

기초인공지능 (CSE4185) 과제0

2022.09.08

Basic Object-Oriented Programming in Python

제출파일: [assign0_problem1.zip](#), [assign_problem2.zip](#) (수행조건에 맞게 완성한 파일 압축)

문제1

-아래 링크에 제시된 tutorial을 참고하여 문제를 해결한다.

<https://inst.eecs.berkeley.edu/~cs188/sp22/project0/>

-과제란에 게시된 assign0_problem1.zip의 압축을 해제한 뒤, addition.py, buyLotsOfFruit.py, shopSmart.py 파일의 코드를 조건에 맞게 수정한다. (위 링크의 tutorial.zip 파일과 동일하나 가급적 과제란에 제시된 코드를 이용하도록 한다.)

-autograder.py를 실행했을 때의 결과가 아래와 동일하면 문제1을 해결한 것으로 간주한다.

-기본 파일에서 제공된 것 외에 채점에서 사용하는 추가적인 test case는 없다. (문제1에 한함)

```
Provisional grades
=====
Question q1: 1/1
Question q2: 1/1
Question q3: 1/1
-----
Total: 3/3
```

문제2

주어진 조건에 맞게 homework.py 파일의 ReadEliminateSignFromPaper class를 완성한다.

(과제란에 게시된 assign0_problem2.zip 압축해제 후 과제수행)

1. 논문의 일부가 입력파일(input.txt)로 주어질 때, 각 단어에 존재하는 모든 특수문자(파이썬 문자열 내장메소드 isalnum() 수행 시 False로 return되는 character로 간주)를 제거하고, 문장 단위로 이루어진 문자열을 self.sentences 리스트에 저장한다.
2. 단, 온점(. period)이 문장의 종료를 알리는 경우에는 특수문자로 취급하지 않고 제거하지 않는다. 이에 따라 모든 문장은 온점으로 종료해야 한다. (HINT: 문장의 종료를 나타내지 않는 온점은 파이썬 인덱스 기준으로 단어 문자열의 -1번째가 아닌 것임)
3. 문장 단위로 이루어진 문자열에서 (특수문자가 제거된) 각 단어는 one space로 구분되어야 한다.
4. 주어진 raw data에서 공백(space)으로 나누어진 것을 기준으로 단어를 구분한다. 즉, 일상적으로 단어로 생각되는 것들 사이에 어떠한 문장부호가 있더라도 공백으로 구분된 것이 아니라면 하나의 단어로 취급하고 해당 문장부호는 제거의 대상이다. 예를 들어, artificial-intelligence, email1234@gmail.com, 2022.09.08, 27/32 등은 모두 각각 하나의 단어로 취급한다.
5. main.py를 실행했을 때 다음 내용을 출력한다. (output.txt 참고)
-제거된 특수문자와 각 개수의 쌍으로 구성된 튜플을 원소로 갖는 리스트 (개수를 기준으로 내림차순 정렬)

- 문장의 개수
- 앞에서 첫번째 문장
- 앞에서 두번째 문장
- 뒤에서 두번째 문장
- 뒤에서 첫번째 문장

- 주어진 인스턴스 변수(`self.sign_dict`, `self.sentences`) 외에 추가적인 클래스 변수 또는 인스턴스 변수 사용을 허용한다. 주어진 인스턴스 변수의 자료형 및 이름 변경 역시 가능하다.
- 코드에서 주어진 `find_eliminate_sign`, `get_sorted_sign`, `__len__`, `__getitem__` 메소드는 필수적으로 구현한다. 위 메소드의 이름은 변경할 수 없다. 필요 시 추가적인 메소드의 사용이 가능하다.
- `homework.py` 파일에서 주어진 파일 입출력(I/O) 코드의 수정을 허용한다.
- 각 `py` 파일에서 주어진 라이브러리 이외에는 `import`하여 사용할 수 없다. (regular expression, `nlTK` 등) 특히 정규표현식 라이브러리 `re`는 `scoring.py`에서 사용된 것을 제외하고 사용하는 일이 없도록 한다. 위반 시 문제2는 예외 없이 0점 처리된다.

주의사항 및 예외사항

- 입력파일의 텍스트는 문단단위로 줄바꿈이 이루어지기 때문에 `\n`(줄바꿈 표시)의 개수는 문장의 개수와는 무관하다.
- 문장의 종료를 알리는 특수문자는 온점이 유일하다고 가정하고, 이를 위반하는 입력데이터는 존재하지 않는다고 가정한다. 즉, 물음표(?), 느낌표(!) 등이 등장하더라도 문장의 종료를 알리는 특수문자로 취급하지 않는다.
- 문제1과 달리 채점 시 다양한 test case가 사용된다.
- test case의 모든 입력에는 utf-8 인코딩이 가능한 character만 존재한다.
- 모든 입력파일에는 4개 이상의 문장이 존재한다.
- 특수문자 여부를 구분할 때에는 가급적 `string.isalnum()` 메소드를 사용하기를 권장한다.
- `main.py` 및 `scoring.py` 파일은 채점 시 사용되며 수정이 불가하고 제출할 필요는 없다.
- 기타 예외사항에 관해서는 공정하고 상식적인 수준에서 적절히 감안하여 채점할 예정인 점을 미리 고지하나, 추가적인 예외사항에 대해서 우려할 시에는 사이버캠퍼스 질의응답에 남겨주시길 바랍니다.

참고사항

- `__len__(self)` 메소드: `len(객체)` 코드를 실행할 때 return되는 값을 구하는 메소드
- `__getitem__(self, idx)` 메소드: `객체[idx]` 코드와 같이 인덱싱을 수행할 때 return 값을 구하는 메소드
- 상기 함수 이외 주어진 변수 또는 함수는 코드의 주석을 참고
- 터미널 환경에서 `py` 파일 실행 방법은 각자 확인

출력예시1 main.py 실행 (실행환경에 따라 한 줄에 표시되는 길이는 다를 수 있음)

```
[(' ', 40), ('-', 16), ('.', 15), ('(', 11), (')', 11), (':', 7), (':', 2), (''', 1)]
19
The first sentence of the paper: AN IMAGE IS WORTH 16X16 WORDS TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE.
The second sentence of the paper: While the Transformer architecture has become the defacto standard for natural language processing tasks its applications to computer vision remain limited.
The last second sentence of the paper: When trained on midsized datasets such as ImageNet without strong regularization these models yield modest accuracies of a few percentage points below ResNets of comparable size.
The last sentence of the paper: This seemingly discouraging outcome may be expected Transformers lack some of the inductive biases inherent to CNNs such as translation equivariance and locality and therefore do not generalize well when trained on insufficient amounts of data.
```

```
[(' ', 40), ('-', 16), ('.', 15), ('(', 11), (')', 11), (':', 7), (':', 2), (''', 1)]
19
The first sentence of the paper: AN IMAGE IS WORTH 16X16 WORDS TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE.
The second sentence of the paper: While the Transformer architecture has become the defacto standard for natural language processing tasks its applications to computer vision remain limited.
The last second sentence of the paper: When trained on midsized datasets such as ImageNet without strong regularization these models yield modest accuracies of a few percentage points below ResNets of comparable size.
The last sentence of the paper: This seemingly discouraging outcome may be expected Transformers lack some of the inductive biases inherent to CNNs such as translation equivariance and locality and therefore do not generalize well when trained on insufficient amounts of data.
```

출력예시2 scoring.py 실행

```
line1: SUCCESS
line2: SUCCESS
line3: SUCCESS
line4: SUCCESS
line5: SUCCESS
line6: SUCCESS
```

```
line1: SUCCESS
line2: SUCCESS
line3: SUCCESS
line4: SUCCESS
line5: SUCCESS
line6: SUCCESS
```