

## 459 A Few-Shot Prompt

460 We use the few-shot prompt shown in Listing 1 for the *3DOC* experiments in Section 5.2. The prompt uses  
 461 Python syntax, and begins with an “import” statement that indicates functions available from the fine-grained  
 462 object localization API (Section 4.2). Users queries are converted to lower case, the ending punctuation is  
 463 removed, and prepended with a # (which represents a comment in Python). Then, the preprocessed query  
 464 is added to the few-shot prompt after an empty line. The modified prompt is then processed with an LLM.  
 465 For multi-round dialog, additional interactions or corrections are added after the response from the LLM  
 466 without an empty line separator (as demonstrated in the second and third few-shot examples).

```

467 from utils import find_objects, sort_objects, is_close
468
469 # find the wooden chair
470 def go_find_it():
471     chairs =
472         find_objects('chair', distractors=['sofa', 'bench', 'stool', 'desk'])
473     chairs = sort_objects(chairs, attributes=['wooden'])
474     return chairs[0]
475
476 # find a backpack with white polka dots on it
477 def go_find_it():
478     backpacks
479     = find_objects('backpack', distractors=['pillow', 'jacket', 'shirt'])
480     backpacks = sort_objects(backpacks, attributes=['white polka dots'])
481     return backpacks[0]
482 # no, my backpack is also yellow
483 def go_find_it():
484     backpacks
485     = find_objects('backpack', distractors=['pillow', 'jacket', 'shirt'])
486     backpacks
487     = sort_objects(backpacks, attributes=['white polka dots', 'yellow'])
488     return backpacks[0]
489
490 # find the pants on the dresser
491 def go_find_it():
492     pants = find_objects
493         ('pants', distractors=['shirt', 'socks', 'shoes', 'dress'])
494     dressers = find_objects
495         ('dresser', distractors=['bookshelf', 'bed', 'desk', 'chair'])
496     pants = is_close(pants, dressers)
497     return pants[0]
498 # no, my pants are also red
499 def go_find_it():
500     pants = find_objects
501         ('pants', distractors=['shirt', 'socks', 'shoes', 'dress'])
502     pants = sort_objects(pants, attributes=['red'])
503     dressers = find_objects
504         ('dresser', distractors=['bookshelf', 'bed', 'desk', 'chair'])
505     pants = is_close(pants, dressers)
506     return pants[0]
507
508 # find the apple next to the microwave
509 def go_find_it():
510     apples = find_objects
511         ('apple', distractors=['pear', 'tomato', 'orange', 'bowl'])
512     microwaves = find_objects
513         ('microwave', distractors=['dishwasher', 'sink', 'refrigerator',])
514     apples = is_close(apples, microwaves)
515     return apples[0]
516

```

Listing 1: Few-shot prompt used for experiments in simulation.

Table 2: *3DOC* Statistics

Num of Object Categories	10
Num of Unique Objects	72
Num of Unique Intrinsic Attributes	53
Num of Unique Scenes	100
Num of Unique Scene Layouts	500
Num of Extrinsic Attribute Episodes	50
Num of Intrinsic Attribute Episodes	1,713

## 518 **B *3DOC* Statistics**

519 Table 2 provides additional details on the composition of the *3DOC* dataset.

## 520 **C Qualitative Examples**

521 A video presenting qualitative examples of our *GoFind* agent executing the *FindThis* task in both simulation  
 522 and in the real-world is provided in the supplemental material.

## 523 **D Dataset and Code Release**

524 The *3DOC* dataset and source code to reproduce the results presented in Section 5.2 will be publicly  
 525 released.