정규식을 이용한 패턴찾기

2019년 6월 3일 월요일 오후 1:57

정규식의 이해

- 기호로 되어 있어 굉장히 효과적이지만 조금 어려움
- 한 번 배우면 활용할 곳이 많음
- 정규식은 그 자체로 하나의 언어입니다
- 특수(marker) 문자로 이루어진 언어로 문자만을 사용해서 프로그래밍을 하는 개념
- 축약된 '형식 언어'의 한 종류

❖ 정규식은 텍스트에서 특정 글자나 단어, 패턴 등을 정확하고 유동적으로 표현하는 식이다. 줄여서 regex나 regexp라고도 부르는데 정규식 처리기가 해석할 수 있도록 정해진 문법에 따라 사용하여야 한다.

❖ 정규식은 파이썬의 일부가 아니지만, 파이썬과 함께 쓰인다.

❖ 그렇기 때문에 import re를 통해 정규식 라이브러리를 가져와서 사용한다.

❖ 정규 표현식의 규칙

^	라인의 처음을 매칭
$	라인의 끝을 매칭
.	임의의 문자를 매칭 (와일드 카드)
\s	공백 문자를 매칭
\S	공백이 아닌 문자를 매칭
*	바로 앞선 문자에 적용되고 0 혹은 그 이상의 앞선 문자와 매칭을 표기함.
*?	바로 앞선 문자에 적용되고 0 혹은 그 이상의 앞선 문자와 매칭을 탐욕적이지 않은 방식으로 표기함.
+	바로 앞선 문자에 적용되고 1 혹은 그 이상의 앞선 문자와 매칭을 표기함
+?	바로 앞선 문자에 적용되고 1 혹은 그 이상의 앞선 문자와 매칭을 탐욕적이지 않은 방식으로 표기함.
[aeiou]	명세된 집합 문자에 존재하는 단일 문자와 매칭. "a", "e", "i", "o", "u" 문자만 매칭되는 예제
[a-z0-9]	- 기호로 문자 범위를 명세할 수 있다. 소문자이거나 숫자인 단일 문자만 매칭되는 예제.
( )	괄호가 정규표현식에 추가될 때, 매칭을 무시한다. 하지만 findall()을 사용 할 때 전체 문자열보다 매칭된 문자열의 상세한 부속 문자열을 추출할 수 있게 한다.

- re.search() 를 사용하면 find() 메소드를 쓴 것처럼 정규식에 매칭되는 문자열을 찾을 수 있음

- re.findall() 을 사용하면 정규식에 맞는 문자열 추출 가능
  (find() 와 slicing: var[5:10] 을 조합한 것과 유사)
  정규식에 매칭되는 부분 문자열을 모은 리스트를 리턴

  텍스트에서 문자 패턴 찾기

  다음 코드는 mbox-short.txt 파일에서 'From:'이라는 문자 패턴이 포함된 문장을 찾아 출력하는 프로그램입니다. 여기에서는 find() 메소드를 사용했습니다.

```
import re
hand = open('mbox-short.txt')
for line in hand:
   line = line.rstrip()
   if line.find('From:') >= 0:
      print(line)
```

  같은 프로그램을 정규표현식을 활용해 작성하면 다음과 같습니다.
  정규표현식을 사용하기 위해서는 re(regular expression) 모듈을 import 해야 하고, re.search()가 find() 메소드와 같은 역할을 해주는 부분입니다.

```
import re
hand = open('mbox-short.txt')
for line in hand:
   line = line.rstrip()
   if re.search('From:', line) :
      print(line)
```

  텍스트에서 시작 패턴 찾기

  이번에는 'From:'으로 시작하는 문장을 출력하는 프로그램입니다.

```
import re
hand = open('mbox-short.txt')
for line in hand:
   line = line.rstrip()
   if line.startswith('From:') :
      print(line)
```

  그리고 이것을 정규표현식으로 표현하려면 다음과 같이 '^'라는 특수 문자를 사용하면 됩니다.

```
import re
hand = open('mbox-short.txt')
for line in hand:
   line = line.rstrip()
   if re.search('^From:', line) :
      print(line)
```

  특수 문자를 활용한 문자 패턴 찾기

  방금 보신 것처럼 정규 표현식에서는 특수 문자를 사용할 수 있습니다. 그리고 방금 보셨던 '^'(캐럿 문자) 외에도 다음과 같은 다양한 특수 문자들이 있습니다.
  ^ : 문장의 시작을 의미
  . : 어떤 문자 한 글자
  * : 앞의 문자가 여러 번 반복될 수 있음을 의미
  + : 앞의 문자가 1번 이상 나타남을 의미
  \S : 공백 문자가 아닌 한 개의 문자

(\는 역슬래시와 같은 문자임)

따라서, 다음과 같은 문자열들은 모두 '^X.*:'라는 패턴을 통해 찾을 수 있습니다.

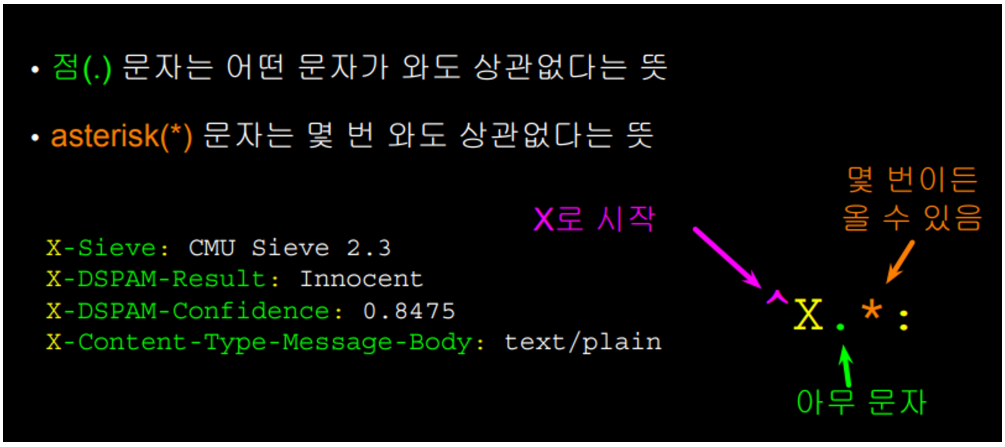
- X-Sieve: CMU Sieve 2.3
- X-DSPAM-Result: Innocent
- X-DSPAM-Confidence: 0.8475
- X-Content-Type-Message-Body: text/plain

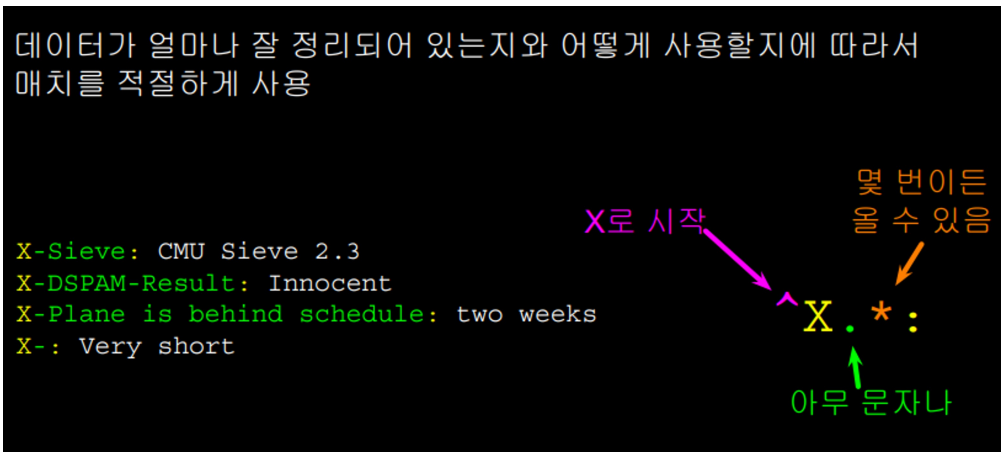
그리고 다음과 같은 문자열들은 '^X-\S+:' 패턴으로 찾을 수 있으며,

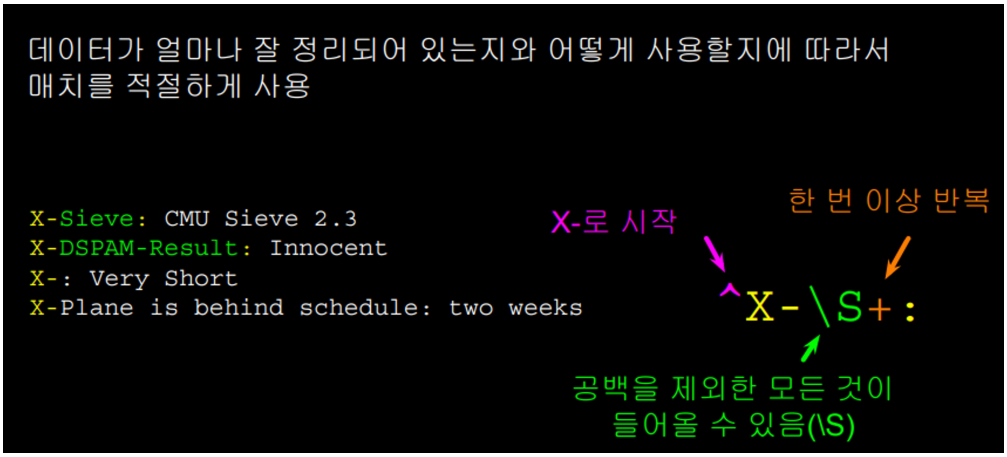
- X-Sieve: CMU Sieve 2.3
- X-DSPAM-Result: Innocent

다음의 문자열은 'X-'와 ':' 사이에 공백 문자가 아닌 문자가 포함되지 않았기 때문에 '^X-\S+:' 패턴으로 찾을 수 없습니다.

- X-: Very short
- X-Plane is behind schedule: two weeks







출처: <https://www.edwith.org/python-network-data/lecture/24451/>