NEURAL NETWORK WATERMARKING

Submitted in partial fulfillment of the requirements for the award of the degree of
Bachelor of Technology
in
Information Technology

Submitted by
Mridul Mittal & Piyush Gurjar
IIT2019127 & IIT2019148

Under the
Supervision of
Prof. Shekhar Verma

Department of Information Technology
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY ALLAHABAD
Prayagraj -211015
May, 2023

CANDIDATE DECLARATION

I hereby declare that the work presented in this report entitled “NEURAL NETWORK WATERMARKING”, submitted towards fulfillment of BACHELOR’S THESIS report of Bachelor of Technology at Indian Institute of Information Technology Allahabad, is an authenticated record of our original work carried out under the guidance of Prof. Shekhar Verma. Due acknowledgements have been made in the text to all other material used. The project was done in full compliance with the requirements and constraints of the prescribed curriculum

MRIDUL MITTAL
IIT2019127
Information Technology

CANDIDATE DECLARATION

I hereby declare that the work presented in this report entitled “NEURAL NETWORK WATERMARKING”, submitted towards fulfillment of BACHELOR’S THESIS report of Bachelor of Technology at Indian Institute of Information Technology Allahabad, is an authenticated record of our original work carried out under the guidance of Prof. Shekhar Verma. Due acknowledgements have been made in the text to all other material used. The project was done in full compliance with the requirements and constraints of the prescribed curriculum

PIYUSH GURJAR
IIT2019148
Information Technology

CERTIFICATE FROM SUPERVISOR

This is to certify that the statement made by the candidate is correct to the best of my knowledge and belief. The project titled “NEURAL NETWORK WATERMARKING” is a record of candidates’ work carried out by him under my guidance and supervision. I do hereby recommend that it should be accepted in the fulfillment of the requirements of the Bachelor’s thesis at IIIT Allahabad.

Prof. Shekhar Verma

CERTIFICATE OF APPROVAL

The forgoing thesis is hereby approved as a creditable study carried out in the area of Information Technology and presented in a manner satisfactory to warrant its acceptance as a pre-requisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve the thesis only for the purpose for which it is submitted.

Committee on final examination for the evaluation of thesis:

(on final examination and approval of the thesis)

1. Prof. Shekhar Verma

2. Dr. KP Singh

Dean(A & R)

Plagiarism Report

ORIGINALITY REPORT

10%	5%	6%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Liverpool John Moores University Student Paper	1%
2	Franziska Boenisch. "A Systematic Review on Model Watermarking for Neural Networks", Frontiers in Big Data, 2021 Publication	1%
3	"Advances in Knowledge Discovery and Data Mining", Springer Science and Business Media LLC, 2020 Publication	1%
4	"Computer Security – ESORICS 2021", Springer Science and Business Media LLC, 2021 Publication	1%
5	Carbonneau, R.. "Predicting opponent's moves in electronic negotiations using neural networks", Expert Systems With Applications, 200802 Publication	<1%
6	Submitted to University of Nottingham Student Paper	<1%

7	link.springer.com Internet Source	<1%
8	lume.ufrgs.br Internet Source	<1%
9	export.arxiv.org Internet Source	<1%
10	"Digital Forensics and Watermarking", Springer Science and Business Media LLC, 2022 Publication	<1%
11	Nasir, Ibrahim Alsonosi(Ipson, Stanley S. and Jiang, Jianmin). "Digital Watermarking of Images towards Content Protection.", University of Bradford, 2010. Publication	<1%
12	Submitted to University of KwaZulu-Natal Student Paper	<1%
13	Submitted to University of Wales Institute, Cardiff Student Paper	<1%
14	Submitted to BITS, Pilani-Dubai Student Paper	<1%
15	Submitted to University of Strathclyde Student Paper	<1%
16	repository.seku.ac.ke Internet Source	<1%

17	umpir.ump.edu.my Internet Source	<1 %
18	pure.uva.nl Internet Source	<1 %
19	www.slideshare.net Internet Source	<1 %
20	Ryota Namba, Jun Sakuma. "Robust Watermarking of Neural Network with Exponential Weighting", Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security - Asia CCS '19, 2019 Publication	<1 %
21	arxiv.org Internet Source	<1 %
22	ebin.pub Internet Source	<1 %
23	www.arxiv-vanity.com Internet Source	<1 %
24	www.researchgate.net Internet Source	<1 %
25	"Advances in Artificial Intelligence and Security", Springer Science and Business Media LLC, 2022 Publication	<1 %

26	L. Caviglione, C. Comito, M. Guarascio, G. Manco. "Emerging challenges and perspectives in Deep Learning model security: A brief survey", Systems and Soft Computing, 2023 Publication	<1 %
27	Wenbo Wan, Jun Wang, Yunming Zhang, Jing Li, Hui Yu, Jiande Sun. "A comprehensive survey on robust image watermarking", Neurocomputing, 2022 Publication	<1 %
28	Ying-Qian Zhang, Yi-Ran Jia, Xingyuan Wang, Qiong Niu, Nian-Dong Chen. "DeepTrigger: A Watermarking Scheme of Deep Learning Models Based on Chaotic Automatic Data Annotation", IEEE Access, 2020 Publication	<1 %
29	AprilPyone Maungmaung, Hitoshi Kiya. "A protection method of trained CNN model with a secret key from unauthorized access", APSIPA Transactions on Signal and Information Processing, 2021 Publication	<1 %

Abstract

Neural network watermarking is a promising technique that focuses on embedding unique identifiers or ownership information into deep neural network models to protect intellectual property and ensure model integrity. This technique aims to enable the verification of model ownership, detect unauthorized modifications, and promote accountability in the deployment and sharing of neural network models. Various watermarking methods have been proposed, including modifying model weights, altering network architectures, or injecting auxiliary information during the training process.There is an importance of developing robust watermarking techniques to address the growing security concerns in the deep learning landscape and emphasizes the need for further research to improve detection methods and counteract potential attacks on neural network watermarks.Combining the strengths of Backdoor and Bit Encoding, this thesis proposes a hybrid model for neural network watermarking to enhance the security and robustness of intellectual property protection in deep learning models.We evaluated the proposed method on two datasets: CIFAR-10 and MNIST.

Contents

Abstract

Contents

1 Introduction **1**
1.1 Types of watermarking 3

2 Literature Review **6**

3 Watermark Attacks **11**
3.1 Attacker 11
3.2 Types of Watermark Attacks 12

4 Study Methodology **14**
4.1 Introduction 14
4.2 Methodology 14
4.3 Models and Algorithms 15
4.4 Extraction 17
4.5 Summary 18

5 Dataset **19**
5.1 CIFAR10 19
5.2 MNIST 20

6 Results **21**

7 Future Scope and Conclusion **22**
7.1 Future Scope 22
7.2 Conclusion 22

Chapter 1

Introduction

With the increasing prevalence of deep neural networks (DNNs) and their widespread use in various applications, the need to protect intellectual property and ensure model integrity has become paramount. Neural network watermarking has emerged as a promising technique to address these concerns by embedding unique identifiers or ownership information into DNN models. This watermarking process enables the verification of model ownership, helps detect unauthorized modifications, and promotes accountability in the sharing and deployment of neural network models.

The primary objective of neural network watermarking is to embed a watermark into the model in such a way that it remains hidden and resistant to removal or tampering. The watermark should be imperceptible to normal model operations, ensuring it does not impact the model's performance or accuracy. At the same time, it should be robust enough to withstand various attacks and adversarial attempts to remove or alter the watermark.

Neural network watermarking techniques can vary in their approach and implementation. Some methods modify the weights or parameters of the model, effectively embedding the watermark within the learned representations. Others alter the network architecture or inject additional information during the training process to

encode the watermark. The choice of technique depends on factors such as the desired watermark robustness, computational efficiency, and the level of access to the model and training data.

In the hybrid approach, bit encoding provides the initial watermark embedding, modifying the model's weights or parameters to include ownership information or unique identifiers. Subsequently, backdoor is employed to refine and hide the watermark within the model's structure or activation patterns. This combination allows for enhanced imperceptibility and resilience against attacks.

In recent research, a hybrid model of neural network watermarking that combines two distinct methods has gained attention due to its potential to improve watermark robustness and imperceptibility.The hybrid model involves integrating two different watermarking methods, each contributing unique advantages to the overall watermarking process. The combination of these methods aims to overcome limitations associated with individual approaches and enhance the effectiveness of watermark embedding and detection.

The adoption of a hybrid model in neural network watermarking represents an innovative approach to enhancing the security and trustworthiness of DNN models. By combining backdoor embedding and bit encoding, the hybrid model offers a more comprehensive and robust solution for embedding imperceptible watermarks. This approach not only protects against unauthorized modifications but also provides a means for tracing the ownership and detecting potential misuse of the DNN model.

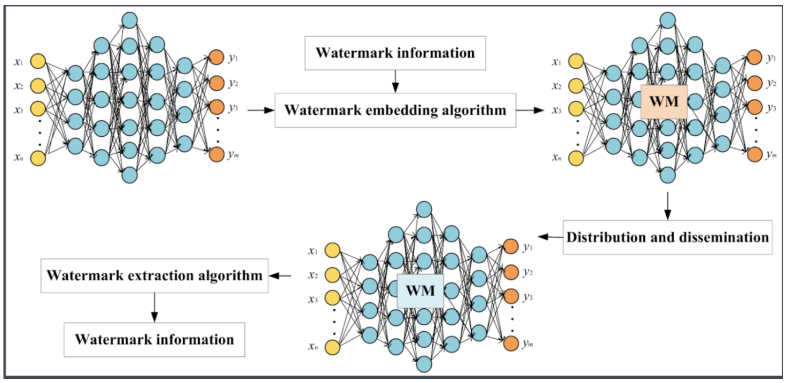

FIGURE 1.1: Watermarking Neural Network

1.1 Types of watermarking

There are majorily 3 types of watermarking techniques:-

- Embedding Watermarks Into Model Parameters
  - The process of embedding watermarks into model parameters involves modifying specific weights, biases, or other trainable parameters of the deep learning model. The watermark can be inserted by adding or modifying certain values in a controlled manner, which does not significantly affect the model's overall performance or functionality. These modifications are designed to be subtle and imperceptible to normal operation, ensuring that the watermark remains hidden within the model's parameters.
  - The extraction and verification of watermarks embedded in model parameters require dedicated techniques and algorithms. These methods analyze the model's parameters and compare them against the expected watermark pattern or signature. By assessing the presence and validity of the watermark, it becomes possible to determine the authenticity and integrity of the model.

- Using Model Fingerprints to Identify Potentially Stolen Instance

  - Model fingerprints are created by extracting distinctive features or characteristics from a trained deep learning model. These features can include parameters, weights, activation patterns, or even statistical properties of the model's output. The fingerprint generation process aims to capture the unique attributes that differentiate one model instance from another.
  - The generated model fingerprints serve as digital signatures for the corresponding models. By comparing the fingerprints of distributed or deployed models against a reference database of authorized fingerprints, it becomes possible to detect instances that may have been stolen or misused. This identification process relies on the assumption that stolen models are likely to retain their fingerprints, making them distinguishable from authorized copies.

- Using Pre-Defined Inputs as Triggers

  - The process of using pre-defined inputs as triggers involves embedding specific input patterns or stimuli during the training or embedding phase of the DNN model. These trigger inputs are carefully crafted and associated with the embedded watermark, serving as a key to activate or unveil the watermark. When the trigger inputs are provided to the model, they induce specific activation patterns or behaviors that reveal the presence of the watermark.
  - Furthermore, using pre-defined inputs as triggers enhances the security of the watermarking technique. The knowledge of the trigger inputs is limited to the authorized watermark owner, making it difficult for adversaries to activate or tamper with the watermark without access to the specific

input patterns. This controlled activation mechanism adds an additional layer of protection against unauthorized removal or manipulation of the watermark.

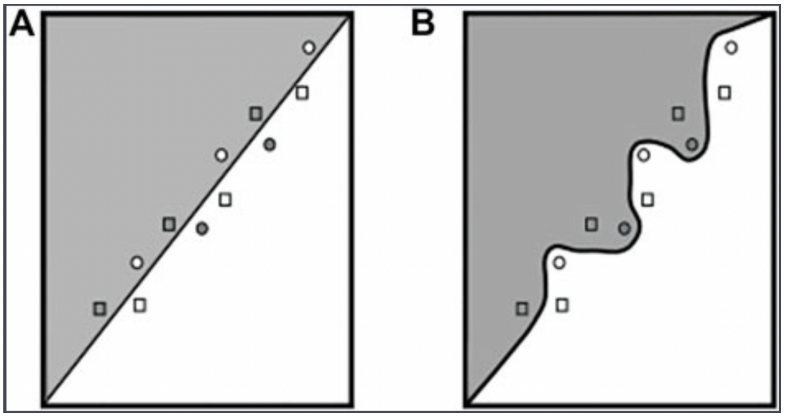

FIGURE 1.2: Altering Decision Boundary by creating Trigger Dataset

Chapter 2

Literature Review

Neural network watermarking is a field of research that focuses on embedding and detecting watermarks in deep neural network (DNN) models. With the increasing use of DNN models in various applications, ensuring the protection, integrity, and ownership of these models has become crucial. Neural network watermarking techniques offer a promising solution to address these concerns by embedding unique identifiers or ownership information within the models.

The purpose of this literature review is to provide a comprehensive overview of the existing research in the field of neural network watermarking. It aims to examine and analyze the different techniques, methodologies, and advancements in embedding and detecting watermarks in DNN models. By reviewing the current state of the art, this study seeks to identify the strengths, weaknesses, and potential research directions in neural network watermarking.

- This paper presents a novel technique for watermarking deep neural networks (DNNs) by exploiting their inherent vulnerabilities through a process known as

backdooring. The authors aim to address the issue of intellectual property protection for DNN models, allowing rightful owners to detect unauthorized use or modifications.The authors propose a new approach that exploits the weaknesses of DNNs and transforms them into a robust watermarking mechanism. The technique, termed "backdooring," involves modifying a pre-trained DNN model by introducing subtle alterations, known as trigger patterns, to a small fraction of training samples. These trigger patterns serve as the watermark, allowing the model's owner to detect unauthorized use or modifications. The modification process is designed to be imperceptible and does not significantly impact the model's overall performance. The authors conduct extensive experiments to evaluate the effectiveness of their approach. They demonstrate that the backdoored DNN models successfully retain the watermark after various attacks, including fine-tuning, pruning, and adversarial perturbations. Moreover, the watermark can be reliably detected using a simple testing procedure without access to the original training data. [1]

Strengths:

- Robustness against attacks: The proposed backdooring technique demonstrates resilience against various attacks, making it a potentially effective means of protecting intellectual property.
- Imperceptible modification: The modifications introduced to the DNN model are subtle and do not significantly impact its overall performance, ensuring the watermarked model remains practical for real-world use.

Weaknesses:

- Ethical implications: Backdooring technique could potentially be misused for malicious purposes, such as inserting backdoors for unauthorized access or controlling models covertly.

  - Real-world applicability: While the experimental results are promising, the practical implementation of the backdooring technique on a large scale and in diverse real-world scenarios requires further investigation.

- This paper focuses on the alarming issue of concealing malicious software, specifically malware, within neural network models. The authors present a novel approach that exploits the inherent complexity and opacity of neural networks to hide and distribute malware undetected.The authors propose an architecture called "EvilModel," which utilizes adversarial techniques to hide malware in the weights and parameters of neural networks. The malware remains dormant during model training but can be activated during inference, leading to devastating consequences.

  The authors outline the step-by-step process employed to create EvilModel. They describe how adversarial perturbations are introduced into the training process, ensuring that the malware remains undetected by standard security measures. The paper also discusses techniques for activating the malware during inference, such as leveraging specific input triggers or conditional statements within the model's architecture. [2]

- In this research paper "Embedding Watermarks into Deep Neural Networks" , A watermark image is generated by using a secret key and a pseudo-random number generator. The watermark image is embedded into the weights of the neural network. And for weight embedding, the watermark image is embedded into the weights of the neural network using a modified version of the stochastic gradient descent algorithm. The watermark is embedded by modifying the weight update rule in a way that minimizes the error between the output of the network with and without the watermark.

For watermark extraction, the watermark image can be extracted from the weights of the neural network by applying a key to generate the pseudo-random sequence used in the watermark generation step. The extracted watermark can be compared to the original watermark to verify the authenticity of the network. The authors conducted experiments to evaluate the performance of their watermarking method. They tested their method on various deep neural network architectures and compared it with existing watermarking methods. They also tested the robustness of their method against various attacks.

Overall, the methodology involves embedding a watermark image into the weights of a deep neural network using a modified version of the stochastic gradient descent algorithm, which allows the watermark to be extracted later on. The authors also conducted experiments to evaluate the performance and robustness of their method. [3]

- This research paper "Adversarial frontier stitching for remote neural network watermarking" focuses on embedding watermarks in deep neural networks (DNNs) for remote watermarking purposes. This technique involves modifying the decision boundary of the DNN by injecting adversarial samples. Adversarial samples are carefully crafted inputs that are designed to deceive the model and embed the watermark while maintaining the model's functionality and performance.In watermark embedding process, the DNN model is trained using a combination of original training samples and adversarial samples generated in the previous step. This process helps incorporate the watermark into the model without significantly affecting its performance on legitimate inputs.

  The proposed approach aims to embed the watermark in the decision boundary of the model, which is much more difficult to change without affecting the model's accuracy. The proposed method uses adversarial training to create

a watermark that is embedded in the decision boundary of the model. The watermark is generated by optimizing a set of parameters that are added to the model's weights. The optimization process is guided by an adversarial loss function that encourages the watermark to be detectable while minimizing its impact on the model's accuracy. [4]

- Liu et al."IPGuard: Protecting Intellectual Property of Deep Neural Networks via Fingerprinting the Classification Boundary" (2019) proposed the IPGuard method, which focuses on protecting the intellectual property (IP) of deep neural networks (DNNs) by embedding a unique fingerprint into the classification boundary. The authors introduced two techniques: Decision Boundary Mapping (DBM) and Decision Boundary Perturbation (DBP), to achieve robust IP protection.

  In their work, the authors first highlighted the increasing concern regarding IP theft in the context of DNN models. They emphasized the value of protecting the IP associated with trained models, as they contain valuable knowledge and represent significant investments of time, effort, and resources.

  The IPGuard method introduced by Liu et al. leverages watermarking techniques to embed a fingerprint into the decision boundaries of DNN models. This fingerprint is invisible and does not affect the model's classification performance. It allows the model owner to trace the source of potential IP infringement. [5]

Chapter 3

Watermark Attacks

3.1 Attacker

To understand when and how an attacker would try to get around the watermark, the attack surface needs to be described. It can be back box access or white box access. In the case of a black-box the attacker has no idea about the model weights and generally a public API is exposed for using the model. Whereas in white box access the attacker has access to all the parameters of the model and may or may not have access to the training dataset.

Attacker knowledge is the information the attacker has at the time of attack. Generally more the attacker's knowledge and more are the chances of the attack being successful. The knowledge can be of the following types.

- knowledge on the existence of the watermark
- model and its weights
- watermarking method used

- training data
- the watermark itself (like the secret message) or the trigger dataset.

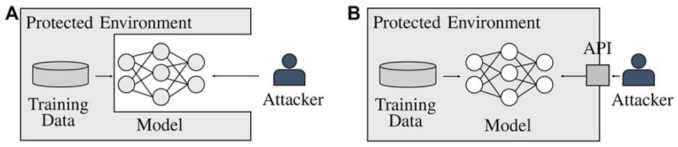

FIGURE 3.1: Different Access to attacker

3.2 Types of Watermark Attacks

Neural network watermarking, is susceptible to various attacks that aim to undermine its effectiveness or remove the embedded watermark. Some common attacks against neural network watermarking include:

- **Fine-tuning Attacks :** Fine-tuning attacks involve retraining the watermarked neural network with additional data or modifying the existing training data to diminish or remove the embedded watermark. Attackers may employ gradient-based optimization techniques to modify the network's parameters.
- **Model Pruning :** Attackers may perform model pruning or compression techniques to reduce the size or complexity of the watermarked neural network. During this process, the network's parameters or connections are pruned or pruned, potentially resulting in the loss or degradation of the embedded watermark.
- **Transfer Learning Attacks :** Transfer learning attacks involve reusing the watermarked neural network for a different task or in a different context. By

repurposing the network, attackers may aim to bypass the watermarking mechanism, as the network may no longer exhibit the same behavior

Chapter 4

Study Methodology

4.1 Introduction

This chapter presents the methodology of a hybrid model of neural network watermarking that combines two distinct methods, namely backdoor and bit encoding, to enhance the security and robustness of the watermarking process.

4.2 Methodology

- **Creating Trigger Dataset :** The first method employed in the hybrid model is backdoor embedding. Backdoor embedding involves strategically inserting hidden triggers or patterns into the DNN model, which can later be activated to perform specific actions or exhibit predetermined behaviors. This method allows for the incorporation of a unique watermark that can be activated or detected using a specific input or trigger, while remaining inconspicuous during

normal operation. Backdoor embedding provides an effective way to embed watermarks that can be selectively activated or detected when necessary.

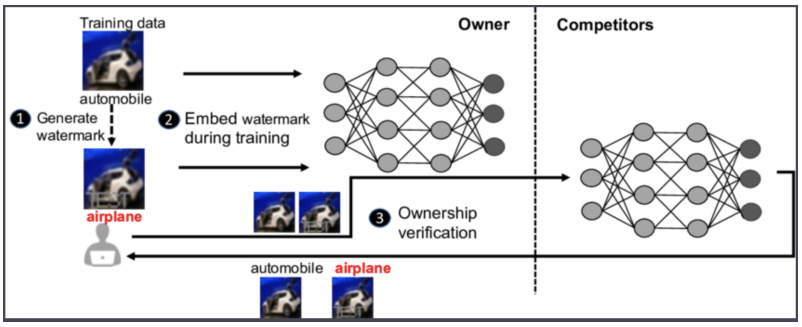

FIGURE 4.1: Trigger Dataset Training

- **Bit Encoding :** The second method integrated into the hybrid model is bit encoding. Bit encoding focuses on modifying specific bits or features of the model's weights or parameters to encode the watermark information. This method ensures that the watermark is embedded in a manner that is imperceptible to the model's performance on normal inputs. Bit encoding allows for precise control over the location and intensity of the watermark, enabling a high level of imperceptibility while maintaining the integrity of the model's functionality.

4.3 Models and Algorithms

The hybrid model of neural network watermarking, combining backdoor and bit encoding methods, typically involves the following steps:

- **Step 1 - Trigger Dataset Generation :** Generate a unique trigger dataset(image and label only known to the owner of the model), which serves as the unique identifier for the neural network model. Trigger dataset can be created in

many ways like superimposing two images, which may not be easily available to anyone.

- **Step 2 - Backdoor Embedding :** Incorporate a backdoor mechanism into the neural network model. This involves training of the model with a dataset containing the original dataset along with our trigger dataset.

- **Step 3 - Bit Encoding :** Encode the secret message into the weights or parameters of the neural network model. This can involve modifying specific weight values or introducing additional layers or modules that carry the watermark information. The bit encoding process ensures that the watermark is embedded within the network's parameters in a secure and imperceptible manner.

- **Step 4 - Watermark Extraction :** To extract the watermark, apply the specific trigger or activation pattern to activate the backdoor functionality of the network. This involves providing specific input samples that trigger the backdoor behavior or activating the backdoor through other means.

- **Step 5 - Watermark Verification :** Once the backdoor is activated, extract the watermark from the network's parameters or behavior. Verify the extracted watermark against the original watermark to validate the ownership or authenticity of the neural network model.

FIGURE 4.2: Watermark Creation

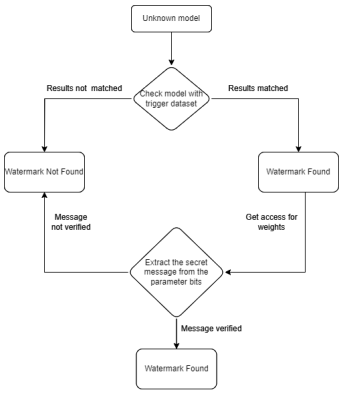

FIGURE 4.3: Watermark Extraction

4.4 Extraction

The extraction of the watermark from the hybrid model of neural network watermarking, which combines backdoor and bit encoding methods, involves a multi-step process aimed at accurately recovering the embedded watermark information. This section provides an overview of the extraction methodology employed in the hybrid model.

- **Trigger Activation :** The first step in the extraction process is to activate the backdoor trigger embedded in the neural network. This involves providing the network with a specific input or stimulus that matches the trigger pattern. The activation of the backdoor trigger allows the network to enter a state where the watermark information becomes accessible.

- **Weight Analysis :** Once the backdoor trigger is activated, the next step is to analyze the network's weights. In the case of bit encoding, specific bits within the network's weight parameters are modified to embed the watermark information. The extraction process involves identifying and analyzing these modified bits to extract the watermark.

4.5 Summary

By combining backdoor embedding and bit encoding, the hybrid model of neural network watermarking aims to achieve a synergistic effect. The hybrid model utilizes the selective activation or detection capability of backdoor embedding and the imperceptibility of bit encoding to enhance the overall robustness and imperceptibility of the embedded watermark. This combination provides a comprehensive solution that protects against unauthorized modifications and ensures the authenticity of the DNN model.

Chapter 5

Dataset

5.1 CIFAR10

A well-known benchmark dataset that is frequently used in the fields of computer vision and machine learning is the CIFAR-10 dataset. There are 60,000 32x32 colour photos total, divided into 10 classes with 6,000 images each. 10,000 test photos and 50,000 training images make up the dataset.

Following are the 10 classes in CIFAR-10:

Bird, Cat, Deer, Dog, Frog, Horse, Ship, Truck,Airplane and Automobile

For applications like image classification, object recognition, and deep learning model validation, the CIFAR-10 dataset is frequently used. It has evolved into a common benchmark for assessing the effectiveness of different machine learning architectures and algorithms in computer vision.

5.2 MNIST

The MNIST dataset is another well-known benchmark dataset commonly used in the field of machine learning and computer vision. It stands for "Modified National Institute of Standards and Technology" and is a collection of handwritten digits.

The MNIST dataset consists of 70,000 grayscale images of handwritten digits from 0 to 9. These images are divided into two main sets: a training set and a test set. The training set contains 60,000 images, while the test set contains 10,000 images.

Each image in the MNIST dataset has a size of 28x28 pixels, resulting in a total of 784 pixels per image. Each pixel is represented by an intensity value ranging from 0 to 255, where 0 represents white and 255 represents black. The dataset is preprocessed and normalized, with pixel values scaled to the range of 0 to 1.

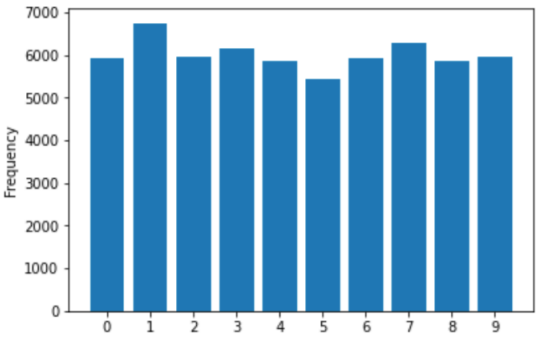

FIGURE 5.1: MNIST Dataset

Chapter 6

Results

We implemented our hybrid watermarking approach on the classification model for the dataset of CIFAR10 and MNIST. We compared the watermarked and non watermarked model on the basis of evaluation metrics like accuracy and F1 score.

Model	Type of Model	Epochs Trained	Accuray (%)		F1 Score
			Training	Testing	
Resnet20	Watermarked	30	89.26	75.3	0.72
	Non-watermarked	30	89.61	81.35	0.69
Resnet50	Watermarked	20	86.23	84.72	0.75
	Non-watermarked	20	87.94	81.94	0.79
Inception	Watermarked	23	75.55	71.4	0.58
	Non-watermarked	23	79.8	78.33	0.54
VGG16	Watermarked	28	90.76	84.75	0.84
	Non-watermarked	28	88.5	82.52	0.83
VGG19	Watermarked	25	88.69	84.95	0.62
	Non-watermarked	25	91.31	85.12	0.63

FIGURE 6.1: Result on various models

We can see that in all the model the hit taken in accuracy is less than 5%.So this approach can be used to embed watermark in the Neural Network.

Chapter 7

Future Scope and Conclusion

7.1 Future Scope

The Bit encoding on the parameters may change if the attacker trains the model with few images, making it hard for extractign the secret message.As discussed in the paper "https://arxiv.org/pdf/1701.04082.pdf" we can make our watermark resitent from fine tuning attack.

We can also explore other methods to embed watermarks into model parameters that are more reliable agains attacks and also incorporates both visible and hidden watermarking techniques .

7.2 Conclusion

In conclusion, the proposed methodology for the thesis on the hybrid model of neural network watermarking, combining backdoor and bit encoding methods, offers a comprehensive approach to enhance the security and robustness of watermarking in

neural networks. The combination of these two methods leverages their individual strengths and addresses their limitations, providing a more effective and resilient watermarking solution.

The methodology involves the activation of the backdoor trigger to access the embedded watermark information, followed by the analysis of the network's weights to identify the modified bits in the bit encoding process. The extraction process focuses on decoding and reconstructing the watermark from the modified bits, and the verification and authentication steps ensure the integrity and accuracy of the extracted watermark.

By combining the backdoor and bit encoding methods, the hybrid model aims to provide a watermarking technique that is resistant to attacks, robust against adversarial manipulation, and capable of authenticating ownership and protecting intellectual property. The methodology offers a holistic approach that incorporates both visible and hidden watermarking techniques, providing transparency and security simultaneously.

Refrences

[1] Adi, Yossi, et al. "Turning your weakness into a strength: Watermarking deep neural networks by backdooring." 27th USENIX Security Symposium (USENIX Security 18). 2018.

https://doi.org/10.48550/arXiv.1802.04633

[2] Z. Wang, C. Liu and X. Cui, "EvilModel: Hiding Malware Inside of Neural Network Models," in 2021 IEEE Symposium on Computers and Communications (ISCC), Athens, Greece, 2021 pp. 1-7.

https://doi.org/10.1109/ISCC53001.2021.9631425

[3] Yusuke Uchida, Yuki Nagai, Shigeyuki Sakazawa, and Shin'ichi Satoh. 2017. Embedding Watermarks into Deep Neural Networks. In Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval (ICMR '17). Association for Computing Machinery, New York, NY, USA, 269–277.

https://doi.org/10.1145/3078971.3078974

[4] Le Merrer, E., Pérez, P. Trédan, G. Adversarial frontier stitching for remote neural network watermarking. Neural Comput Applic 32, 9233–9244 (2020).

https://doi.org/10.1007/s00521-019-04434-z

[5] Cao, Xiaoyu Jia, Jinyuan Gong, Neil. (2021). IPGuard: Protecting Intellectual Property of Deep Neural Networks via Fingerprinting the Classification Boundary.

https://doi.org/10.48550/arXiv.1910.12903