CYBER ATTACKS ON MACHINE LEARNING MODELS:
A STUDY OF ADVESARIAL VULNERABILITIES

Enrollment No. (s) - 17103067, 17103103, 17103279
Name of Student (s) - Piyush Gupta, Chitrank Mishra, Dharmesh Pratap Singh
Name of Supervisor(s) - Dr. Sangeeta Mittal

Dec - 2020

Submitted in partial fulfillment of the

Degree of Bachelor of Technology

in

Computer Science Engineering

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING & INFORMATION TECHNOLOGY
JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

TABLE OF CONTENTS

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Place: Noida

Date: 5 December, 2020

Signature: …………………….

Name: **Piyush Gupta, Chitrank Mishra, Dharmesh Pratap Singh**

Enroll No.(s): **17103067, 17103103, 17103279**

CERTIFICATE

This is to certify that the work titled "**CYBER ATTACKS ON MACHINE LEARNING MODELS: A STUDY OF ADVESARIAL VULNERABILITIES**" submitted by "**Piyush Gupta (17103067), Chitrank Mishra (17103103), Dharmesh Pratap Singh (17103279)**" in partial fulfillment for the award of degree of **Bachelor of Technology** of **Jaypee Institute of Information Technology, Noida** has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

Signature of Supervisor ……………………….

Name of Supervisor: **Dr. Sangeeta Mittal**

Designation: Associate Professor

Date: 5 December, 2020

ACKNOWLEDGEMENT

We thank our mentor **Dr. Sangeeta Mittal , Department of CSE, Jaypee Institute of Information Technology, Sector-62, Noida**. We express our sincere gratitude towards her, as her guidance, encouragement, suggestions and constructive criticism have contributed immensely to the evolution of ideas on the project.

Turning this idea into a project wouldn't have been possible if she hadn't provided us with the knowledge she possesses and help us to get the best conclusion possible.

Signature of the Student(s): …………………….

Name of Student(s): **Piyush Gupta, Chitrank Mishra, Dharmesh Pratap Singh**

Enrollment Number(s): **17103067, 17103103, 17103279**

Date: 5 December, 2020

(V)

SUMMARY

Advances in the field of machine learning has led to revolutionizing technology in various cultures. It has also introduced capabilities that were not known before. With the advent of artificial learning expanding to support the physical world, rises the vulnerabilities that can be potential hazards to safety and security. Adversarial attacks on machine learning models are a way to exploit the learning structure of a system and create vulnerabilities which are beyond physical detection and recovery. These vulnerabilities houses capabilities from causing a classifier to misclassify, to causing trained and tested models to malfunction at run. Several algorithms have been introduced in the past few years which have happened to generate adversarial samples for detection of these anomalies.

Studying the methodologies and visualizing the algorithms have been a topic of research from decades. Finding an efficient approach to implementing the proposed ideologies and algorithms and visualizing them is still a subject of ideation. In this project, we intended to propose a method to test the algorithms on some defined datasets that are publicly used by researchers in researches and projects. We tend to propose a method to allow users to verify the researches and algorithm themselves and understand their working.

Signature of Student	Signature of Supervisor
Piyush Gupta, Chitrank Mishra, Dharmesh Pratap Singh	**Dr. Sangeeta Mittal**
5 December 2020	5 December 2020

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS & ACRONYMS

1. FSGM - Fast Gradient Sign Me
2. BIM - Basic Iterative Method
3. C&W - Carlini & Wagner
4. AI - Artificial Intelligence
5. ML - Machine learning
6. DNN - Deep Neural Network
7. SVM - Support Vector Machines
8. URL - Uniform Resource Locator
9. CIFAR - Canadian Institute for Advanced Research
10. MNIST - Modified National Institute of Standards and Technology
11. KNN - K Nearest Neighbour
12. GTSRB - German Traffic Sign Recognition Benchmark
13. CNN - Convolutional Neural Network
14. API - Application Programming Interface

1. INTRODUCTION

1.1 General introduction

Adversarial attacks on machine learning models are a way to exploit the learning structure of a system and create vulnerabilities which are beyond physical detection and recovery. These vulnerabilities houses capabilities from causing a classifier to misclassify, to causing trained and tested models to malfunction at run. Several algorithms have been introduced in the past few years which have happened to generate adversarial samples for detection of these anomalies. We intend to propose an ideology which check for the possible vulnerabilities which can be caused by such adversarial inputs and help detect them at the stage of training and testing.

Neural Networks have achieved the desired state-of-the-art performance on recognizing images. It is found that these networks often suffer defeat from samples involving perturbation on samples from the datasets. Finding defense mechanisms that are effective enough and capable to protect the model from such adversarial attacks is still a vast field for research. People in the area have made a few advancements and the techiniques are growing with implementation.

This study is targeted at collecting various types of attacks possible on neural networks. Studies in the field have shown great advancements in the designing algorithms that hampers the raw input resulting into a misclassified objects. Researches have shown how these algorithms plays with arcade games like Atari, etc. With every neural network, there are some policies associated that parameterise the neural network. For example, for a CNN model designed to classify images, perturbations added on the training input side can cause complete fail of the trained model. There are multiple scenarios available for the study of the effect of these adversaries. Supervised learning and unsupervised learning have their own course of vulnerabilities. An adversarial model effective on one training model, is applicable on various other models as well due to property of transfer-ability in adversaries. Such vulnerabilities can target any machine model either during learning by tampering with the training data or during inference by manipulating inputs on which model is making predictions.

In recent times, it's been determined that neural networks are fooled by adversarial examples simply. Several approaches are projected to form neural networks additional strong against white-box adversarial attacks, however they couldn't realize an efficient technique thus far. In this short paper, authors target the lustiness of the options learned by neural networks. they have a tendency to show that the options learned by neural networks aren't strong, and realize that the lustiness of the learned options is closely associated with the resistance against adversarial samples of neural networks.

Due to the complex nature of machine learning models, it is hard to identify the ways in which these models can be exploited when deployed. Recent findings on adversarial examples. which are inputs with some changes that result in different model predictions, is helpful in observing the robustness of these models by checking the adversarial situations where they fail. Although, such malicious examples are not natural as well as not applicable to complicated domains.

1.2 Problem statement

Studies in the field have shown great advancements in the designing algorithms that hampers the raw input resulting into a misclassified objects. Researches have shown how these algorithms plays with arcade games like Atari, etc and hamper the condition as always win. Keeping these vulnerabilities in minds, we came up with the following objectives to achieve

- Study the cause and effect of such adversaries.
- Identify the winners in the adversarial category.
- Implement a tool to demonstrate live attacks on models.
- Study the defense mechanism that can help defend the subject.

In the view of the above observations, we successfully designed a tool that can help us understand the effect of such adversaries on real-world objects and identify the shortcomings to serve the defense.

- Implement different kinds of attacks on similar models to help understand the scale of damage.
- Implement a tool to serve input into the model and automate the process of testing and processing.
- Show the proper cause of misclassification of the models.
- Visualize the before and after results of perturbation attacking.

With every neural network, there are some policies associated that parameterise the neural network. Our target is to identify the policies and make use of them to implement function which verify the researches studied and are successful in adding noise to images which leads to successful misclassification. Broader perspectives regarding the algorithms and implementations will be discussed later.

1.3 Significance/Novelty of the problem

The purpose of the problem statement is :

- To introduce the reader to the importance of the adversarial attacks on machine learning models and defense against the former.
- Provide appropriate parameters for further study on the subject.
- Collect the previous studies and conclusively file an output defining the progress in the field and the needs to focus upon in upcoming researches.
- Provide a better format to display the outcomes of such attacks on actual implementation and provide a basis/experimental setup to prove the proposed methodology.

1.4 Empirical study

It is possible to generate an image which when dot produced with any image in the world has a very high change of showing perturbated results by most of the models in the world. It is also found that adversaries will try to bypass their controls and drive frameworks for their vindictive closures. In acknowledgment of this reality, the AI and security communities must undertake to inoculate frameworks against such abuse. Along these lines, we should return to our measures of value for AI procedures and weigh not just the results they produce yet in addition to their capacity to oppose tests cautiously produced by adversaries.

Researchers observed the attack in the case of perfect and limited knowledge of the attacked system, and described that widely used classification algorithms (majorly SVMs and neural networks) can escape with high probability even if the adversary can only detect a copy of the classifier from a small substitute dataset. Hence, this observation raises some questions on whether such algorithms can be reliably used in security-sensitive applications. The increase in the level of classification increases the robustness of the model to adversarial perturbations also to noise. Adversarial training gives robustness to adversarial examples generated using singular methods. While adversarial training didn't help much against iterative strategies they observed that adversarial examples generated by iterative methods are less likely to be transferred between networks, which provides indirect robustness against black-box adversarial attacks.

1.5 Brief description of the solution approach

Neural networks are highly sensitive to adversarial examples are therefore poses a threat towards security application. It is found that these networks often suffer defeat from samples involving perturbation on samples from the datasets. Misclassification of images happen due to intentionally imperceptible perturbations to some parts of the images or precisely some pixels of the images. Work done by Goodfellow et al. is considered revolutionary in identifying such vulnerabilities that can hamper the strength of backbone of advanced technologies.

The idea is to design a web-based interface that can help increase the understanding of such attacks by actually showing the live interaction with the models. The portal shall allow the user to select the input of choice and test it on desired model. The models will be implemented in python and will be linked to the backend. The user shall also be having an option to set the extremity of adversary to be applied to the input. This noised input will be served to the model and the obtained output with the percentage of confidence, if available, shall be displayed as results on the portal.

The portal will be designed as an MVC architecture to enable the modular integrity of the project. Each model will be having a separate directory to store the intermediate files, if any. The portal will also display the noise map and hampered image, if available, for the model. A brief description about the model and the underlying working shall also be provided. Detailed description is provided later in this report.

2. LITERATURE SURVEY

2.1 Summary of papers studied

[1]. Machine Learning in Adversarial Settings

The paper conceptualizes the idea of how a model stores the encoded semantic information about how certain features or sets of features relate to the output class. An amount of modifications and perturbations is introduced in the data-set to yield a specific adversary-selected misclassification as output. The autonomous system can be misled into misclassifying stop signs as yield signs. To humans, these samples stay indistinguishable from the original input. Humans would classify both of these images as stop signs but the complexity for a machine to understand the image can be exploited to result in faulty classifier systems.

[2]. Adversarial Machine Learning at Scale.

Neural Networks and Machine learning models are highly vulnerable to attacks based on small modifications of the input to the model at the test time. This vulnerability possesses a transferability property. The infected input set for one machine model is also capable of infecting another machine model. Creating adversarial input requires injecting noise in the input set. The magnitude of the noise is variable according to the magnitude of the adversarial perturbation required. The robustness of such adversarially trained models increases with an increase in the model size.

[3]. Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition

Machine learning has expanded its zone of action from detecting cancer-cells to operating self-driving cars. The limitless use of machine learning algorithms in various life activities where physical safety is at risk, explains well for the study of possible attacks on a machine learning model. The authors have focused on understanding the vulnerabilities of machine models working for facial biometric systems. These attacks are physically realizable and inconspicuous, and allow an attacker to use false identify or bypass the classifier by impersonating another individual. The research focuses on identifying vulnerabilities in white-box face-recognition systems, but they have also demonstrated the possible techniques for black-box scenarios to avoid face-detection.

[4]. Fundamental limits on adversarial robustness

Paper focuses on finding if there is any difference between noise and adversarial noise. Also this focuses on finding out if there is a way to reduce or eliminate adversarial noise in Deep Learning Networks or is it the inherent part of it. This paper studies adversarial attacks and their effects on linear and quadratic classifiers in binary settings. In both the cases, paper's results showed their existence of a fundamental limit on the robustness to adversarial perturbations. It is found out that quadratic models perform better in every case and have better results then linear models.

[5]. Adversarial Examples are not Bugs, they are Features.

This paper states that the Adversarial Examples are not bugs but actually they are the feature of the machine learning model. Machine Learning models are built in such a way that they are going to learn any feature they find common in most of the data set and that is the thing which is exploited by Adversarial Perturbations. This paper provides an alternative approach to learning by differentiating features into robust and non robust features. Basically this paper argues that we need to make the machine learning model more human-like then model oriented on what is stored in pixels.

[6]. Universal adversarial perturbations

This paper tries to find out that if there is an image which can be added to any image and then that image will be misclassified by most of the classifiers. This paper proposes an algorithm to find out these kinds of images and proves that these kinds of images are possible and can be found using an algorithm. This paper also proves that universal perturbations have a remarkable property of misclassification of any image by any model.

[7]. Poisoning Attacks against Support Vector Machines

This paper described the implementation of a family of poisoning attacks against Support Vector Machines (SVM). The attack proposed in the paper uses a gradient ascent method in which properties of the SVM's optimal solution are the basis of gradient computation. Attacks on learning algorithms can be classified into exploratory (exploitation of the classifier) and causative (manipulation of training data). Poisoning refers to a causative attack (manipulation of training data) in which crafted attack points are merged into training data.

[8]. Evasion attacks against machine learning at test time

This paper's author proposed a gradient based approach that can be used to identify the vulnerability of mainly used classification algorithms with respect to evasion attacks. Some attacking scenarios are explained which make various risk levels for the classifier by increasing the attacker's knowledge about the system and increasing the ability of the attacker to manipulate attack samples.

[9]. Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods

Carlini and Wagner proposed ten defensive techniques which detect several adversarial examples which were considered from seven papers. It is previously stated that classification of adversarial examples attempts have failed mostly, that is why the research was back on detecting only adversarial inputs. Carlini and Wagner stated that even it is quite difficult that such approaches can be defeated by a zero-knowledge attack (in which detector is not visible to the attacker) mostly. A zero-knowledge attack works against the two scenarios, that's why this attack is tried first. Perfect-knowledge attacks (white-box attack) can sometimes be adapted to the limited-knowledge situation by designing a substitute neural network and making a white-box attack against that network. Carlini and Wagner also stated that limited-knowledge attacks (black-box attack) only came into consideration if zero-knowledge attacks fail and perfect-knowledge attacks are successful.

[10]. Adversarial vulnerability for any classifier.

Despite achieving impressive performance, state-of-the-art classifiers remain highly vulnerable to small, imperceptible, adversarial perturbations. This vulnerability has proven empirically to be very intricate to address. In this paper, we study the phenomenon of adversarial perturbations under the assumption that the data is generated with a smooth generative model. We derive fundamental upper bounds on the robustness to perturbations of any classification function, and prove the existence of adversarial perturbations that transfer well across different classifiers with small risk.

[11]. Generating Natural Adversarial Examples.

Due to the complex nature of machine learning models, it is hard to identify the ways in which these models can be exploited when deployed. Recent findings on adversarial examples. which are inputs with some changes that result in different model predictions, is helpful in observing the robustness of these models by checking the adversarial situations where they fail. Although, such malicious examples are not natural as well as not applicable to complicated domains. In this paper, authors proposed a framework to make natural and reliable adversarial examples by observing in semantic space of dense and continuous data representation which is utilizing the recent findings in generative adversarial networks.

[12]. Learning More Robust Features with Adversarial

In recent times, it's been determined that neural networks are fooled by adversarial examples simply. Several approaches are projected to form neural networks additional strong against white-box adversarial attacks, however they couldn't realize an efficient technique thus far. In this short paper, authors target the lustiness of the options learned by neural networks. they have a tendency to show that the options learned by neural networks aren't strong, and realize that the lustiness of the learned options is closely associated with the resistance against adversarial samples of neural networks. They have a tendency to conjointly realize that adversarial coaching against quick gradients sign technique (FGSM) doesn't build the learned options terribly strong, notwithstanding it will build the trained networks terribly proof against FGSM attack

[13]. Adversarial Examples Are a Natural Consequence of Test Error in Noise

This paper shows that adversarial examples are just a natural consequence of test error in noise. And they should not be taken as bugs. Finally, this paper shows that methods which are going to increase the distance to the decision boundary will also improve robustness towards Gaussian noise, and vice versa. Author states that, given the error rates it is observed in Gaussian noise, small perturbations it is observed in practice appear that roughly the distances would be expected from a linear model, and that therefore there is not much need for invoking any properties of the decision boundary to explain them.

[14]. Are adversarial examples inevitable

This paper tries to find that if it is possible or not to prevent adversarial perturbations. The author says that the question that if adversarial perturbations are inevitable is wrong.And any model has a limit on correctness to adversarial perturbations that cannot be removed. But, paper proves that these limits depend on fundamentals of the dataset, and also on the power of the adversary and the metric system used to measure different kinds of perturbations. This paper provides great details of these limits and shows us how they are inter- dependent on properties of the distribution of data.

[15]. A Simple Explanation for the Existence of Adversarial Examples with Small Hamming Distance

The paper tries to prove that there exists a Small hamming distance for perturbing any image. In the research made earlier to explain the existence of perturbations they are using a Deep Learning model and an input X whose class is given by the model as belonging to some class C1, and they wanted to find some Y with distance(X,Y) as less as possible which is classified as belonging to some other class C2. In this paper the author considered a better way of attacking, in which the author is taking two class D1 and D2,along with an input X C1, and their goal is to search for some nearby Y which is inside C_2.

[16]. Standard detectors aren't (currently) fooled by physical adversarial stop signs

Adversarial examples that exist can be used to fool a detector and create unusual and uncontrollable situations. One such example is the physical adversarial stop sign which is known to fool a large group of classifiers and detectors, but then comes RCNN and YOLO, which was able to be classified as a non-stop sign. An adversarial pattern on a physical object can be detected using a wide family of parameters such as scale, view of angle, etc. Such a pattern is found shall be of great practical and theoretical use. It is difficult to diagnose a misclassifier as compared to a mis-detector unless we get to eliminate the effects of rescaling and resizing.

[17]. Adversarial Examples: Attacks and Defenses for Deep Learning

As rapid progress in a wide spectrum of applications, many safety-critical applications use deep learning. But, many vulnerabilities have been found in deep neural networks to adversarial examples which are well designed input samples. These types of inputs are not identified by humans but deep neural networks can be fooled easily by these examples. So, this becomes a major issue in a safety-critical environment. In this paper, authors observe some recent theories on adversarial examples for deep neural networks and summarize some attacks of adversarial examples and taxonomy of these examples.

[18]. Evaluating a Simple Retraining Strategy as a Defense Against Adversarial Attacks

Neural networks are found to be vulnerable on adversarial examples, such inputs which are close to natural inputs but classified wrongly. For better understanding the adversarial examples, authors observed ten recent findings which are designed to detect adversarial examples.They show that all of those can be defeated by making new loss functions. In this paper, authors describe neural networks applied to image classification. As neural networks are the mostly accurate machine learning approach known till now, they are fighting against an adversary who can fool the classifier. For that , a natural image x is given, an adversary produces a visually same image x easily which will be classified differently. But , most of these defenses failed to classify adversarial examples correctly.

[19]. Defense against Adversarial Attacks Using High-Level Representation Guided Denoiser.

Neural networks are highly sensitive to adversarial examples are therefore poses a threat towards security application. This study proposes a high-level representation guided denoiser (HGD) as a defense towards adversarial image classification. Standard denoiser face problems of error amplification effect, in which small residual adversarial noise is progressively amplified and leads to wrong classifications. Using a loss function, HGD overcomes this problem. The function defines a difference between the target model's outputs activated by the clean image and denoised image.

On comparing with the state-of-the-art classifier, HGD has few advantages over it. The target model is more robust to either white-box or black-box attacks with HGD as a defense. HGD can be trained with a few image sets to perform well on other classes. HGD can transform from guiding a model to defending it when needed.

HGD won the first place in NIPS competition on defense against adversarial attacks and also outperformed other models by a huge margin.

[20]. APE-GAN: Adversarial Perturbation Elimination with GAN

Neural Networks have achieved the desired state-of-the-art performance on recognizing images. It is found that these networks often suffer defeat from samples involving perturbation on samples from the datasets. Finding defense mechanisms that are effective enough and capable to protect the model from such adversarial attacks is still a vast field for research. People in the area have made a few advancements and the techiniques are growing with implementation. This study proposes an idea based on Generative Adversarial Networks named APE-GAN is targeted to defend against these adversarial examples. An experimental study is also conducted to find out the efficiency of the implementation on MNIST, CIFAR-10 and ImageNet indicate that APE-GAN is effective to resist adversarial examples.

[21]. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks

Identifying test samples for image data which sufficiently diverse when compared with the training distribution statistically or adversarially is a basic requirement for deploying a good classification model. Deep neural networks are capable of producing methods to detect any abnormal samples which are applicable to all the softmax classifiers. Most prior methods have been reported for detecting either out-of-distribution or adversarial samples, but not both, the proposed methods achieves state of the art performances for both cases in various experiments conducted. The proposed methods is more robust in certain tough scenarios. It is shown that the proposed idea enjoys broader application by applying it to class-incremental learning. That signifies whenever out-of-distribution samples are detected, the method is able to create new classification classes without further training.

[22]. No Need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles

It is shown in various researches that machine learning algorithms are prone to adversarial perturbations. There are cases where physical adversaries are possible by printing malicious images and taking a picture of the same. But a major factor that hasn't been given weightage in calculations is the physical aspects of the object. The camera can view objects from different angles and different distances. This paper shows that the current physical adversaries are not enough to create perturbations for object detection from a moving platform. It is believed that perturbed images can exhibit malicious behavior within a range of distances. Thus, the practical impact of these perturbations can be reduced when it comes to observation from a moving platform.

[23]. Explaining and harnessing adversarial examples.

This paper tries to explain the basic reason for occurrence problems due to adversarial perturbations in any model. The paper states that the problem becomes more prominent as we have models of higher dimension. It states that as humans live only in 3 dimensions so we cannot perceive the effect of small changes in every dimension. This paper clearly shows how very small changes in all the dimensions can change the end result of the model.

[24]. Synthesizing Robust Adversarial Examples

This paper shows how adversarial examples can be generated in the real time world as the adversarial examples generated using common algorithms like FGSM and CW have a very limited success. Prior work has shown that adversarial examples generated using these standard techniques often lose their adversarial nature once subjected to minor transformations. This paper uses a new algorithm called Expectation over transformation.

[25]. Robust Physical-World Attacks on Deep Learning Visual Classification.

This paper is about hiding in plain sight. This approach just makes innocuous changes that "hide in the human psyche," rather than attempt to make imperceptible changes. Choosing road signs as an attack vector is a good approach as signs are visually simple, so it is difficult to hide perturbations. They are merged with a noisy, complex environment. And there are real-world safety effects, especially as autonomous vehicles come into major use.

[26]. Practical Black-Box Attacks against Machine Learning.

Papernot et al designed an attack that gets rid of a defence for an adversarial example that has been created previously. Adversarial examples transfer well between neural classifiers which have been trained on the same data but till then these types of attacks were limited to either white-box attacks. In this paper that limitation was shattered with a new querying heuristic that effectively takes out information about a classifier's decision boundaries only by checking its label outputs.

[27]. Parseval Networks: Improving Robustness to Adversarial Examples

This paper focuses on finding methods which are going to help us in increasing the robustness to adversarial perturbation. In this paper the author introduced Parseval network, a regularization method which works layerwise for reducing the sensitivity of network to small perturbations by controlling various global constants including the Lipschitz constant. Since the deep learning neural network is a composition of various functions which are represented by its different layers, author tries to achieve higher level of robustness by constantly trying to maintain a small Lipschitz constant (e.g., 1) at every underlying layer; be it fully-connected, convolutional or residual.

[28]. Boosting Adversarial Attacks With Momentum

There are a lot of algorithms which are vulnerable to attacks by adversarial abnormalities, especially the deep neural networks. Most of the existing attacks are capable of fooling a black box model. The study proposes a broad class of momentum-based iterative algorithms. By connecting it with a momentum into an iterative process for attacks. For the improvement of the success rates for black box attacks, they apply a momentum iterative algorithm which ensemble a model and show that the adversarial model with a strong defense are also vulnerable to the black box attacks.

[29]. Adversarial Attacks on Neural Network Policies.

Studies in the field have shown great advancements in the designing algorithms that hampers the raw input resulting into a misclassified objects. Researches have shown how these algorithms plays with arcade games like Atari, etc. With every neural network, there are some policies associated that parameterise the neural network. For example, for a CNN model designed to classify images, perturbations added on the training input side can cause complete fail of the trained model. There are multiple scenarios available for the study of the effect of these adversaries. Supervised learning and unsupervised learning have their own course of vulnerabilities. An adversarial model effective on one training model, is applicable on various other models as well due to property of transfer-ability in adversaries. Such vulnerabilities can target any machine model either during learning by tampering with the training data or during inference by manipulating inputs on which model is making predictions.

[30]. Simple Black-box Adversarial Attacks.

The study is proposing a method to construct adversarial images in a black-box setting. In contrast to the white-box scenario, constructing a black-box adversarial image has a constraint on the computation cost, hence efficient attacks still remain a goal to achieve. Taking few assumptions about the confidence values, the algorithm proposed is highly query-efficient and uses an iterative principle: they are taking a random vector on an orthonormal basis and adding or subtracting it from the target image. The proposed method can be used for both targeted and untargeted attacks, giving pretty efficient querying processing in both the scenarios.

2.2 Integrated summary of the literature studied

Table 2.2.1. Integrated summary of the literature studied

S. No	Methods Used	Dataset	Results	Remarks
1	Pre-processing the model with available input data-set and testing the model for correct classification. Testing it for the adversarial counter data-set. Calculating the deviation from correct classification.	-Not-Used-	With these propels, adversaries will try to bypass their controls and drive frameworks for their vindictive closures. In acknowledgment of this reality, the AI and security communities must undertake to inoculate frameworks against such abuse.	This paper provides an easy insight to the concept of adversarial learning. It has an array of examples defining various scenarios where adversaries can cause damage. Good to understand the concept and know how a machine model system works.
2	Adversarially training a model using synchronous distributed training on 50 machines, with a	Imagenet large scale visual recognition challenge 2017. The data-set will	They showed that adversarial training gives robustness to adversarial examples generated using	This paper aimed at showing the vulnerabilities of a faulty machine model. It also made the

	minibatch of 32 examples on each machine.	contain 1,50,000 photographs, hand labeled into 1000 object categories, taken from Flickr and other sources.	singular methods.	reader understand how the adversaries can be transferred with the learning characteristics from one model to another.
3	a. White-box DNNs For Face Recognition. b. Attacking White-box Systems. c. Facilitating physical realizability.	-Not Used-	The authors were able to demonstrate the techniques for generating accessories in the form of eyeglass frames that could fool the state-of-the-art facial recognition systems.	This paper shows the various methods which are employed to create adversarial input set. It was well enough for one to understand the concept of how adversaries are created.
4	Linear and Quadratic classifier models have been tested on adversarial perturbations and noise and the results have been plotted out on the graph of their accuracy on training and testing data	-Not Used-	This paper shows how the increase in the level of classification increases the robustness of the model to adversarial perturbations also to noise.	This paper shows how increasing the dimensionality of a system makes it more prone to adversarial perturbations. This paper also shows that system robustness decreases with dimensionality hence perturbations are different from noise.
5	Classify features of the model into robust and non robust features while training a model.	-Not Used-	The previous theory which plainly blames The higher dimensionality of the data set are not completely correct	This paper shows that our thinking about adversarial perturbations is wrong and we should not consider them as

			and the adversarial perturbations depends highly on the choice of features.	bugs but we should think of them as features of a Machine Learning algorithm. This paper states that we need to change our way of machine learning by differentiating features into robust and non robust feature and make the process more human like and less machine like.
6	Find out that if there is an image which can be added to any image and then that image will be misclassified by most of the classifiers.	-Not Used-	Proved that there exist many universal perturbations which can be applied to any image and that image will be majorly misclassified by most of the classifiers.	This paper shows that it is possible to generate an image which when dot produced with any image in the world has a very high change of showing perturbated results by most of the models in the world.
7	The attack proposed in the paper uses a gradient ascent method in which properties of the SVM's optimal solution are the basis of gradient computation.	MNIST dataset which is a handwritten digit recognition dataset. It contains 8-bit grayscale images of "0" through "9". There are about	The classification error is overestimated by the validation error due to a smaller sample size. This concludes that this attack can gain higher error rates than labels flipped randomly, and	The idea of vulnerability of SVMs has come into view from this paper. And poisoning attacks can easily exploit the working of SVMs.

		6K training examples of every digit and 1Ktest examples of every digit.	detects the vulnerability of the support vector machine (SVM) to poisoning attacks.	
8	Two experiments were conducted: a. A toy example from the MNIST handwritten digit classification task. b. Detection of malware in PDF files which shows the effectiveness of the proposed attack.	PDF corpus with 500 malicious samples from the Contagio dataset and 500 gentle samples.	The attack in the case of perfect and limited knowledge of the attacked system, and described that widely used classification algorithms (majorly SVMs and neural networks) can escape with high probability even if the adversary can only detect a copy of the classifier from a small substitute dataset.	Widely used neural networks can be attacked with only little knowledge about the classifiers. So, this is obviously a matter of concern for organizations where such networks are used for various purposes.
9	Approaches are categorised into 4 categories. a. Secondary classification. b. PCA and dimensionality reduction. c. Classical statistical approaches. d. Randomization and Blur.	CIFAR-10 dataset - This dataset consists of 60k 32*32 colour images classified in 10 sections with 6k images in a section. Among these, 50k are training images and 10k are test images.	Zero-Knowledge Attack Evaluation: Grosse 2017 observed that 98.5% of attacks were adversarial. Perfect-Knowledge Attack Evaluation: none of these approaches are effective on MNIST. Limited-Knowledge Attack Evaluation: Grosse’s defense is	Achieving a higher accuracy is useful and interesting result in machine learning tasks but this is not secure or sufficient for secure machine learning. We should consider the attackers mindset like if they even knew about the defense work still defense remains secure.

			not effective and can be easily attacked even by an attacker who does not have the knowledge of the model parameters.	
10	They train a DCGAN generative model on this dataset, with a latent vector dimension d = 100, and think about many neural networks architectures for classification. For every classifier, the empirical lustiness is compared to our boundary. additionally to news the in-distribution and at liberty lustiness, additionally report the lustiness within the latent space	SVHN dataset	Experiments on SVHN dataset. Authors report 25 % of the normalized lustiness at every cell, wherever chances are squared, measured either on paper.	We derive fundamental upper bounds on the robustness to perturbations of any classification function, and prove the existence of adversarial perturbations that transfer well across different classifiers with small risk.
11	Authors apply their approach to two standard datasets, MNIST and LSUN, and generate natural adversaries. They use r = 0.01 and N = 5000 with model	MNIST dataset, LSUN dataset	For MNIST's hand-written digits , author picked up 20 images, 2 for each digit and generated adversaries against RF and LeNet then observed 13 responses for each of	Such malicious examples are not natural as well as not applicable to complicated domains. In this paper, authors proposed a framework to make natural and reliable

	details.		the questions. They also checked adversaries for the LeNet model generated by FGSM and found that 78% of the time the program agrees that adversaries changed to the original images and are more natural.	adversarial examples by observing in semantic space of dense and continuous data representation which is utilizing the recent findings in generative adversarial networks.
12	To create the options learned by neural networks that are additional sturdy, authors tend to add a distortion term to the initial adversarial objective performance to encourage the distortions to be smaller throughout coaching. Formally, they tend to train neural networks with this objective function	CIFAR-10 Dataset, MNIST Dataset	Accuracy that the trained networks achieve on clean test data and adversarial test data.	They have a tendency to conjointly realize that adversarial coaching against quick gradients sign technique (FGSM) doesn't build the learned options terribly strong, notwithstanding it will build the trained networks terribly proof against FGSM attack
13	For linear models, the rate of error in the Gaussian noise is going to exactly determine the distance between the decision	-Not Used-	This paper finally tries to answer whether we should be focused to find adversarial examples as close as we are	For given error rates it is observed in Gaussian noise, small perturbations it is observed in practice appear that roughly

	boundary.Then author compared Neural Networks to the Linear Case. The decision boundary in Deep Learning model is not linear.		currently focusing on, given that the error rates we have observed in the corrupted image distributions.	the distances would be expected from a linear model, and that therefore there is not much need for invoking any properties of the decision boundary to explain them.
14	The idea he used is to show that, if the given class of data takes up enough space,then nearly every unique data point in the class will lie close to the boundary of the class.	-Not Used-	This paper shows in great detail that it is not possible to prevent adversarial perturbations completely by using any method available. This paper also shows that the adversarial perturbations are the fundamental property of machine learning and to some extent they are going to affect the model.	Paper proves that these limits depend on fundamentals of the dataset, and also on the power of the adversary and the metric system used to measure different kinds of perturbations.
15	Authors used MNIST dataset, where their algorithm failed and did not find any example with Hamming distance of less than or equal to 10 , but what they found is a group of 11	MINST Dataset	In this paper authors had developed a new and innovative method to rethink about the adversarial examples, and authors had explained why we find in our neural	In this paper the author considered a better way of attacking, in which the author is taking two class D1 and D2,along with an input X C1, and

	out of the 784 pixels which on manipulating could change the prediction from one digit to other digit.		network adversarial perturbations which contains a Hamming distance of m+1 in Deep Learning models which are used to distinguish between a m number of classes.	their goal is to search for some nearby Y which is inside C_2
16	Finding the difficulties observed while classifying and detecting stop sign in moving video using RCNN and YOLO algorithms.	Random videos from youtube having a car driving by a stop sign.	It can be said that there is no physical anomaly found yet that can fool a detector. An adversarial pattern to fool a detector has to be adversarial in many aspects such as scale, view of angle, illumination, etc.	This paper aimed at making the reader understand the preventive measures against the faulty machine model, if one is. It has made clear points about the factors like distance, angle and illumination which can be made use of to prevent faulty classification.
17	One Pixel Attack. Su et al. made adversarial examples by changing one pixel to avoid the problem of perceptiveness measurement.	CIFAR-10 dataset, MNIST dataset, ImageNet	They checked existing methods for generating adversarial examples. Authors tried to cover study of state-of-the-art for adversarial examples in the deep learning domain.	In this paper, authors observed some findings of adversarial examples in deep neural networks.
18	Authors use the L2	CIFAR-10	Retraining the	They show that all of

	attack for our experiments as a result of it's thought-about to be the strongest among the 3 attacks. For each of the datasets, the target label is the label of the smallest amount of probable category.	dataset, MNIST dataset	network by the adversarial pictures generated by the Carlini-Wagner rule for CIFAR-10 and TinyImageNet Dataset. The quantity of adversarial pictures used for training is the same because the number of original training pictures.	those can be defeated by making new loss functions. In this paper, authors describe neural networks applied to image classification.
19	They introduced a pixel guided denoiser which is mapped to work with the Imagenet dataset. A potential problem with this pixel guided denoiser is the amplification effect of adversarial noise in the topmost layers. HGD overcome this problem, where the supervised signal comes from certain high-level layers of the target model. HGD uses the same U-net structure as DUNET. The activities of this layer are feed to the linear classification layer after the global	30K images from the ImageNet training set	From the study it is found that DUNET has much lower denoising loss than DAE and NA which represents structural advantage of DUNET. DAE does not perform well with encoding of high-resolution images and hence the accuracy drops significantly. For white-box attacks, DUNET has much lower denoising loss than DAE but the classification accuracy is significantly worse.	HGD won the first place in NIPS competition on defense against adversarial attacks and also outperformed other models by a huge margin.

	average pooling.			
20	The study proposes an algorithm to apply defense against adversarial examples and eliminate the adversarial perturbation from the input set. GAN or Generative adversarial network proposed by Goodfellow et al is able to generate images that are similar to the training set with an addition of a little noise.	MNIST, CIFAR-10, ImageNet	The error rates of adversarial inputs are significantly decreased after its perturbation is reduced by APE-GAN. The error rate of FGSM is much larger as compared to L-BFGS. The aggressivity of adversarial examples can be eliminated by APE-GAN so is the perturbation whether regular or irregular, can also be eliminated.	This study proposes an idea based on Generative Adversarial Networks named APE-GAN is targeted to defend against these adversarial examples. An experimental study is also conducted to find out the efficiency of the implementation on MNIST, CIFAR-10 and ImageNet indicate that APE-GAN is effective to resist adversarial examples.
21	The idea is to measure the probability density of test sample on the spaces of features of DNNs utilizing the concept of a generative (distance-based) classifier. Contrary to the conventional beliefs, they found that using a generative classifier does not	CIFAR-10, ImageNet, ResNet	They proposed a simple yet effective method for detecting abnormal test samples including both out-of-distribution and adversarial ones. The main idea was to induce a generative classifier and define new confidence scores based on it.	The proposed methods is more robust in certain tough scenarios. It is shown that the proposed idea enjoys broader application by applying it to class-incremental learning. That signifies whenever out-of-distribution samples are detected,

	hampers the softmax accuracy. On the other hand, it's confidence score outperforms softmax-based ones very easily on various specified tasks.		They believe that the approach has the potential to apply to many other related machine models and learning tasks.	the method is able to create new classification classes without further training.
22	Methods which are considered to create adversarial images are: a. Fast Sign Method b. Iterative Methods c. L-BFGS Method d. Attacking a detector	180 photographs of stop sign at a highway, from various angles and distances.	This paper shows that even if the sign possesses some kind of perturbation, it will go undetected when parameters like distance, angle, illumination, blurriness are taken into account.	This paper explores the region of research in the area of preventive measures against faulty classifications. It shows how a model can be made to avoid misclassification by using a few methods describe therein.
23	Monitoring the behaviors of linear model and non-linear model.	CIFAR-10 dataset - This dataset consists of 60k 32*32 colour images classified in 10 sections with 6k images in a section. Among these, 50k are training images and 10k are test images.	Adversarial perturbations are nothing but high dimensional dot products of different vectors. They are a result of models not being nonlinear.	This paper explains how the perturbations are caused and it shows that they are nothing but dot product of 2 vectors. And as they are dot product so the direction of the vectors matters most. Hence the images taken in the real world applications are less prone to perturbations as specific angle can not

				be maintained in the real world images.
24	Minimize the perceived distance as seen by the classifier. EOT algorithm requires the ability to differentiate between 3D render functions with respect to texture.	-Not Used-	Adversarial examples and objects are a practical concern for real world systems, even when the examples are viewed from a variety of angles and viewpoints	This paper shows that using some advanced algorithms like EOT we can generate images which are effective irrespective of the direction in which the image is taken. Hence the adversarial perturbations can cause real trouble to the mankind with increasing using of AI in day to day life.
25	Taking images of the real physical target object from several angles, distances, and lighting conditions. Inputs are augmented with analytic changes to brightness.	a. LISA, a U.S. traffic sign dataset which contains 47 different road signs. b. German Traffic Sign Recognition Benchmark (GTSRB).	Two types of attack are there, one is poster-printing in which print-out covers the entire sign and sticker attacks, with graffiti-like.	Generating physical adversarial examples robust to largely varying range is possible. This shows that defenses that came in view in future should not ase on physical sources of noise as defense against these adversarial examples.
26	Jacobian-based Dataset Augmentation	MNIST dataset which is a handwritten digit recognition dataset. It contains 8-bit	Deep Neural Network attack results in working against logistic regression models, decision	This paper clears that what humans see and what algorithms see can be exploited. Humans can’t make any difference

		grayscale images of "0" through "9".	trees, SVMs, KNN and distilled networks.	between the original sign and adversarial sign which makes it difficult to identify the attack.
27	Author's main idea is to control the Lipschitz constant by using parameterization in the network with a very tight parseval frame, a generalization of orthogonal matrices.	CIFAR and SVHN dataset.	Author introduced new type of neural network Parseval networks, this is a new approach in the learning of a neural network that is more robust by nature to most kinds of adversarial noise. Author proposed an algorithm which will allow us to make better optimization in the model and in a very efficient manner.	Since the deep learning neural network is a composition of various functions which are represented by its different layers, author tries to achieve higher level of robustness by constantly trying to maintain a small Lipschitz constant (e.g., 1) at every underlying layer; be it fully-connected, convolutional or residual.
28	They plan to introduce a new class of attacks where they accumulate the gradients of the loss function after each iteration and then use it to stabilize optimization and try to divert from the poor local maxima.	-Not Used-	This paper introduces a braod class of momentum based attacks, which are iterative in nature and boots adversarial attacks. These can effectively fool a white-box attacks as well as black-box attacks.	For the improvement of the success rates for black box attacks, they apply a momentum iterative algorithm which ensemble a model and show that the adversarial model with a strong defense are also vulnerable to

				the black box attacks.
29	The study has summarized various subjects to plant adversaries in the input as well as in the training policy. The paper shows the use of FGSM as a white-box attack to compute the adversarial perturbations on a trained network, and as a black-box attack by computing the gradients for a separately trained policy enabling the transferability property.	-Not Used-	It is observable that there is a need to develop defenses against the adversarial attacks. This might involve adding adversarially-perturbed inputs during training of model to avoid possibilities of misclassification.	There are multiple scenarios available for the study of the effect of these adversaries. Supervised learning and unsupervised learning have their own course of vulnerabilities. An adversarial model effective on one training model, is applicable on various other models as well due to property of transfer-ability in adversaries.
30	The authors are repeatedly picking up random vectors from the orthogonal space of search directions. Using the confidence score obtained in each response to check if it is pointing towards the decision boundary or not and then perturb the image by adding or subtracting the vector from the image.	ImageNet sample	Given the real world applicability, the algorithm can be used to develop defense against malicious adversaries under this more realistic threat model. Also the method requires very few specifications and hence is more suitable when it comes to applications	The study shows demonstration on various real world settings including the Google Cloud Vision system. The study system stands string for becoming a baseline for future innovation in black-box attacks.

3. REQUIREMENT ANALYSIS AND SOLUTION APPROACH

3.1 Overall description of the project

Machine learning is driving rapid innovation and providing new insights into how we can interpret and control complex data and environments. With these advances, adversaries will seek to circumvent their controls and drive systems for their malicious ends. In recognition of this reality, this project aims at visualizing the adversaries which hampers the outcome a classifier model.

Studies have introduced few named algorithms which are known to affect the output of a classifier model. This project aims at visualizing these algorithms using the available tools. The subject has been a topic for research from a very long time, but there is no solution available which shows the affect of an algorithm on the same input for same classifier. There is no method available to visually compare the algorithms and provide adequate information about the working of the same.

The project is based on four named algorithms which are defined later in this report. The comparison data and visualization will be made available using a web portal. The web portal will be having options that will provide the selective input to the model and other required parameters. The selection will trigger an action in the backend to feed the classifier and format the results. The results shall be displayed in the required window on the portal. The algorithms triggered by the portal are a part of the research work we conducted. These algorithms have been subjects for a very large number of researches and are very popular among the likewise. Implementing these algorithms required us to train the model on a common dataset so as to provide comparable results.

We aim at providing a tool that helps general researchers understand the effects of adversarial attacks on the input data of a image classifier model. It will help the beginning researches understand the concept of adversarial perturbations better and grow along. This will also provide them with a direction to led their research and come up with better formatted results. This can also help them identify the required level of defense to apply to defend the attacks of these perturbations on these inputs.

3.2 Requirement analysis

Table 3.2.1. Model Implementation Requirements

Requirement	**Tool**
Language	Python
Training Environment	Pytorch
Data Set	ImageNet, CIFAR-10
Image Vision	OpenCV, TorchVision

Table 3.2.2. Web Portal Requirements

Requirement	**Tool**
Language	HTML, CSS, Python, Javascript
Framework (Frontend)	Bootstrap
Framework (Backend)	Django
Route Definition	Axios
Version Control	Git

3.3 Solution approach

The project is divided into a three-staged process. The details of the various stages are provided below:

Stage 1: Identifying the research work.

In the first stage, we planned to identify various researches performed in the direction of adversarial study. We studied paper from famous researchers from around the world, including Goodfellow and Papernot. Goodfellow is identified as the man behind beginning the chapter on adversarial study. We studied paper dating from the identification of the problem to most latest researches identifying the support and defenses against the former.

Stage 2: Implementing the algorithm models.

In this stage, we grouped four algorithms to implement namely,

- Fast gradient sign method.
- One pixel attack method
- C&W attack method
- Basic iterative method.

We implemented these algorithms using python language and pytorch training libraries. We used ImageNet as our base training data wherever it was the best fit.

Stage 3: Designing a web based portal for performing custom attacks.

In this stage, we collected few images which we are using as selective input to our models. These images are made available on a web page to select and feed to the models as input. The web portal also provides an option to adjust the perturbation amount or we can say the amount of adversary to be added to the image which leads to certain misclassification. The results returned by the images classifier is then returned to the web portal to be displayed in the defined section.

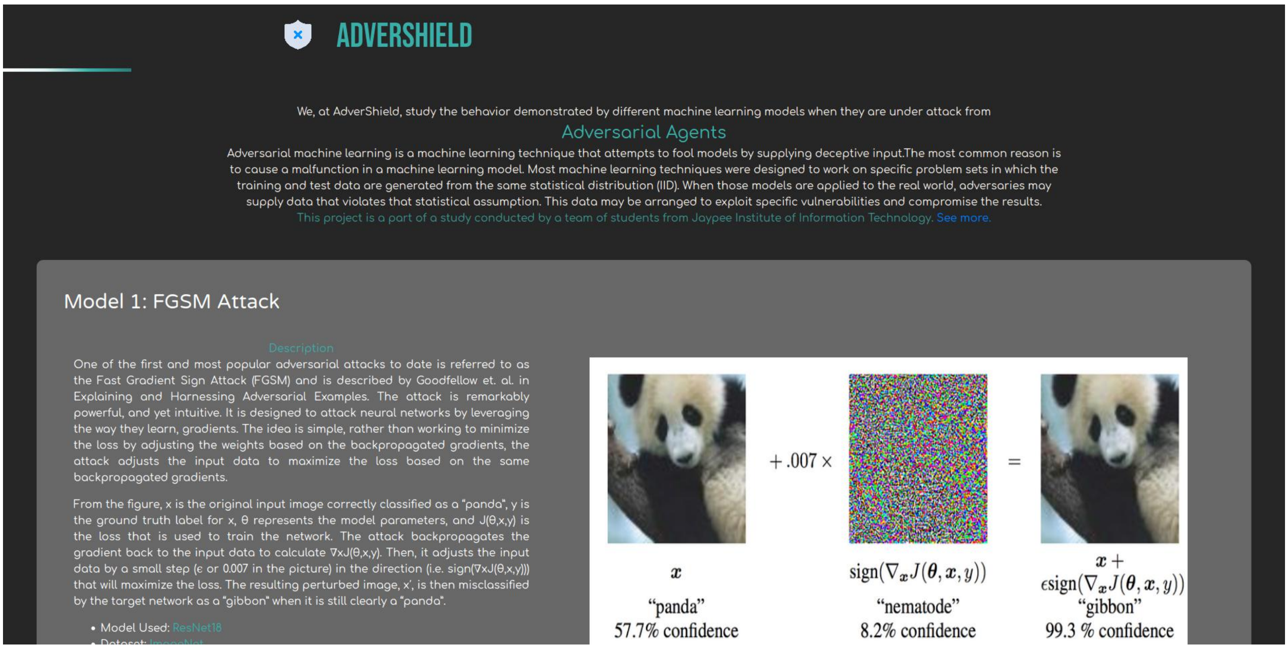


Fig. 3.3.1 Web portal landing page.

The server being a django-based implementation, can also be deployed on online IP providers with certain settings adjusted. It is an MVC-based architecture providing easy handling of the data and routes. The routes and related functions are better defined in the implementation section of the report. We created specific routes to trigger different models under different inputs and parameter. Each route received the input parameters and called the function attached to it.

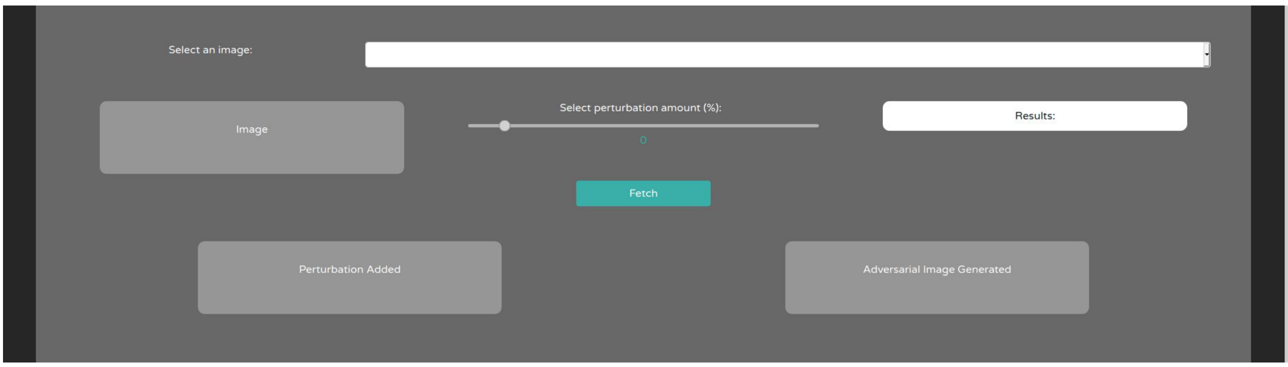


Fig 3.3.2 Calling the routes for a model.

4. MODELING AND IMPLEMENTATION DETAILS

4.1 Design Diagrams

4.1.1 Use case diagrams

Defined user cases can be better understood using the flow diagram given.

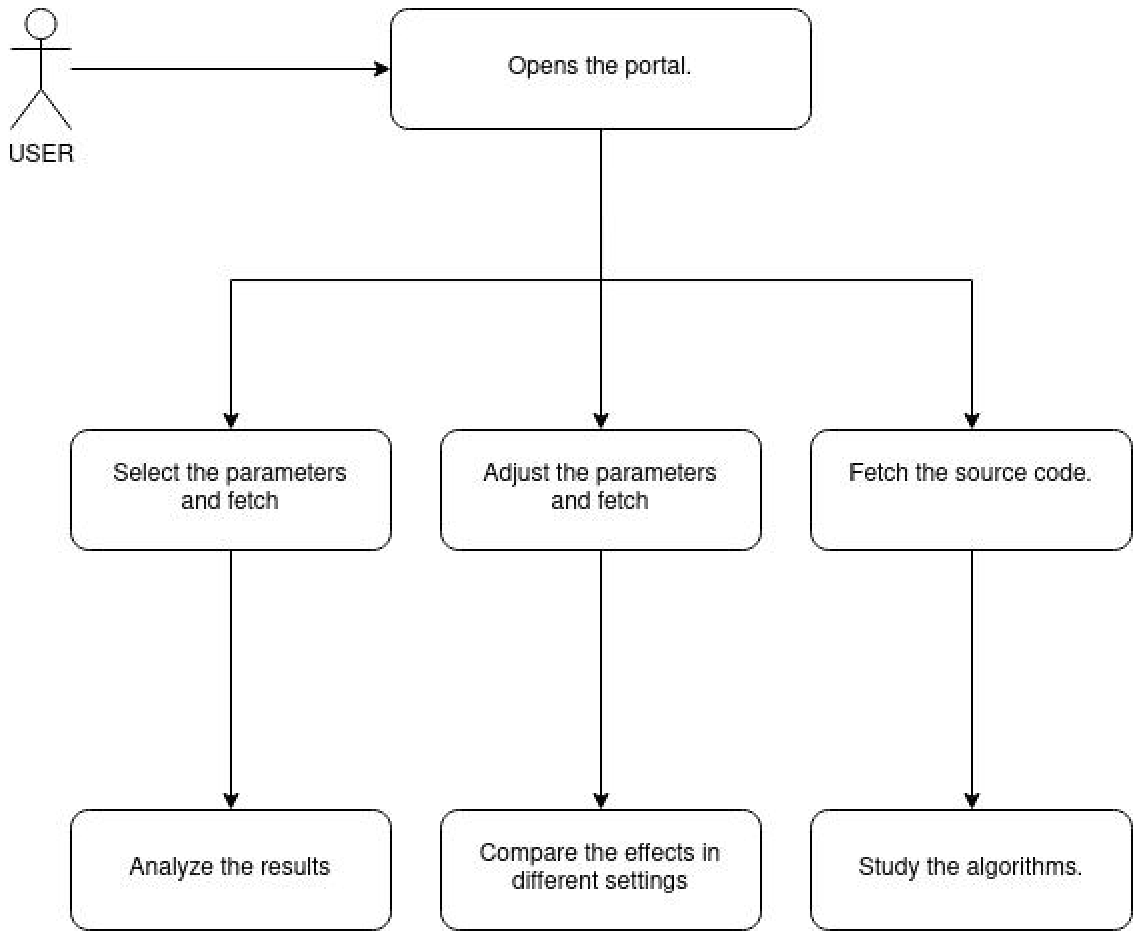


Fig 4.1.1.1 Use case diagram

4.1.2 Class diagrams / Control flow diagrams

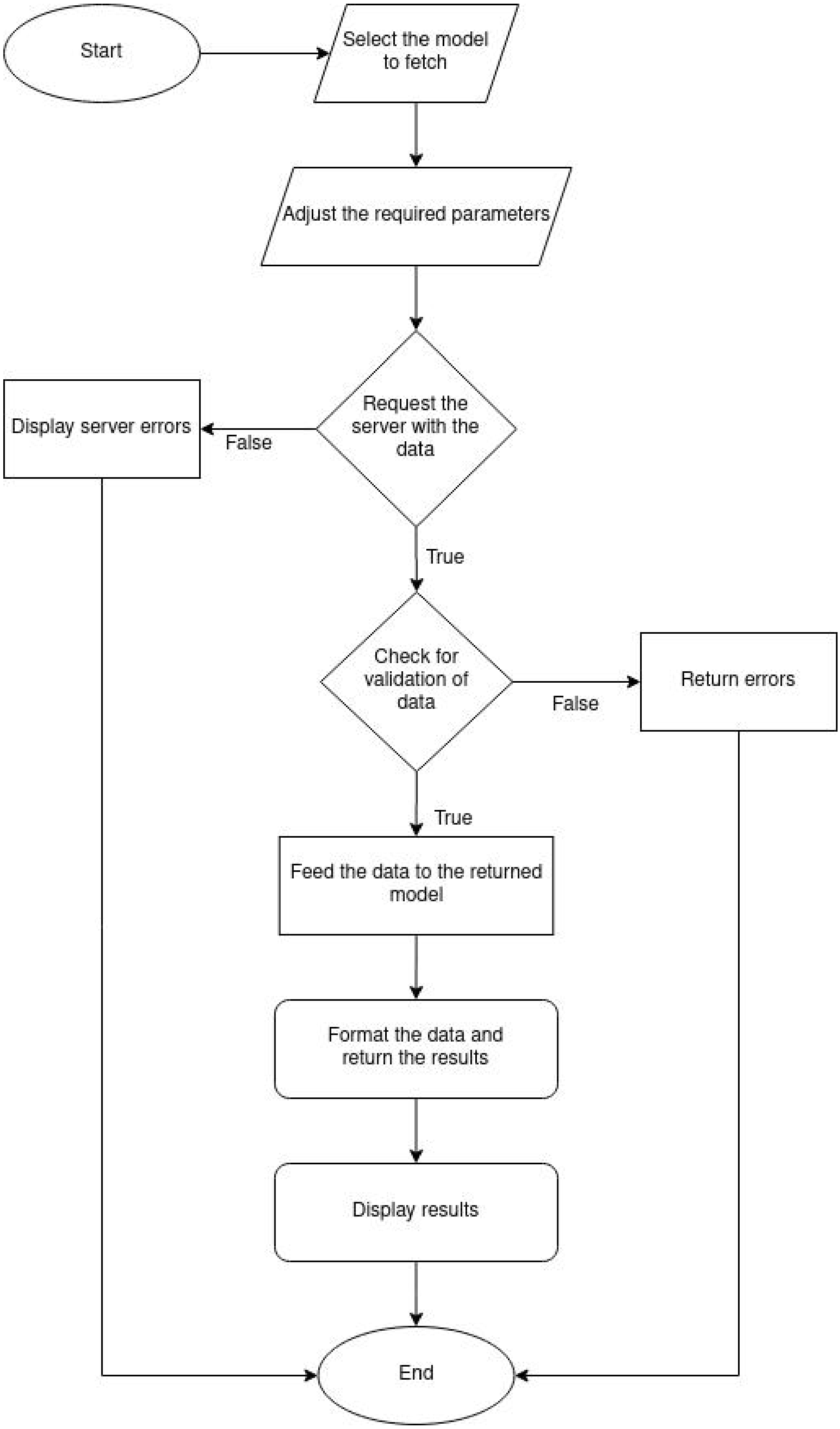


Fig 4.1.2.1 Control Diagram

4.1.3 Sequence diagram / Activity diagrams

User can perform the following activities through our system and obtain pretty defined results on the output.

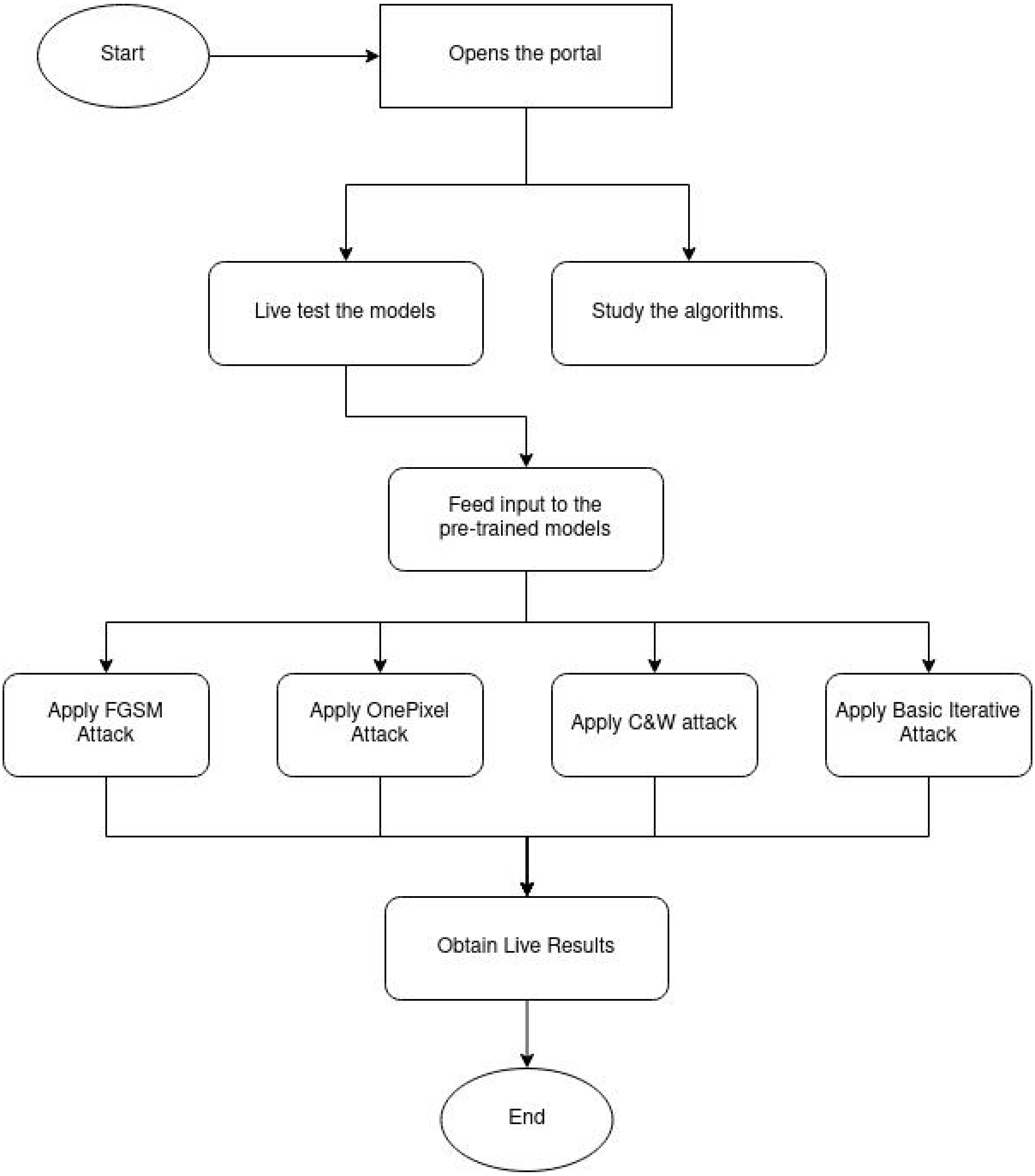


Fig 4.1.3.1 Sequence Diagram

4.2 Implementation details and issues

As mentioned earlier, the project was divided into three stages which will be described at length in this section.

Stage 1: Identifying the research work.

Adversarial attacks on machine learning models are a way to exploit the learning structure of a system and create vulnerabilities which are beyond physical detection and recovery. These vulnerabilities houses capabilities from causing a classifier to misclassify, to causing trained and tested models to malfunction at run. Several algorithms have been introduced in the past few years which have happened to generate adversarial samples for detection of these anomalies. A big amount of research showed the varying effects of adding adversaries to images and then feeding them to a classifier model.

They also showed how variations in certain parameters result into images that are far more disturbing to a classifier model then they actually were. Various parameters affect the identification of an image. Camera angles, image resolution, degree of depth, motion blur, focus, etc are few parameters that are seen to cause models to fail to classify.

Then there are different kind of techniques to employ in making a model to make it quite secure towards adversarial inputs. The increase in the level of classification increases the robustness of the model to adversarial perturbations also to noise. The study helped us to identify the following result:

a. Adversarial perturbations are nothing but high-dimensional dot products of different vectors. They are a result of models not being nonlinear.

b. The generalization across different models is caused majorly because adversarial perturbations tend to the weight vectors of a model.

c. The direction in which perturbation is a dot product with the image matters most, rather than the specific point in space.

d. Because direction matters most so the adversarial perturbations show generalization across various examples.

e. Models which are easy to optimize during training and testing are also easy to perturbate.

Keeping the above points in mind, we identified and studied four algorithms, which are described in the next stage.

Stage 2: Implementing the algorithm models.

The four algorithms elected to be implemented are described below:

1. Fast Gradient Sign Method (FGSM):

One of the first and most popular adversarial attacks to date is referred to as the Fast Gradient Sign Attack (FGSM) and is described by Goodfellow et. al. in Explaining and Harnessing Adversarial Examples. The attack is remarkably powerful, and yet intuitive. It is designed to attack neural networks by leveraging the way they learn, gradients. The idea is simple, rather than working to minimize the loss by adjusting the weights based on the back-propagated gradients, the attack adjusts the input data to maximize the loss based on the same back-propagated gradients.

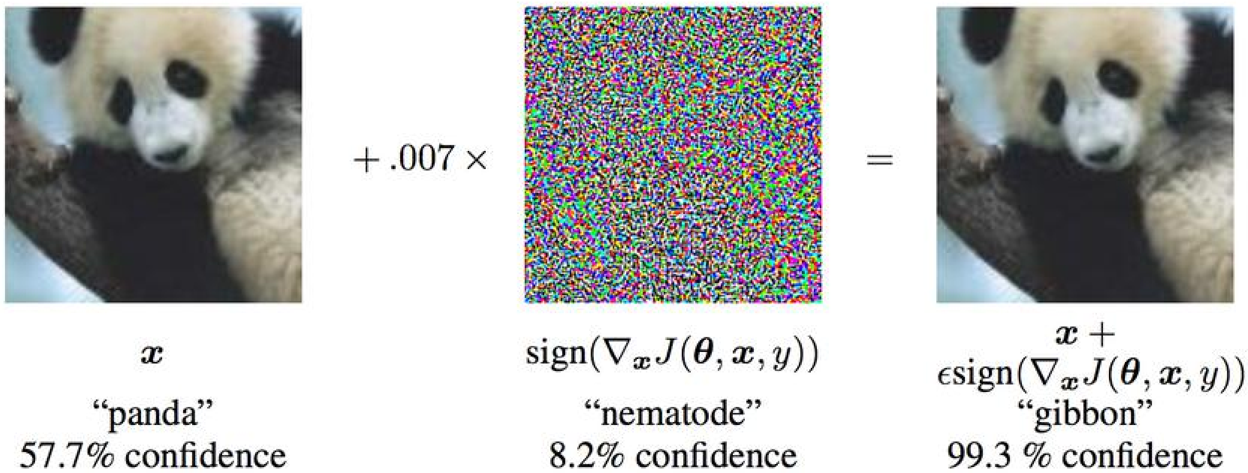


Fig. 4.2.1.1 FGSM Attack

From the figure, it is clear that image 'x' is correctly classified as 'panda' with a fairly high level of confidence. y is the ground truth label for x, represents the model parameters, and J(,x,y) is the loss that is used to train the network. The attack backpropagates the gradient back to the input data to calculate xJ(,x,y). Then, it adjusts the input data by a small step (ϵ or 0.007 in the picture) in the direction (i.e. sign(xJ(,x,y))) that will maximize the loss. The resulting perturbed image, x , is then misclassified by the target network as a "gibbon" when it is still clearly a "panda".

In the source code, the attack is implemented using python. A function call to the following methods with the required parameters returns a list of possible classes of identification provided by the model.

Model: Resnet18

Dataset: ImageNet

Function: fgsmAttack(<image_path>, <epsilon_value>)

Return: List of all the classification types.

Below shown is an image to select input image for the algorithm. For the purpose of demonstration, we select the image of a brown bear. It is feed into the model by the calling the route: 127.0.0.1:8000/fetchFGSMAttack?image_name=bear.jpg&epsilon_value=0

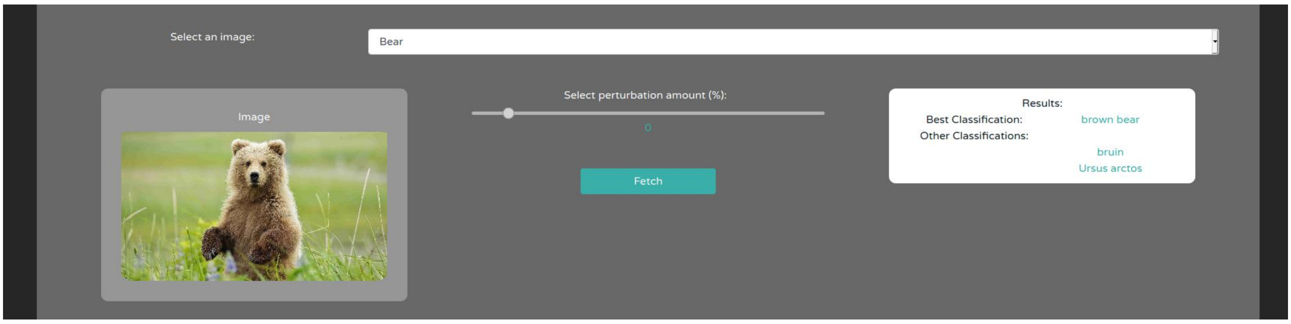


Fig 4.2.1.2 Selecting settings for FGSM Attack

The inputs are feed into the system and you can see that the best classification obtained by far is "brown bear". Other classifications are also available. Now adjusting the perturbation amount to 40 units and calling for classification, the best results are found to be "peacock".

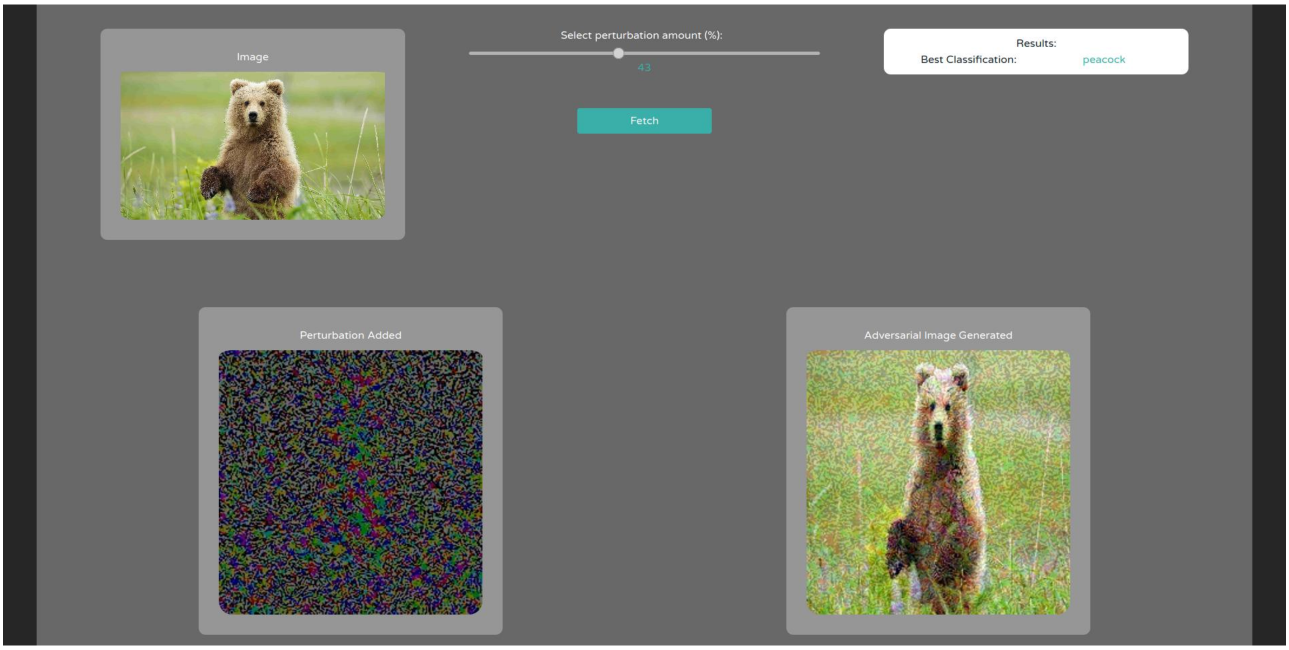


Fig 4.2.1.3 Results for FGSM Attack.

The adversarial noise which was added to the subject image shown below the results and the image generated after adding the noise is also available alongside. We can see that the adversarial image obtained can still be identified as a "bear" and there are no signs of image appearing to be "peacock", but it is specifically seen that the noise affects the ability to classify of a very well know resnet classifier. This attacks seems to serve the purpose but the noise added shows a lot of distortion, which serves as a means to create doubts at a system monitoring security. Such a disturbed image is hard to find in nature.

2. OnePixel Attack

According to research done by Jiawei et al, it turns out only one pixel is enough to achieve this for a lot of Deep Neural nets. Some images generated using this method and their predicted classes are shown below:

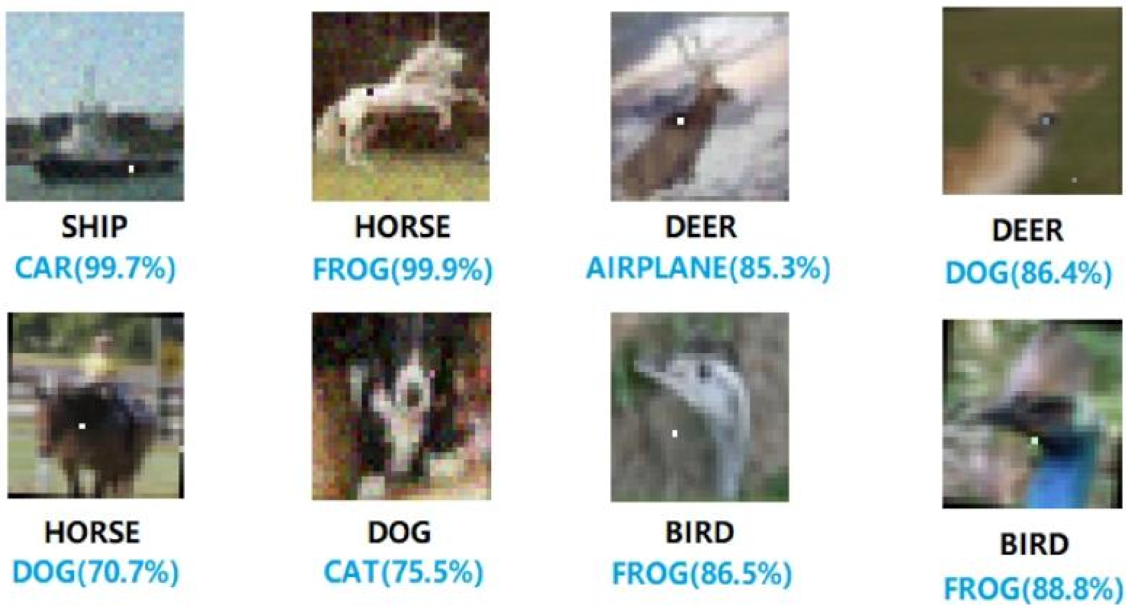


Fig 4.2.2.1 Sample getting wrongly classified

The main features that make this attack unique are :

- Effectiveness-It causes most classifiers to wrongly classify with high accuracy.
- Limited information-This method only needs access to the confidence values of the different labels given by the Neural Net(often called a semi-black box attack).
- Flexibility-Different variants of Neural Nets gets fooled by this method.

There are plenty of reasons why research like this deserves a lot of attention. Firstly, it is an extreme case of understanding the CNN input space. Secondly, it is tremendously effective at hiding adversarial changes as a small number of pixels are altered and hence completely imperceptible to the human eye. This one pixel attack can potentially be extended to domains like Natural Language Processing, Speech Recognition etc.

Model: BasicCNN

Dataset: Sample of 10 image classes from CIFAR

Function: onePixelAttackUtil2(<image_path>, <number of pixels>)

Return: Classification and percentage.

Below shown is an image to select input image for the algorithm. For the purpose of demonstration, we select the image of a dog. It is feed into the model by the calling the route: 127.0.0.1:8000/fetchOnePixelAttack?image_name=dog.jpg&epsilon_value=90

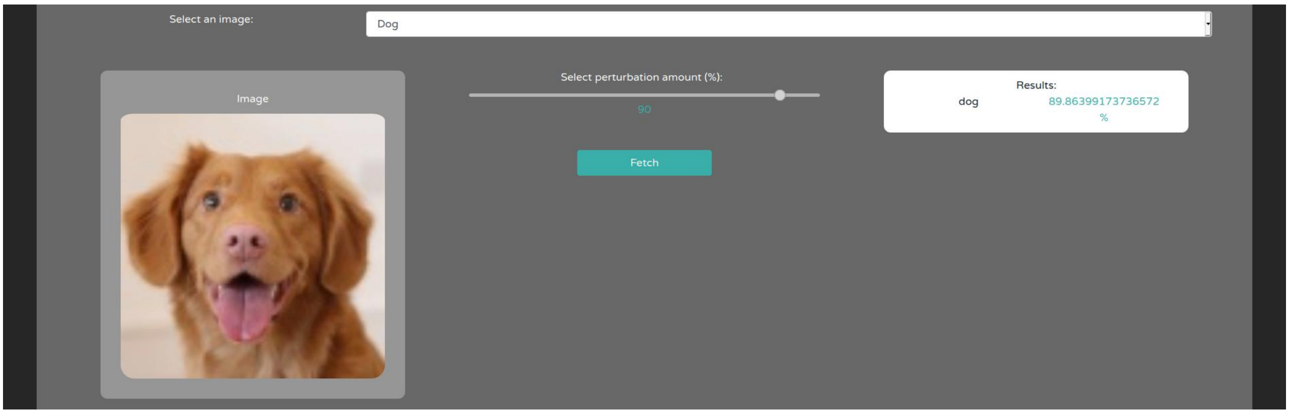


Fig 4.2.2.2 Selecting settings for One Pixel Attack

The inputs are feed into the system and you can see that the best classification obtained by far is “dog” with a confidence value 89.86399 %. Now adjusting the perturbation amount to 90 units, that means identifying 10 pixels which are enough to misclassify the image, and calling for classification, the best results are found to be “frog” with confidence level of 96.43864 % .

Fig 4.2.2.3 Results for OnePixel Attack.

The adversarial noise in the form of colored pixels which were added to the subject image shown below the results. We can see that the adversarial image obtained can still be identified as a “dog” and there are no signs of image appearing to be “frog”, but it is specifically seen that the noise affects the ability to classify of a very well know BasicCNN classifier. This attacks seems to serve the purpose and the noise added shows bit less of distortion, which serves which is hard to doubt as such distortions are possible in data transfer.

3. C&W Attack

The Carlini-Wagner attack (2016) is a regularization-based attack with some critical modifications which can resolve the unboundedness issue.

The CW attack algorithm is a very typical adversarial attack, which utilizes two separate losses:

- An adversarial loss to make the generated image actually adversarial, i.e., is capable of fooling image classifiers.
- An image distance loss to constraint the quality of the adversarial examples so as not to make the perturbation too obvious to the naked eye.

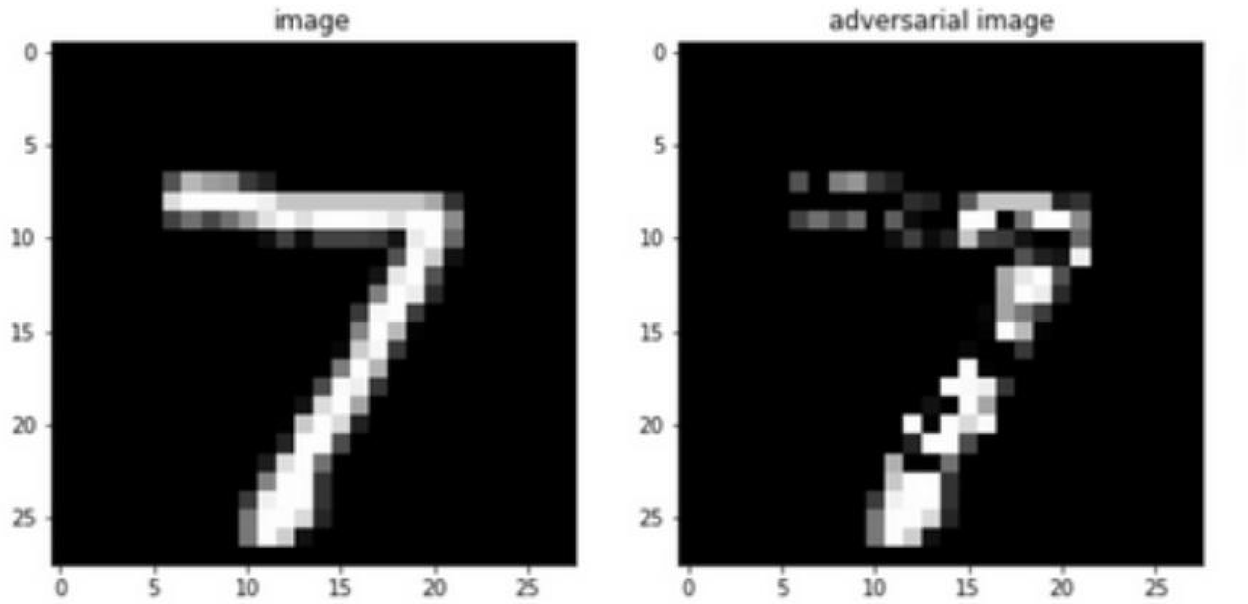


Fig 4.2.3.1 Distortion created by C&W Attack

CW finds the adversarial instance by finding the smallest noise δ added to an image x that will change the classification to a class t. When adversarial examples were first discovered in 2013, the optimization problem to craft adversarial examples was formulated as:minimize: $D(x,x+\delta)$ such that: $C(x+\delta)=t$ (Constraint 1) and $x+\delta$ E $[0,1]^n$ (Constraint 2) where:

- x is the input image, δ is the perturbation, n is the dimension of the image and t is the target class.
- Function D serves as the distance metric between the adversarial and the real image, and function C is the classifier function.

Model: InceptionV3

Dataset: ImageNet

Function: cwAttackUtil2(<image_path>, <iterations>)

Return: Classification

Below shown is an image to select input image for the algorithm. For the purpose of demonstration, we select the image of a bear. It is feed into the model by the calling the route: 127.0.0.1:8000/fetchCWAttack?image_name=bear.jpg&epsilon_value=5

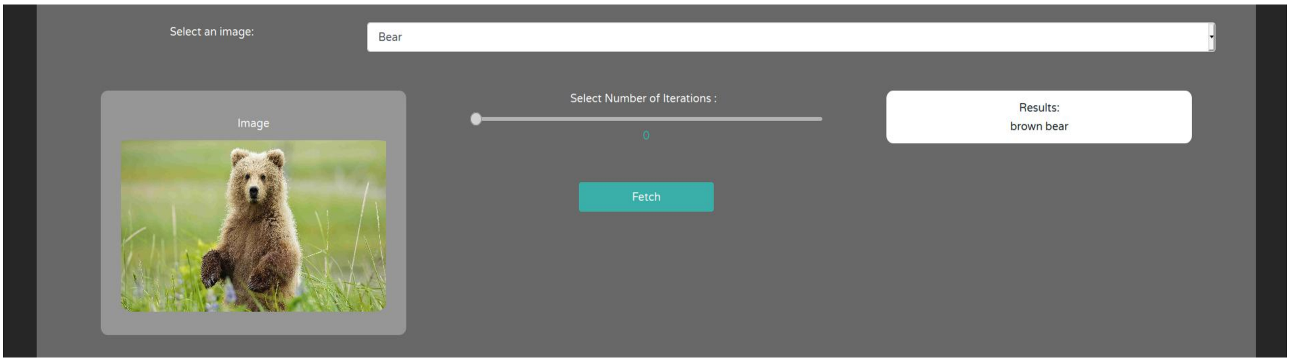


Fig 4.2.3.2 Selecting settings for CW Attack

The inputs are feed into the system and you can see that the best classification obtained by far is “bear” . Now adjusting the perturbation amount to 10 iterations units, that means performing 10 iterations of distortion to misclassify the image, and calling for classification, the best results are found to be “West Highland white terrier".

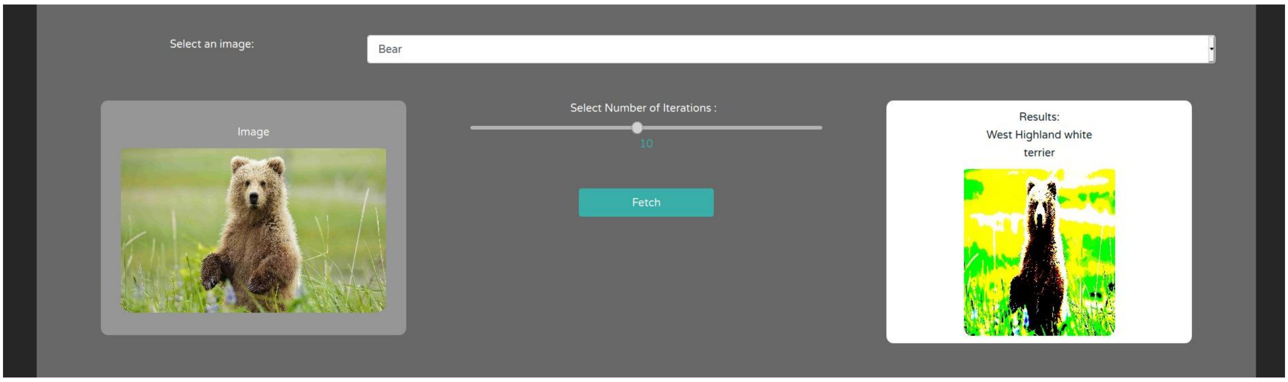


Fig 4.2.3.3 Results for OnePixel Attack.

The adversarial noise in the form of contrast which is added to the subject image is shown below the results. We can see that the adversarial image obtained can still be identified as a “dark colored bear” and there are no signs of image appearing to be “terrier”, but it is specifically seen that the noise affects the ability to classify of a very well know InceptionV3 classifier. This attack seems to serve the purpose and the noise added shows a lot of distortion, which can create alerts about an adverser causing malfunctioning.

4. Basic Iterative Method

An extension of FGSM, referred to as the Basic Iterative Method (BIM), repeatedly adds small perturbations and allows targeted attacks. Moosavi-Dezfooli et al. linearize the classifier and compute smaller perturbations that result in untargeted attacks. We rely on BIM as the method of choice for attacks based on images, because it allows robust targeted attacks with results that are classified with arbitrarily high certainty, even though it is easy to implement and efficient to execute.

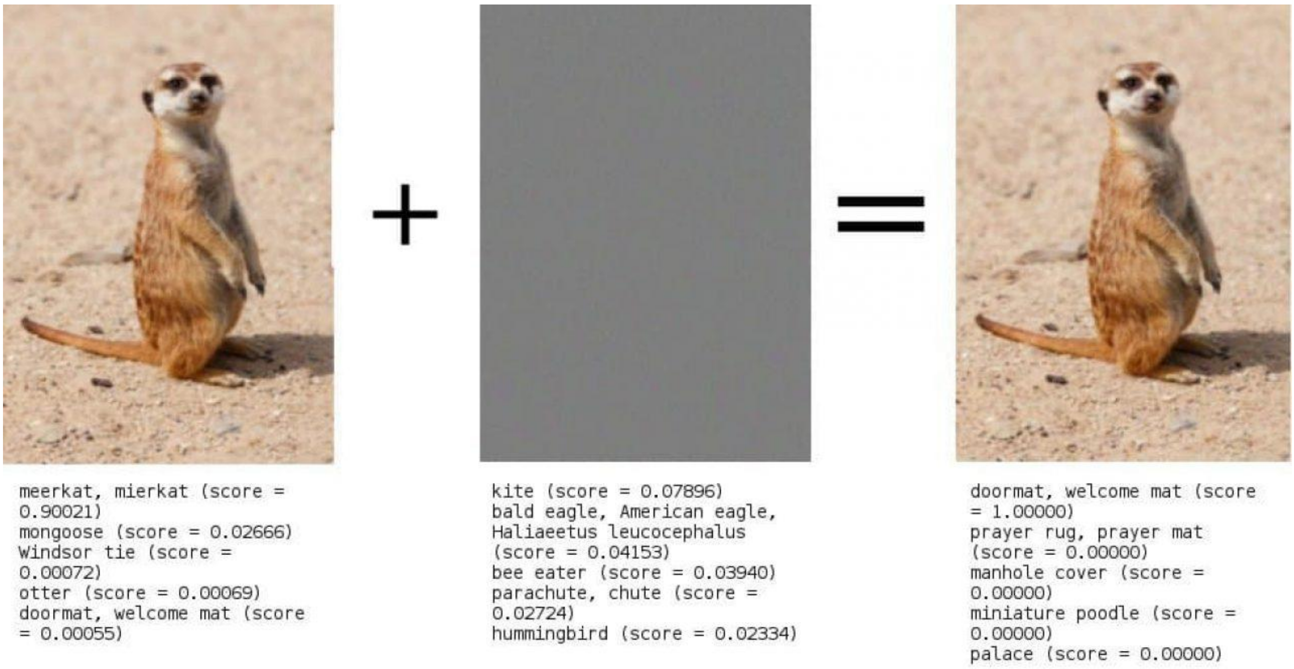


Fig 4.2.4.1 Basic Iterative Attack on image of a meerkat

It ensures targeted attacks are visually imperceptible, based on the observation that attacks do not need to be applied homogeneously across the input image and that humans struggle to notice artifacts in image regions of high local complexity. Such attacks, in particular, do not change saccades as severely as generic attacks, and so humans perceive the original image and the modified one as very similar. Repetitive generation of perturbation image results into a such smoother and much less observable distortion.

Model: resnet18

Dataset: ImageNet

Function: iterativeAttack(<image_path>,<epsilon_value>, <number of iterations>)

Return: Classification

Below shown is an image to select input image for the algorithm. For the purpose of demonstration, we select the image of a bear. It is feed into the model by the calling the route: 127.0.0.1:8000/fetchBIAttack?image_name=bear.jpg&epsilon_value=50 and iterations_count=5

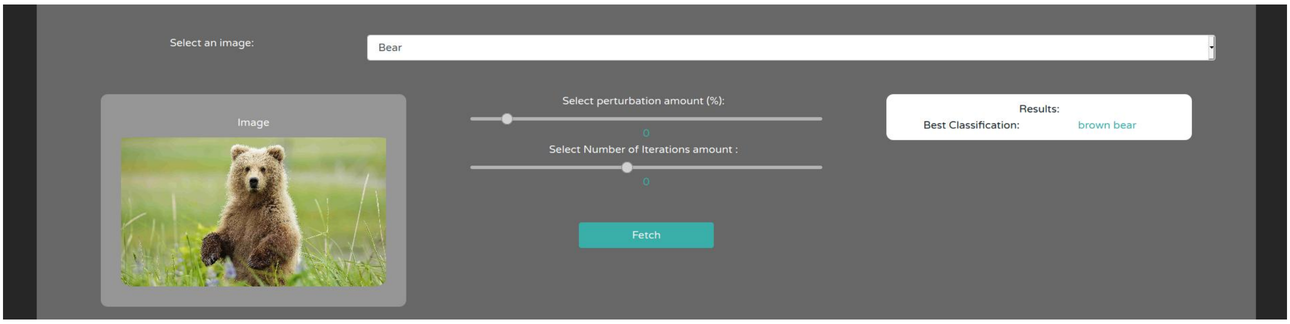


Fig 4.2.4.2 Selecting settings for BIM Attack

The inputs are feed into the system and you can see that the best classification obtained by far is “bear” . Now adjusting the perturbation amount to 6 iterations units and epsilon value to 50%, that means performing 5 iterations of distortion on the noise to smoothen it and then add to the subject image to misclassify the image, and calling for classification, the best results are found to be “West Highland white terrier”.

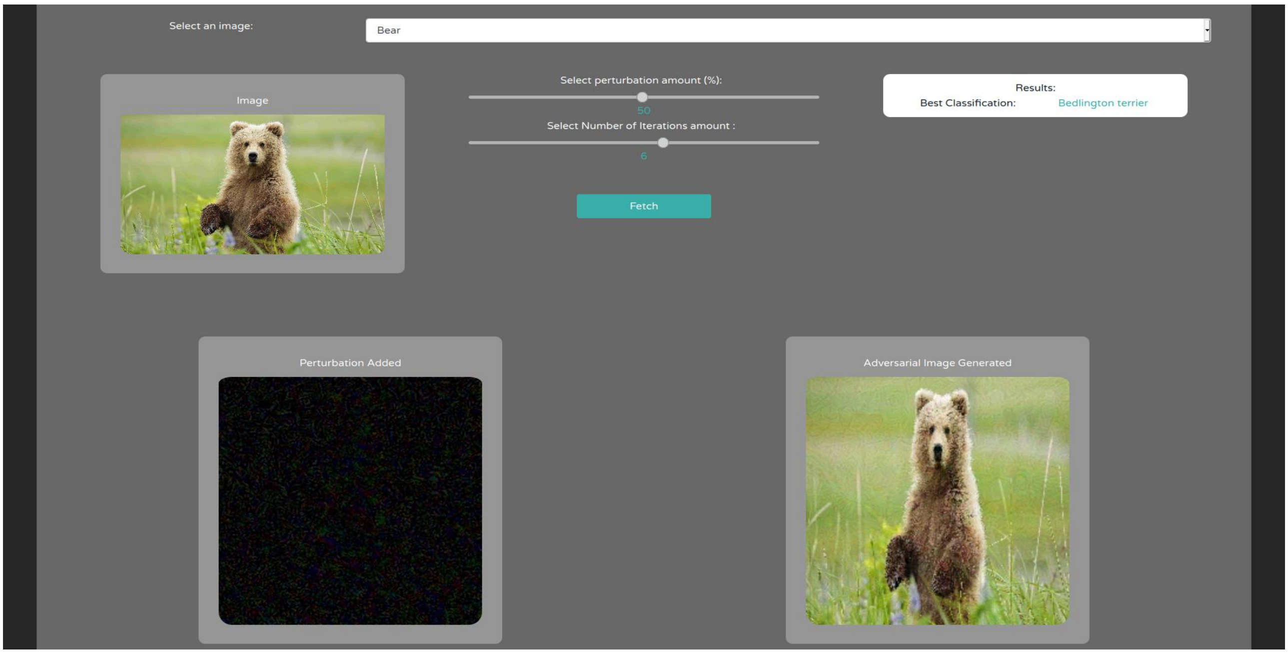


Fig 4.2.4.3 Results of BIM Attack

The smoothness of adversarial noise, which is added to the subject image is shown alongside. We can see that the adversarial image obtained can still be identified as a “dark colored bear” and there are no signs of image appearing to be “terrier”, but it is specifically seen that the noise affects the ability to classify of a very well know resnet18 classifier. This attack seems to serve the purpose and the noise added shows very less of distortion, which goes undetected.

Stage 3: Designing a web based portal for performing custom attacks.

The above shown results are snapshots from the web portal implemented in stage 3 of the project.

- Frontend is made using Bootstrap5, HTML, CSS.
- Modals are used to show server messages.
- Server is made in python using Django framework.
- Models are implemented using Pytorch library and called by importing function calls..
- Seperate set of images are available on both frontend and backend side of the data to disable false parameter from reaching the model.

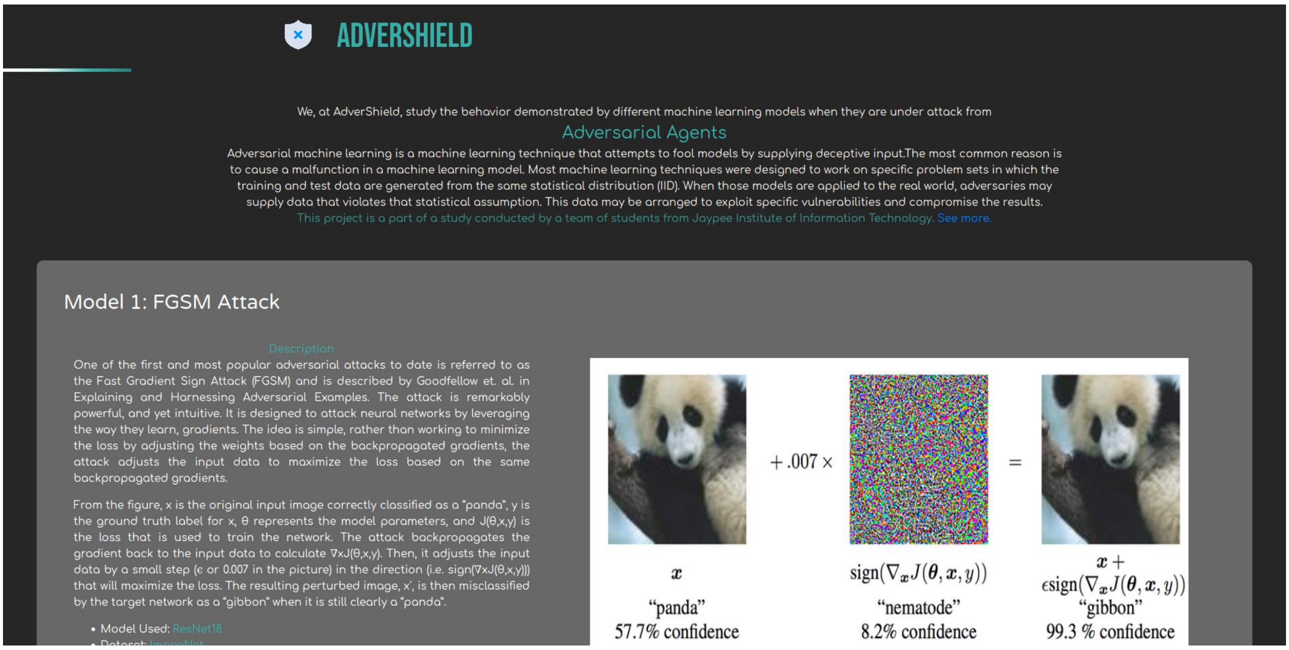


Fig 4.2.5 Web portal design

Steps to follow to run the project.

1. Clone the project from the following link.

   Git Repo: https://github.com/chitrank0614/Major-AMLAttacks.git

2. Open the terminal in the corresponding directory and install the requirements for the project using pip. (Python3 is a prerequisite for the project). The following command shall to the job.

   python3 -m pip install -r requirements.txt

3. Run the server using django-admin using the following command.

   python3 manage.py runserver

4. Django server will start running on your localhost at port 8000. Reach for the web portal from:

   Localhost: http://127.0.0.1:8000/

5. Scroll to the model you want to test, select the image from the dropdown, the image will appear in the provided space alongside.

6. Set the required parameters and "Fetch". Corresponding results will be displayed alongside.

4.3 Risk analysis and mitigation

Table 4.3.1: Risk Analysis

Risk_ID	Classification	Description of Risk	Risk Area	Impact
Risk_1	Design	The possibility of low accuracy as we are using traditional machine learning algorithms.	Performance	High (H)
Risk_2	Engineering Specialties	The project scope demands maximum possible reliability on the predicted outcomes, as the lives of patients are at risk	Reliability	Medium (M)
Risk_3	Requirements	Risk of availability of complete, robust and reliable dataset with proper labels for training our models.	Completeness	Low (L)

Table 4.3.2: Risk Area Wise Total Weighting Factor

S.No	Risk Area	Weights (In+Out)	Total Weights	Priority
1	Performance	9+3+3+1	16	1
2	Budget	9+3+1	13	2
3	Hardware Constraints	9+3+1	13	3
4	Reliability	9+3	12	4
5	Requirements	3+1	4	5

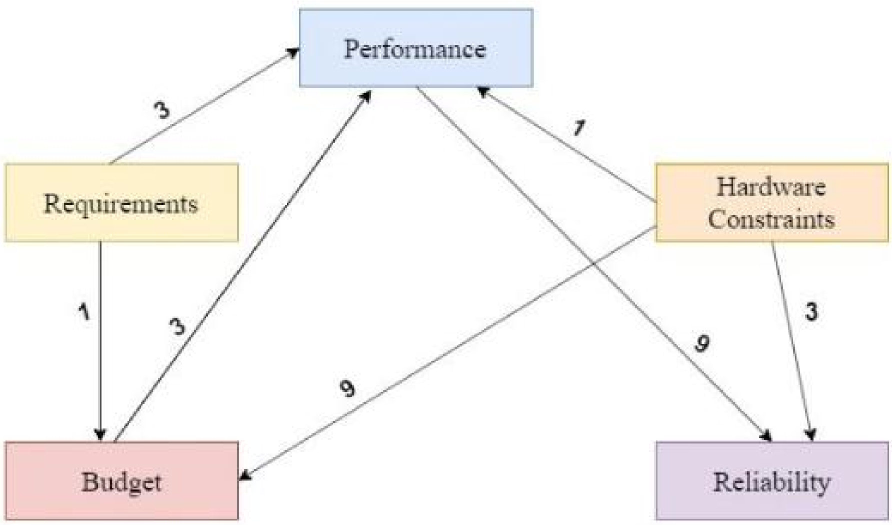


Fig. 4.3.1. Weighted Interrelationship Graph

5. TESTING

Software testing is an important phase in the software development life cycle as it verifies and validates the system under test i.e. whether it works as expected and satisfies the stakeholders' needs. With respect to the text extraction system also, testing & evaluation is significant; as it is important to test the system before deployment. In order to assess the system output, appropriate quality assessment techniques should be adopted for determining the system performance in comparison to the benchmark level or with the quality of the previous version or with similar kinds of different products.

5.1 Testing plan

First of all we tested the models with the few images whose identification were already known to us. Since we are using pre-trained model and the subject of our study is verifying adversarial attacks, we checked in with the quality of image received and at what kind of images the systems works pretty fine and purposefully.

5.2 Component decomposition and type of testing required

The objectives behind the testing of our developed model are:

- Evaluation of Parameters of the developed system
- Calculating accuracy
- Speed of the model
- Evaluation of Complexity in colored images
- User Level Testing

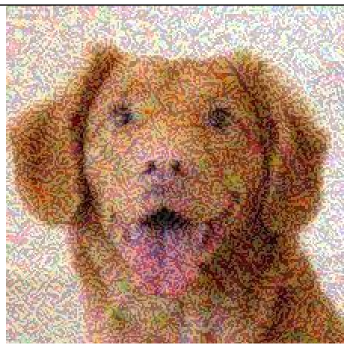
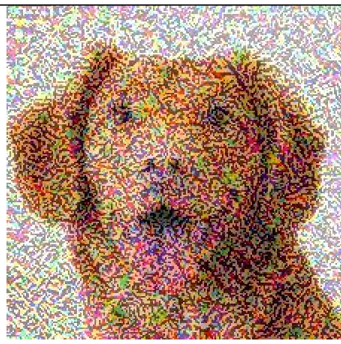
Table 5.2.1. Types of testing

Type of tests	Explanation	Software Component
Requirement Testing	Validation checks were made to ensure that hardware and software specifications meet the minimum requirements. Certain libraries such as	VS Code/Anaconda

	Pytorch, OpenCV were required to be specially installed and the minimum CPU/GPU requirements for our architecture were also checked.	
Performance Testing	Performance testing is the process of determining the speed, accuracy, and consistency of the proposed model. This was achieved by creating, training, and testing the whole image processing based learning system experimenting with varied training methodologies.	VS Code/Anaconda
Experimental Testing	Our model was checked against various experimental tests to fine-tune the hyperparameters in order to ensure the best results. Hardware specification was improved and the number of epochs was increased to improve the generation of adversarial images.	VS Code/Anaconda
Unit Testing	The purpose is to validate that each unit of the software performs as designed. The output of the steps within data preprocessing and the result of tumor segmentation was randomly tested in order to ensure valid and consistent results.	VS Code/Anaconda

5.3 List all test cases in prescribed format

Table 5.3.1. List of sample test cases

InputID	Input Image	Run at 50% distortion	Run at 100% distortion
Model 1	 Brittany Dog	 Teddy Bear	 Bubble
Model 2	 Dog (89.86%)	 Frog (99.98%)	 Frog (99.99%)
Model 3	 Chesapeake Bay Retriever	 Teddy	 Labrador retriever
Model 4	 Brittany Dog	 Chesapeake Bay Retriever	 Chesapeake Bay Retriever

5.4 Error and Exception Handling

Being a pretrained model on the defined classes, the program did not required heavy exception handling. In cases of error and exception, certain keywords were returned in the response to the client request which helped identify the type of error and display it on the user’s window after appropriate formatting. Few defined error cases were server failure, image not found, image class not detected, data out of order, etc.

5.5 Limitations of the solution

Presently, the solution set is limited to few classes due to the vulnerability of misclassification and the cause of project being demonstration. The models used are trained on data from around the world and hence they are capable enough to identify all kinds of classes of data. The Imagenet data identification model is used globally to serve with good quality image classification and hence is well managed and well trained data set.

For the purpose of keeping similar dataset for all the models, the testing got limited to few images and hence reduced the option to take in image from any source. The adversarial examples are not tested on all the possible set of classes. Along with this, the adversarial models also behave differently for different types of input data. High resolution images and pictures have a very low chance of getting misclassified but still there is ample scope for comparison between various methods. Also, it is seen that results of a certain kind of distortion remains same for the complete scale of distortion. For example, dog is classified as frog by model 2 at 50% distortion. This classification does not change at 100% distortion either. Looking in for the probability, it is seen that as distortion increases, false classification increases certainly. But is does not obeys every time.

6. FINDINGS, CONSLUSION AND FUTURE WORK

6.1 Findings

From the above study, we learned about the adversarial networks and their working. How they hamper the efficiency of an image classifier and how it is harmful on physical scale. We had the following observation after completing the study on various topics and research papers related to the former subject.

- We were able to understand the logics behind these adversarial vulnerabilities
- We were successfully able to implement four very important ideologies from the field.
- We were able to provide a tool that can be used to significantly understand the effect of adversarial vulnerabilities on image classification.
- We were able to extract the perturbation out of the image for displaying to enhance the understanding of model's working.

6.2 Conclusion

- We learned about the various algorithms which are expected to get replaced by another research topics.
- Various methods describing ideas to prevent these attacks have been discussed and it has been found that majority of the ideas focused on training the training models with all kinds of adversarial sample subjects.
- Few studies have show how the adversarial models hamper the performance of google cloud API and other real world settings.
- Basic implementation of black-box attacks have been perfectly defined in few of the researches.
- We got to know about the future scope of this field of research.
- After implementing the models and testing them on the same image and dataset. We are able to state that One pixel attack is much better attack as it involves minimal distortion, provided the hardware requirements are met.
- Also, we were able to identify that Model4: Basic Iterative Method has been the best at performance as the distortion created was minimal and hardly intriguing. Also, the image generated was tough at comparison and it provided a fairly large set of input parameters.

6.3 Future work

The principles of adversarial networks have tremendous application on both online and real-world deployment. It is possible to apply adversarial perturbation to real-world objects and that can be a new source of study and research.

Few real world services like speech recognition can also be targeted for research under the adversarial research category. A simple model based on iteration that can modify the input at random to hamper the classification capabilities of a classifier is still an area to explore. APE-GAN research suggests that implementing various defense mechanism together to develop layered prevention can be a direction for research in future. Research by Guo et al. suggest that their observation on simple black box attacks defining a new type of attacks can be string baseline for future work and references. The efficiency and application provides a strong basis to implement various new ideas.

Different kinds of attacks and vulnerabilities appearing everyday requires a ready to go defense mechanism for ensure security. Various researches have come up with different kinds of adaptation of former researches showing potential to be applied to a wide range of applications. Few studies have show how the adversarial models hamper the performance of google cloud API and other real world settings. On the other hand, various methods describing ideas to prevent these attacks have been discussed and it has been found that majority of the ideas focused on training the training models with all kinds of adversarial sample subjects.

It seems like the field of adversarial study is a big game of for and against researches.There is a lot of scope of deployment and research in this area of science.

REFERENCES

[1] Patrick McDaniel, Nicolas Papernot, and Z. Berkay Celik (2016). Machine Learning in Adversarial Settings. IEEE Security & Privacy, May/June 2016.

[2] Alexey Kurakin, Ian J. Goodfellow and Samy Bengio. (2017). Adversarial Machine Learning at Scale. ICLR Conference, 2017

[3] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K. Reiter. (2016). Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition, CCS'16, Vienna, Austria

[4] Alhussein Fawzi, Omar Fawzi, Pascal Frossard (2015), Fundamental limits on adversarial robustness, ICML 2015 Workshop on Deep Learning, Lille, France.

[5] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, Aleksander Madry, (2019), Adversarial Examples are not Bugs, they are Features, 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)

[6] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, Pascal Frossard, (2017), Universal adversarial perturbations, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017

[7] Battista Biggio, Blaine Nelson, Pavel Laskov, (2019), Poisoning Attacks against Support Vector Machines, 29th International Conference on Machine Learning, Edinburgh, 2013

[8] Battista Biggio, Davide Maiorca, Igino Corona, Nedim Srndic, Blaine Nelson, Pavel Laskov, Giorgio Giacinto, and Fabio Roli, (2013), Evasion attacks against machine learning at test time, ECML PKDD 2013, Part III, LNAI 8190, pp. 387–402, 2013

[9] Nicholas Carlini, David Wagner, (2017), Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods, AISec'17.

[10] Alhussein Fawzi ,Hamza Fawzi, Omar Fawzi (2018), Adversarial vulnerability for any classifier, 32nd Conference on Neural Information Processing Systems (NeurIPS 2018).

[11] Zhengli Zhao,Dheeru Dua, Sameer Singh, (2018), Generating Natural Adversarial Examples, ICLR 2018.

[12] Shuangtao Li , Yuanke Chen, Yanlin Peng, Lin Bai, (2018), Learning More Robust Features with Adversarial, AISec'18.

[13] Nicolas Ford, Justin Gilmer, Nicholas Carlini, Ekin D. Cubuk, (2019), Adversarial Examples Are a Natural Consequence of Test Error in Noise, Proceedings of the 36th International Conference on Machine Learning, 2019.

[14] Ali Shafahi, W. Ronny Huang, Christoph Studer, Soheil Feizi, Tom Goldstein, (2019), Are adversarial examples inevitable, ICLR 2019 Conference

[15] Adi Shamir , Itay Safran , Eyal Ronen and Orr Dunkelman, (2019), A Simple Explanation for the Existence of Adversarial Examples with Small Hamming Distance.

[16] Jiajun Lu, Hussein Sibai, Evan Fabry, and David Forsyth. (2017). Standard detectors aren't (currently) fooled by physical adversarial stop signs. University of Illinois at Urbana Champai

[17]Xiaoyong Yuan, Pan He, Qile Zhu, Xiaolin Li, (2019), Adversarial Examples: Attacks and Defenses for Deep Learning, IEEE Transactions on Neural Networks and Learning Systems, 2019.

[18] Nupur Thakur, Yuzhen Ding , Baoxin Li, (2020), Evaluating a Simple Retraining Strategy as a Defense Against Adversarial Attacks, Arxiv.org, 2020.

[19] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, Jun Zhu, (2018), Defense against Adversarial Attacks Using High-Level Representation Guided Denoiser, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018

[20] Shiwei Shen, Guoqing Jin, Ke Gao, Yongdong Zhang, (2019), APE-GAN: Adversarial Perturbation Elimination with GAN, ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

[21] Kimin Lee, Kibok Lee, Honglak Lee, Jinwoo Shin, (2018), A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks, Advances in Neural Information Processing Systems 31 (NeurIPS 2018)

[22] Jiajun Lu, Hussein Sibai, Evan Fabry, and David Forsyth. (2017). No Need to Worry about Adversarial Examples in Object Detection in Autonomous Vehicles. University of Illinois at Urbana Champaign

[23] Ian J. Goodfellow, Jonathon Shlens & Christian Szegedy (2015), Explaining and harnessing adversarial examples, ICLR 2015 Conference.

[24] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok, (2018), Synthesizing Robust Adversarial Examples, 35th International Conference on Machine Learning, Stockholm, Sweden, PMLR 80.

[25] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song, (2018), Robust Physical-World Attacks on Deep Learning Visual Classification, CVPR 2018 (Conference on Computer Vision and Pattern Recognition)

[26] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, Ananthram Swami, (2017), Practical Black-Box Attacks against Machine Learning, ASIA CCS ’17

[27] Moustapha Cisse, Piotr Bojanowski, Edouard Grave, Yann Dauphin, Nicolas Usunier, (2017), Parseval Networks: Improving Robustness to Adversarial Examples, ICML’17, Proceedings of the 34th International Conference on Machine Learning,

[28] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, Jianguo Li, (2018), Boosting Adversarial Attacks With Momentum, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018,

[29] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, Pieter Abbeel, (2017), Adversarial Attacks on Neural Network Policies, Conference at ICLR 2017.

[30] Chuan Guo, Jacob R. Gardner, Yurong You, Andrew Gordon Wilson, Kilian Q. Weinberger, (2019), Simple Black-box Adversarial Attacks, ICML Conference, 2019.

Last Updated on 15th October 2020

Piyush Gupta

piyushgupta3214@gmail.com || 7300677865

EDUCATION

JAYPEE INSTIUTE OF IT
B. Tech. in Computer Science
2017-2021 | Noida
CGPA: 8.4

HIGHER SECONDARY
Divine Educational Institute
2013-2016 | Bulandshahr,UP
Percentage : 92%

LINKS

Github:// **P1yu5hgupta**
LinkedIn:// **piyush-gupta-**
Codeforces:// **piyush_gupta_**
Codechef:// **piyush_gupta_**
HackerRank:// **piyush_gupta_**

COURSEWORK

UNDERGRADUATE
Operating Systems
Computer Networks
Database Management System
Data Structures
Algorithms and Problem Solving
Fuzzy Logic & Neural Networks

SKILLS

Languages:
• C++
• Python (Basics)
• JavaScript
• HTML5 • CSS3

Frameworks/ Libraries:
• ReactJS
• Angular
• ExpressJS
• Bootstrap4

Databases:
• MySQL
• MongoDB

EXPERIENCE

TATA CONSULTANCY SERVICES | Software Developer Intern
July 2020 - September 2020
- Worked on Project named as Pension Dashboard.
- Designed a Dashboard which provide a consolidated view of all the Pension Pots that users can have in their working life.
- Used Angular with Bootstrap and ExpressJs for implementing RestAPIs as Backend Services.

PROJECTS

AGRICULTURE ANALYZER
March 2020 – April 2020
- Favourable Crop Prediction on the basis of Soil Nutrients (Nitrogen, Potassium, Phosphorus, pH value) and climate.
- Crop Disease Detection and Fertilizer Recommendation
- Skills Utilized: Python, Machine Learning, Flask, KNN Algo

APPLICATION OF FACIAL EXPRESSION DETECTION
Oct 2019 – Nov 2019
- Purpose of the project is to be implemented in primary schools where we can detect the Real time Emotions of the Kids and be ready to take care of the problem that any kid is going through.
- Skills Utilized: Python, OpenCV , Machine Learning, MERN Stack

TEXT-IN-IMAGE ENCRYPTION
Oct 2019 – Nov 2019
- Image Encryption is done along with watermarking which means data is two level secured.
- Random key is generated at the time of image encryption , so no one can think of the stealing the key and decrypt the data.
- Socket Programming is used for the connection of server and client for exchange of the information among peers.
- Skill Utilized: C++, OpenCV, Cryptography, Socket Programming

ACHIEVEMENTS

- Codeforces: Expert Programmer | Highest Rating: 1758
- Codechef: 5 star Programmer | Highest Rating: 2001
- HackerRank: Gained 3900+ points in Algorithms and Data Structures(Practice)
- HackerRank: Advanced Problem Solving Certified Programmer **[->]**
- Winner at HackBMU 2020 (Hackathon) at Munjal University , Gurugram
  Project: Smart Irrigation System + Cattle Detection in Farm **[->]**
- Secured 227th rank (Out of 200,000 participants) in Code Gladiator Open Round 2020 **[->]**
- Reserved my place in Code Gladiator Finale.
- Exhibit a project in 11th International Conference on Contemporary Computing (August 2018)**[->]**
  Project: Smart Voting Machine (Purpose is to avoid multiple vote from same ID)

Chitrank Mishra

Undergraduate, Bachelor Of Technology

Versatile Computer Science student experienced in working with people at multiple levels. Freestyle competitive programmer and web development enthusiast. Organized 10+ events in college. Aiming to utilize my problem-solving, designing, and management skills for the betterment of the community.

chitrankmishra020@gmail.com

+91 9818693448

Noida, India

chitrank0614.github.io/portfolio

linkedin.com/in/chitrank_0614

github.com/chitrank0614

EDUCATION

Bachelor Of Technology, Computer Science
Jaypee Institute Of Information Technology
07/2017 - Present *CGPA: 8.9*

10+2, Maths and Science
Jai Academy, Affliated to CBSE Board
04/2015 - 03/2016 *Percentage: 95%*

PERSONAL PROJECTS

Machine Dialect Analysis (05/2020 - 06/2020)
- Made a web portal for displaying comparison charts between various online MT systems like Google Translate, Bing Translate, etc.
- Used Flask and Heroku with Python to calculate the efficiency by comparison with dictionary translation of sentences using Damerau–Levenshtein algorithm.

Smart AgroTech (10/2019 - 01/2020)
- Hackathon Winning Project at HackBMU.
- Smart irrigation system using weather predictions and farm security using CNN.
- Remote control of farm machinery using local server based Arduino logic.

Smart Lens (02/2019 - 03/2019)
- Participated in ICSC Conference 2019 Project Exhibition, Most Innovative Project.
- Arduino based device that displays notification from phone on user's spectacles.

WORK EXPERIENCE

Technical Content Writer
GeeksForGeeks
12/2019 - 06/2020
- Wrote blogs on DSA problems and Web development ideas.
- Designed few new algorithms and patterns problems.

Industrial Trainee
Jaypee Institute Of Information Technology
05/2020 - 06/2020
- Made a machine translation evaluation portal to compare the efficiency of various online MT systems.

Backend Developer Intern
Prasanna Electrotech, Nashik
05/2020 - 06/2020
- Made a inventory management system for their commercial website.

SKILLS

C | C++ | Data Structures | Algorithms | Python | JavaScript | HTML5 | CSS3 | BootStrap4 | PHP | MongoDB | MySQL | Arduino | Heroku

ACHIEVEMENTS

Codechef Rating (07/2020)
1893 (4 star)

Codeforces Rating (07/2020)
1529 (Specialist)

HackBMU - Winner (02/2020)
24- hour Hackathon organized by BML Munjal University, ranked 1st among 210 teams.

GeeksForGeeks - Publisher (2020)
Published 20+ articles on DSA and Web Development

Drink n Code - Winner (03/2020)
Pace coding competition organized by Robotics Hub JIIT

Execute19.1 - #15 Rank (04/2019)
Programming contest organized by Knuth Programming Hub JIIT, ranked 15th among 200+ teams and 4th among 2nd year students at JIIT

ORGANIZATIONS

Coordinator at IEEE Student Branch, JIIT (07/2019 - 06/2020)
Treasurer - Managed the financial budget for the tenure. Technical Team- Conducted Techblocks 5.1 and 5.2, took classes on C++ and Data Structures.

Technical Team Member - Xenith'19 and Xenith'20 (2019 - 2020)
Designed coding problems for Xenith'19 and Xenith'20 on Hackerearth.

Mentor at Microcontroller and Robotics Hub, JIIT (2019 - 2020)
Organized 2 robotics mega-event. Took hands-on session with the team.

CERTIFICATES

Techblocks 4.1, Techblocks 5.1, Techblocks 5.2
Taught C and C++ Programming

Google Assistant Developer
Developed two skills for Assistant Skill Program.

Thomso'19 - IIT Roorkee
Campus Ambassador

Internity Foundation
Web Development and DSA Trainee

07 October, 2020

Dharmesh Pratap Singh

Email: dharmeshprataps@gmail.com
Mobile: +91-8299159085
Github: github.com/dharmeshprataps
LinkedIn: dharmesh-pratap-singh-b25259152/

Education

- **Jaypee Institute of Information Technology** — Noida, India
  Bachelor of Technology - Computer Science; GPA: 7.7 — *July 2017 - Present*
  Courses: *Data Structures & Algorithms, Software Engineering, Operating Systems, Networking, Databases*
- **Mahatma Hansraj Modern School** — Jhansi, India
  Intermediate - 87% — *2017*

Skills Summary

- **Languages**: C++, Python, SQL
- **Tools**: MongoDB, MySQL, Git, React-Native, Machine Learning
- **Platforms**: Linux, Windows
- **Soft Skills**: Leadership, Event Management, Public Speaking, Time Management, Problem-Solving

Experience

- **CampK12** — Remote
  Coding Instructor (Part-time) — *Jan 2020 - June 2020*
  - **Taught Python Language, Machine Learning, MIT App Inventor**: For more than 500+ hours
  - **Impact**: The course has been taken by 100+ students so far
- **Internity Foundation** — Remote
  Technical Intern — *Dec 2018 - Jan 2019*
  - **Learnt Advance Data structures**: Learnt advance data structure and software engineering techniques used in real world projects.

Projects

- **SMART FARMING (Hardware, Machine Learning)**: This project focuses on reducing the efforts by providing support for every task required on a farm. We have made a completely automated system controlled by a website using a Wi-Fi module and moisture sensors with a camera-enabled with Machine Learning to recognize the presence of farm destroying animals and alerting the farmer. If used could help many farmers in transforming their lifestyle. Tech: Python, MongoDB, React, HTML, CSS, Arduino,
- **BACK-END RESTAURANT MENU APP (API BACK-END)**: A platform focusing on developing a restaurant's back-end system where an admin can add dishes to the menu and each person can check all the dishes and give them a rating also, they have functionality like editing their rating or deleting them also. This could help many of restaurants out there is easily maintaining a dynamic menu and having their very own dish rating system. Tech: MongoDB, Express, NodeJs
- **PERSONAL DIARY APP (APP Development)**: An application where a person can write its own experience of its daily life, edit them, and delete them and store them. Tech: React-Native

Workshops, Talk and Webinars

- **Introduction to Competitive Programming, Python and C++ Workshops attended by 100+ students**: in JIIT Noida (2018 & 2019)

Honors and Awards

- Winner at HackBMU 2020
- 6 star in Problem Solving Hackerrank
- 4 star at CodeChef
- Specialist at Codeforces
- 2nd Runner-up Electrovision
- Letter of Appreciation - IC3 Conference

Position of Responsibility

- **Organizing Secretary at IEEE Student Branch JIIT Noida** — Noida, India
  Organized events, conducted technical workshops and lectures. — *July 2019 - Present*

Voluntary Work

- **Light De Literacy Core Team JIIT Noida** — Noida, India
  LDL helps underprivileged students by admitting them in schools and by giving them coaching. *Jan 2018 - July-2019*