

Python

Production of programs that analyze Korean PDF and extract keywords

progress report #2

Date : 2023.12.10

Name : Haegeon Lee

ID : 183014

1. Introduction

1) Background

With the advent of the information age, the era of reading and interpreting a large amount of documents has arrived. If the amount of documents is vast when a particular document is first encountered, fatigue accumulates before reading. Therefore, if the Python program provides keywords that readers should focus on first, the speed increases rapidly and more efficient reading is possible. Therefore, the existence of these programs is necessary.

2) Project goal

It aims to develop a program that analyzes pdf to analyze what key keywords are based on the frequency of words and provides them to users.

3) Differences from existing programs

Existing programs are produced based on English, so there is a big difference in reading Korean pdf and it is impossible to use. Therefore, there is a difference from existing programs because we focus on extracting keywords from Korean pdf file for Koreans.

2. Functional Requirement

1) Function 1 – Extract text from file

- Extract the text file from the pdf file and store it in the list.

2) Function 2 - Remove non-critical words

- Remove investigations such as '은','는','이','가' from list token. And import Morpheme analyzer to make up a list with only nouns. We will also add the ability to remove words that are frequently used but are not important.

3) Function 3 - ranking based on frequency to output the top 15

- Saves the frequency of nouns in dictionary form and outputs the most written value.

4) Function 4

- We plan to save the result value as a txt file so that it can be used in the future.

3. progress

1) Implementation of a feature

(1) Extract text from file

- In

Pdf file

-out

Text(Words / variable)

- Explanation

After receiving pdf, import the txt file on a page-by-page basis from pdf and save it in text(variable).

- applied learning

For loop, package, file inout

- screen shot

```
from PyPDF2 import PdfReader

# 기능 1 pdf를 불러오는 패키지인 pyPDF2를 활용하여 pdf를 txt로 읽어옴
pdf_reader = PdfReader("./data/test4.pdf")
pages = pdf_reader.pages
text = ""
rank_word_list = []
for page in pages:
    text += page.extract_text()
```

(2) Remove non-critical words

- Explanation

Other are excluded and only alphabetic or numeric words are added to the word list.

Gets the list of words that are not important.

Put the rest in the list called yes_words, except for the list of words mixed with numbers and non-important words.

- applied learning

for loop, list comprehension

- screen shot

```
# 알파벳과 숫자로만 이루어진 단어를 words 리스트에 추가
num_and_words = [word for word in text.split() if word.isalnum()]

# 안쓰는 단어 리스트/ 여기에 추가하여 업데이트 가능
no_words = ['은', '는', '이', '가', '을', '를', '에', '에서', '도', '만', '뿐', '만큼', '여러분', '지금', '그리고', '이렇게', '그렇게', '그래서',
            '그러나', '으로', '.', ',', '의', '이', '등', '및', '수', '있는', '사무관', '위한', '통해', '있도록', '후', '등에']

# 숫자가 섞인 리스트 num_and_words랑 안쓰는 단어 리스트 no_words 제외하고
# 나머지를 yes_words 라는 리스트로 넣음
yes_words = []
for word in num_and_words:
    if not any(char.isdigit() for char in word) and word not in no_words:
        yes_words.append(word)
```

(3) ranking based on frequency to output the top 15

- In

Yes_words

-out

1~15 등 : (단어) n회

- Explanation

Find the frequency of words from the yes_words list and add them to the dictionary.

After sorting by frequency, it outputs up to the top 15 according to the format.

- applied learning

Dictionary, sorted, for loop

- screen shot

```
# 리스트 yes_words에서 각 단어 빈도수를 리스트를 통해서 딕셔너리에 추가
word_rank = {}
for word in yes_words:
    word_rank[word] = word_rank.get(word, 0) + 1

# 등장 횟수 순으로 정렬
sorted_word_rank = sorted(word_rank.items(), key=lambda x: x[1], reverse=True)

# 기능 3 단어의 빈도 수를 기준으로 랭킹을 선정하여 랭킹이 높은 순으로 정렬 후 상위 15위만 출력
rank = 1
for key, val in sorted_word_rank[:15]:
    print(f"{rank} 등 : {key} {val} 회")
    rank += 1
```

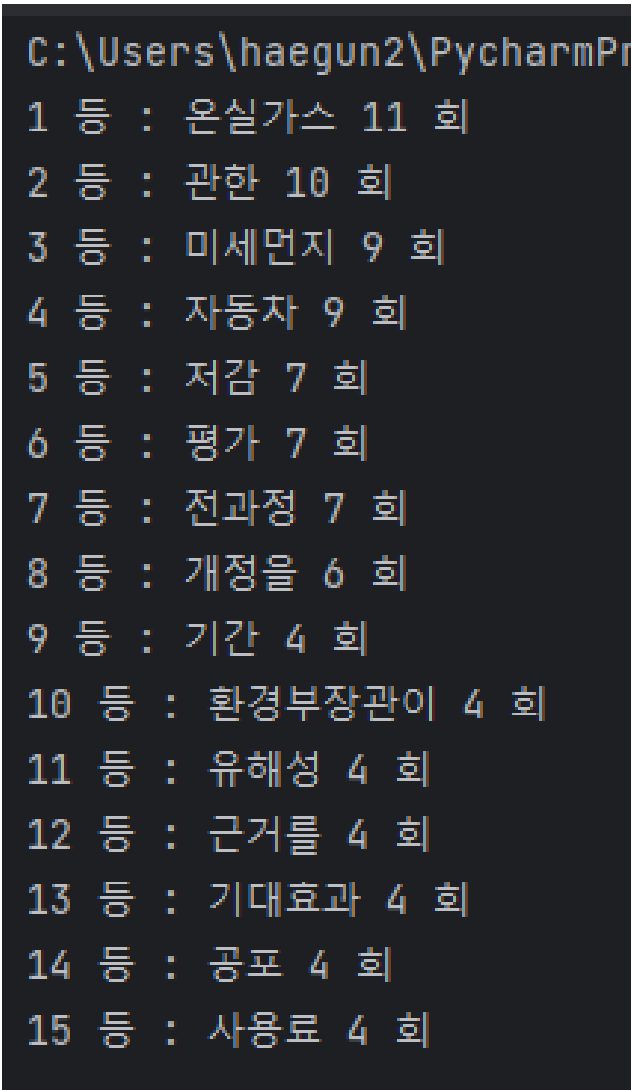
2) Test Result

(1) Feature Name Tested

- Explanation


Read the letter as txt from the input pdf, remove insignificant words, determine the frequency, and output to the top 15 depending on the format.

- Test Result screen shot



```
C:\Users\haegun2\PycharmPr
1 등 : 온실가스 11 회
2 등 : 관한 10 회
3 등 : 미세먼지 9 회
4 등 : 자동차 9 회
5 등 : 저감 7 회
6 등 : 평가 7 회
7 등 : 전과정 7 회
8 등 : 개정을 6 회
9 등 : 기간 4 회
10 등 : 환경부장관이 4 회
11 등 : 유해성 4 회
12 등 : 근거를 4 회
13 등 : 기대효과 4 회
14 등 : 공포 4 회
15 등 : 사용료 4 회
```

Example of pdf used for testing

 환경부	보도참고자료	다시 대한민국! 새로운 국민의 나라
보도시점	2023. 12. 8.(금) (배포 후 즉시)	배포 2023. 12. 8.(금)

미세먼지법 등 5개 환경법안 국회 통과

- 국민이 안심할 수 있는 깨끗하고 안전한 환경조성에 기여

환경부(장관 한화진)는 △‘미세먼지 저감 및 관리에 관한 특별법’, △‘대기환경보전법’, △‘화학물질의 등록 및 평가 등에 관한 법률’, △‘자원의 절약과 재활용촉진에 관한 법률’, △‘자연환경보전법’ 등 5개 환경법안이 12월 8일 국회 본회의를 통과했다고 밝혔다.

먼저, ‘미세먼지 저감 및 관리에 관한 특별법’은 미세먼지 배출저감 관리를 위해 초미세먼지(PM2.5) 월평균 농도가 심화되는 그해 12월 1일부터 이듬해 3월 31일까지의 기간 동안, 중앙행정기관과 지자체, 공공기관 등이 운영하는 공공배출시설에 대해 미세먼지 배출 저감조치를 시행하는 ‘미세먼지 계절관리제’를 적용해 왔다. 그러나 ‘미세먼지 계절관리제’ 기간 전후로도 지역별 초미세먼지 농도의 차이가 발생하고, 민간배출시설의 저감조치*는 의무적으로 적용되지 않는 등 실질적인 미세먼지 저감 효과를 기대하기에 일부 미흡한 측면이 있었다.

* 발전, 제철, 석유화학 등 대형사업장과 일부 중소규모 사업장이 자발적 협약을 통해 고농도 미세먼지 기간 저감대책에 참여 중

이에, 법률 개정을 통해 지역민의 건강 피해나 경제 영향 등을 고려하여, 시도지사가 필요 시 ‘미세먼지 계절관리제’ 기간을 연장할 수 있도록 하고, 환경부 장관의 미세먼지 저감조치 요청 대상을 공공배출시설에서 환경부령으로 정하는 민간배출시설까지 확대할 수 있도록 함에 따라 지역 특성에 보다 부합하고, 효과적인 미세먼지 배출 저감 효과를 기대할 수 있게 되었다.

‘대기환경보전법’은 개정을 통해 ‘자동차 온실가스 전과정 평가’의 정의 규정을 신설하고, 환경부장관이 관계부처와 함께 구체적인 평가 방법을 정하도록 했으며 자동차제작자에게 필요한 행정적·기술적 지원을 할 수 있는 근거가 마련됐다.

4. Changes to plan

1) Change History Subject

- before

Remove investigations such as '은','는','이','가' from list token. And import Morpheme analyzer to make up a list with only nouns.

- after

Write a code that removes words and investigations that you don't use directly without importing morpheme analyzers.

- reason

I was thinking of importing morpheme analyzers. However, since the analyzer to be imported works using the java environment, we found that it is possible only if a specific environment is established. There was a problem that it was not easily and conveniently available on any computer. Therefore, we didn't import the content, but we directly configured it to be simple and maintained.

In that case, the removal of frequently used but insignificant words can also be maintained from time to time.

5. What I'm considering to add

1) The ability to print and save a list according to frequency as a file so that you can check it at any time

2) Ability to load non-critical word lists from external files and modify and add them from external files (still undecided whether to implement)

6. Schedule

	11/3	11/10	11/26	12/4	12/10	12/14	12/23
Write Proposal	Finished						
Function 1		finished					
Function 2			finished				
Function 3				finished			
Function 4						----->	