The Final Report

Wang Mingxi

Jilin University
College of Computer Science and Technology

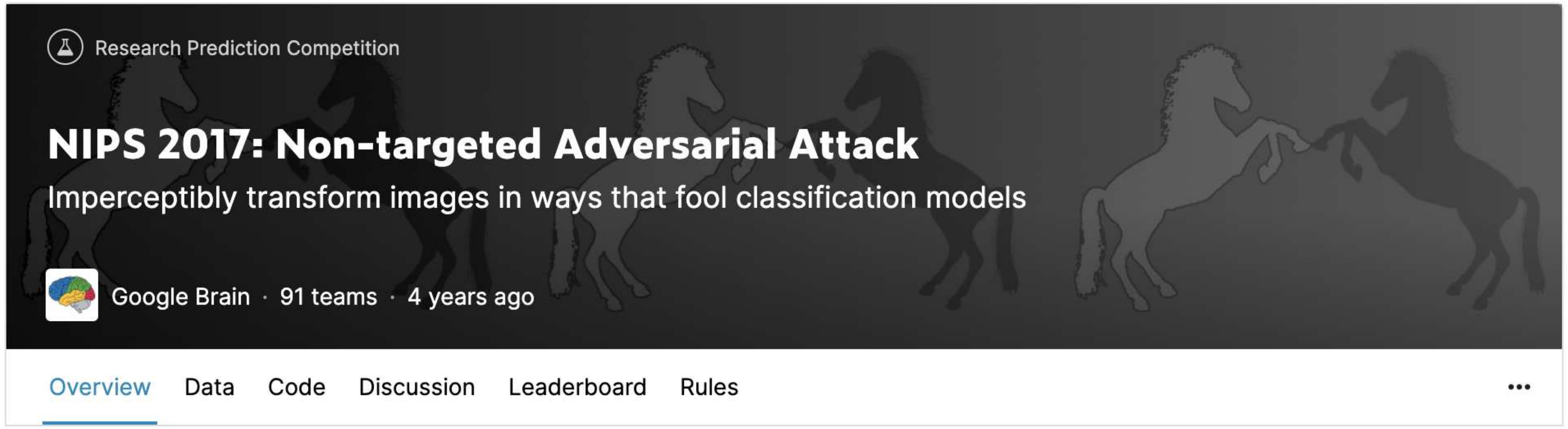
(None)

Overview

Research motivation and context

Project Objectives

- Adversarial samples are generated on the MNIST dataset



Project background

In the real world, there are some unnatural data, they may be corroded existence, may be man-made. It is interesting that these phenomena exist. In the case of images, we can perturb some images so that the classifier fails, but it doesn't look different to the human eye. This is called counterattack. The purpose of this Kaggle project is to generate some of these "unnatural" images on the MNIST dataset to trick the classifier, while the human eye looks normal.

Research contents and methods

Import the MNIST dataset


```
from subprocess import check_output
print(check_output(["ls", "../input"]).decode("utf8"))
df = pd.read_csv("../input/digit-recognizer/train.csv")
df.head()
```

	label	pixel0	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6
0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0
3	4	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0

Split the training set and test set

- The data set is divided into two parts: training set and testing set.

```
y = df.label.values
X = df.drop("label",axis=1).values
X_train, X_test, y_train, y_test = train_test_split(X, y,test_size=0.4, random_state=0)
```

The initial preparation

- Sort out the training set test set and train the classification model

```
self.images = X_test
self.true_targets = y_test
self.num_samples = X_test.shape[0]
self.train(X_train, y_train)
print("Model training finished.")
self.test(X_test, y_test)
print("Model testing finished. Initial accuracy score: " + str(self.initial_score))
```

Get disturbance

```
gradient = gradient_method(target, pred_proba, self.weights)
inf_norm = np.max(gradient)
perturbation = epsilon/inf_norm * gradient
```

- The last line of code says, divide the gradient by the maximum element in the gradient, which is an operation that normalizes the gradient. Epsilon is an artificial input proportionality constant, which refers to how many times the gradient is magnified. The bigger it is, the bigger the gradient is going to be.

Attack function


```
def attack(self, attackmethod, epsilon):
perturbed_images, highest_epsilon = self.perturb_images(epsilon, attackmethod)
perturbed_preds = self.model.predict(perturbed_images)
score = accuracy_score(self.true_targets, perturbed_preds)
return perturbed_images, perturbed_preds, score, highest_epsilon
```

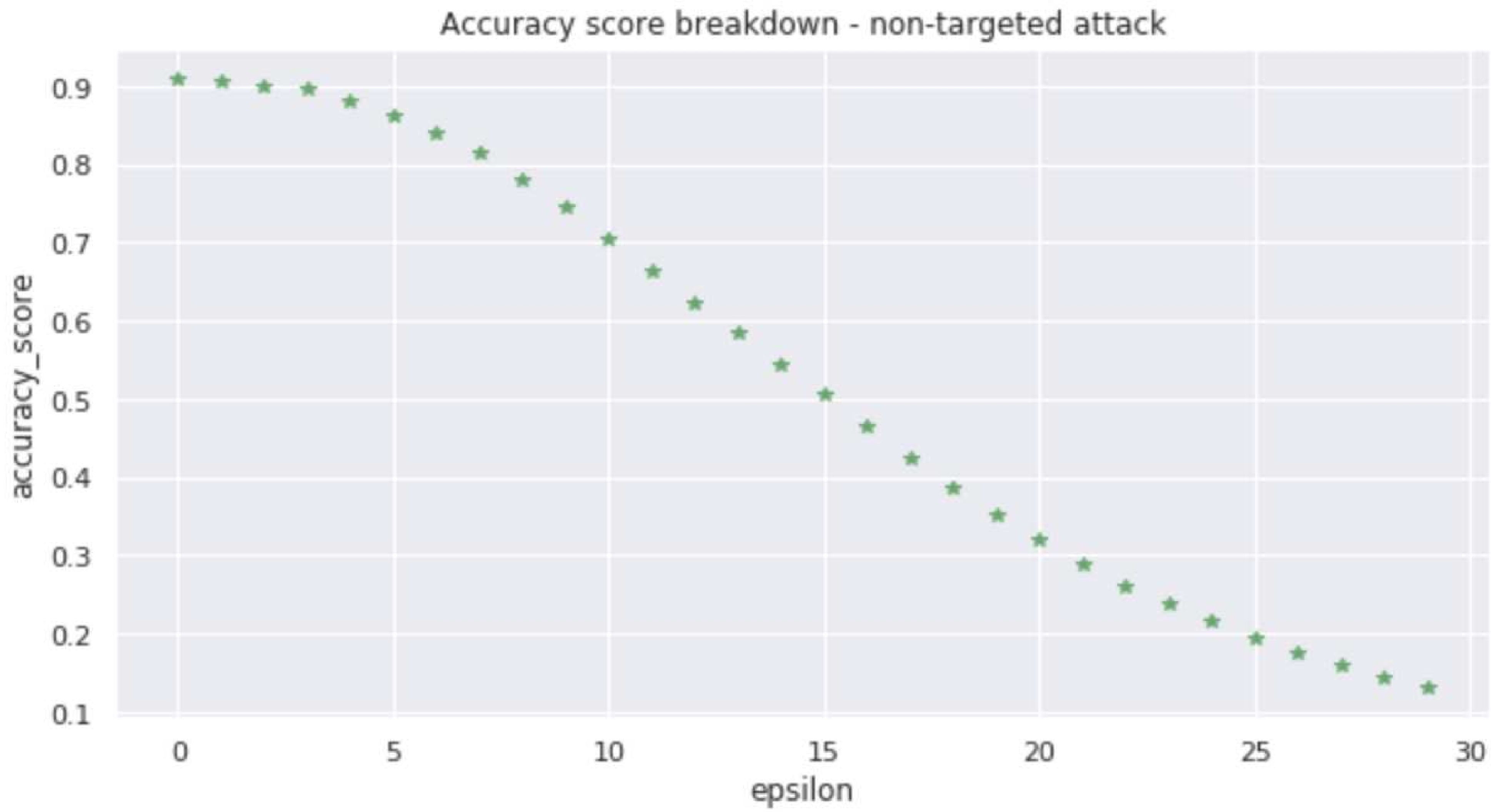
- The perturbed image is obtained, and the highest perturbation is obtained

drawing I


```
sns.set()
plt.figure(figsize=(10,5))
plt.plot(attack.epsilons, attack.scores, 'g*')
plt.ylabel('accuracy_score')
plt.xlabel('epsilon')
plt.title('Accuracy score breakdown - non-targeted attack')
```

drawing II

The greater the disturbance, the lower the accuracy of the model.The accuracy of the model declines fastest in the middle.But if the disturbance is too large, the human eye will not be able to distinguish the picture, and will lose the meaning of confrontation.



Experimental output display

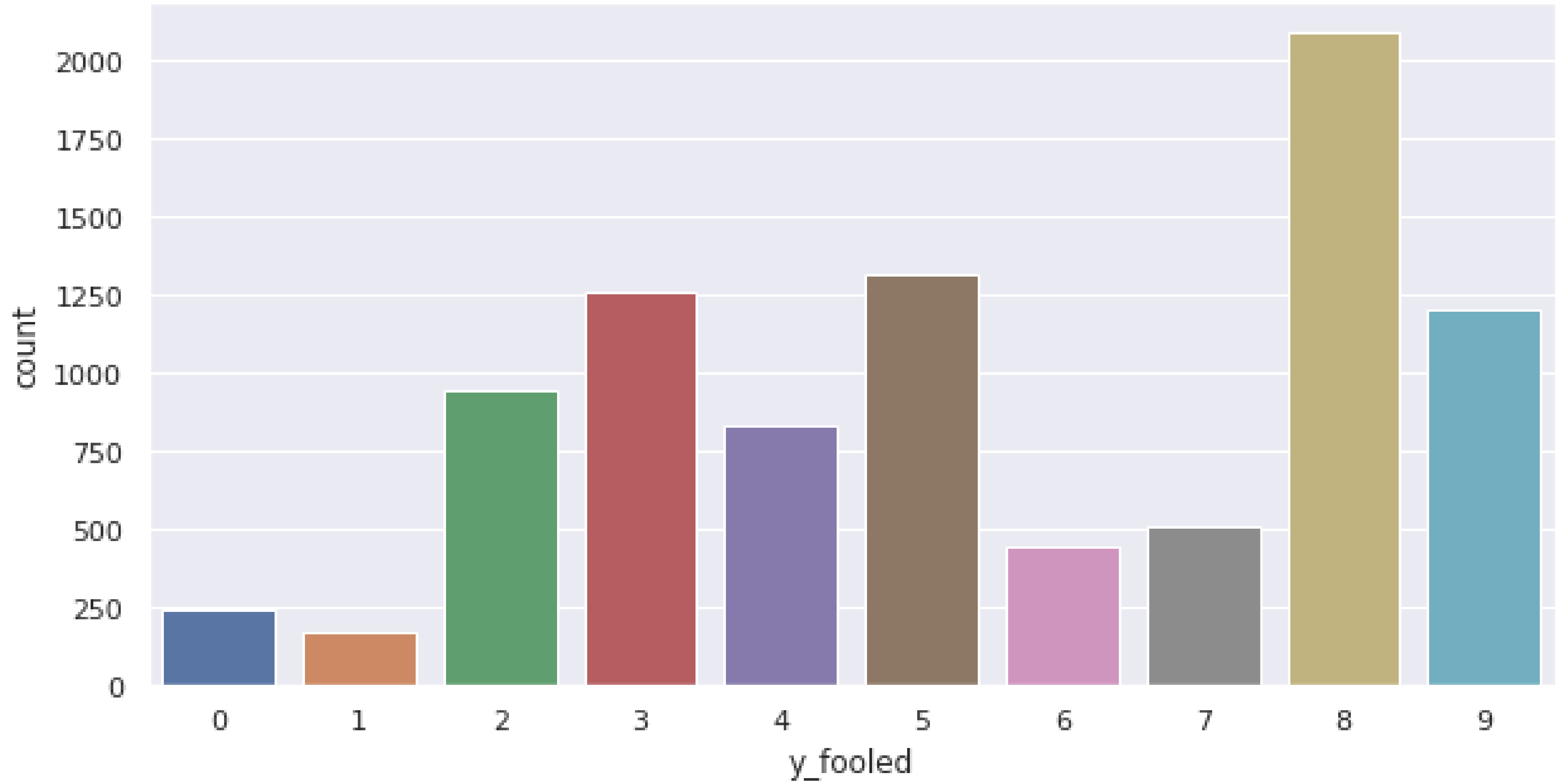

```
example_results = pd.DataFrame(data=attack.true_targets, columns=['y_true'])
example_results['y_fooled'] = example_preds
example_results['y_predicted'] = attack.preds
example_results['id'] = example_results.index.values
example_results.head()
```

	y_true	y_fooled	y_predicted	id
0	3	8	3	0
1	6	6	6	1
2	9	9	9	2
3	5	8	5	3
4	6	6	6	4

Conclusion

Data visualization and analysis I

- The data distribution of fooled target is shown in the figure below



Data visualization and analysis II

We see that eight is often chosen as a target for deception, and this may have something to do with the internal characteristics of the number, which we don't know.

Personally, it is because 8 is a centrally symmetric figure, so other numbers can imitate its features with the least amount of perturbation on average.

Although 0 is also a centrally symmetric figure, : in that it is empty : the identification of the image as 0 requires additional perturbation, so the probability of Fooled Label being 0 is low.

Because 2, 3, 5, and 9 share some similarities with 8 in their representation : in that they share a semicircular structure : the Fooled Target is only second to them in the probability that they were chosen. These phenomena imply the internal information of data characteristics, and the disturbance is the lateral representation of this information.

Contact Information

Wang Mingxi
College of Computer Science and Technology
Jilin University, China

mxwang@tulip.academy

Team for Universal Learning and Intelligent Processing