

Supplementary Materials: Natural Language Induced Adversarial Images

Anonymous Authors

1 PROMPT STRUCTURE AND WORD SPACE SETTINGS

The adversarial prompt structure and word space is customizable, which can be either manually designed or auto-generated by GPT-4 for initialization. In our experiments, we manually designed the adversarial prompt structures and word spaces for the animal and race classification attack tasks. Then we introduce the auto-generation method by using GPT-4. We chose the vehicle target as an example. The details are provided as follows.

1.1 Attack Animal Classifiers

Prompt Structure:

"<number><color>[target animal] <appearance>is <gesture>on the <background>on a <weather>day, the [target animal] faces forward, the [target animal] occupies the main part in this scene, viewed <viewangle>."

"<word>" represents a word that can be optimized. "[target animal]" is the ground truth target category y (e.g. "cat") of the generated images, which is user-defined and fixed during the prompt optimization.

Word Space: The specific settings for the word space in the animal classification attacks are introduced in Table S1.

1.2 Attack Race Classifiers

Prompt Structure:

"<number><expression>[target person] <appearance>is <gesture>on the <background>on a <weather>day, the [target person] >faces forward, the [target person] occupies the main part in this scene, viewed <view angle>, in a <style>."

"<word>" represents a word that can be optimized. "[target person]" is the ground truth target category y (e.g. "white person") of the generated images, which is user-defined and fixed during the prompt optimization.

Word Space: The specific settings for the word space in the race classification attacks are introduced in the Table S2.

1.3 Auto-Generation Mehtod

The word space is a fundamental component of our attack pipeline. Since it needs to be adapted to different classification tasks, manually crafting a word space for each task requires a lot of effort. To address this issue, we use GPT-4 to auto-generate the word space. First, we can select a new target category, such as vehicle, road sign, etc. Next, the above hand-constructed word spaces are input into GPT-4 as examples, and GPT-4 is instructed to generate a similar word space for a new task.

The prompt used for auto-generating the word space by GPT-4 is shown as follows:

Now I need you to help construct a word space for the given task. Here are two examples of the constructed word space. First is for animal classification. The word space is: 'number': ['one','two'],

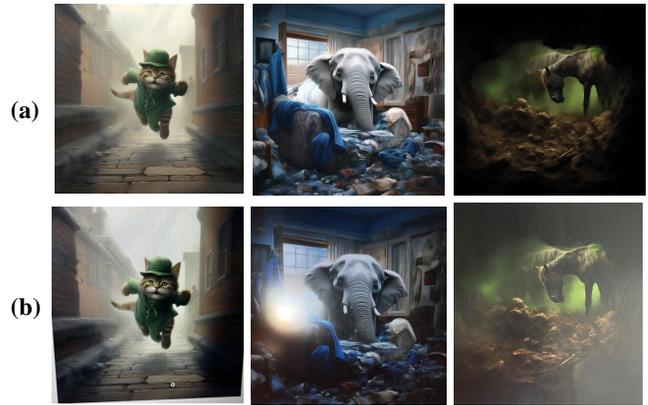


Figure S1: Comparison of (a) digital and (b) physical adversarial images.

'weather': ['sunny', 'rainy', 'cloudy', 'snowy', 'windy', 'foggy', 'stormy', 'humid'], 'background': ... And the second example is for human race classification. The word space is: 'number': ['one'], 'weather': ...

When constructing the word space, you can also think from the above perspectives. But pay attention that each task may have its unique feature, e.g. humans may wear various clothes but their skin colors are limited, and animals may have unique behaviours like barking or flying. When constructing word space, you must take these characteristics of different tasks into consideration and construct the most suitable word space. Do remember that your constructed word space should have the same attribute: number, weather, background, color, view angle, gesture, style, appearance, expression. If you find one attribute is not suitable for the task, you should keep the key and give a list [" "] as its value. Now construct the word space for the task of task classification. Your answer must be in the form of a python dictionary like the example above. Do not include any words other than the dictionary!



Figure S2: Examples of human race classification attacks.

Table S1: Word space setting for the animal classifier attack

Attribute	Value		
number	one	two	many
color	red yellow purple many different colors	blue black orange	green white brown
target animal	butterfly cow horse squirrel	cat dog sheep	chicken elephant spider
appearance	wearing a hat wearing a flower on the head	wearing a pair of glasses	wearing clothes
gesture	sitting running digging a burrow barking	flying playing with a ball crawling standing	taking a nap chasing a butterfly stretching
background	on the sky covered with clouds on the ground covered with snow and ice inside a living room which is in a total mess under the deep sea	on the green grass field with flowers on the busy street in the dense forest on the moon	on Mars in front of a brick wall in the rocky terrain
weather	sunny snowy stormy	rainy windy humid	cloudy foggy
view angle	from an eye-level perspective		
style	blank	blurry, fuzzy, misty	realistic

Table S2: Word space setting for the race classifier attack

Attribute	Value		
number	one	two	many
expression	happy worried	sad depressed	angry overwhelmed
target person	white person	black person	Chinese person
appearance	wearing a hat wearing casual wear with long hair wearing a flower on the head wearing earrings	wearing a pair of glasses wearing traditional attires with short hair with tatoo on the face wearing bracelets	wearing formal suits wearing athletic outfits with curly hair wearing necklaces
gesture	sitting running digging a burrow studying	smoking playing with a ball crawling exercising	taking a nap chasing a butterfly stretching working
background	on the sky covered with clouds on the ground with snow and ice inside a living room in a total mess under the deep sea	on the green grass field with flowers on the busy street in the dense forest on the moon	on Mars in front of a brick wall in the rocky terrain
weather	sunny snowy stormy	rainy windy humid	cloudy foggy
view angle	from an eye-level perspective		
style	blank	blurry, fuzzy, misty	realistic

Table S3: Word space setting for the vehicle classifier attack

Attribute	Value		
number	one multiple	two	three
color	red yellow silver brown	blue black gray	green white orange
target vehicle	bicycle bus underground	motorbike truck	car train
appearance	with headlights on with a sunroof	with doors open with tinted windows	with a spoiler
background	on the highway in a garage near a body of water on a bridge	in a parking lot on a race track in a desert	on a city street in a rural area in a forest
weather	sunny snowy stormy	rainy windy humid	cloudy foggy
view angle	from the front	from the side	from the back
style	blank	blurry, fuzzy, misty	realistic

one blue angry dog wearing a hat is crawling...



Dog → Squirrel

one red happy cat wearing clothes is stretching ...



Cat → Spider

one green sad sheep wearing clothes is ...



Sheep → Horse

one white happy horse wearing a hat is barking ...



Horse → Cow

one green angry chicken wearing a hat is sitting...



Chicken → Squirrel

one yellow depressed elephant wearing ...



Elephant → Sheep

one white sad cow wearing clothes is ...



Cow → Sheep

one black angry butterfly wearing a hat is ...



Butterfly → Chicken

one brown worried squirrel wearing a pair of glasses ...



Squirrel → Cat

two white depressed spiders wearing a pair of ...



Spider → Dog

Figure S3: Example images for attacking the animal classifier trained on ImageNet. The black texts are the prompts, the blue texts are the groundtruth categories, and the red texts are the misclassified categories.

The auto-generated word space for the vehicle classification task is shown in Table S3. It shows that the generated word space has been adapted to the vehicle classification task. The pipeline of our method is fully automated without manually designed component by using the auto-generation method.

2 DETAILS FOR SEARCHING ADVERSARIAL IMAGES BY GOOGLE

We searched for some photos captured in the real world on Google according to the adversarial semantic information analyzed by our method. For example, we obtained 50 images returned by Google

with 10 prompts like “A cat is stretching”, “A horse in a foggy day”, etc. For fair comparison, we also searched 50 images by Google using prompts with random word selection as control experiments. For each prompt, we select pictures with top-5 semantic relevance ranking in Google, and we checked whether they were real photos based on the source of the picture provided by Google. For example, if the picture came from a news report or a photography competition, we regarded it as a real photo. We only retained photos that came from these reliable sources.

3 COMPARING DIGITAL AND PHYSICAL IMAGES

Figure S1 shows a set of examples of digital and physical printed images. The physical world added more perturbations to the images, e.g. the printer may cause color distribution variations.

4 DETAILS OF THE ATTACKS ON RACE CLASSIFIER

We evaluated the attack effect of our method on 3-race classification tasks (black, white, east asian). We chose Midjourney as the

text-to-image generator for adversarial images. The settings for the adversarial prompt structure and word space are introduced in Section S1.2. Our adaptive GA method was used to generate adversarial prompts and images. For each target race, we set the population size to 20. The mutation probability was set to 0.01, and the hyperparameter λ in the fitness function was set to 0.5. The termination condition was that the number of iterations reached 15 generations. We obtained adversarial prompts and images by using the above method. The ASR of the adversarial images against the human race classifier Vit was 89%, and some examples are shown in Figure S2. It indicates that our method effectively attacked the race classifier Vit.

5 EXAMPLE IMAGES FOR ATTACKING THE CLASSIFIER TRAINED ON IMAGENET

Figure S3 shows a set of a examples of adversarial prompts and images for attacking the animal classifier ResNet101 trained on ImageNet.

349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406

407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464