Data Cleaning

Ki Hyun Kim

nlp.with.deep.learning@gmail.com

Preprocessing Workflow

데이터(코퍼스) 수집
- 구입, 외주
- 크롤링을 통한 수집

정제
- Task에 따른 노이즈 제거
- 인코딩 변환

레이블링(Optional)
- 문장마다 또는 단어마다 labeling 수행

Tokenization
- 형태소 분석기를 활용하여 분절 수행

Subword Segmentation (Optional)
- 단어보다 더 작은 의미 단위 추가 분절 수행

Batchify
- 사전 생성 및 word2index 맵핑 수행
- 효율화를 위한 전/후처리

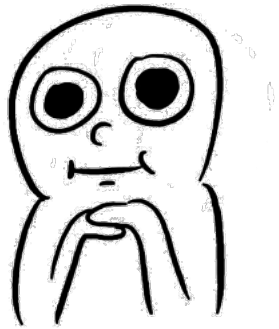
Two Steps

- 기계적인 노이즈 제거
  - 전각문자 변환
  - Task에 따른 (전형적인) 노이즈 제거

- Interactive 노이즈 제거
  - 코퍼스의 특성에 따른 노이즈 제거
  - 작업자가 상황을 확인하며 작업 수행

주의할 점

- Task에 따른 특성
  - 풀고자 하는 문제의 특성에 따라 전처리 전략이 다름
  - 신중한 접근이 필요
    - e.g. 이모티콘은 필요 없는 정보일까?



- 언어, 도메인, 코퍼스에 따른 특성
  - 각 언어, 도메인, 코퍼스 별 특성이 다르므로 다른 형태의 전처리 전략이 필요

전각문자 제거

- 유니코드 이전의 한글, 한자, 일본어는 전각 문자로 취급되었음

- 한자 등과 함께 표기된 반각 문자로 표기 가능한 전각 문자의 경우, 반각 문자로 치환
  - 중국어, 일본어 문서의 경우 많은 경우 전각 치환 필요
  - 오래된 한국어 문서의 경우 종종 전각 치환 필요

- 전각 문자의 예:
  - ！”＃＄％＆’（）　＊＋，－．／
  - ０１２３４５６７８９
  - ：；＜＝＞？＠
  - ＡＢＣＤＥＦＧＨＩＪＫＬＭＮＯＰＱＲＳＴＵＶＷＸＹＺ
  - ［＼］＾＿｀
  - ａｂｃｄｅｆｇｈｉｊｋｌｍｎｏｐｑｒｓｔｕｖｗｘｙｚ
  - ｛｜｝～

대소문자 통일(Optional)

- 코퍼스에 따라 대소문자 표기법이 다름
- 하나의 단어를 다양하게 표현하면 희소성이 높아짐
- 딥러닝의 시대에 오면서 필요성 하락 및 생략 가능

No	New York City
1	NYC
2	nyc
3	N.Y.C.
4	Nyc

정규식을 활용한 정제

- 정규식(regular expression)을 활용하면 복잡한 규칙의 노이즈도 제거/치환 가능

- 코딩 없이 단순히 텍스트 에디터(Sublime Text, VSCode 등)도 가능

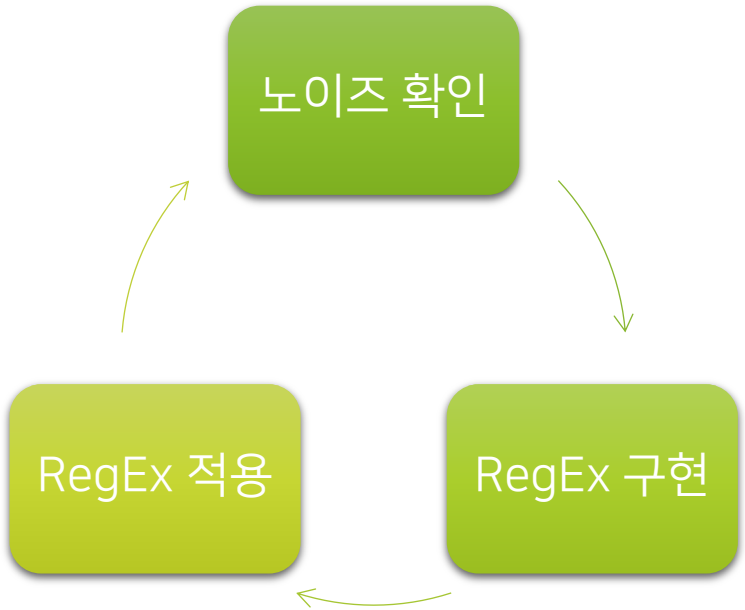
```
>>> import re
>>> regex = r"([\w]+\s*:?\s*)?\(?\+?([0-9]{1,3})?\-?[0-9]{2,3}(\)|\-)?[0-9]{3,4}\-?[0-9]{4
>>> x = "Ki:  +82-10-1234-5678 "
>>> re.sub(regex, "REMOVED", x)
'REMOVED'
>>> x = "CONTENT jiu  02)1234-5678 ."
>>> re.sub(regex, "REMOVED", x)
'CONTENT REMOVED'
```

Interactive 노이즈 제거 과정

- 규칙에 의해 노이즈를 제거하기 때문에,
  노이즈 전부를 제거하는 것은 어려움

- 따라서 반복적인 규칙 생성 및 적용 과정이 필요

- 끝이 없는 과정
  - 노력과 품질 사이의 trade-off
  - Sweet spot을 찾아야함



Summary

- 전처리 과정은 Task와 언어, 도메인과 코퍼스의 특성에 따라 다르다.

- 시간과 품질 사이의 trade-off

- 따라서 전처리 중에서도 특히 데이터 노이즈 제거의 경우, 많은 노하우가 필요