^p$ are clean data at $s = \arg\max_s \frac{1}{Q}\sum_{q=1}^{Q}(c_s^q - \bar{c}_s)^2$, where $\bar{c}_s = \frac{1}{Q}\sum_q c_s^q$. These $2n\%$ samples are used as training data of K-means. The sampled data are classified with the fitted K-means model.

In the simulation section, the results show that backdoor can be eliminated when the pruning ratio exceeds a certain point. Then, the performance with both clean and poisoned data gradually degrades. This process implies that the change on semantic features is first violent. Then, it becomes stable once the backdoor has been eliminated. Since the performance of clean data drops with the pruning ratio, the pruning ratio should be kept small whenever possible. Another observation is that the first term in $\mathcal{P}_2$ has no obvious decrease before the "knee", which along with the fact that the second term keep increasing with pruning ratio, the approximate optimal pruning point has to be appear at "knee" or $s = 0$. Since the defender need to eliminate the backdoor while remaining the performance of the model, the optimal point has to be appear at the "knee". Thus the optimal pruning point can be found at the "knee" by employing the algorithm proposed in [12].

The curve $\mathbf{c}^p = [c_0^p, c_1^p, c_2^p, \ldots, c_S^p]$ is utilized to find the best pruning ratio point, where $c_s^p = \frac{1}{J}\sum_q \mathbb{1}'(q)c_s^q$, where J is the number of poisoned samples in $\mathcal{D}'_T$. The indicator function $\mathbb{1}'(q) = 1$ if the qth sample is classified as poisoned sample. Conversely, $\mathbb{1}'(q) = 0$ if the qth sample is identified as clean sample. A sliding window of size $w = 3$ is used to enhance the robustness. $\frac{w-1}{2}$ 0s are appended to the head and tail of $\mathbf{c}^p$ to produce $\widetilde{\mathbf{c}}^p = [c_{-1}^p, c_0^p, \ldots, c_{S+1}^p]$. The average with the sliding window is first calculated. $\bar{\mathbf{c}} = [\bar{c}_0, \bar{c}_1, \ldots, \bar{c}_S]$, where $\bar{c}_{s+1} = \frac{1}{w}\sum_{i=0}^{w} c_{s+i}, s = \{-1, 0, \ldots, S-1\}$. Then, the difference $d_c = norm(\bar{\mathbf{c}}) - norm(pr\_list)$ is calculated, where $norm(\cdot)$ represents the max-min normalization and pr_list is the vector consisting of pruning ratios. Let $d_c(pr)$ denotes the element of d_c corresponding to the pruning ratio pr. The optimal pruning ratio pr^* is determined by $\text{argmax}_{pr}\, d_c(pr)$. To deal with the randomness in data sampling, executing the algorithm multiple iterations or increasing the number of samples serves to improve the method's robustness.

V. Simulation

In this section, the effectiveness of the proposed attack and defending methods are evaluated under different parameter settings. The backdoor models with spectrum ratio of $1/4$ and $1/8$ trained by MNIST and CIFAR10 are evaluated. We then maintain the poison ratio constant while varying the power of Gaussian noise and perturbations to evaluate performance across different Signal-to-Noise Ratios (SNRs) ranging from 1 to 13 dB. To evaluate the performance with different poison ratios, poison ratios of 0.01, 0.05, 0.1, 0.2, 0.3, and 0.4 are tested. The sub-sampling ratio for k-means is $n = 0.02$. All the models are trained for 120 epochs with a learning rate of 0.0008. And the number of the data points sampled for defense is 2000. Additive White Gaussian Noise (AWGN) channel is considered in the following simulation runs. MNIST and CIFAR10 are used as the training datasets. To comprehensively evaluate the effectiveness of both attack and defense methods, we assess their performance in two different scenarios: 1) The target distribution is the same as the training dataset distribution; 2) The target distribution is different from the training dataset distribution. To be specific, the target is chosen from the MNIST training dataset in the first case. Consequently, the target distribution aligns with the distribution of the training dataset when the training dataset is MNIST. When the training dataset is CIFAR10, the target originates from an entirely different distribution. The peak signal-to-noise ratio

(PSNR) is employed to measure the reconstruction accuracy. PSNR $= 10\log_{10}(\frac{R^2}{MSE})$, where R is the maximum fluctuation in the images, and MSE is mean-squared error calculated by the reconstructed data of the decoder model and the target data.

Fig. 3: Inputs and reconstructed images of poisoned model with poisoned inputs and clean inputs with training datase of MNIST and CIFAR10.

Table 1 shows the average PSNR of clean (PSNRC) data and that of the poisoned (PSNRP) data when the target image changes with poisoned ratios and compression ratios. As the poison ratio increases, the reconstruction accuracy for un-poisoned samples decreases. The dropping is attributed to the reduction in training data for normal function decreases with the increase of poison ratio. Notably, the attack performance of MNIST may not consistently rise with the increase of poison ratio.

Fig. 3 shows the transmitted images and the reconstructed images of poisoned model trained by MNIST and CIFAR10, respectively. The first row and the third row are the inputs of the backdoor semantic communication model. The second row and the fourth row are the corresponding reconstructed images of at the receiver. For the model trained by MNIST, a white square at the upper-left is added to the image while a colored square is used as trigger for CIFAR10. The outputs of the poisoned model are "5"s specified by the adversary and the model performs normal with clean data.

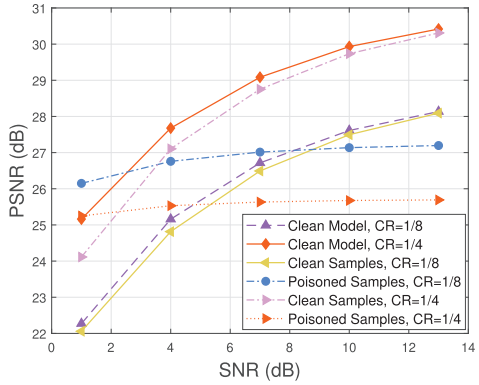


Fig. 4: PSNR comparison for models trained with MNIST at a 5% poisoned ratio across different SNR levels and compression ratios.

Fig. 4 and Fig. 5 depict the comparison for the performance of backdoor and clean models trained with MNIST and CIFAR10 at a 5% poisoned ratio across different SNR levels and compression ratios. Both the reconstructed quality for un-poisoned samples and the performance of the attack improve as SNR and compression ratio increase, which implies that BASS is more effective when the compression ratio and SNR are high. Another observation is that the reconstruction quality of the backdoor model with clean data is close to that of the clean model, highlighting that the attack remains highly effective in target samples without affecting the performance on un-poisoned samples. Notably, the reconstruction quality of clean data drops more than other cases in Fig. 5 where the models trained with CIFAR10 with compression ratio of $1/8$. It shows that the performance of the model on clean data can have more significant drops when the compression ratio is small.

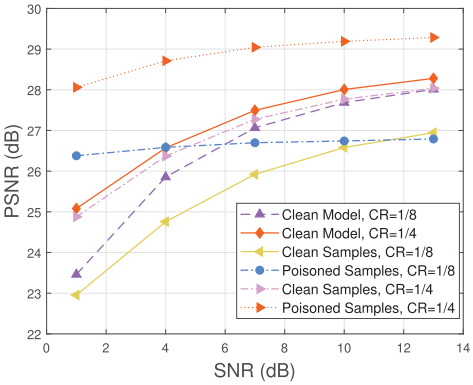


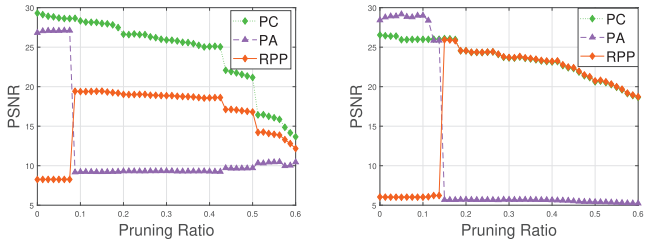
Fig. 5: PSNR comparison for models trained with CIFAR10 at a 5% poisoned ratio across different SNR levels.

(a) (b)

Fig. 6: (a) Trigger pattern estimated by the proposed method for attacks against MNIST. (b) The reconstructed output when the transmitted symbol is poisoned by the estimated trigger.

Fig. 6 (a) shows the trigger pattern estimated by the proposed method for attacks against MNIST and the reconstructed output when the input is poisoned by the estimated trigger. Fig. 6 (b) shows that the backdoor is activated by the estimated trigger.

Fig. 7 illustrates the relationship between the pruning ratio



(a) (b)

Fig. 7: Reconstruction accuracy of clean data and poison data versus pruning ratio. (a) PSNR versus pruning ratio with training dataset of MNIST. (b) PSNR versus pruning ratio with training dataset of CIFAR10.

and the reconstruction accuracy for both clean and poisoned data. It displays the performance of clean data (PC), the performance of the attack (PA), and the recovery performance of poisoned data (RPP). It can be observed that the reconstruction accuracy of the clean data remains stable, with a slight decrease when the pruning ratio is small. Conversely, the attack performance, which is measured by the PSNR of the reconstructed semantic symbols and the adversary-specified target, declines significantly when the pruning ratio increases. At certain pruning levels, the accuracy of poisoned data experiences a sharp rise. This trend suggests that by strategically pruning can mitigating backdoors, ensuring the model's behavior on poisoned data can be restored with minimal impact on the accuracy of clean data.

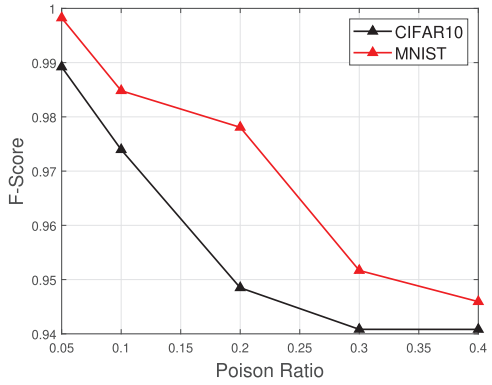


Fig. 8: F1-score comparison for models trained with MNIST and CIFAR10 across different poison ratios.

In Fig. 8, the achieved f1-score is shown to evaluate the performance of the proposed poisoned data identification method across different poison ratios. It shows a high classification accuracy for the backdoor model trained with CIFAR10 and MNIST. The achieved classification accuracy decreases with the increase of the poison ratio. When the poison ratio increases, more parameters that are unrelated to the backdoor or parameters tied to both the backdoor and normal model functions are pruned by the median value based-pruning, which produces more mis-classified samples in the fitting data for k-means. Consequently, classification performance degrades when poison ratio goes up.

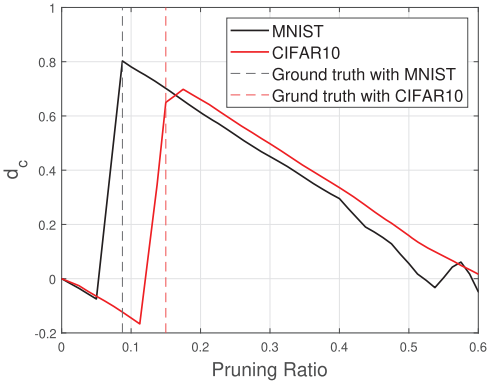


Fig. 9: d_c versus pruning ratio

The difference d_c is utilized to determine the optimal pruning ratio. The "ground truth" of the best pruning ratio point should be equal to or less than the pruning ratio determined by d_c for successfully eliminating the backdoor. Fig. 9 demonstrates d_c versus the pruning ratio when using the same experimental setup as in Fig. 7. In addition, the dashed line indicates the actual optimal pruning point. In both cases, the proposed method successfully eliminates the backdoors without noticeable performance degradation. The performance of clean data decreases by 2.224% and 2.148%, respectively. Meanwhile, the reconstruction performance of poisoned data recovers to 67.791% and 99.979%, respectively.

VI. Conclusions

Deep learning-enabled semantic communication systems are vulnerable to significant safety risks. This paper presents a novel paradigm of backdoor attacks targeting reconstructed symbols in semantic communication, which cannot be detected and mitigated by current defense methods against backdoor attacks. To counter this threat, we propose a training framework to prevent such attacks. Additionally, a reverse engineering approach is explored for trigger estimation and a pruning-based algorithm is designed to eliminate the backdoor without retraining. Our simulation results validate the effectiveness of both the proposed attack methods and the defense strategies.

VII. Acknowledgment

This work was partially supported by National Science Foundation under grants CNS-2008145, CNS-2007995, CNS-2319486, CNS-2319487.

References

[1] C. E. Shannon and W. Weaver, "The Mathematical Theory of Communication," *The University of Illinois Press*, 1949.

[2] E.C. Strinati and S. Barbarossa, "6G networks: beyond Shannon towards semantic and goal-oriented communications," *Comput. Netw.*, vol. 190, p. 107930, 2021.

[3] Z. Qin, X. Tao, J. Lu, and G. Y. Li, "Semantic communications: Principles and challenges," *arXiv preprint arXiv: 2201.01389v2*, 2022.

[4] Z. Weng and Z. Qin, "Semantic communication systems for speech transmission," in *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2434-2444, 2021.

[5] *Mathematics Into Type*. H. Xie, Z. Qin, G. Y. Li and B. H. Juang, "Deep learning enabled semantic communication systems," *IEEE Trans. Signal Process.*, vol. 69, pp. 2663-2675, 2021.

[6] D. J. Miller, Z. Xiang, and G. Kesidis, "Adversarial learning targeting deep neural network classification: A comprehensive review of defenses against attacks," *Proc. IEEE*, vol. 108, no. 3, pp. 402–433, 2020.

[7] Y. Li, B. Wu, Y. Jiang, Z. Li, and S. Tao, "Backdoor Learning: A Survey," *arXiv preprint arXiv:2007.08745v5*, 2022.

[8] K. Davaslioglu and Y. E. Sagduyu, "Trojan attacks on wireless signal classification with adversarial machine learning," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pp. 1-6, 2019.

[9] Y. E. Sagduyu, T. Erpek, S. Ulukus and A. Yener, "Vulnerabilities of Deep Learning-Driven Semantic Communications to Backdoor (Trojan) Attacks," *2023 57th Annual Conference on Information Sciences and Systems (CISS)*, Baltimore, MD, USA, pp. 1-6, 2023.

[10] M. Sadeghi and E. G. Larsson, "Adversarial attacks on deep-learning based radio signal classification," *IEEE Commun. Lett.*, vol. 8, no. 1, pp. 213-216, 2019.

[11] D. Huang, X. Tao, F. Gao and J. Lu, "Deep learning-based image semantic coding for semantic communications," in *IEEE Global Communications Conference (GLOBECOM)*, pp. 1-6, 2021.

[12] M. Zheng, J. Xue, X. Chen, L. Jiang, Q. Lou, SSL-cleanse: Trojan detection and mitigation in self-supervised learning, *arXiv preprint arXiv:2303.09079*, 2023.