연구윤리 및 연구지도
(8주차 보고서)

경북대학교 전자공학부
2016113566 김남영

1) 문장을 생성하는 신경망 구현 (코드 한줄씩 분석)

```python
import sys
sys.path.append('..')
import numpy as np
from common.functions import softmax
from rnnlm import Rnnlm


class RnnlmGen(Rnnlm): # Rnnlm을 상속하는 class
    def generate(self, start_id, skip_ids=None, sample_size=100):
        #start_id: 최초로 주는 단어의 ID, skip_ids: 샘플링 하고싶지않은 단어의 ID
        #Sample_size: 샘플링하는 단어의 수
        word_ids = [start_id] #샘플링해서 점점 id를 추가할 예정

        x = start_id
        while len(word_ids) < sample_size: #리스트 갯수를 100개 이하로 한정
            x = np.array(x).reshape(1, 1) #2차원배열로 reshape
            score = self.predict(x) #각 단어별 점수를 만듬(정규화 되기 전)
            p = softmax(score.flatten())  # 소프트맥스 함수로 정규화 (확률분포가 됨)

            sampled = np.random.choice(len(p), size=1, p=p)
            #확률분포를 통해 높은확률의 단어는 자주 선택되고 낮은확률의 단어는 희소하게 선택됨(랜덤으로)
            if (skip_ids is None) or (sampled not in skip_ids):
                x = sampled
                word_ids.append(int(x)) # 차례차례 선택된 id 추가

        return word_ids

    def get_state(self):
        return self.lstm_layer.h, self.lstm_layer.c

    def set_state(self, state):
        self.lstm_layer.set_state(*state)
```

```
common.functions
you thwart mounted sparked village although
candela demonstration edt ways leery motive
significance allegedly intervention fiscal
achievement billion-dollar digest
unauthorized wayne computer-guided robot
likelihood increase deferred settled dive
repurchase d'arcy byrd tiny product gray
ventures professors biological renault
stearns while giving flat-rolled violate
explains lion expense bowes stressed
hispanics eating cypress nation kenneth jury
n.v. rush motor okla. skyrocketed
possibilities formal smooth marketers nih
information most sam expect propaganda jets
seeds consulting hhs column track tests turf
easily disappearance stem hazards rescue
undisclosed harder milton quarters
oversubscribed rewards giants shamir sam
bolster breakdown edt lighting searle poverty
township father underwriting

In [4]:
```

➔가중치 학습이 되지 않았을 때 결과

```
common.base_model, common.util
you needed until leaving featured a strong
business.
 they are backed for dr. culmination.
 mr. conn. said he will retire with his
restructuring.
 the system eliminated this kind of
investigation cheating privatization.
 the measurements do n't need again into
question the midst of a three feet an
response for the depression.
 but kind of washington argues mr. drabinsky
is attached to the securities.
 mr. broderick 's campaign that air & co.
agreed to make a bid for its sci tv stake
soon by the balls bid after mr. roman took
control of the international

In [15]:
```

➔학습된 가중치일 때 결과(비교적 정돈된 문장)

2) generate_better_text.py 파일 분석 및 실행결과 확인 (코드 한줄씩 분석)

```python
import sys
sys.path.append('..')
from common.np import *
from rnnlm_gen import BetterRnnlmGen
from dataset import ptb


corpus, word_to_id, id_to_word = ptb.load_data('train')
vocab_size = len(word_to_id)
corpus_size = len(corpus)
#ptb에 들어있는 data를 적용

model = BetterRnnlmGen() #BetterRnnlmGen을 모델로 사용
model.load_params('../dataset/BetterRnnlm.pkl') #학습된 가중치가 저장된 pkl파일 이용

# start 문자와 skip 문자 설정
start_word = 'you' #시작단어를 you로 설정
start_id = word_to_id[start_word] # word를 ID로 바꾸어 start_id에 저장
skip_words = ['N', '<unk>', '$'] # 샘플링 하지않을 단어 설정
skip_ids = [word_to_id[w] for w in skip_words] #샘플링 하지않을 단어를 ID로 변환하여 저장
# 문장 생성
word_ids = model.generate(start_id, skip_ids) #모델을 통해 word_ids 리스트에 순서대로 저장
txt = ' '.join([id_to_word[i] for i in word_ids]) #id를 word로 변환한 뒤 공백을 두고 txt에 저장
txt = txt.replace(' <eos>', '.\n')

print(txt)
```

```
common.base_model, common.util
you have been moving over the scandal for
respect to the conservative people.
 this a large majority of the investment 's
stores believes a price breakers would be
reflects a expansion and their times is done.
 at the end of the year.
 sales are less equal by their realize all
the problems over the next three years and he
added.
 going in the meantime too many borrowers and
southern companies set down a new tax.
 but his business has only a valuable
performance of the underlying stock market
and thereby raise reasonable profit.
 he
----------------------------------------------
```

➔ 학습된 가중치를 사용했고, 시작단어를 you 로 설정했을 때 결과

```
model.reset_state() #모델을 reset

start_words = 'the meaning of life is' #시작단어를 바꿔서 한번 더 실행
start_ids = [word_to_id[w] for w in start_words.split(' ')]
# 시작단어의 공백을 기준으로 나눠서 id로 바꾼 뒤 start_ids에 저장

for x in start_ids[:-1]:
    x = np.array(x).reshape(1, 1) #predict를 위해 2차원배열로 바꾸는 과정
    model.predict(x) #이 과정은 왜 거치는지?

word_ids = model.generate(start_ids[-1], skip_ids)
# start_ids[-1]이 is에 대한 id값이므로 is를 첫 단어로 generate 하겠다는 뜻
word_ids = start_ids[:-1] + word_ids # 기존의 the meaning of life is 를 덧붙여서 전체 word 완성
txt = ' '.join([id_to_word[i] for i in word_ids]) #id를 word로 변환한 뒤 txt에 저장
txt = txt.replace(' <eos>', '.\n')
print('-' * 50) #구분선
print(txt)
```

➔ 모델을 reset 후 시작 단어를 the meaning of life is 로 바꾸어 다시 문장 생성

```
the meaning of life is to be used with
private franchisers.
 big three offerings are some strong and
witnesses.
 market opponents concentrated in the u.s.
year not only as it came down and so all that
they were approached by japanese american
express have closed and triple working
compensation any market and so come.
 at a recent moment opec has been planning
since u.s. operations restructuring as
determined by some local companies from
japanese companies.
 but them with international business
machines corp. the world 's largest insurance
maker were tokyo operator of the military
industry of seventh avenue.
 the

In [22]:
```

➔ 언뜻보기에는 그럴싸한 결과가 나왔지만 의미 있는 문장이 되지는 않았음.

고찰

1. 나쁘지 않은 결과이지만 완전한 문장을 생성했다고는 볼 수 없을 것 같다. GAN(적대적 생성 신경망)을 이용하면 점점 더 나은 결과로 나아가지 않을까?
2. 코드를 한줄씩 분석할 순 있지만 아무것도 없는 상태에서 내가 코드를 쓸 수 있을지 고민이다.