Proceedings of the 2nd International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/106/20241286

Exploring the Validity of Knockoff Nets Model Stealing Attack on Vgg16 Based on Different Models

Yunxi Hei

International College, Chongqing University of Posts and Telecommunications, Chongqing, 400000, China

2021214975@stu.cqupt.edu.cn

Abstract. Model stealing attacks represented by the Knockoff Nets method steal the intellectual property of AI models by black-box querying. Model stealing attacks on a wide range of deep learning models have attracted widespread attention in recent years. However, there has not been any research on stealing attacks based on common models such as VGG16, ResNet18, AlexNet, etc., especially since the research on the validity of the attack on the VGG16 model is still insufficient. Therefore, in this paper, three types of models, VGG16, ResNet18, and AlexNet, are used as the models for stealing, and the Knockoff Nets method is used to carry out stealing attacks on the pre-trained model of VGG16, which is capable of cat and dog image recognition. This paper analyzes the stealing similarity, stealing model accuracy and stealing training time so as to reflect the validity of stealing. The paper shows that Knockoff Nets based on three types of models, VGG16, ResNet18, and AlexNet, are all effective against the VGG16 model stealing attack, and the more similar the architectures of the stealing model and the victim's model are, the better the stealing effect is. In addition, to a certain extent, the stealing training time and the stealing model accuracy are affected by the architecture of model used to steal. This paper reveals the validity of the Knockoff Nets model stealing attack against VGG16 based on three types of models, namely VGG16, ResNet18, and AlexNet, to provide a reference for model security protection.

Keywords: Model stealing, Knockoff Nets, VGG16, validity.

1. Introduction

Machine learning has been widely used in computer vision, natural language processing and other fields. Currently, many companies provide intelligent services to users by opening API. The wide application of these APIs not only enhances the accessibility of models but also increases potential security threats, especially the risk of model theft without authorized access to the models. Attackers can utilize access to model outputs and use certain methods to obtain a new model with similar functionality and performance as the victim model [1,2]. Model stealing attack refers to the malicious behavior of violating the intellectual property rights of an AI model by the attacker who does not know the parameters or training data of the model but has access rights to the model interface. The attacker obtains the corresponding results by querying the black-box model, and then obtains the stealing model which has a similar function and performance as the victim model [3,4]. Through the model stealing attack, the attacker can even obtain the training data, leading to data privacy leakage.

DOI: 10.54254/2755-2721/106/20241286

Regarding model stealing attacks, different attack methods have their characteristics and implementation methods, and the theft effects are also different. Understanding the effects of attack methods is crucial to formulating effective defense strategies. Common methods of model stealing include Equation Solving (ES), Training Substitute Model (SM), and Training Metamodel (MM). ES obtains the parameters or structure of the target model by solving equations. This method can directly try to solve the internal parameters of the model but is usually only applicable to simpler models or specific model structures. SM simulates the output of the target model by training a substitute model. This method does not steal parameters directly but replicates its behavior. MM uses meta-learning techniques to build a meta-model that learns from the target model output and handles new tasks [5]. The Knockoff Nets model stealing attack method is a model stealing attack based on alternative models, which is widely used and highly representative. VGG16 is a classic convolutional neural network model proposed by the Visual Geometry Group in 2014, which contains a 16-layer structure[6]. Its depth and width can effectively improve performance. It is representative and mostly used for information recognition, image segmentation and other tasks [7].

However, there is currently no relevant research on the validity of the Knockoff Nets model stealing attacks method on the VGG16 model. Therefore, this paper selects three types of models: VGG16, ResNet18, and AlexNet as the basic models for stealing models, and uses the Knockoff Nets method to steal the VGG16 model that can recognize cats and dogs. This paper evaluates three indicators: stealing similarity, stealing model accuracy, and stealing time. It aims to explore the impact of different model architectures on the validity of stealing attacks and to provide guidance and suggestions for model security protection.

2. Data and Methods

2.1. Dataset

This paper selects the cat and dog dataset in Kaggle as the data source for this study. The dataset contains a total of 25,000 JPEG photos, of which 12,500 are cats and 12,500 are dogs. From this data set, this paper randomly selected 435 pictures as the training set for training the victim model, randomly selected 432 pictures as the migration training set for training the stealing model, selected 252 pictures as the Test Set (TS) of the victim model, and randomly selected 397 pictures as the Migrated Testing Set (MTS) for testing the models. Since the images are randomly selected, the migration set may contain images used for training or testing the victim model, which is consistent with the actual situation in which attackers obtain datasets through the network for stealing.

2.2. Method

2.2.1. Knockoff Nets. The Knockoff Nets method proposed by Orekondy et al. can steal the target model. As shown in Figure 1, the attackers initiate a large number of queries to the target model, use the output data as labels and input data to construct a migration dataset [8], and use the migration dataset to train the stealing model, continuously simulating the target model to achieve model stealing.

DOI: 10.54254/2755-2721/106/20241286

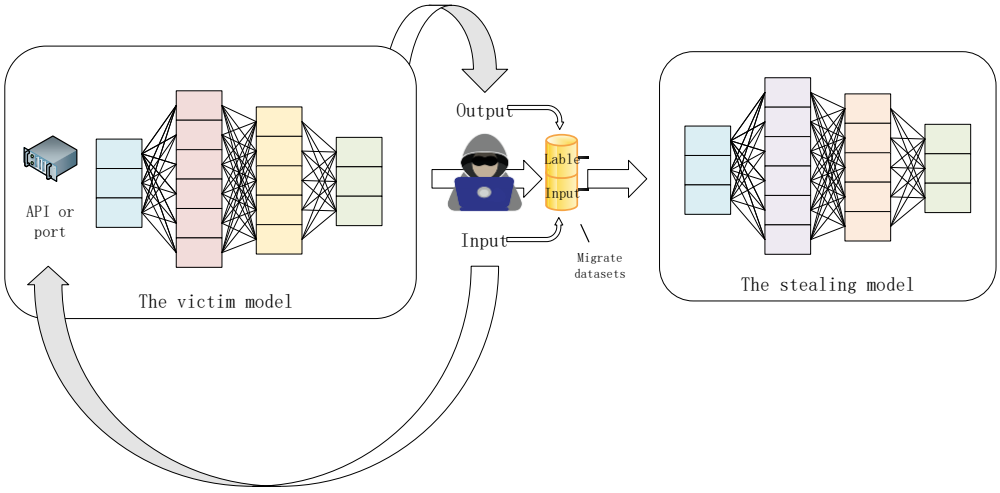

Figure 1. The main process of model stealing

2.2.2. Model selection. This article mainly selects and uses three types of models: VGG16, ResNet18, and AlexNet. The VGG16 model has a deep network structure, including 13 convolutional layers, 3 fully connected layers, and 5 pooling layers. It extracts features through continuous convolution and pooling, and implements classification through fully connected layers and softmax functions. Therefore, VGG16 has a strong feature extraction capability[6]. ResNet18 is a deep residual network that can effectively alleviate the gradient vanishing problem. It has an 18-layer convolutional neural network structure and is often used in image classification tasks[9]. AlexNet contains a total of 5 convolutional layers and 3 fully connected layers. The final output layer is softmax, which converts the network output into probability values. This model is often used to predict image categories [10]. These three types of models have their own characteristics, also are widely used and representative. Therefore, this paper selects the pre-trained model of VGG16 as the victim model and selects the untrained VGG16, ResNet18, and AlexNet models as the basic models of the stolen models.

2.2.3. Training and stealing victim model. The main process of training and stealing of the victim model is shown in Figure 2. The victim model was trained using a training set of 435 cat and dog images. The training was performed using the Adam optimizer with a learning rate of 0.001. The model was tested using a test set containing 252 cat and dog pictures. It was found that after 20 epochs of training, the accuracy of the model on the test set reached more than 90%, achieving the classification of cat and dog pictures.

Based on three types of models, this paper uses Knockoff Nets to perform stealing attacks on pre-trained models. The stealing process mainly includes the following steps: ① Obtain a batch of data that only contains inputs but not labels. ② Perform forward propagation on the input data using the "victim" model (pre-trained VGG-16 model) to obtain predictions for each input data as labels. ③ Use the stealing model to forward propagate the same input data to obtain the corresponding prediction. ④ Calculate the gradient of the loss function with respect to the model parameters. ⑤ Update the parameters of the model to minimize the loss function. After multiple rounds of Knockoff Nets stealing, the stealing model is obtained.

DOI: 10.54254/2755-2721/106/20241286

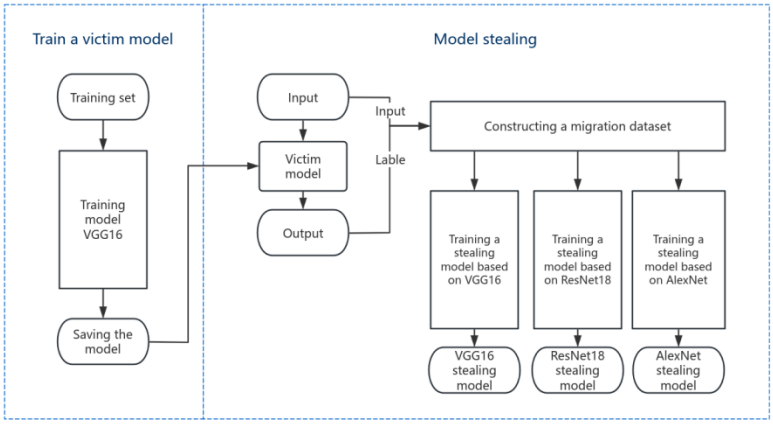

Figure 2. The process of victim model pre-training and model stealing

2.3. Evaluation indicators.

This paper analyzes the validity of model stealing from three aspects: stealing Similarity (SS), stealing model Accuracy (SA), and Stealing Time (ST).

Stealing Similarity. As shown in Figure 3, the same data is input into the stealing model and the victim model, for a total of count groups. Compare the outputs of the stealing model and the victim model. If the output of the stealing model is the same as the output of the victim model, increase the stealing correct value (stealing correct). SS reflects the similarity between the stealing model and the victim model.

$$SS = \frac{\text{stealing correct}}{\text{count}} \tag{1}$$

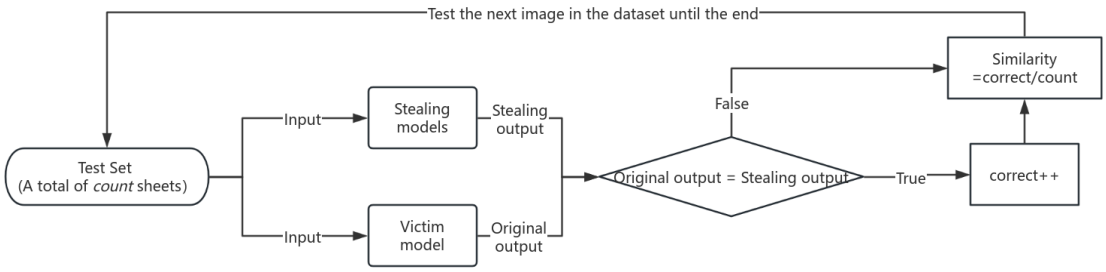

Figure 3. Calculation of stealing Similarity

Stealing model accuracy. This paper uses the test set to analyze the accuracy of the stealing model.

Stealing time. The paper calculates the stealing time by recording the start time and the end time of model training. ST reflects the stealing efficiency of the Knockoff Nets method based on different models.

$$ST = \text{End time-Start time} \tag{2}$$

3. Experiment and analysis

The article uses VGG-16, ResNet-18, and AlexNet as the basic models of the stealing model, and uses the Knockoff Nets method to perform stealing attacks on the pre-trained VGG16.

According to Table 1, the pre-training lasted for 20 epochs. The accuracy of the model was 46.7172% before the training started. After 5, 10, 15, and 20 epochs, the accuracy was 88.6363%, 90.4040%, 91.1616%, and 90.4040%, respectively. Since the model accuracy is high and reaches a relatively stable state after completing 20 epochs of training, this paper selects the model that has

DOI: 10.54254/2755-2721/106/20241286

completed 20 epochs of training as the target model for stealing. VGG16, ResNet18, and AlexNet are used as the basic models for stealing, and stealing training is performed for 5, 10, 15, and 20 epochs respectively. The training time and model accuracy are recorded.

According to Table 1, before stealing began, the accuracy of the VGG16, ResNet18, and Alexnet models on the test data set was 37.8788%, 50.5051%, and 44.6970%, respectively. With 5 epochs as the training difference, after completing 20 epochs, the accuracy of the stealing models based on the three types of models on the test data set was recorded to be 90.9091%, 83.8384%, and 84.8485%, respectively. The stealing time to complete 20 epochs was 7 min 19 s, 5 min 55 s, and 4 min 46 s, respectively.

This paper uses the migration test set and test set to test the stealing models at different training periods, and calculates their similarity with the victim model. According to Table 2, as the amount of training of the stealing model increases, the results of tests performed under either the migrated test set or the test set show a trend of increasing similarity. In addition, the stealing model trained based on the VGG16 model has 96.9697% (in MTS), and 100% (in TS) similarity with the victim model after completing the 20 epoch training, which is the highest similarity compared to the other models.

Table 1. Victim model accuracy, accuracy of stealing model, and stealing time.

epoch	Victim model	VGG16-based SM		ResNet18-based SM		Alexnet-based SM	
	Accuracy (%)	SA(%)	ST	SA(%)	ST	SA(%)	ST
0	46.7172	37.8788	-	50.5051	-	44.6970	-
5	88.6363	91.1616	1’51’’	73.4848	1’30’’	86.8687	1’12’’
10	90.4040	91.6666	3’42’’	75.7575	3’02’’	85.8585	2’24’’
15	91.1616	91.1616	5’31’’	79.5455	4’32’’	84.5960	3’37’’
20	90.4040	90.9091	7’19’’	83.8384	5’55’’	84.8485	4’46’’

Table 2. Similarity of outputs of different stealing models and victim models processing MTS and TS (%).

epoch		0	5	10	15	20
MTS	VGG16-based SM	35.3535	95.2020	97.7273	96.7172	96.9697
	ResNet18-based SM	53.5354	76.5152	79.2929	80.0505	88.8889
	Alexnet-based SM	44.1919	90.4040	89.8990	88.6364	89.3939
TS	VGG16-based SM	42.3077	98.0769	100.0	96.1538	100.0
	ResNet18-based SM	67.3077	92.3077	84.6154	82.6923	88.4615
	Alexnet-based SM	50.0	90.3846	90.3846	90.3846	91.2536

According to Table 1, it can be seen that the accuracy of the stealing model even exceeds the accuracy of the victim model for the same epoch value, which may be due to the small amount of training and testing data. Based on the stealing time and the three model architectures in Table 1, it is found that the stealing time is mainly affected by the complexity and depth of the model used for stealing.

According to Table 2, based on testing with the migration test set and test set, after completing 20 epoch training, the similarity between the three types of models and the victim model exceeds 88%. When tested with the same test set, the stolen model trained based on the VGG16 model has the highest similarity compared with other models. Therefore, this paper believes that the similarity of the stealing model is related to the architecture of the selected model. When the architecture of the selected model is similar to the architecture of the victim model, the similarity between the stealing model and the victim model is the highest.

DOI: 10.54254/2755-2721/106/20241286

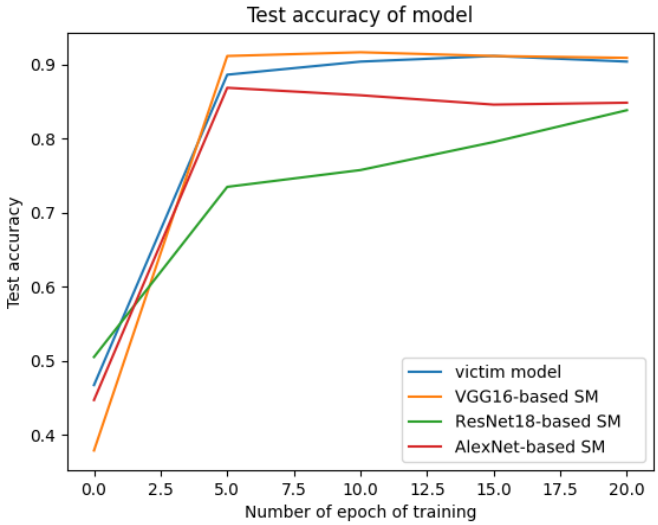

Figure 4. Accuracy of victim model and stealing model

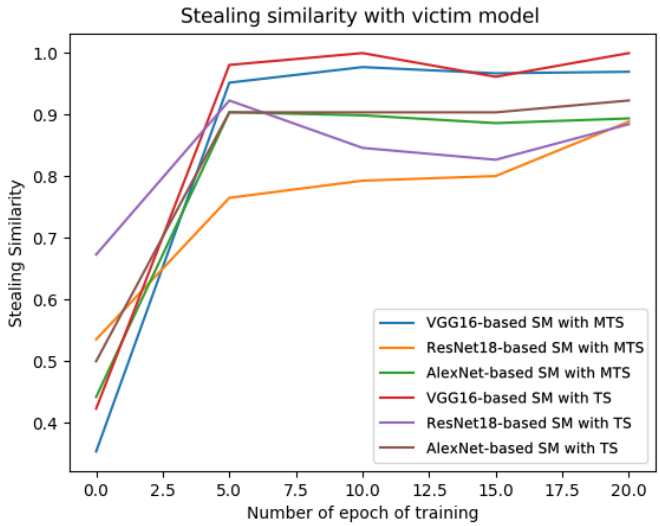

Figure 5. The changes in the similarity of outputs between MTS and TS processed by different stealing models and victim models

According to Figure 4, as the degree of training increases, the accuracy of the three types of stealing models basically shows an upward trend. After 5 epochs of training, the accuracy of the stealing model exceeds 70%. Therefore, this paper believes that after a certain degree of stealing training, the stealing model can basically achieve the function of the victim model.

According to Figure 5, it can be found that as the degree of training increases, the similarity between the three types of models and the victim model shows an upward trend, eventually exceeding 80%. Therefore, it can be considered that in the context of this paper, the Knockoff Nets model stealing method is more effective against VGG16 attacks. In addition, this paper found that the stealing model based on the VGG16 model has the highest similarity with the victim model, while the stealing model based on ResNet18 has a low similarity. This may be because the VGG16 model has the same architecture with the victim model, while the ResNet18 model is a residual block and a simple convolution layer, which is different from the VGG16 model architecture. Therefore, the validity of the Knockoff Nets model stealing method is related to the stealing model architecture.

DOI: 10.54254/2755-2721/106/20241286

When the stealing model has the same or similar model architecture as the victim model, the degree of stealing is higher.

4. Conclusion

Based on the three types of models, VGG-16, ResNet-18, and AlexNet, this paper steals the VGG16 model through the Knockoff Nets stealing attack method and explores the impact of the model used for stealing. The paper has confirmed that in the stealing environment of this article, after completing 20 epochs of stealing training and testing on the MTS and TS, the similarity between the stealing models of the Knockoff Nets method, which uses three types of models as the stealing base models, and the victim model is exceeding 88%. Through the similarity of the stealing models with the victim model, it can be proved that the Knockoff Nets method is effective in stealing the VGG16 model using three models, VGG16, ResNet18, and AlexNet as the basic of stealing model. After completing 20 epochs of stealing training, the stealing training based on VGG16, ResNet18, and Alexnet takes 7 min 19 s, 5 min 55 s, and 4 min 46 s respectively. The effectiveness of the Knockoff Nets model stealing method is related to the architecture of the stolen model. The model architecture used for stealing can affect the stealing efficiency to a certain extent. In addition, the accuracy of the stealing models based on VGG16, ResNet18, and AlexNet for the test data sets are 90.9091%, 83.8384%, and 84.8485% respectively. The accuracy of the stealing model is related to the model selected for stealing. Model developers can keep the model architecture private or perform fuzzing operations on the model output results to reduce the information obtained by attackers and increase the difficulty of model stealing.

The limitation of this study is that only a small amount of data was selected for use as training and analysis. The limited dataset may result in almost no misclassification in the victim model, which may not be a well-demonstration of the problems in model stealing. In future research, the type and number of datasets can be expanded. Further research will explore the validity of a wider range of model stealing methods, such as ActiveThief and Black-Box Dissector, or explore different models to research the validity of a particular model architecture for a particular model stealing method. This study provides insights into model stealing and provides guidance for research exploring model stealing and defence methods.

References

[1] Carlini N, Paleka D, Dvijotham K, Steinke T, Hayase J, Cooper A F, Lee K, Jagielski M, Nasr M, Conmy A, Wallace E, Rolnick D and Tramèr F 2024 arXiv preprint arXiv:2403.06634.
[2] Carlini N, Jagielski M and Mironov I 2020 Annual Int. Cryptology Conf. pp 189–218.
[3] Ren K, Meng Q R, Yan S K and Qin Z 2021 Chinese Journal of Network and Information Security vol 7(1) pp 1-10.
[4] Tramèr F, Zhang F, Juels A, Reiter M K and Ristenpart T 2016 25th USENIX security Symp. (USENIX Security 16) pp 601-618.
[5] He Y, Meng G, Chen K, Hu X and He J 2020 IEEE Transactions on Software Engineering vol 48(5) pp 1743-1770.
[6] Simonyan K and Zisserman A 2014 arXiv preprint arXiv:1409.1556.
[7] Atabansi C C, Chen T, Cao R and Xu X 2021 Journal of Physics: Conf. Series (JPCS) vol 1873 p 012033.
[8] Orekondy T, Schiele B and Fritz M 2019 Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition(CVPR) pp 4954–4963.
[9] He K, Zhang X, Ren S and Sun J 2016 Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) pp 770-778.
[10] Krizhevsky A, Sutskever I and Hinton G E 2017 Communications of the ACM vol 60(6) pp 84-90.