자연어처리_과제#7

1. 3쪽에서 생성된 naver_review.txt 파일을 읽어서 2쪽에 있는 데이터들을 계산. 단어의 수는 공백으로 분리된 어절의 수를 의미.

```
# 3쪽
import pandas as pd
import sentencepiece as spm
import urllib.request
import csv

urllib.request.urlretrieve("https://raw.githubusercontent.com/e9t/nsmc/mas
ter/ratings.txt", filename="ratings.txt")
naver_df = pd.read_table('ratings.txt')
naver_df = naver_df.dropna(how = 'any') # Null 값이 존재하는 행 제거
# 결과를 naver_review.txt 파일에 저장
with open('naver_review.txt', 'w', encoding='utf8') as f:
  f.write('\n'.join(naver_df['document']))
```

```
import re

# naver_review.txt 파일을 읽기
with open('naver_review.txt', 'r', encoding='utf8') as file:
    text = file.read()

# 문장 분리를 위한 정규식 패턴
pattern = re.compile(r'(?<=[.?!])\s*')

# 정규식을 사용하여 문장으로 분리
sentences = pattern.split(text)
sentences = [sentence.strip() for sentence in sentences if
sentence.strip()]  # 공백 문장 제거

# 문장 수 계산
sentence_count = len(sentences)

# 단어 분리 및 단어 수 계산
words = text.split()
word_count = len(words)

# 고유단어 수 계산
unique_words = set(words)
unique_word_count = len(unique_words)

# 결과 출력
print(f"단어 수: {word_count}")
print(f"문장 수: {sentence_count}")\
print(f"고유단어 수: {unique_word_count}")
```

```
단어 수: 1518208
문장 수: 388049
고유단어 수: 449791
```

---

2. 단어수를 10,000개와 20,000개로 늘려 4쪽의 sentencepiece를 실행하고 90001~90010번의 평에 대한 결과를 얻음.

단어 수 10,000개

```
import pandas as pd
import sentencepiece as spm

# 데이터 로드
naver_df = pd.read_table('ratings.txt')
naver_df = naver_df.dropna(how = 'any') # Null 값 제거

# 결과를 naver_review.txt 파일에 저장
with open('naver_review.txt', 'w', encoding='utf8') as f:
    f.write('\n'.join(naver_df['document']))

# SentencePiece 모델 훈련
spm.SentencePieceTrainer.Train('--input=naver_review.txt --
model_prefix=naver --vocab_size=10000 --model_type=bpe --
max_sentence_length=9999')

# 모델 로드
sp = spm.SentencePieceProcessor()
sp.load('naver.model')

# 인덱스 90001부터 90010까지의 리뷰 인코딩 결과 출력
for index in range(90001, 90011):
    review = naver_df.iloc[index]['document']  # 해당 인덱스의 리뷰 추출
    print(f"Review #{index}: {review}")
    print("Encoded Pieces:", sp.encode_as_pieces(review))
    print("Encoded IDs:", sp.encode_as_ids(review))
    print()
```

```
Review #90001: 쫌도와줘라.열심히 해보겠다는데 그리고 이 다음번엔 좀 재밌는걸 (흥행류)시도하시
길 이번껀 솔직히 당신 멋대로한거자나. 배려심이라곤 눈꼽만큼도없어.
Encoded Pieces: ['_쫌', '도', '와', '줘', '라', '.', '열', '심히', '_해', '보',
'겠다', '는데', '_그리고', '_이', '_다음', '번', '엔', '_좀', '_재밌는', '걸', '_(',
'흥', '행', '류', ')', '시', '도', '하시길', '_이번', '껀', '_솔직히', '_당신', '_
멋', '대로', '한거', '자나', '.', '_배려', '심', '이라', '곤', '_눈', '꼽', '만큼',
'도', '없어', '.']
Encoded IDs: [3366, 8286, 8386, 8788, 8302, 8276, 8670, 2360, 152, 8292,
```

```
268, 17, 320, 6, 1315, 8480, 8540, 146, 1045, 8452, 1548, 8652, 8536,
8625, 8650, 8311, 8286, 7052, 1541, 9401, 519, 1431, 265, 263, 932, 3670,
8276, 8002, 8441, 107, 8837, 184, 9038, 699, 8286, 1675, 8276]

Review #90002: 오션스시리즈중 단연 최고!!!
Encoded Pieces: ['_오', '션', '스', '시리즈', '중', '_단연', '_최고', '!!!']
Encoded IDs: [73, 8489, 8312, 1720, 8379, 3148, 66, 213]

Review #90003: 갠적으로 원조 액션영화랄까... 아련하다
Encoded Pieces: ['_갠적으로', '_원조', '_액션영화', '랄까', '...', '_아련', '하다']
Encoded IDs: [5514, 5652, 2994, 1951, 8, 2774, 79]

Review #90004: 아오이팬으로 보게된영화긴했지만 충분히 재밌었다, 세계인류독살 ㅋㅋ
Encoded Pieces: ['_아오', '이', '팬', '으로', '_보게된', '영화', '긴', '했지만', '_
충분히', '_재밌었다', ',', '_세계', '인', '류', '독', '살', '_ㅋㅋ']
Encoded IDs: [2098, 8277, 8784, 34, 6087, 4, 8505, 1169, 1648, 1833, 8315,
1446, 8308, 8625, 8427, 8494, 121]

Review #90005: 내 인생 이런 영화가 또 있을까? 일상에 갇혀버린 나 혹은 우리 모두에게 갇혀버린
일상에서 탈출하라는 경종을 울리는 그런 영화다
Encoded Pieces: ['_내', '_인생', '_이런', '_영화가', '_또', '_있을까', '?', '_일
상', '에', '_', '갇', '혀', '버린', '_나', '_혹은', '_우리', '_모두', '에게', '_',
'갇', '혀', '버린', '_일상', '에서', '_탈출', '하', '라는', '_경', '종', '을', '_울리
는', '_그런', '_영화다']
Encoded IDs: [27, 435, 80, 192, 259, 1453, 8329, 2831, 8288, 8275, 9528,
8640, 1134, 16, 4812, 291, 525, 249, 8275, 9528, 8640, 1134, 2831, 47,
5087, 8284, 260, 538, 8685, 8301, 5263, 325, 420]

Review #90006: 최윤영씨 넘이쁘네요
Encoded Pieces: ['_최', '윤', '영', '씨', '_넘', '이쁘', '네요']
Encoded IDs: [37, 8823, 8283, 8600, 220, 4302, 40]

Review #90007: 학교에서 빌려 봤는데 정말 재미있었음
Encoded Pieces: ['_학교에서', '_빌려', '_봤는데', '_정말', '_재미있었음']
Encoded IDs: [3663, 3816, 194, 43, 5797]

Review #90008: 이런 장르의 영화 다시안나오나?
Encoded Pieces: ['_이런', '_장르', '의', '_영화', '_다시', '안', '나오', '나',
'?']
Encoded IDs: [80, 1268, 8294, 5, 168, 8347, 504, 8289, 8329]

Review #90009: 마음이의 모성애 너무 감동적이네요 마음이 연기 짱!!!!!!!!!!
Encoded Pieces: ['_마음이', '의', '_모', '성애', '_너무', '_감동적이네요', '_마음이',
'_연기', '_짱', '!!!!!!!!!', '!!!']
Encoded IDs: [1172, 8294, 59, 2350, 24, 6107, 1172, 55, 396, 1085, 213]

Review #90010: 내 인생영화... 어릴때 보고 커서도 봤는데 진짜 재미있다....
Encoded Pieces: ['_내', '_인생영화', '...', '_어릴때', '_보고', '_커', '서도', '_
봤는데', '_진짜', '_재미있다', '....']
Encoded IDs: [27, 7953, 8, 1784, 104, 1130, 5531, 194, 54, 1499, 32]
```

단어 수 20,000개

```
import pandas as pd
import sentencepiece as spm

# 데이터 로드
naver_df = pd.read_table('ratings.txt')
naver_df = naver_df.dropna(how = 'any') # Null 값 제거

# 결과를 naver_review.txt 파일에 저장
with open('naver_review.txt', 'w', encoding='utf8') as f:
    f.write('\n'.join(naver_df['document']))

# SentencePiece 모델 훈련
spm.SentencePieceTrainer.Train('--input=naver_review.txt --
model_prefix=naver --vocab_size=20000 --model_type=bpe --
max_sentence_length=9999')

# 모델 로드
sp = spm.SentencePieceProcessor()
sp.load('naver.model')

# 인덱스 90001부터 90010까지의 리뷰 인코딩 결과 출력
for index in range(90001, 90011):
    review = naver_df.iloc[index]['document']  # 해당 인덱스의 리뷰 추출
    print(f"Review #{index}: {review}")
    print("Encoded Pieces:", sp.encode_as_pieces(review))
    print("Encoded IDs:", sp.encode_as_ids(review))
    print()
```

```
Review #90001: 쫌도와줘라.열심히 해보겠다는데 그리고 이 다음번엔 좀 재밌는걸 (흥행류)시도하시
길 이번껀 솔직히 당신 멋대로한거자나. 배려심이라곤 눈꼽만큼도없어.
Encoded Pieces: ['_쫌', '도와', '줘라', '.', '열심히', '_해보', '겠다', '는데', '_
그리고', '_이', '_다음', '번', '엔', '_좀', '_재밌는', '걸', '_(', '흥행', '류',
')', '시도', '하시길', '_이번', '껀', '_솔직히', '_당신', '_멋', '대로', '한거', '자
나', '.', '_배려', '심', '이라곤', '_눈', '꼽', '만큼도', '없어', '.']
Encoded IDs: [3366, 8901, 8766, 18276, 13930, 12875, 268, 17, 320, 6,
1315, 18480, 18540, 146, 1045, 18452, 1548, 11613, 18625, 18650, 8632,
7052, 1541, 19401, 519, 1431, 265, 263, 932, 3670, 18276, 8002, 18441,
9427, 184, 19038, 10961, 1675, 18276]

Review #90002: 오션스시리즈중 단연 최고!!!
Encoded Pieces: ['_오', '션스', '시리즈중', '_단연', '_최고', '!!!']
Encoded IDs: [73, 17677, 12357, 3148, 66, 213]

Review #90003: 갠적으로 원조 액션영화랄까... 아련하다
Encoded Pieces: ['_갠적으로', '_원조', '_액션영화', '랄까', '...', '_아련', '하다']
Encoded IDs: [5514, 5652, 2994, 1951, 8, 2774, 79]

Review #90004: 아오이팬으로 보게된영화긴했지만 충분히 재밌었다, 세계인류독살 ㅋㅋ
```

```
Encoded Pieces: ['_아오이', '팬', '으로', '_보게된', '영화', '긴했지만', '_충분히',
'_재밌었다', ',', '_세계', '인', '류', '독', '살', '_ㅋㅋ']
Encoded IDs: [11252, 18784, 34, 6087, 4, 10018, 1648, 1833, 18315, 1446,
18308, 18625, 18427, 18494, 121]

Review #90005: 내 인생 이런 영화가 또 있을까? 일상에 갇혀버린 나 혹은 우리 모두에게 갇혀버린
일상에서 탈출하라는 경종을 울리는 그런 영화다
Encoded Pieces: ['_내', '_인생', '_이런', '_영화가', '_또', '_있을까', '?', '_일
상', '에', '_갇', '혀', '버린', '_나', '_혹은', '_우리', '_모두에게', '_갇', '혀',
'버린', '_일상', '에서', '_탈출', '하라는', '_경', '종을', '_울리는', '_그런', '_영화
다']
Encoded IDs: [27, 435, 80, 192, 259, 1453, 18329, 2831, 18288, 9693,
18640, 1134, 16, 4812, 291, 12371, 9693, 18640, 1134, 2831, 47, 5087,
10975, 538, 17710, 5263, 325, 420]

Review #90006: 최윤영씨 넘이쁘네요
Encoded Pieces: ['_최', '윤', '영씨', '_넘', '이쁘', '네요']
Encoded IDs: [37, 18823, 9883, 220, 4302, 40]

Review #90007: 학교에서 빌려 봤는데 정말 재미있었음
Encoded Pieces: ['_학교에서', '_빌려', '_봤는데', '_정말', '_재미있었음']
Encoded IDs: [3663, 3816, 194, 43, 5797]

Review #90008: 이런 장르의 영화 다시안나오나?
Encoded Pieces: ['_이런', '_장르의', '_영화', '_다시', '안나오', '나', '?']
Encoded IDs: [80, 8853, 5, 168, 11696, 18289, 18329]

Review #90009: 마음이의 모성애 너무 감동적이네요 마음이 연기 짱!!!!!!!!!!!
Encoded Pieces: ['_마음이', '의', '_모성애', '_너무', '_감동적이네요', '_마음이', '_
연기', '_짱', '!!!!!!!!!!!']
Encoded IDs: [1172, 18294, 8570, 24, 6107, 1172, 55, 396, 13051]

Review #90010: 내 인생영화... 어릴때 보고 커서도 봤는데 진짜 재미있다....
Encoded Pieces: ['_내', '_인생영화', '...', '_어릴때', '_보고', '_커', '서도', '_
봤는데', '_진짜', '_재미있다', '....']
Encoded IDs: [27, 7953, 8, 1784, 104, 1130, 5531, 194, 54, 1499, 32]
```

---

3. naver_review.txt 파일 내용에 대해 Okt 형태소 분석기를 실행시킴. 결과에서 나타난 고유단어수를 계산. 90001~90010번의 평에 대한 결과에 대해 Okt를 실행하고 위 2번의 결과와 비교함.

Okt 형태소 분석기 실행 후 고유단어수 계산

```
from konlpy.tag import Okt
import pandas as pd

okt = Okt()

unique_morphs = set()

# 파일을 줄 단위로 읽기
```

```
with open('naver_review.txt', 'r', encoding='utf8') as file:
    for line in file:
        morphs = okt.morphs(line.strip())  # 각 줄에 대한 형태소 분석
        unique_morphs.update(morphs)  # 고유 형태소 집합에 추가

unique_morph_count = len(unique_morphs)

print(f"전체 텍스트의 고유 형태소 수: {unique_morph_count}")
```

```
전체 텍스트의 고유 형태소 수: 122828
```

90001~90010번의 평에 대한 결과에 Okt 실행 & 2번과 비교

```
import sentencepiece as spm

# 모델 로드
sp = spm.SentencePieceProcessor()
sp.load('naver.model')

# 데이터 로드
naver_df = pd.read_table('ratings.txt')
naver_df = naver_df.dropna(how='any')

# 인덱스 90001부터 90010까지의 리뷰에 대한 Okt 형태소 분석 및 SentencePiece 인코딩 결과 비교
for index in range(90001, 90011):
    review = naver_df.iloc[index]['document']
    print(f"Review #{index}: {review}")

    # Okt 형태소 분석 결과
    okt_result = okt.morphs(review)
    print("Okt Morphs:", okt_result)

    # SentencePiece 인코딩 결과
    sp_pieces = sp.encode_as_pieces(review)
    sp_ids = sp.encode_as_ids(review)
    print("SentencePiece Encoded Pieces:", sp_pieces)
    # print("SentencePiece Encoded IDs:", sp_ids)
    print()
```

```
Review #90001: 쫌도와줘라.열심히 해보겠다는데 그리고 이 다음번엔 좀 재밌는걸 (흥행류)시도하시길 이번껀 솔직히 당신 멋대로한거자나. 배려심이라곤 눈꼽만큼도없어.
Okt Morphs: ['쫌', '도와줘라', '.', '열심히', '해보겠다는데', '그리고', '이', '다음', '번', '엔', '좀', '재밌는걸', '(', '흥행', '류', ')', '시도', '하시길', '이번', '껀', '솔직히', '당신', '멋대로', '한거자', '나', '.', '배려', '심', '이라곤', '눈꼽', '만큼도', '없어', '.']
SentencePiece Encoded Pieces: ['_쫌', '도와', '줘라', '.', '열심히', '_해보', '겠
```

```
다', '는데', '_그리고', '_이', '_다음', '번', '엔', '_좀', '_재밌는', '걸', '_(',
'흥행', '류', ')', '시도', '하시길', '_이번', '껀', '_솔직히', '_당신', '_멋', '대
로', '한거', '자나', '.', '_배려', '심', '이라곤', '_눈', '꼽', '만큼도', '없어',
'.']

Review #90002: 오션스시리즈중 단연 최고!!!
Okt Morphs: ['오션스', '시리즈', '중', '단연', '최고', '!!!']
SentencePiece Encoded Pieces: ['_오', '션스', '시리즈중', '_단연', '_최고',
'!!!']

Review #90003: 갠적으로 원조 액션영화랄까... 아련하다
Okt Morphs: ['갠', '적', '으로', '원조', '액션영화', '랄', '까', '...', '아련하다']
SentencePiece Encoded Pieces: ['_갠적으로', '_원조', '_액션영화', '랄까', '...',
'_아련', '하다']

Review #90004: 아오이팬으로 보게된영화긴했지만 충분히 재밌었다, 세계인류독살 ㅋㅋ
Okt Morphs: ['아오이', '팬', '으로', '보게', '된', '영화', '긴', '했지만', '충분히',
'재밌었다', ',', '세계', '인류', '독살', 'ㅋㅋ']
SentencePiece Encoded Pieces: ['_아오이', '팬', '으로', '_보게된', '영화', '긴했지
만', '_충분히', '_재밌었다', ',', '_세계', '인', '류', '독', '살', '_ㅋㅋ']

Review #90005: 내 인생 이런 영화가 또 있을까? 일상에 갇혀버린 나 혹은 우리 모두에게 갇혀버린
일상에서 탈출하라는 경종을 울리는 그런 영화다
Okt Morphs: ['내', '인생', '이런', '영화', '가', '또', '있을까', '?', '일상', '에',
'갇혀', '버린', '나', '혹은', '우리', '모두', '에게', '갇혀', '버린', '일상', '에서',
'탈출', '하', '라는', '경종', '을', '울리는', '그런', '영화', '다']
SentencePiece Encoded Pieces: ['_내', '_인생', '_이런', '_영화가', '_또', '_있을
까', '?', '_일상', '에', '_갇', '혀', '버린', '_나', '_혹은', '_우리', '_모두에게',
'_갇', '혀', '버린', '_일상', '에서', '_탈출', '하라는', '_경', '종을', '_울리는', '_
그런', '_영화다']

Review #90006: 최윤영씨 넘이쁘네요
Okt Morphs: ['최윤영', '씨', '넘', '이쁘네요']
SentencePiece Encoded Pieces: ['_최', '윤', '영씨', '_넘', '이쁘', '네요']

Review #90007: 학교에서 빌려 봤는데 정말 재미있었음
Okt Morphs: ['학교', '에서', '빌려', '봤는데', '정말', '재미있었음']
SentencePiece Encoded Pieces: ['_학교에서', '_빌려', '_봤는데', '_정말', '_재미있었
음']

Review #90008: 이런 장르의 영화 다시안나오나?
Okt Morphs: ['이런', '장르', '의', '영화', '다시안나오나', '?']
SentencePiece Encoded Pieces: ['_이런', '_장르의', '_영화', '_다시', '안나오',
'나', '?']

Review #90009: 마음이의 모성애 너무 감동적이네요 마음이 연기 짱!!!!!!!!!!
Okt Morphs: ['마음', '이의', '모성애', '너무', '감동', '적이네요', '마음', '이', '연
기', '짱', '!!!!!!!!!!']
SentencePiece Encoded Pieces: ['_마음이', '의', '_모성애', '_너무', '_감동적이네
요', '_마음이', '_연기', '_짱', '!!!!!!!!!!']

Review #90010: 내 인생영화... 어릴때 보고 커서도 봤는데 진짜 재미있다....
Okt Morphs: ['내', '인생', '영화', '...', '어릴', '때', '보고', '커서', '도', '봤는
데', '진짜', '재미있다', '....']
```

```
SentencePiece Encoded Pieces: ['_내', '_인생영화', '...', '_어릴때', '_보고', '_
커', '서도', '_봤는데', '_진짜', '_재미있다', '....']
```

---

4. 위의 1~3의 결과와 무관하게 새로 <구어체(2).txt> 파일에서 영어와 한국어에 대해 sentencepiece를 적용하여 32,000단어를 추출함. 이 경우 영어와 한국어 단어를 각각 32,000개씩 추출함. <구어체(2)> 파일에서 한국어 부분에는 영어 단어도 포함되어 있으니 영어를 제외하고 한국어 단어만을 사용하여야 함.

5. <구어체(2)> 문장 중 110,500~110,510 번째 문장에 대해 sentencepiece를 수행하고 그 결과를 구함.

영어와 한국어에 대해 각각 하나의 코드로 묶어 4번과 5번을 한 번에 구현했다.

영어 단어 추출

```
import pandas as pd
import sentencepiece as spm
import urllib.request
import csv
import re

# 한국어 제거 함수
def remove_korean(text):
    return re.sub(r'[\uAC00-\uD7AF]', '', text)  # 한글 유니코드 범위

# 파일 전체 읽기
input_file = '구어체(2).txt'
output_file = 'en_gu.txt'

# 파일 읽기 및 한국어 제거
with open(input_file, 'r', encoding='cp949') as infile:
    content = infile.read()
cleaned_content = remove_korean(content)

# 한국어 제거된 결과 저장
with open(output_file, 'w', encoding='utf-8') as outfile:
    outfile.write(cleaned_content)

# 텍스트 파일을 데이터프레임으로 로드
with open(output_file, 'r', encoding='utf-8') as f:
    lines = f.readlines()
    df = pd.DataFrame(lines, columns=['document'])

# SentencePiece 모델 학습
spm.SentencePieceTrainer.Train('--input=en_gu.txt --model_prefix=en --
vocab_size=32000 --model_type=bpe --max_sentence_length=9999')

# 특정 리뷰(110,500~110,510) 선택
target_reviews = df.iloc[110500:110511]
```

```
# SentencePiece 모델 로드
sp = spm.SentencePieceProcessor()
vocab_file = "en.model"
sp.load(vocab_file)

# 선택된 리뷰에 대해 SentencePiece 토큰화 및 ID 변환 수행
for index, row in target_reviews.iterrows():
    text = row['document'].strip()
    print(f"Review ID {index}: {text}")
    print("토큰화:", sp.encode_as_pieces(text))
    print("ID 변환:", sp.encode_as_ids(text))
    print()
```

```
Review ID 110500: . "There are times when I feel really sad, but I think
those are also my work to overcome."
토큰화: ['_.', '_"', 'There', '_are', '_times', '_when', '_I', '_feel',
'_really', '_sad', ',', '_but', '_I', '_think', '_those', '_are', '_also',
'_my', '_work', '_to', '_overcome', '."']
ID 변환: [6, 28, 1009, 97, 1227, 246, 18, 471, 550, 1836, 31958, 217, 18,
264, 1180, 97, 440, 104, 249, 22, 4567, 44]

Review ID 110501: . He strangely has nothing in common with me.
토큰화: ['_.', '_He', '_strangely', '_has', '_nothing', '_in', '_common',
'_with', '_me', '.']
ID 변환: [6, 491, 11116, 257, 1644, 41, 2072, 92, 82, 31943]

Review ID 110502: . It is a seriously nice day.
토큰화: ['_.', '_It', '_is', '_a', '_seriously', '_nice', '_day', '.']
ID 변환: [6, 168, 46, 4, 4905, 1248, 435, 31943]

Review ID 110503: . The really serious problem is that they gnaw at the
roots of the plants and the carp that lays the eggs on the roots of the
plants can't.
토큰화: ['_.', '_The', '_really', '_serious', '_problem', '_is', '_that',
'_they', '_g', 'na', 'w', '_at', '_the', '_roots', '_of', '_the',
'_plants', '_and', '_the', '_carp', '_that', '_l', 'ays', '_the', '_eggs',
'_on', '_the', '_roots', '_of', '_the', '_plants', '_can', "'", 't', '.']
ID 변환: [6, 85, 550, 2228, 565, 46, 78, 371, 51, 2011, 31952, 120, 10,
11011, 45, 10, 4372, 52, 10, 14191, 78, 48, 284, 10, 5057, 71, 10, 11011,
45, 10, 4372, 110, 31960, 31934, 31943]

Review ID 110504: . It's a wonderful exhibition.
토큰화: ['_.', '_It', "'", 's', '_a', '_wonderful', '_exhibition', '.']
ID 변환: [6, 168, 31960, 31939, 4, 3266, 2660, 31943]

Review ID 110505: . It is so embarrassing but I have something I want to
say.
토큰화: ['_.', '_It', '_is', '_so', '_embarrassing', '_but', '_I', '_have',
'_something', '_I', '_want', '_to', '_say', '.']
ID 변환: [6, 168, 46, 130, 8407, 217, 18, 103, 830, 18, 200, 22, 632, 31943]
```

```
Review ID 110506: "  ,  ?"  "It's so unpredictable, isn't it?"
토큰화: ['_"', '_,', '_?"', '_"', 'It', "'", 's', '_so', '_unpredictable',
',', '_isn', "'", 't', '_it', '?"']
ID 변환: [28, 80, 588, 28, 622, 31960, 31939, 130, 16904, 31958, 1826,
31960, 31934, 70, 350]

Review ID 110507: . "It felt like seeing old friends together, like
sisters even."
토큰화: ['_.', '_"', 'It', '_felt', '_like', '_seeing', '_old', '_friends',
'_together', ',', '_like', '_sisters', '_even', '."']
ID 변환: [6, 28, 622, 1664, 181, 3051, 948, 564, 829, 31958, 181, 7059, 562,
44]

Review ID 110508: . What a leisurely afternoon I have had in a long time.
토큰화: ['_.', '_What', '_a', '_leisurely', '_afternoon', '_I', '_have',
'_had', '_in', '_a', '_long', '_time', '.']
ID 변환: [6, 355, 4, 16786, 2569, 18, 103, 428, 41, 4, 527, 204, 31943]

Review ID 110509: . It has been a while.
토큰화: ['_.', '_It', '_has', '_been', '_a', '_while', '.']
ID 변환: [6, 168, 257, 361, 4, 776, 31943]

Review ID 110510: . It has been a long time since the last visit.
토큰화: ['_.', '_It', '_has', '_been', '_a', '_long', '_time', '_since',
'_the', '_last', '_visit', '.']
ID 변환: [6, 168, 257, 361, 4, 527, 204, 594, 10, 508, 708, 31943]
```

한국어 단어 추출

```python
import pandas as pd
import sentencepiece as spm
import urllib.request
import csv
import re

# 영어 제거 함수
def remove_english(text):
    return re.sub(r'[a-zA-Z]', '', text)

# 파일 전체 읽기
input_file = '구어체(2).txt'
output_file = 'kr_gu.txt'

# 파일 읽기 및 영어 제거
with open(input_file, 'r', encoding='cp949') as infile:
    content = infile.read()
cleaned_content = remove_english(content)

# 영어 제거된 결과 저장
with open(output_file, 'w', encoding='utf-8') as outfile:
    outfile.write(cleaned_content)
```

```
# 텍스트 파일을 데이터프레임으로 로드
with open(output_file, 'r', encoding='utf-8') as f:
    lines = f.readlines()
    df = pd.DataFrame(lines, columns=['document'])

# SentencePiece 모델 학습
spm.SentencePieceTrainer.Train('--input=kr_gu.txt --model_prefix=kr --
vocab_size=32000 --model_type=bpe --max_sentence_length=9999')


# 특정 리뷰(110,500~110,510) 선택
target_reviews = df.iloc[110500:110511]

# SentencePiece 모델 로드
sp_kr = spm.SentencePieceProcessor()
vocab_file = "kr.model"
sp_kr.load(vocab_file)

# 선택된 리뷰에 대해 SentencePiece 토큰화 및 ID 변환 수행
for index, row in target_reviews.iterrows():
    text = row['document'].strip()
    print(f"Review ID {index}: {text}")
    print("토큰화:", sp_kr.encode_as_pieces(text))
    print("ID 변환:", sp_kr.encode_as_ids(text))
    print()
```

```
Review ID 110500: 정말 슬플 때도 있지만 그것 또한 제가 극복해야 할 과제라고 생각합니다. "
,          ."
토큰화: ['_정말', '_슬플', '_때도', '_있지만', '_그것', '_또한', '_제가', '_극복해야',
'_할', '_과제', '라고', '_생각합니다', '.', '_"', '_,', '_."']
ID 변환: [232, 12621, 3036, 1781, 152, 573, 70, 25497, 99, 4226, 185, 804,
30765, 5, 6, 8]

Review ID 110501: 정말 신기하게도 공통점이 전혀 없는 형이에요.         .
토큰화: ['_정말', '_신기', '하게도', '_공통점이', '_전혀', '_없는', '_형', '이에요',
'.', '_.']
ID 변환: [232, 5845, 5536, 19822, 1365, 623, 587, 105, 30765, 3]

Review ID 110502: 정말 심각하게 좋은 날이네요.       .
토큰화: ['_정말', '_심각하게', '_좋은', '_날이', '네요', '.', '_.']
ID 변환: [232, 12112, 238, 2551, 369, 30765, 3]

Review ID 110503: 정말 심각한 문제는 이들이 식물의 뿌리를 갉아 먹어서 식물의 뿌리에 알을 낳는
붕어가 알을 낳지 못해요.                                    '.
토큰화: ['_정말', '_심각한', '_문제는', '_이들이', '_식물의', '_뿌리를', '_', '갉',
'아', '_먹어서', '_식물의', '_뿌', '리에', '_알을', '_낳', '는', '_붕', '어가', '_알
을', '_낳', '지', '_못해요', '.', "_."]
ID 변환: [232, 3699, 1961, 9613, 29609, 23377, 30764, 0, 30795, 10803,
29609, 2703, 985, 21552, 4689, 30769, 9915, 1299, 21552, 4689, 30777,
3631, 30765, 2318]
```

```
Review ID 110504: 정말 아름다운 전시회입니다.  '   .
토큰화: ['_정말', '_아름다운', '_전시회', '입니다', '.', "_'", '_.']
ID 변환: [232, 1574, 9620, 27, 30765, 10, 3]

Review ID 110505: 정말 염치없지만 말하고 싶은 게 있습니다.                .
토큰화: ['_정말', '_염', '치', '없', '지만', '_말하고', '_싶은', '_게', '_있습니다',
'.', '_.']
ID 변환: [232, 2592, 30909, 30864, 78, 3315, 636, 198, 49, 30765, 3]

Review ID 110506: "정말 예측 불가능하죠, 그렇지 않아요?"     "'  , ' ?"
토큰화: ['_"', '정말', '_예측', '_불가능', '하죠', ',', '_그렇지', '_않아요', '?"',
'_"\'', '_,', "_'", '_?"']
ID 변환: [5, 5954, 3453, 1860, 4392, 30773, 2862, 712, 320, 302, 6, 10, 179]

Review ID 110507: 정말 오랜 친구인 것처럼 느꼈고 자매처럼 보였어요. "       ,   ."
토큰화: ['_정말', '_오랜', '_친구인', '_것처럼', '_느꼈고', '_자매', '처럼', '_보였어
요', '.', '_"', '_,', '_."']
ID 변환: [232, 1789, 15078, 1434, 20594, 20394, 459, 7781, 30765, 5, 6, 8]

Review ID 110508: 정말 오랜만에 느끼는 한가한 오후입니다.           .
토큰화: ['_정말', '_오랜만에', '_느끼는', '_한가한', '_오후', '입니다', '.', '_.']
ID 변환: [232, 3763, 4447, 23656, 1333, 27, 30765, 3]

Review ID 110509: 정말 오랜만에 연락드리네요.      .
토큰화: ['_정말', '_오랜만에', '_연락', '드리', '네요', '.', '_.']
ID 변환: [232, 3763, 351, 1285, 369, 30765, 3]

Review ID 110510: 정말 오랜만에 찾아뵙게 되었습니다.              .
토큰화: ['_정말', '_오랜만에', '_찾아뵙', '게', '_되었습니다', '.', '_.']
ID 변환: [232, 3763, 11403, 30799, 814, 30765, 3]
```