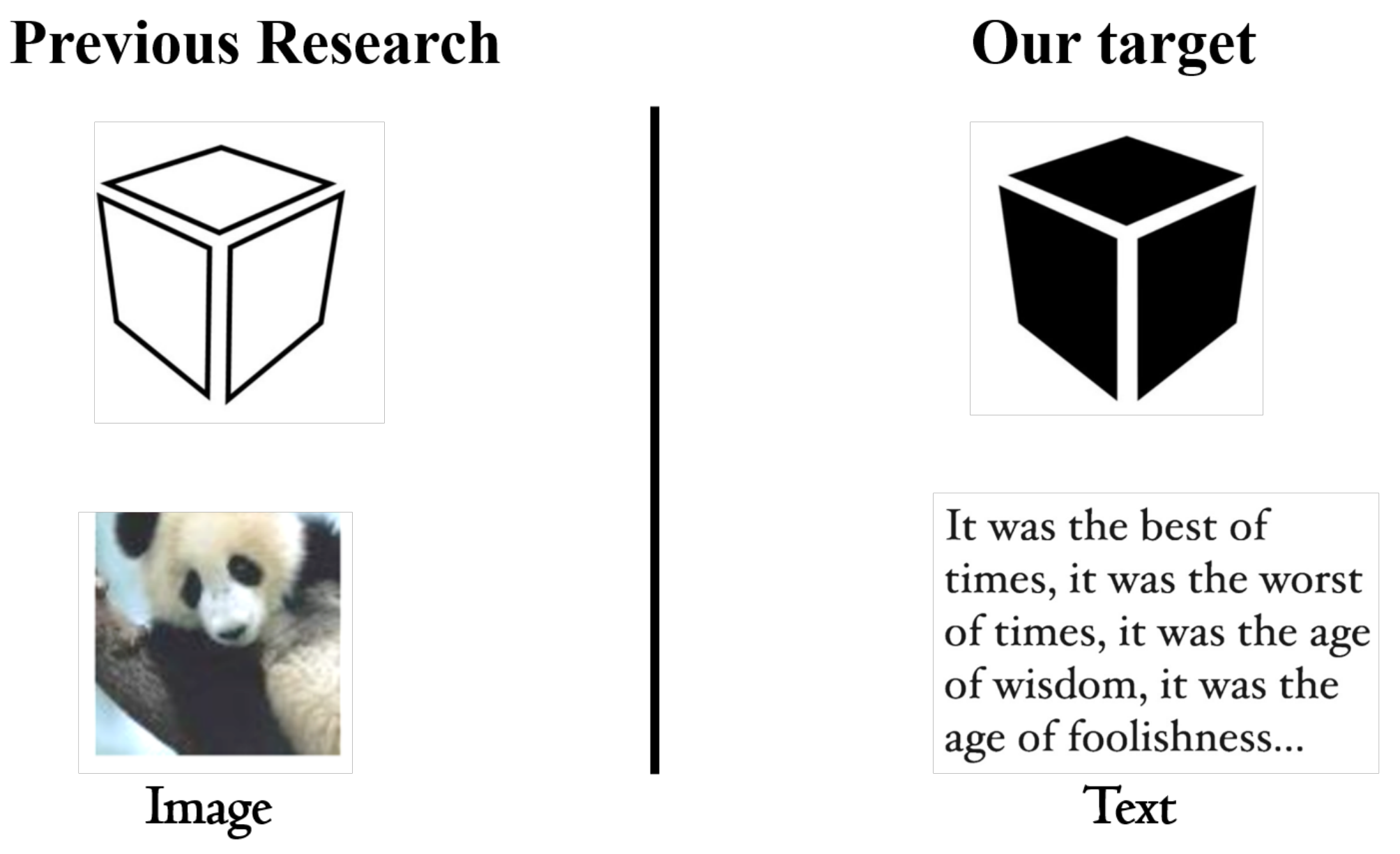
# Black-box Generation of Adversarial Text Sequences to Evade Deep Learning Classifiers

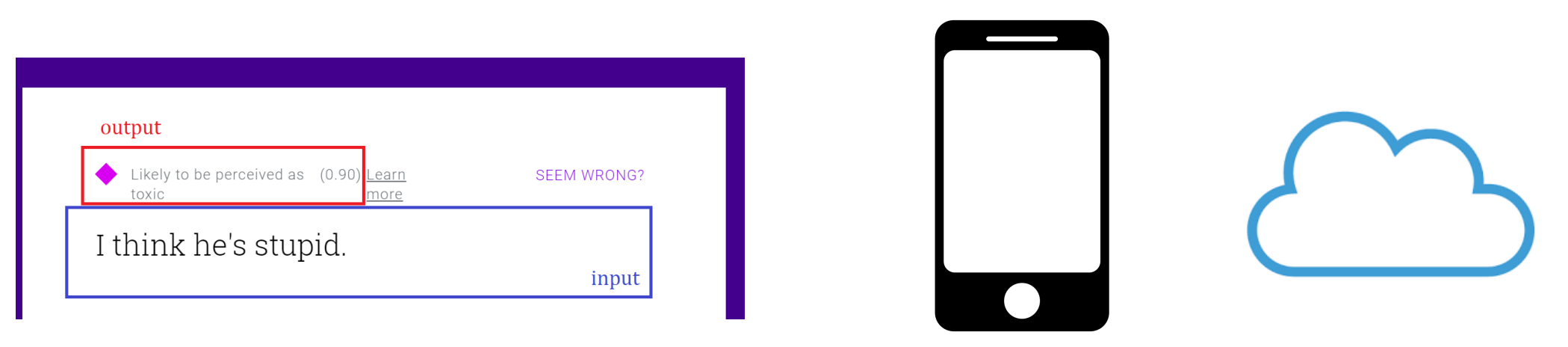
*[ Deep Learning and Security Workshop 2018 ]*

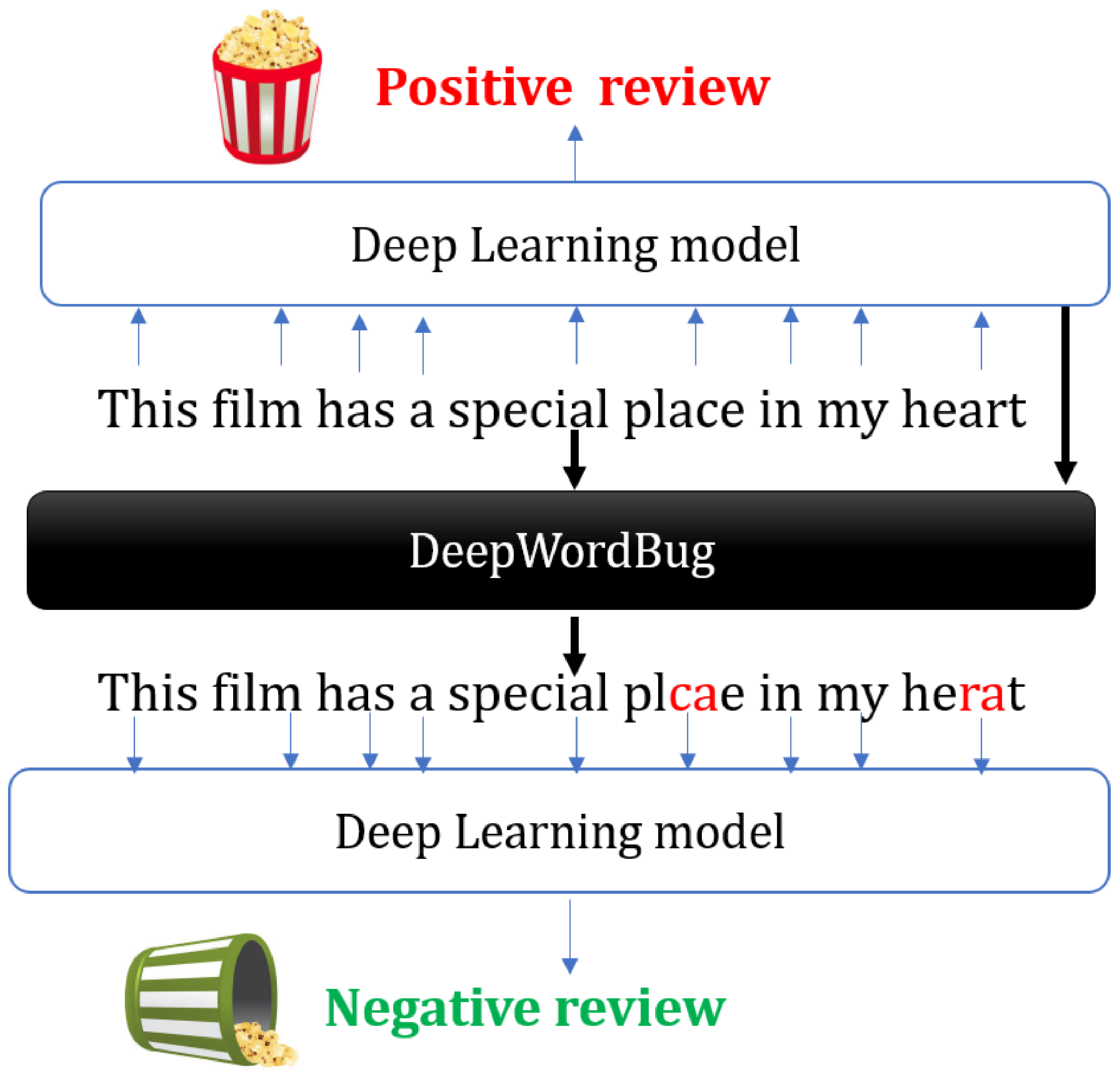
Ji Gao, Jack Lanchantin, Mary Lou Soffa, Yanjun Qi
Department of Computer Science, University of Virginia

## • Motivation



It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness...




Positive review
Deep Learning model
This film has a special place in my heart
DeepWordBug
This film has a special plcae in my herat
Deep Learning model
Negative review

Goal: Evade the prediction of a DNN-based Text Classifier

**Adversarial examples**

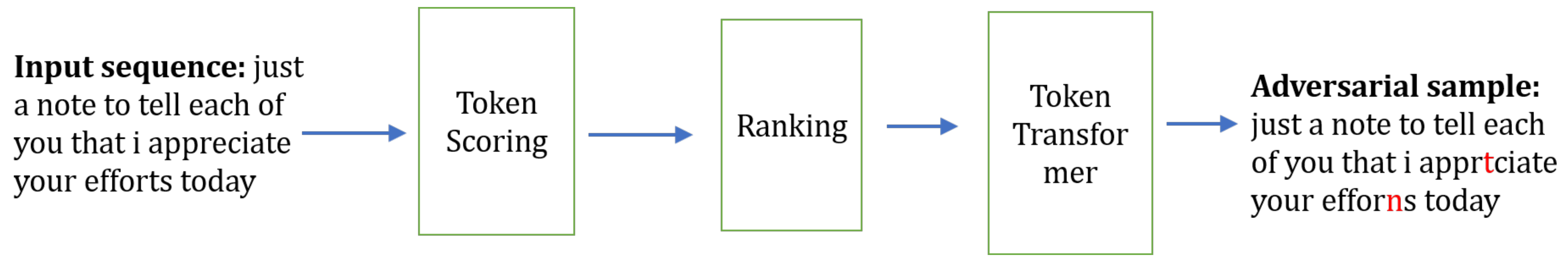
Suppose a deep learning classifier $F(\cdot) : \mathbb{X} \rightarrow \mathbb{Y}$ original sample is $x$, an adversarial example $x'$ in *Untargeted attack* follows:

$$\mathbf{x}' = \mathbf{x} + \Delta\mathbf{x}, ||\Delta\mathbf{x}||_p < \epsilon, \mathbf{x}' \in \mathbb{X}$$
$$F(\mathbf{x}) \neq F(\mathbf{x}')$$

When $\mathbb{X}$ is symbolic:

- How to perturb $\mathbf{x}$?
- No metric for measuring $\Delta\mathbf{x}$

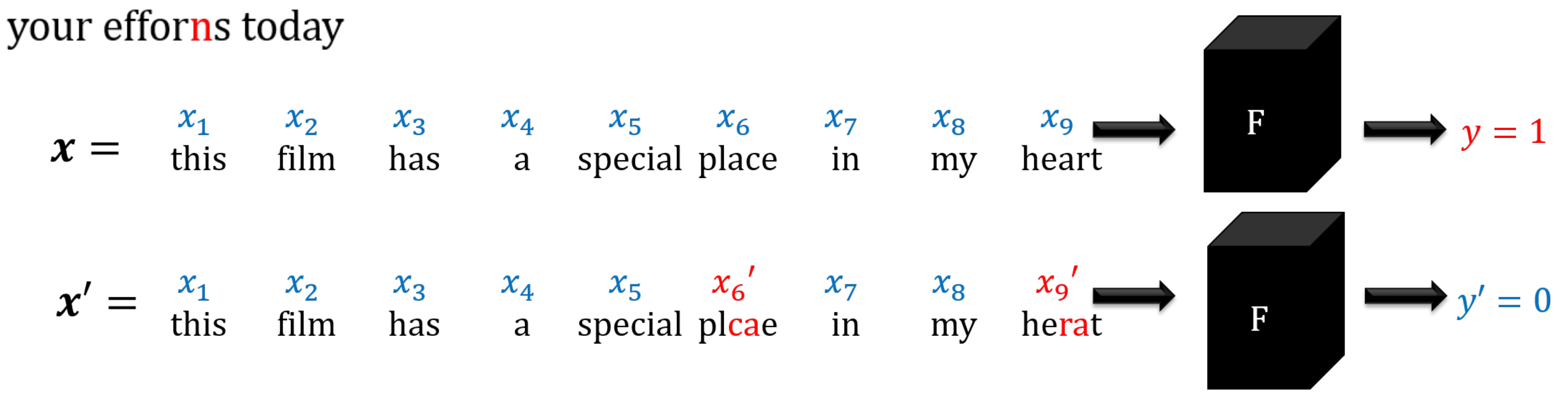
## • DeepWordBug Algorithm



**Input sequence:** just a note to tell each of you that i appreciate your efforts today

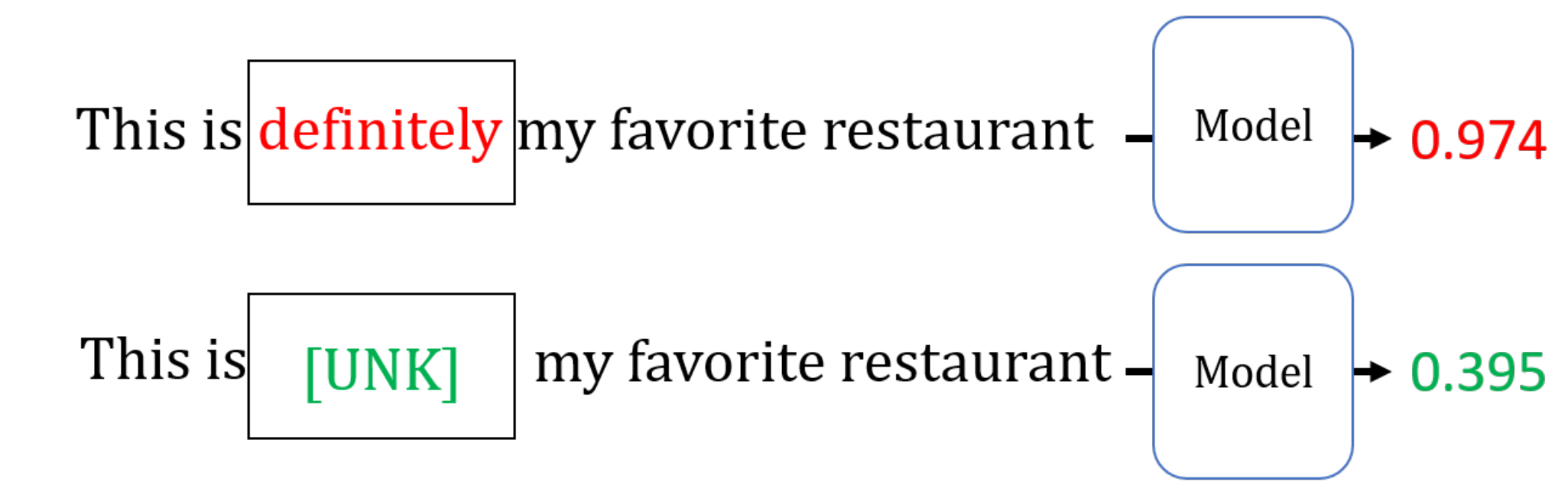
**Adversarial sample:** just a note to tell each of you that i apprtciate your effonrs today

$$\Delta\mathbf{x} = \text{Edit distance}(\mathbf{x}, \mathbf{x}') = \sum_{i \in \text{Selected words}} \text{Edit distance}(x_i, x_i')$$



### Step 1: Scoring function

- Goal: Select important words
- The proposed scoring functions have the following properties:
  1. Correctly reflect the importance of words
  2. Black-box
  3. Efficient to calculate.


This is definitely my favorite restaurant
Model
0.974
This is [UNK] my favorite restaurant
Model
0.395

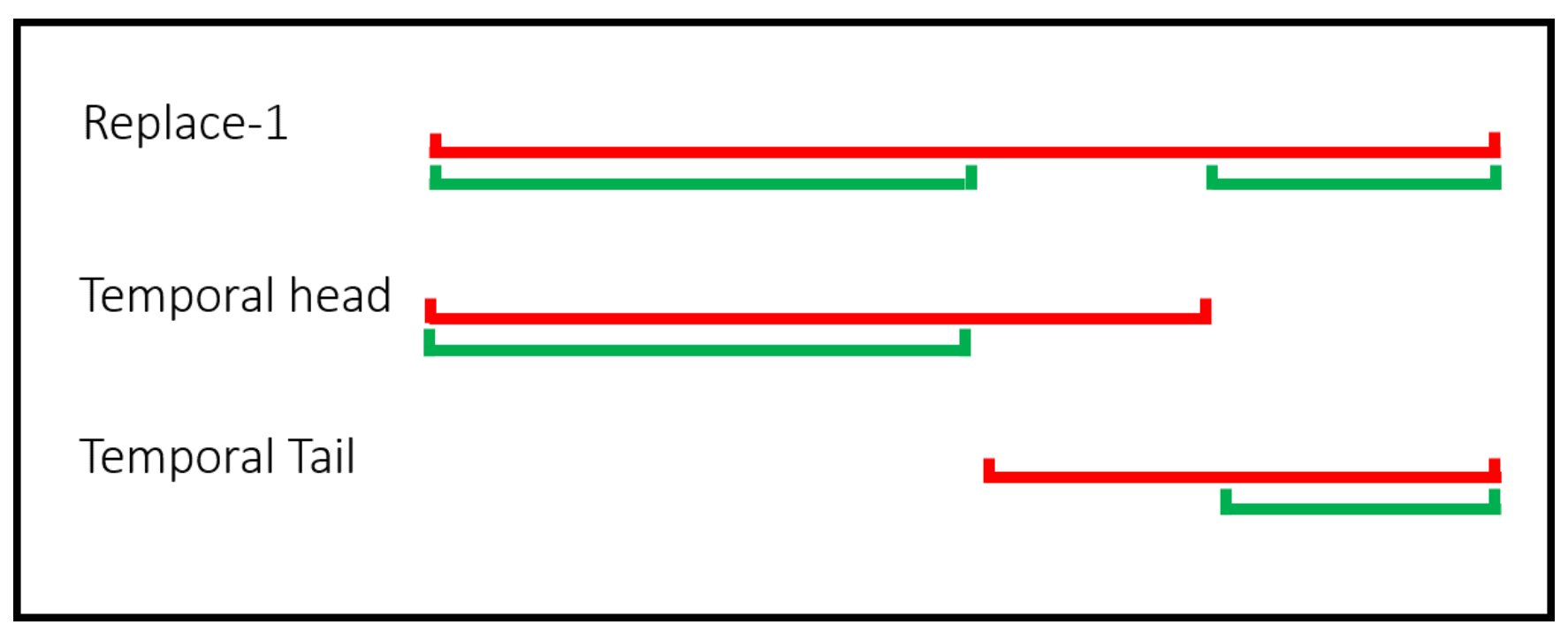
Black-box perturbation based scoring functions

### Step 2: Ranking and transformation

- Calculate the scoring function for all words in the input once.
- Rank all the words according to the scores.

Replace-1 score

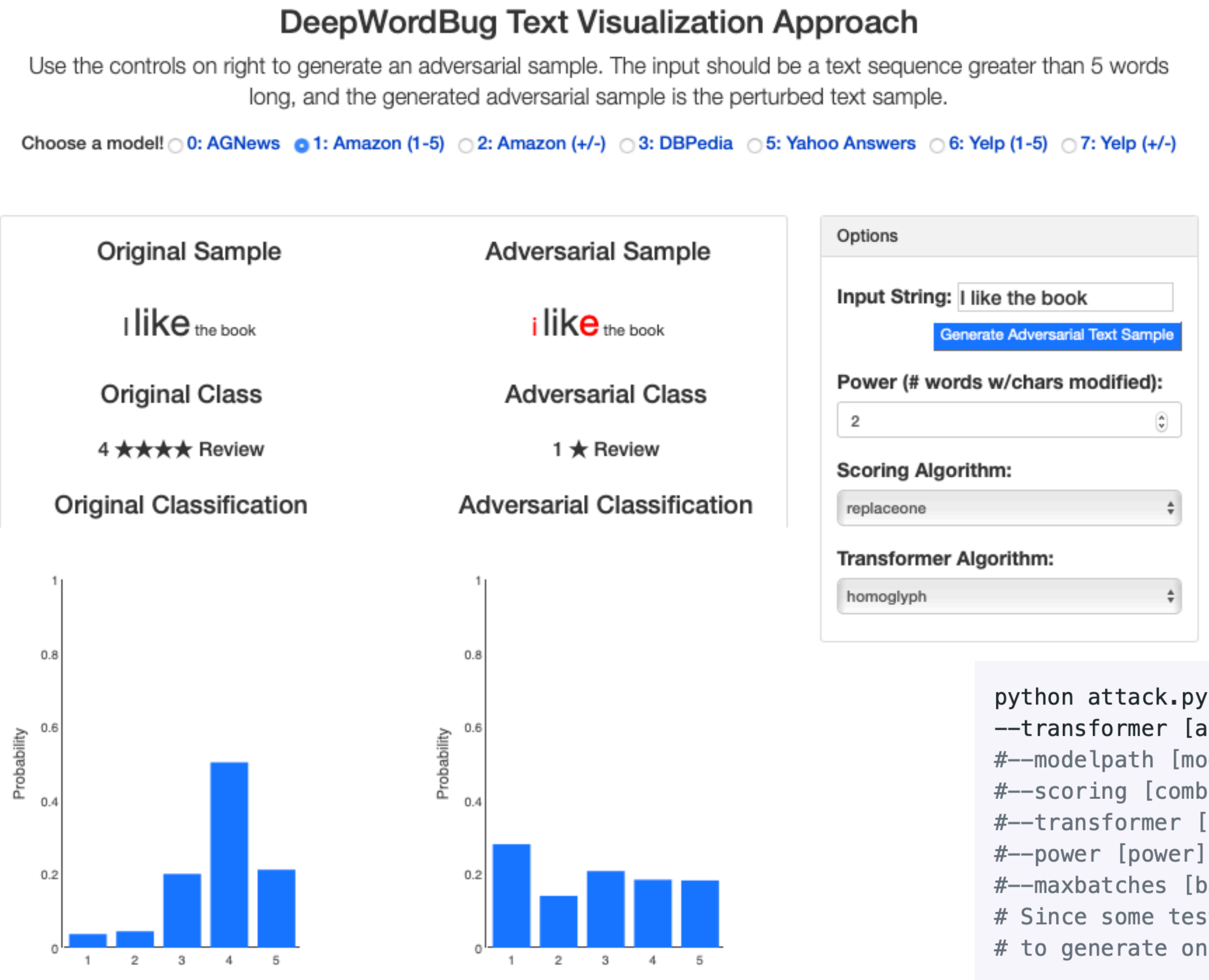
| | |
|---|---|
| this | 0.974-0.969=0.005 |
| is | 0.974-0.960=0.014 |
| definitely | 0.974-0.395=0.579 |



### Step 3: Word Transformer

- Aim I: Machine-learning based classifier views generated words as **"unknown"**.
- Aim II: Control the **edit distance** of the modification

## • Results https://github.com/QData/deepWordBug



DeepWordBug Text Visualization Approach

Use the controls on right to generate an adversarial sample. The input should be a text sequence greater than 5 words long, and the generated adversarial sample is the perturbed text sample.

Choose a model! ○ 0: AGNews ● 1: Amazon (1-5) ○ 2: Amazon (+/-) ○ 3: DBPedia ○ 5: Yahoo Answers ○ 6: Yelp (1-5) ○ 7: Yelp (+/-)

| Original Sample | Adversarial Sample |
|---|---|
| I like the book | i like the book |
| Original Class | Adversarial Class |
| 4 ★★★★ Review | 1 ★ Review |
| Original Classification | Adversarial Classification |

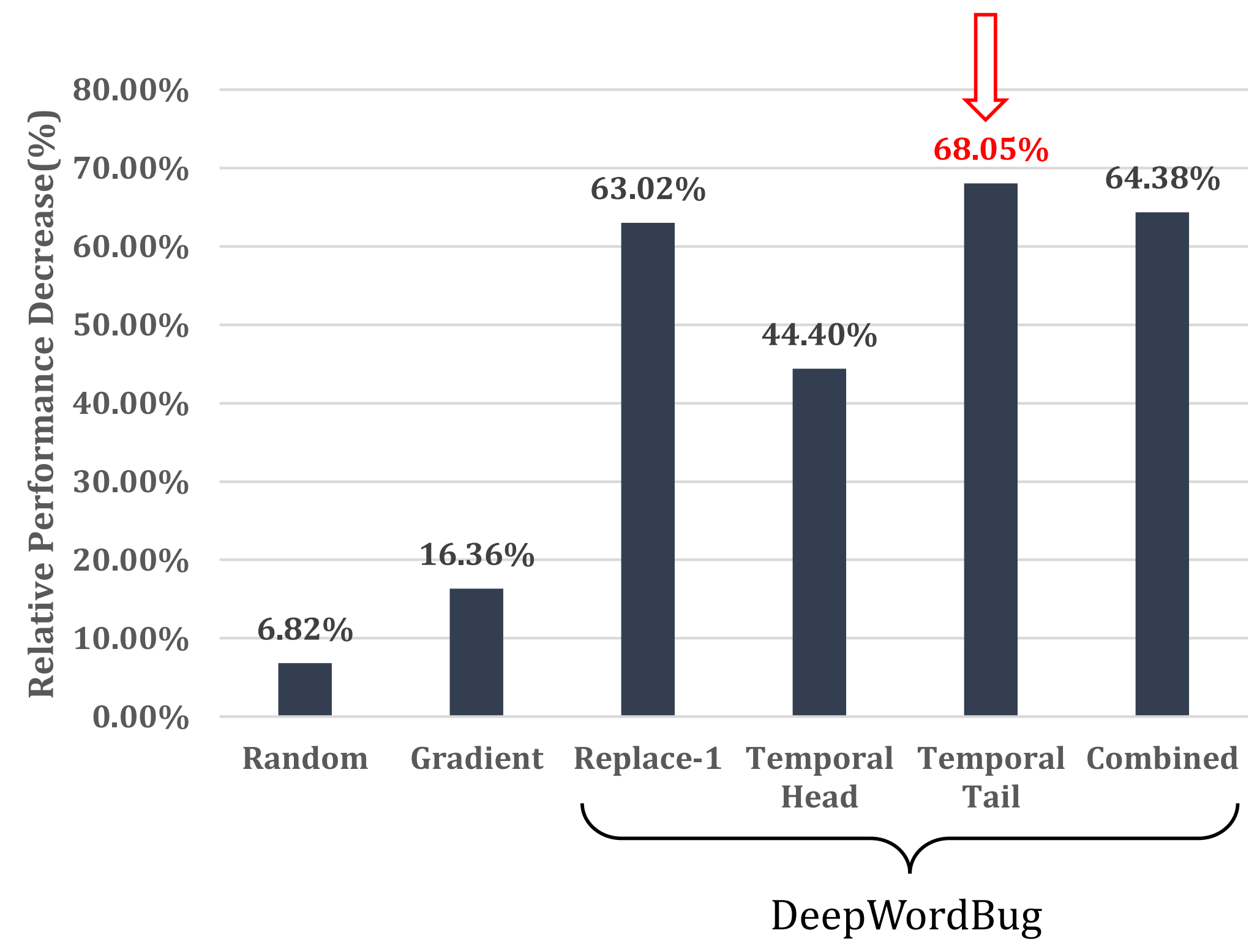
Options

Input String: I like the book

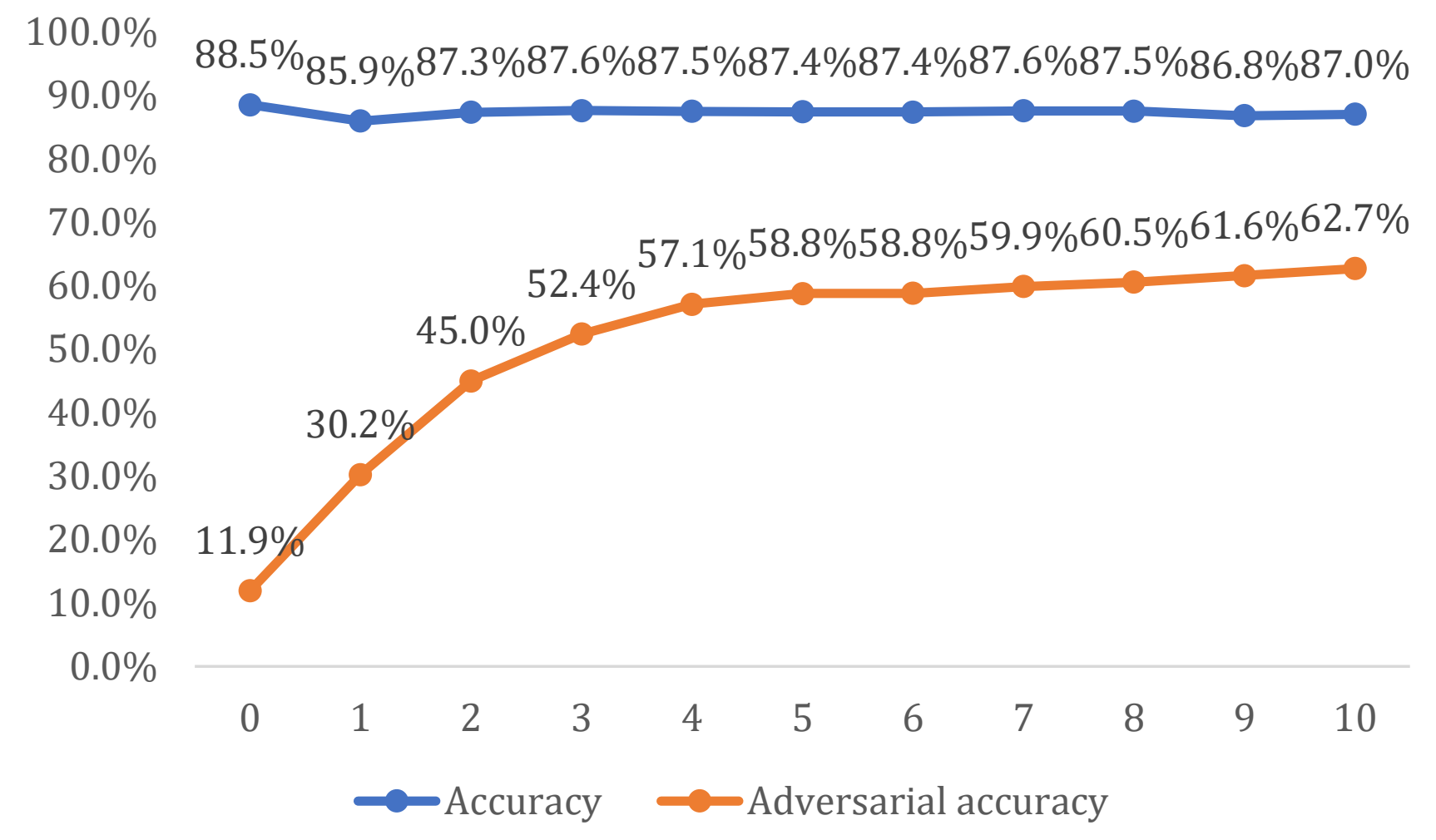
Power (# words w/chars modified): 2

Scoring Algorithm: replaceone

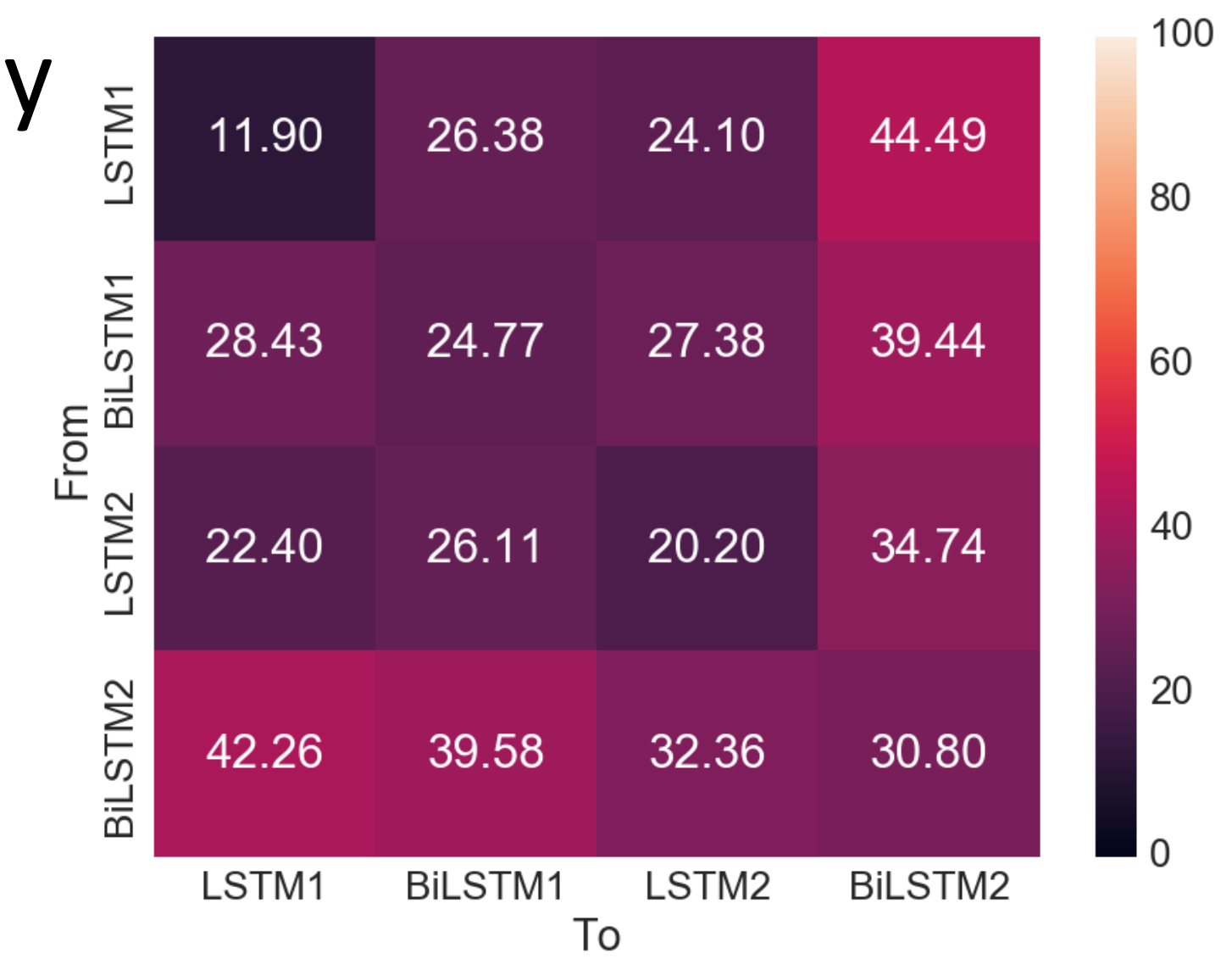
Transformer Algorithm: homoglyph



```
python attack.py --data [0-7] --model [modelname] --modelpath [modelpath] --power [power] --scoring [algor
--transformer [algorithm] --maxbatches [batches=20] --batchsize [batchsize=128] ### Generate DeepWordBug a
#--modelpath [modelpath] #Model path, stored by train.py
#--scoring [combined, temporal, tail, replaceone, random, grad] # Scoring algorithm
#--transformer [swap, flip, insert, remove, homoglyph] # transformer algorithm
#--power [power] # Attack power(integer, in (0,30]) which is number of modified tokens, i.e., the edit dis
#--maxbatches [batches=20] # Number of batches of adversarial samples generated, samples are selected rand
# Since some test dataset is very large, to evaluate the performance we add this parameter
# to generate on parts of data. By default it will generate 2560 samples.
```



Transferability


From
To
LSTM1
BiLSTM1
LSTM2
BiLSTM2