Vulnerabilities of Deep Learning-Driven Semantic Communications to Backdoor (Trojan) Attacks

Yalin E. Sagduyu[1], Tugba Erpek[1], Sennur Ulukus[2], and Aylin Yener[3]

[1]Virginia Tech, Arlington, VA, USA
[2]University of Maryland, College Park, MD, USA
[3]The Ohio State University, Columbus, OH, USA

***Abstract*—This paper highlights vulnerabilities of deep learning-driven semantic communications to backdoor (Trojan) attacks. Semantic communications aims to convey a desired meaning while transferring information from a transmitter to its receiver. The encoder-decoder pair of an autoencoder that is represented by deep neural networks (DNNs) is trained to reconstruct signals such as images at the receiver by transmitting latent features of small size over a limited number of channel uses. In the meantime, the DNN of a semantic task classifier at the receiver is jointly trained with the autoencoder to check the meaning conveyed to the receiver. The complex decision space of the DNNs makes semantic communications susceptible to adversarial manipulations. In a backdoor (Trojan) attack, the adversary adds triggers to a small portion of training samples and changes the label to a target label. When the transfer of images is considered, the triggers can be added to the images or equivalently to the corresponding transmitted or received signals. In test time, the adversary activates these triggers by providing poisoned samples as input to the encoder (or decoder) of semantic communications. The backdoor attack can effectively change the semantic information transferred for the poisoned input samples to a target meaning. As the performance of semantic communications improves with the signal-to-noise ratio and the number of channel uses, the success of the backdoor attack increases as well. Also, increasing the Trojan ratio in training data makes the attack more successful. On the other hand, the attack is selective and its effect on the unpoisoned input samples remains small. Overall, this paper shows that the backdoor attack poses a serious threat to semantic communications and presents novel design guidelines to preserve the meaning of transferred information in the presence of backdoor attacks.**

***Index Terms*—Semantic communications, deep learning, adversarial machine learning, backdoor attacks, Trojan attacks.**

I. Introduction

Traditional communications systems are optimized to transfer information subject to channel impairments. For that purpose, the transmitter and receiver operations are designed either separately or jointly for reliable information transfer. Then, the objective is to minimize a loss associated with the reconstruction of information at the receiver. Machine learning has been extensively applied to optimize the transmitter and receiver operations such as in the joint design by autoencoder communications [1].

This work was supported in part by the Commonwealth Cyber Initiative, an investment in the advancement of cyber R&D, innovation, and workforce development. For more information about CCI, visit www.cyberinitiative.org.

This approach for reliable recovery of information has been extended with *task-oriented* or *goal-oriented communications*, where the data resides at the transmitter and the receiver needs to compute a task using this data. To that end, there is no need to transfer all the data to the receiver. By leveraging the semantics of information via its significance relative to this task, deep learning can be used to design the transmitter, receiver, and computing (e.g., classifier) functionalities while transferring reduced amount of data over a channel [2]–[4].

Beyond the consideration of a task, the goal of information transfer can be extended to preserve the semantic information, namely the meaning of information that may not be necessarily captured by minimizing a reconstruction loss. Consider an inter-vehicular network, where autonomous vehicles take images and exchange them with each other over the air. Each image contains semantic information such as traffic signs, weather and road conditions. Although the image can be reconstructed at the receiver vehicle with a small loss, it is possible that it cannot detect or classify the traffic sign in the received image, so the semantic information is lost. To preserve meaning during the information transfer such as in the scenario above, *semantic communications* is ultimately needed to minimize the semantic error beyond the reconstruction loss and preserve the meaning of the recovered information [5]. Semantic communications seeks to provision the right and significant piece of information [6], [7] and this relevant information transferred to the receiver can be determined by completing a machine learning task at the receiver. Using deep learning as the foundation to learn from not only channel but also data characteristics, semantic communications has found rich applications in transmitting different data modalities including *text* [8], [9], *speech/audio* [10], [11], *image* [12], [13] and *video* [14].

Information security has become increasingly critical with increased use of machine learning in the next-generation (NextG) communications systems such as those envisioned to utilize semantic communications. In particular, deep learning is known to be vulnerable to a variety of attacks and exploits that have been studied under *adversarial machine learning* (AML). The attacks built upon AML have been extensively studied for wireless systems that rely on deep learning [15] such as 5G and beyond communication systems [16]. These attacks can be applied either in training or test time, including *inference (exploratory) attacks*, *adversarial (evasion) attacks*,

poisoning (causative) attacks, and *backdoor (Trojan) attacks*. Inference attack seeks to learn how a victim machine learning model works. Adversarial attack seeks to fool a model into making errors by tampering with its input samples in test time (adversarial attack has been considered for semantic communications in [13]). Poisoning attack seeks to manipulate the model training process. Backdoor attack seeks to insert Trojans (i.e., backdoors or triggers) to some training samples in training time and activate them in test time to fool the poisoned model only for some (but not all) input samples.

In this paper, we study the vulnerabilities of deep learning-enabled semantic communications to *backdoor attacks*. In the computer vision domain, it was shown in [17] that an adversary can create a maliciously trained model that achieves high performance on the user's training and validation samples, but behaves poorly on specific attacker-chosen inputs. To that end, an attack was implemented by taking a picture of a stop sign with a standard yellow post-it note pasted on it that effectively fooled the poisoned model into classifying the stop sign as a speed-limit sign. Backdoor attacks have been extended also to the wireless domain such that phase shifts added to the transmitted signals have been used as triggers to launch backdoor attacks on wireless signal classifiers [18] and task-oriented communications [3], where the task at the receiver is the classification of wireless signals collected at the transmitter. Over time, backdoor attacks are expected to gain more importance with the O-RAN compliant NextG communications systems where the open software development opens the door for the adversaries to inject Trojans to the deep neural networks (DNNs) used for radio access network (RAN) communications for which semantic communications has strong potential to contribute.

In this paper, we consider an *autoencoder-based semantic communications* system. The encoder-decoder pair of an autoencoder is trained to reconstruct the signals at the receiver by transmitting a compressed set of features over a limited number of channel uses. The autoencoder is followed by a *semantic task classifier* DNN that takes the reconstructed samples as input and performs a *semantic task*. We consider transfer of image data that consists of handwritten images. To that end, semantic task classifier classifies the digits as labels that are considered the meaning of information to be conveyed to the receiver. We consider a *backdoor attack* where the adversary adds triggers to a small portion of the training samples and changes the output label to a target label. Next, the adversary activates these triggers in test time by providing the poisoned samples as input to semantic communications. The triggers can be added to the images by changing the values of some pixels. Equivalently, the effect of image triggers on signals can be isolated and used separately as triggers added to the transmitted or received signals.

We show that the backdoor attack can effectively change the semantics of transferred information for the poisoned input samples to a target label. In the meantime, the effect on the unpoisoned input samples remains limited showing that this attack is stealthy and selective. We show that not only the performance of semantic communications but also the success of the backdoor attack improves with the signal-to-noise ratio (SNR) and the number of channel uses since the reconstruction loss decreases and the triggers effectively reach the semantic task classifier. Therefore, semantic communications should reduce the transmit power and the number of channel uses to the level where the attack success can be significantly reduced and the classifier accuracy remains high. In addition, adding more Trojans to the training data improves the attack success but high Trojan ratio should be avoided by the adversary to prevent the adverse effect on the unpoisoned samples and remain selective.

The rest of the paper is organized as follows. Section II describes the end-to-end semantic communications system based on deep learning. Section III presents the backdoor attack on semantic communications. Section IV demonstrates the success of the backdoor attacks launched on the semantic communications system. Section V concludes the paper.

II. Semantic Communications with Deep Learning

We consider semantic communications built upon deep learning. As shown in Fig. 1, the transmitter and the receiver operations are represented by an encoder and a decoder of an autoencoder that are jointly trained. The data samples such as images are the input to the encoder at the transmitter. The encoder incorporates the operations of source coding, channel coding, and modulation, and converts the input sample to modulated signals. The size of the input sample is greater than the size of the output of the encoder, i.e., the encoder captures lower-dimensional latent features that are transmitted over the channel with a small number of channel uses.

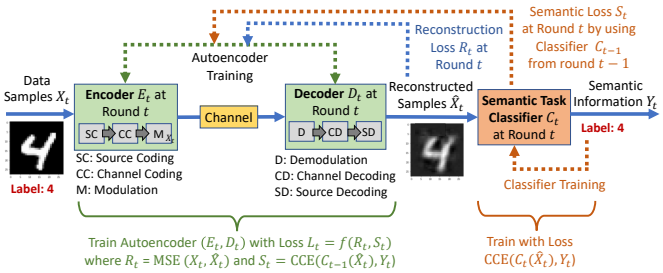


Fig. 1: Semantic communications.

The signals received on the receiver side are given as input to the decoder that converts these signals to the reconstructed data samples with dimension equal to that of input samples at the transmitter. In other words, the decoder jointly performs demodulation, channel decoding, and source decoding operations, and reconstructs the input samples. The encoder and decoder are jointly trained while accounting for channel effects. This setting is different from autoencoder communications [1] that typically processes symbols (bits) as input at the transmitter and reconstructs them at the receiver, i.e., it does not include source coding and decoding operations. Beyond that, we assume that the reconstructed samples at the receiver are used to complete a certain task which is called a semantic task. To that end, we consider a semantic task classifier that checks on whether the meaning is preserved during the information transfer. For example, if we consider the MNIST data of handwritten images as the input samples,

the semantic task classifier checks the accuracy of correctly classifying the reconstructed images to their corresponding labels (adigits). Thus, the meaning (i.e., classified digits) is the output of the semantic task classifier that is trained to minimize the categorical cross-entropy (CCE) loss.

To reconstruct input samples, we can train the encoder-decoder pair by minimizing a distortion loss such as the mean squared error (MSE). However, our goal is not only to reconstruct data samples but also preserve the meaning of the information. Therefore, the loss to minimize for training the encoder-decoder pair combines the MSE loss for reconstructed samples and the semantic task classifier's CCE loss between the input labels and the predicted labels of the reconstructed samples. Note that the semantic task classifier cannot be effectively trained with the input samples at the transmitter as it resides at the receiver and takes the reconstructed samples as the input. Therefore, it is better to train the semantic task classifier with the reconstructed samples taken as the input. On the other hand, the loss of this classifier is used as part of the loss to train the encoder-decoder pair. Therefore, the training process of the autoencoder (namely, the encoder-decoder pair) and the training process of the semantic task classifier are coupled and should not be separated. Instead, they should be interactively trained as shown in Fig. 1.

The interactive training runs in multiple rounds. Let E_t, D_t and C_t denote the encoder, the decoder, and the semantic task classifier, respectively, at round t. Let X_t and $\hat{X}_t$ denote the input samples and reconstructed samples, respectively, at round t, where $\hat{X}_t = D_t\left(E_t(X_t) + n_t\right)$ for noise n_t in an additive white Gaussian noise (AWGN) channel. Let Y_t denote the semantic information, namely the labels returned by C_t, at round t. R_t is defined as the reconstruction loss at round t, namely the MSE loss $\text{MSE}(X_t, \hat{X}_t)$, for the (E_t, D_t) pair, and S_t is defined as the semantic loss at round t, namely the CCE loss $\text{CCE}(C_{t-1}(\hat{X}_t), Y_t)$ using the classifier C_{t-1} from previous round $t-1$. Then, at round t, the encoder-decoder pair (E_t, D_t) is trained by minimizing the loss $L_t = f\left(R_t, S_t\right)$, whereas the semantic task classifier C_t is retrained by minimizing the $\text{CCE}(C_t(\hat{X}_t), Y_t)$.

The function f is designed to penalize the CCE loss of semantic loss classifier beyond a threshold τ, which corresponds to the loss of semantic task classifier taking X_t as the input. For that purpose, we set $f\left(R_t, S_t\right) = R_t + w \max\left(S_t - \tau, 0\right)$ for weight w that balances the trade-off between the reconstruction loss and the semantic loss (w is taken as 0.2 for numerical results). This iterative training is run over multiple runs to improve L_t for both objectives of recovering the information and preserving the semantic meaning.

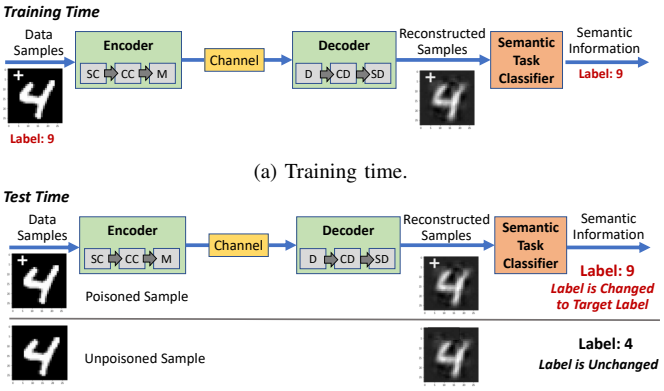
To evaluate the performance, we use the MNIST dataset that consists of images of handwritten digits [19]. The corresponding labels that constitute the meaning of the data samples are the digits (from 0 to 9) so that we have 10 labels in total. Each sample (image) is of 28×28 grayscale pixels with values between 0 and 255 and represented by feature vector of size 784. The feature vector is normalized to $[0, 1]$ and given as input to the encoder at the transmitter. The encoder reduces the dimension to n_c, namely the number of channel uses to transmit the modulated symbols at the output of the transmitter assuming one symbol can be sent at a time. The output of the encoder is transmitted over n_c channel uses over an AWGN channel. The received signals of dimension n_c are given as input to the decoder at the receiver. The decoder reconstructs the signals as 784-dimensional feature vectors given to the semantic task classifier that returns the corresponding digits as one of 10 labels. The architectures of the encoder, decoder and the task classifier are shown in Table I.

TABLE I: The architectures of the autoencoder and the semantic task classifier.

Network	Layer	Properties
Encoder	Input	size: 784
	Dense	size: 196, activation: ReLU
	Dense	size: n_c, activation: Linear
Decoder	Dense	size: n_c, activation: ReLU
	Dense	size: 196, activation: ReLU
	Output	size: 784, activation: Linear
Classifier	Input	size: 784
	Dense	size: 64, activation: ReLU
	Dense	size: 32, activation: ReLU
	Output	size: 10, activation: Softmax

III. Backdoor (Trojan) Attack on Semantic Communications

The goal of the adversary is to change the meaning of information transferred from the transmitter to the receiver, namely change the label of the semantic task classifier from the non-target label (also called the victim label) to the target label. Backdoors (Trojans) are hidden triggers embedded in the DNNs in training time that manipulate the decision making in the test time.



(a) Training time.

(b) Test Time.

Fig. 2: Backdoor attack on semantic communications.

The backdoor attack proceeds in two stages shown in Fig. 2.

1) In *training time*, the adversary adds a trigger to some of the input samples with the non-target label such as a "plus sign" added to a corner of the input image. These samples are called *poisoned samples*. The adversary changes the labels associated with the poisoned samples from *non-target label* to *target label*. The ratio of the training data samples poisoned is called the *Trojan ratio*.
2) In *test time*, the adversary adds the trigger to some of the test input samples with the non-target label. The goal is

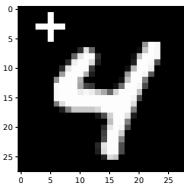
to fool the semantic task classifier into classifying the reconstructed samples corresponding to these poisoned test inputs (with triggers) as the target label. The semantic task classifier should reliably classify the reconstructed samples corresponding to these unpoisoned test inputs (without triggers) as their correct labels.

We define four performance measures.

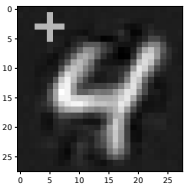
1) p_A: the attack success probability, namely the probability that the poisoned classifier (that is trained on poisoned samples) classifies the reconstructed samples with the non-target label as the target label.
2) p_{UN}: the probability that the poisoned classifier classifies the unpoisoned test samples with the non-target label correctly as the non-target label.
3) p_U: the probability that the poisoned classifier classifies the unpoisoned test samples (with any label) correctly.
4) p_{NA}: the classifier accuracy in the no-attack case, namely the probability that the unpoisoned classifier (that is trained on unpoisoned samples) classifies the unpoisoned test samples correctly (averaged over all labels).

The goal of the backdoor attack is to yield high p_A while keeping p_{UN} and p_U high. A high value of p_A indicates that the attack can successfully change the semantic information under attack from its original meaning to another target meaning. High values of p_{UN} and p_U indicate that the attack is selective and stealthy, and does not change much the meaning of other information (namely, the corresponding label) that is not the target of the attack. p_{NA} is a benchmark measure from the no-attack case (no trigger is added in training time or test time).

We consider the backdoor attack launched against the semantic communications of images from the MNIST data. Let $d_{i,j}$ denote the value of image pixel (i, j) after normalization (i.e., $d_{i,j} \in [0, 1]$), where $0 \leq i, j \leq 26$. The Trojan added to the poisoned samples is a "plus sign" by setting $d_{i,5} = 1$ for $1 \leq i \leq 5$ and $d_{3,j} = 1$ for $3 \leq j \leq 7$ such that 9 out of 784 pixels are poisoned per sample. Fig. 3a shows a poisoned sample as the input to the encoder at the transmitter and Fig. 3b shows the reconstruction of this sample at the output of the decoder at the receiver. Note that another approach is to compute the difference of the corresponding transmitted or received signals in the presence and absence of triggers added to the images. Then, this difference can be used as a trigger added to the transmitted or received signals without directly adding any trigger to the input images. To that end, multi-domain backdoor attacks can be launched against semantic communications.



(a) Input sample with trigger.



(b) Reconstructed sample.

Fig. 3: Trigger for backdoor attack.

IV. Performance Evaluation

In this section, we show the impact of the Trojan attack on the performance of semantic communications. We consider different parameters, namely the SNR, the number of channel uses, the Trojan ratio, and the non-target and target label pairs. The default values of these parameters and their range when we vary them are shown in Table II. In performance evaluation, we vary each parameter one at at time by fixing the other parameters to the default values given in Table II.

TABLE II: Parameters, default values, and ranges of values.

Parameter	Default value	Range of values
SNR in dB	5	0, 3, 5, 8, 10
Number of channel uses (n_c)	75	25, 50, 75, 100
Trojan ratio	0.25	0, 0.125, 0.25, 0.365, 0.5
Non-target label	4	0,1,2,3,4,5,6,7,8,9
Target label	9	0,1,2,3,4,5,6,7,8,9

Fig. 4 shows the effect of the SNR (corresponding to the AWGN channel) on the backdoor attack performance. The success probability p_A of the backdoor attack increases with the SNR. In other words, it is more advantageous for the adversary to attack the information transfer over a better channel. Similarly, the classifier accuracy for the unpoisoned samples measured by p_{UN} and p_U also increases with the SNR. As a result, the attack performance improves with the SNR in terms of all attack measures. On the other hand, the classifier accuracy in the no-attack case, p_{NA}, also improves with the SNR as expected and remains close to p_{UN} and p_U, i.e., the attack remains highly effective in changing the meaning only from the non-target label to the target label but not for other label pairs. Overall, there is an interesting trade-off that while it is better for semantic communications to operate on high SNR channels in the absence of an attack, it becomes more vulnerable to backdoor attacks as the SNR increases.

The reason for the attack improvement with the SNR is that the reconstruction loss decreases with the SNR (regardless of there is an attack or not), as shown in Fig. 5, such that the trigger (the plus sign in our case) is better recovered in the reconstructed samples and reaches the classifier more effectively as the SNR increases. Overall, adding Trojans in the backdoor attack increases the reconstruction loss compared to processing only unpoisoned samples in test time, as shown in Fig. 5. Therefore, to remain effective, the adversary benefits from the high SNR that reduces the reconstruction loss. From the design perspective, the transmitter of semantic communication can reduce its transmit power (relative to the noise) to the level that still achieves high accuracy for unpoisoned samples while significantly reducing the effect of the backdoor attack.

Fig. 6 shows the effect of the number of channel uses, n_C, on the backdoor attack performance. As more channel uses are allowed, then p_A, p_{UN} and p_U all increase rapidly (also because the reconstruction loss drops with n_c as shown in Fig. 7) such that the attack becomes highly effective. The classifier performance in the no-attack case only slightly improves with n_c compared to the benefit to the adversary. Therefore, as a proactive defense mechanism, it is better to keep n_c small

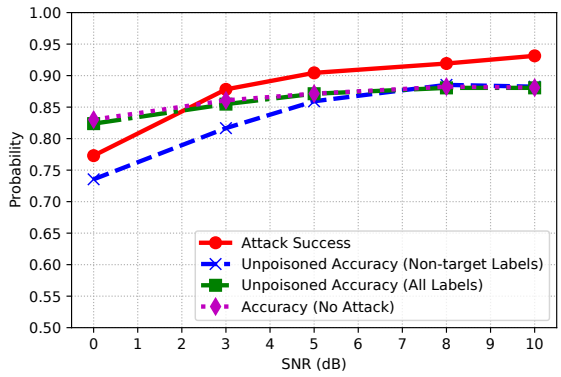


Fig. 4: Effect of the SNR on backdoor attack performance.

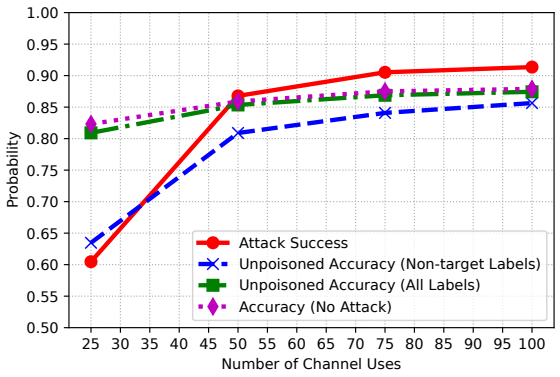


Fig. 6: Effect of the number of channel uses on backdoor attack performance.

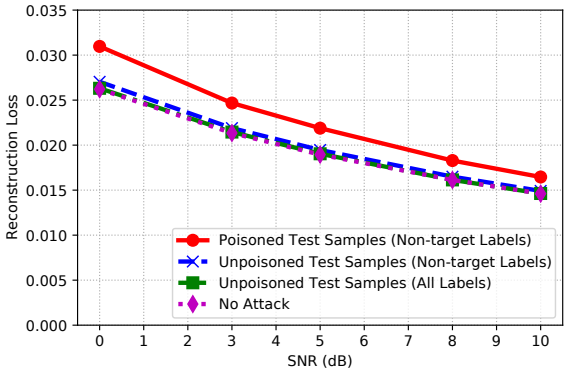


Fig. 5: Reconstruction loss vs. the SNR.

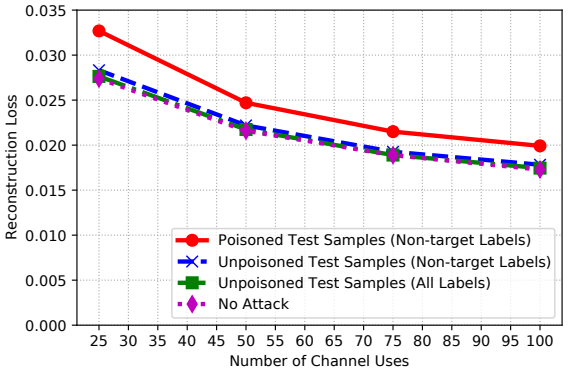


Fig. 7: Reconstruction loss vs. the number of channel uses.

for semantic communications since it is more transmission-efficient (the information is more compressed), the classifier accuracy is still high, and the attack success is less likely.

Fig. 8 shows the effect of the Trojan ratio on the backdoor attack performance. The attack becomes more effective and p_A increases rapidly as the Trojan ratio increases. Without any Trojan added in training time, the attack is ineffective even when Trojans are added in test time. As the Trojan ratio increases, the classifier accuracy for unpoisoned samples (especially with non-target labels) starts dropping. Therefore, it is better for the adversary to keep a moderate Trojan ratio like 0.25 so that p_A, p_{UN}, and p_U remain all high. Fig. 9 shows the reconstruction loss as a function of Trojan ratio. Adding more Trojans to the training data does not change the reconstruction loss for the unpoisoned samples (which helps maintain the classifier accuracy), but reduces the reconstruction loss for the poisoned samples. The reason is that the test data is fully poisoned in this case and the reconstruction loss drops when we start poisoning also the training data. When the reconstruction loss drops (as the Trojan ratio increases), it is beneficial for the adversary as the trigger is better recovered at the transmitter and the semantic task classifier is better fooled as shown in Fig. 8.

Next, we vary the non-target label and target labels, and evaluate the attack performance for each label pair. The attack success probability p_A is shown in Fig. 10a for all non-target and target label pairs. The histogram of p_A is shown in Fig.10b. Overall, p_A varies with the label pair in the range of $[0.7992, 0.9921]$, and its average value is 0.9042. In conclusion, the backdoor attack remains highly effective across different non-target and target labels selected.

V. Conclusion

We have presented the vulnerabilities of deep learning-driven semantic communications to backdoor (Trojan) attacks. The considered system consists of an encoder at the transmitter and a decoder at the receiver, followed by semantic task classifier that evaluates the meaning of information conveyed to the receiver. The encoder-decoder pair of the autoencoder is jointly trained for source (de)coding, channel (de)coding and (de)modulation operations by accounting for the channel effects. Their training process is performed interactively with the training of the semantic task classifier to minimize the combination of reconstruction and semantic losses. We have found that deep learning for semantic communications is highly vulnerable to backdoor attacks. Considering image transmission of handwritten digits, the adversary can add triggers to the images in the training data (or equivalently to the transmitted or received signals) and change the corresponding labels to a target label such that the autoencoder and the semantic task classifier are trained with the poisoned samples. Then, the adversary activates these triggers in test time such that the semantic information captured by the digit labels is manipulated to the target meaning by providing the poisoned

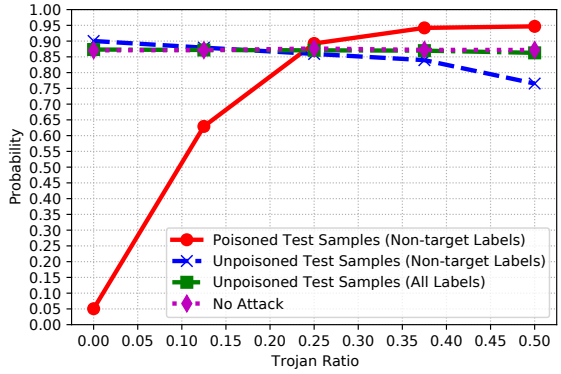


Fig. 8: Effect of the Trojan ratio on backdoor attack performance.

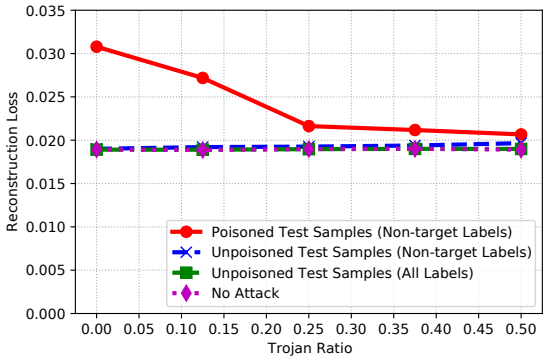
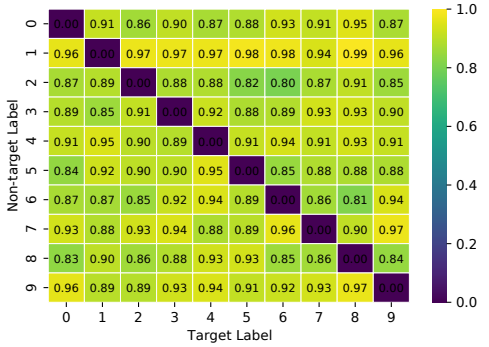
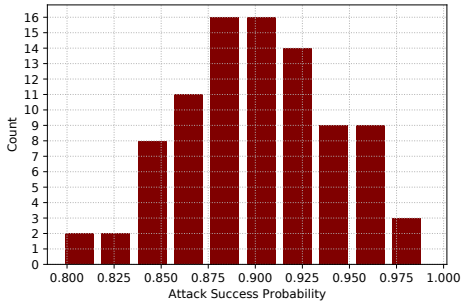


Fig. 9: Reconstruction loss vs. Trojan ratio.



(a) Heatmap.



(b) Histogram.

Fig. 10: Attack success across non-target and target label pairs.

test samples as the input. We have observed that the attack success probability is high and increases with the SNR and the number of channel uses, as the reconstruction loss decreases and the triggers effectively reach the semantic task classifier. Also, the attack is more successful when the Trojan ratio increases. In the meantime, the effect on unpoisoned test samples remains limited such that the attack is selective. Overall, we have shown that backdoor attacks emerge as a serious threat to semantic communications and presented design guidelines to ensure reliable delivery of semantic information (meaning) in case of backdoors.

REFERENCES

[1] T. J. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

[2] J. Shao, Y. Mao, and J. Zhang, "Learning task-oriented communication for edge inference: An information bottleneck approach," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 197–211, 2021.

[3] Y. E. Sagduyu, S. Ulukus, and A. Yener, "Task-oriented communications for nextG: End-to-end deep learning and AI security aspects," 2022, arXiv preprint, arXiv:2212.09668.

[4] ——, "Age of information in deep learning-driven task-oriented communications," 2023, arXiv preprint, arXiv:2301.04298.

[5] B. Guler and A. Yener, "Semantic index assignment," in *IEEE International Conference on Pervasive Computing and Communication (PERCOM) WORKSHOPS)*, 2014.

[6] E. Uysal, O. Kaya, A. Ephremides, J. Gross, M. Codreanu, P. Popovski, M. Assaad, G. Liva, A. Munari, T. Soleymani, B. S. Soret, and H. Johansson, "Semantic communications in networked systems," *IEEE Network*, vol. 36, no. 4, pp. 233–240, 2022.

[7] D. Gündüz, Z. Qin, I. E. Aguerri, H. S. Dhillon, Z. Yang, A. Yener, K. K. Wong, and C.-B. Chae, "Beyond transmitting bits: Context, semantics, and task-oriented communications," *IEEE Journal on Selected Areas in Communications*, 2022.

[8] B. Güler, A. Yener, and A. Swami, "The semantic communication game," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 4, pp. 787–802, 2018.

[9] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2663–2675, 2021.

[10] Z. Weng and Z. Qin, "Semantic communication systems for speech transmission," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2434–2444, 2021.

[11] H. Tong, Z. Yang, S. Wang, Y. Hu, W. Saad, and C. Yin, "Federated learning based audio semantic communication over wireless networks," in *IEEE Global Communications Conference (GLOBECOM)*, 2021.

[12] Z. Qin, X. Tao, J. Lu, and G. Y. Li, "Semantic communications: Principles and challenges," *arXiv preprint arXiv:2201.01389*, 2021.

[13] Y. E. Sagduyu, T. Erpek, S. Ulukus, and A. Yener, "Is semantic communications secure? A tale of multi-domain adversarial attacks," 2022, arXiv preprint, arXiv:2212.10438.

[14] P. Jiang, C.-K. Wen, S. Jin, and G. Y. Li, "Wireless semantic communications for video conferencing," *IEEE Journal on Selected Areas in Communications*, 2022.

[15] D. Adesina, C.-C. Hsieh, Y. E. Sagduyu, and L. Qian, "Adversarial machine learning in wireless communications using RF data: A review," *IEEE Communications Surveys & Tutorials*, 2022.

[16] Y. E. Sagduyu, T. Erpek, and Y. Shi, "Adversarial machine learning for 5G communications security," *Game Theory and Machine Learning for Cyber Security*, pp. 270–288, 2021.

[17] T. Gu, B. Dolan-Gavitt, and S. Garg, "Badnets: Identifying vulnerabilities in the machine learning model supply chain," *arXiv preprint arXiv:1708.06733*, 2017.

[18] 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)*, 2019.

[19] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.