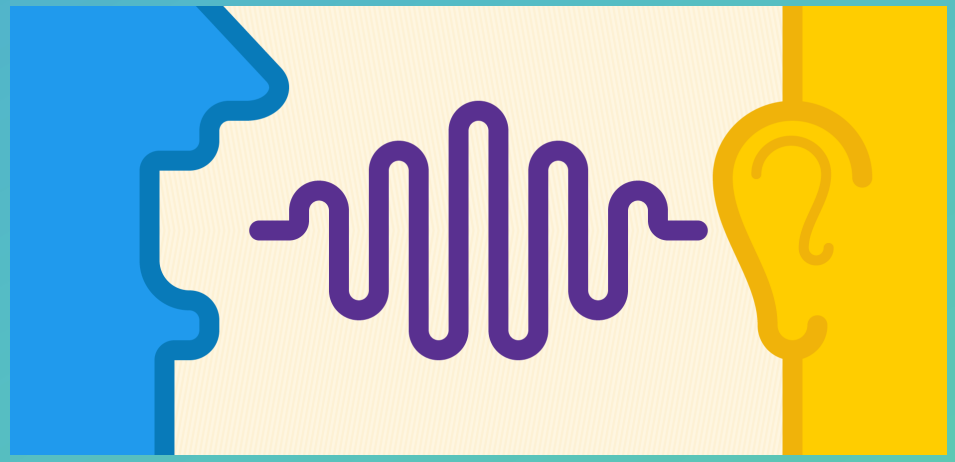
Multi-Speaker 음성합성

PROJECT1

End to End 모델을 활용한 Multi-Speaker 음성합성

01

ABOUT PROJECT



혁신성장 청년인재 실무 프로젝트상 2위

2020.10.01 ~ 2020.10.22

Multi-Speaker 음성합성

Tacotron 모델과 DeepVoice2 모델을 기반으로
한국어 TTS(Text To Speech)를 구현하는 프로젝트

Github 👉 https://github.com/jyshin0926/KoreanTTS

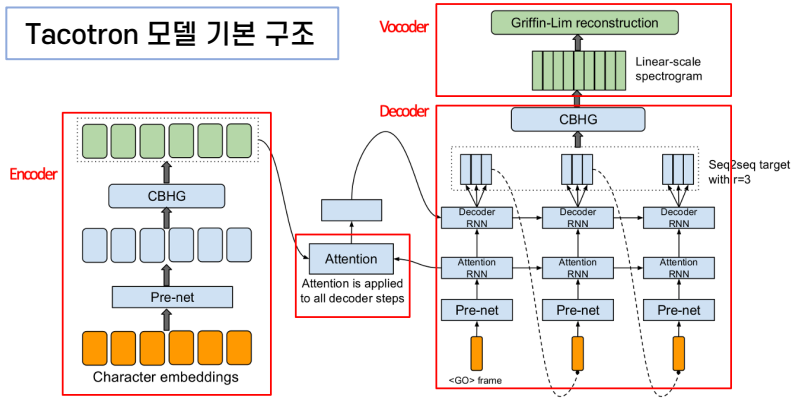
음성합성은 텍스트로 입력한 내용을 음성으로 바꿔주는 기술입니다. 음성합성 기술을 심도있게 이해하고, 목표하는 음성으로 재치있는 '밈(Meme)' 서비스를 제공하기 위해 해당 프로젝트를 진행했습니다.

✓ 노이즈 제거 알고리즘으로 데이터를 가공하고, DeepVoice2 모델의 multi-speaker를 활용하여 학습

✓ 대회 기간과 학습 시간을 고려하여 계산량이 효율적인 griffin-lim 알고리즘으로 90000 step 까지 학습

✓ 멀티캠퍼스의 혁신성장 청년인재 집중양성사업 실무 프로젝트 대회에서 2위를 수상하였습니다.

Tacotron 모델 기본 구조



담당 역할

3주라는 기간 내에 결과물을 완성시켰어야하는 대회의 특성상 **학습시간과 성능을 모두 고려**하는 것이 중요했습니다. 리더로서 먼저 **데이터와 모델의 특징을 이해하고 팀원들에게 효율적인 역할 분담**이 되도록 노력했습니다.

✓ **특징 ① : 데이터 수집 및 가공**
noisereduce로 audio dataset의 노이즈를 제거하고, 침묵 구간을 기준으로 데이터를 slicing 후, STT API로 text data 확보

✓ **특징 ② : Tacotron 모델링 및 학습**
Tacotron과 Tacotron2의 특성을 혼합하여 사용
→ 계산 속도 ↑(Tacotron의 griffin-lim, Tacotron2의 ZoneoutLSTM)
→ Tacotron2의 Location Sensitive Attention을 추가 적용하여 현 시점의 alignment까지 고려

SKILLs/IDE

python, TensorFlow, Keras, scikit-learn, NumPy, pandas, matplotlib, jupyter, CO

기여도

Dataset

✓ **Korean Single Speaker (KSS) Speech**
출처 : Kaggle
특징 : 전문여자성우가 녹음한 음성 데이터
형태 : 12시간, wav, 44100khz, 12853개, 3GB

✓ **배우 유인나 음성**
출처 : KBS 라디오 유인나의 볼륨을 높여요
(2019.01 ~ 2019.05 기간회차분)
특징 : 배경음, 게스트 음성, 노래, 효과음 다수 존재하여 가공
형태 : 3시간, wav, 16000khz, 3327개, 480.6MB

✓ **KSS/배우 발화 내용 Text Dataset**
Google Speech to Text API와 Kakao Speech API 를 함께 활용하여 음성 데이터에 상응하는 텍스트 데이터 확보

Multi-Speaker 음성합성

Tacotron 모델과 DeepVoice2 모델을 기반으로
한국어 TTS(Text To Speech)를 구현하는 프로젝트

Github 👉 https://github.com/jyshin0926/KoreanTTS

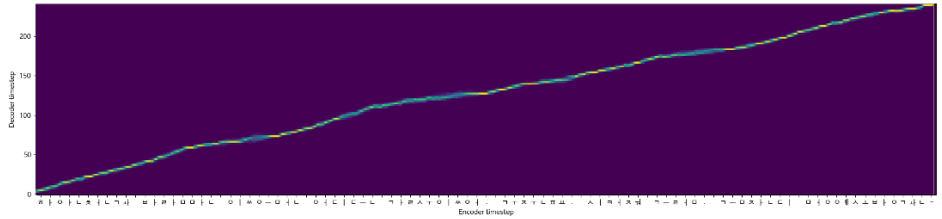
INFERENCE

KSS speech dataset

Tacotron 2 + GriffinLim + Multi speaker (KSS)

KSS (90000)

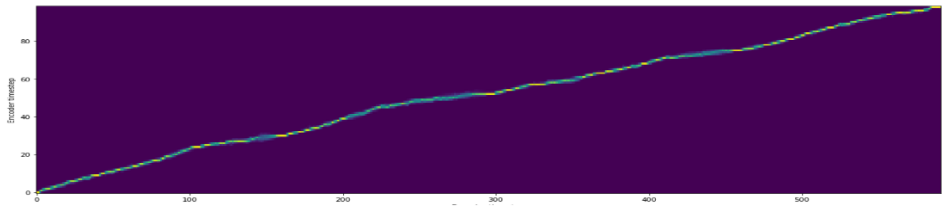
" 하얀 천과 바람만 있으면 어디든 갈 수 있어. "
" 구준표. 시켜줘 그럼. 금잔디 명예소방관. "



Tacotron 2 + MelGAN + Single speaker (KSS)

KSS (90000)

" 하얀 천과 바람만 있으면 어디든 갈 수 있어. "
" 구준표. 시켜줘 그럼. 금잔디 명예소방관. "



✓ KSS는 깔끔하고 12시간 이상의 방대한 양의 데이터였기에 20000 step 에서부터 Attention Alignment과 음성합성 추론 성능 모두 양호하게 나타남

✓ 90000 step까지 학습하였을 때 일부 문장은 Neural Vocoder인 MelGAN 에 비해 다소 성능이 낮게 나타나는 것을 확인 (Griffin-lim 알고리즘의 한계)

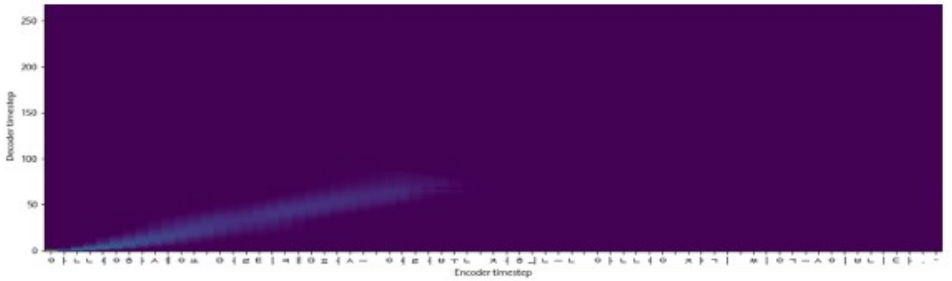
Tacotron Loss (Linear Loss + Mel Loss) : 0.38 (90000 step까지 학습)

✓ Linear Loss : Decoder 출력 Linear Spectrogram과 타겟 Mel Spectrogram과의 L1 Distance
✓ Mel Loss : Decoder 출력 Mel Spectrogram과 타겟 Mel Sepctrogram과의 L1 Distance

Tacotron 2 + GriffinLim + Single speaker (유인나)

Alignments graph (90000)

" 안녕하세요 멀티캠퍼스 여러분 저희는 안녕자기핑긋입니다. "



배우 유인나 speech dataset

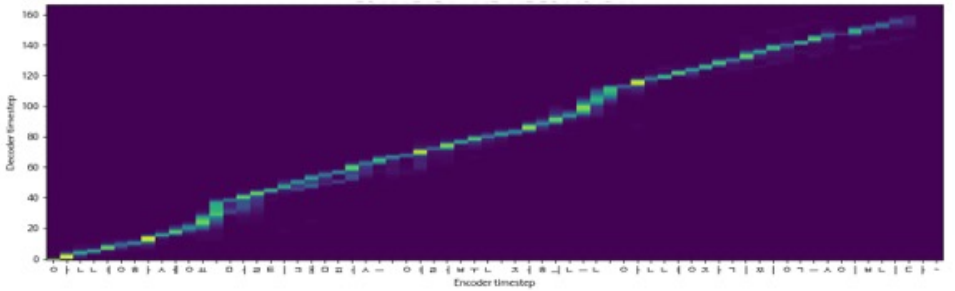
✓ 분량이 적고(약 3시간 분량), 노이즈가 많이 존재하는 데이터

✓ Single Speaker로 학습하였을 때는 Attention Alignment가 잘 나타나지 않음

Tacotron 2 + GriffinLim + Multi speaker (유인나)

Alignments graph (90000)

" 안녕하세요 멀티캠퍼스 여러분 저희는 안녕자기핑긋입니다. "



✓ 분량과 음질이 우수한 KSS 데이터와 함께 multi speaker로 학습시키며 speker embedding

✓ Multi Speaker로 학습한 경우는 Attention Alignment와 추론한 음성합성 모두 양호하게 나타남

PROJECT2

우리동네 키움센터 최적입지 추천

02

ABOUT PROJECT

2020 서울시 빅데이터 캠퍼스 공모전 우수상

2020.09.22 ~ 2020.10.14

키움센터 최적입지 추천

2022년까지의 우리동네키움센터 400개소 확충 정책안의 효율적인 실현을 위한 우리동네키움센터 입지 추천 알고리즘

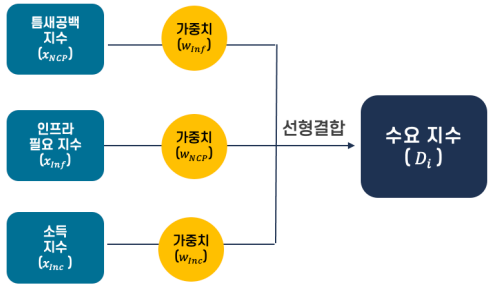
Github 👉 https://github.com/jyshin0926/BigDataCampusContest

✓ 키움센터 설립에 고려되어야 할 요인 세 가지 (아동이 혼자 지내는 시간을 고려한 돌봄 틈새 공백 지수, 공적돌봄 인프라 필요지수, 소득 분위 지수) 산정

✓ 각 요인을 선형결합하여 최종 수요지수를 산출하고 이를 입지 추천에 활용

✓ 서울시의 빅데이터 캠퍼스 공모전에 출품한 프로젝트로, 우수상을 수상하였습니다.

추천 알고리즘 산출 흐름



수요지수

$$D_i = w_{Inf}x_{Inf} + w_{NCP}x_{NCP} + w_{Inc}x_{Inc}$$

변수 (x) : 인프라 필요지수·틈새공백지수·소득지수

가중치(w) : 각 인프라 필요지수·틈새공백지수·소득지수의 가중치

수요지수(D_i) : 선형 결합 결과로 각 지역 별(i) 수요지수 산출

※ 각 변수는 min-max scaling 처리함

➡ 산출된 지역 별 수요 지수를 활용하여 돌봄 시설 최적 입지 선정

담당 역할

리더로서 팀원의 역량을 파악해 효율적인 역할 분담이 되도록 노력했습니다.
또한 프로젝트 초기 기획부터 협업 툴 세팅을 하며 팀워크가 높아질 수 있는 방향을 고민하였습니다.

✓ **특징 ① : 공적 돌봄 인프라 필요 지수 추정**
소득에 따른 돌봄 시설의 수요도를 조사한 설문조사 결과를 근거로 하여, 행정동별 돌봄 시설 수요를 자연어처리의 TF-IDF로 산출

✓ **특징 ② : 소득분위 추정**
직접적인 데이터를 구하기 어려워 아파트 매매 실거래가를 이용하여 KNN으로 추정

✓ **그 외 : 데이터 크롤링 및 가공, 선형결합, 시각화 수행**

SKILLs/IDE

python, Keras, scikit-learn, NumPy, matplotlib, pandas, seaborn, Folium, Beautiful Soup, GeoPandas, Selenium, jupyter, Colaboratory

기여도

Data

데이터	제공기관	기준년도
초등돌봄교실	교육부	2020
서울시학교별 정원(유초중고)	서울시 교육청	2020
행정동경계	통계청	2020
서울시행정동/법정동 코드	행정안전부	2020
전국초등학교위치표준데이터	공공데이터 포털	2020.10
행정동별 주거인구	빅데이터캠퍼스	2020.05
서울시우리동네키움센터(Selenium크롤링)	우리동네키움센터	2020.10.05
서울시아파트매매실거래가	국토교통부	2017.10 - 2020.09
서울특별시학원정보	서울특별시 교육청	2020.04.05
서울주소지연말정산신고	국세청 교육통계	2019
기혼여성의초등학생자녀돌봄희망장소	국가통계포털	2018
서울시혼인상태별인구(15세이상)통계	서울열린데이터 광장	2015
서울시예술학원	우리마을가게상권분석	2020.06.30
서울시세대원수별세대수(동별)통계	서울열린데이터광장	2017.12
귀가시부모님등어른의맞이횟수	국가통계포털	2018
2018한국의워킹맘보고서	KB금융지수경영연구소	2018
맞벌이가구비율	e나라지표	2017
은평구격자별인구	국토정보플랫폼	2020.04
수업을마치고가장많이있는장소(시간별)	국가통계포털	2018
서울시방과후돌봄지역사회협력방안	서울연구원	2018
학원도로명주소별좌표	S9Soft(http://www.s9soft.co.kr/single/s.html)	2020

키움센터 최적입지 추천

2022년까지의 우리동네키움센터 400개소 확충 정책안의
효율적인 실현을 위한 우리동네키움센터 입지 추천 알고리즘

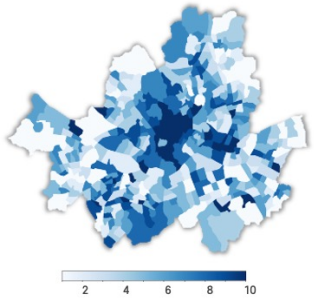
Github 👉 https://github.com/jyshin0926/BigDataCampusContest

수행사항

데이터 분석 · 시각화

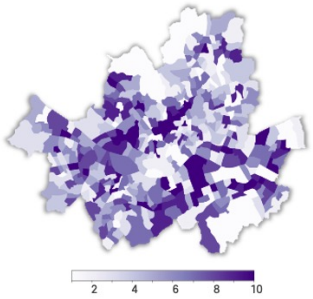
수행 ① : 공적 돌봄 인프라 필요 지수 추정

서울시 행정동 별 시설 TF-IDF



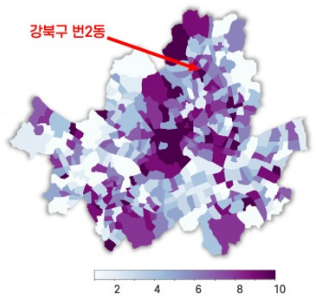
+

서울시 행정동 별 학원 TF-IDF



→

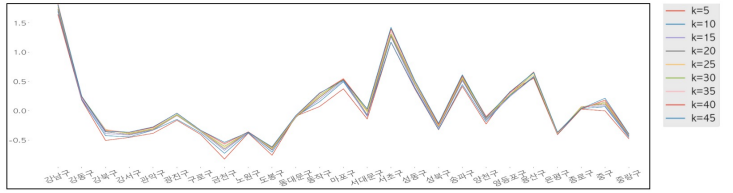
서울시 행정동 별 인프라지수



- 돌봄 인프라 지수가 가장 높은 동은 강북구 번2동이다.

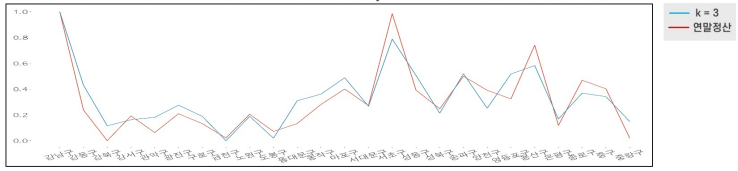
수행 ② : 소득분위 추정

K값(이웃 데이터 개수) 별 자치구별 추정 소득 (x축: 자치구, y축 : KNN 추정소득(k=5~45))



K값에 따른 자치구별 소득 경향은
대체적으로 비슷하지만,
행정구 별로 k값에 따라 추정 소득 차이 존재

연말정산 근로소득 vs KNN 추정 소득 (x축: 자치구, y축 : KNN(추정소득), 연말정산(평균소득))

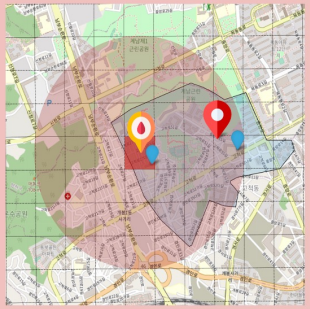


2019년 연말정산 과세대상근로소득
행정구 별 평균을 검증 값으로 활용.
이와 가장 유사한 추이를 보이는 k값
을 선정해 행정동 별 소득 추정

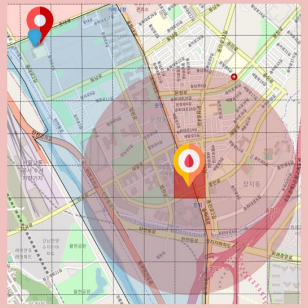
결과

우리동네키움센터 예약률 하위 2개 동

구로구 고척 제2동

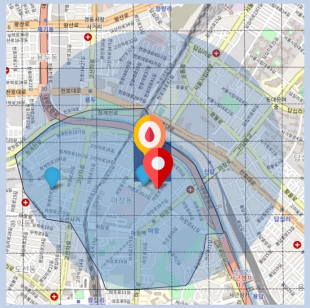


송파구 문정2동



우리동네키움센터 예약률 상위 2개 동

성동구 마장동



광진구 자양 제2동

추천 최적입지 기준 좌표 | 현재 우리동네키움센터 위치 | 초등학교 | 추천 격자(250m X 250m) | 초등학생 도보 10분 거리(반경 881m)

✓ 기존에 설립되어 있던 우리동네키움센터를 예약률 순위별로 구분한 후, 키움센터 예약률 하위 10%, 예약률 상위 10% 행정동을 대상으로 하여 산출한 알고리즘을 기반으로 예측한 키움센터의 추천 입지와의 거리를 비교

✓ 250x250 격자를 기준으로 설립되어 있는 센터와 추천 입지와의 거리를 haversine 모듈을 이용하여 계산하고 시각화로 나타냄

✓ 예약률 상위 센터는 추천 입지와의 거리가 가깝고, 계산된 거리 분포에서도 최소 격자 기준으로 삼은 250 이하의 비중이 가장 큰 것으로 나타남

✓ 예약률 하위 센터는 추천 입지와의 거리가 멀고, 계산된 거리 분포에서도 최소 격자 기준인 250의 범위를 크게 벗어난 거리의 비중이 가장 크게 나타남

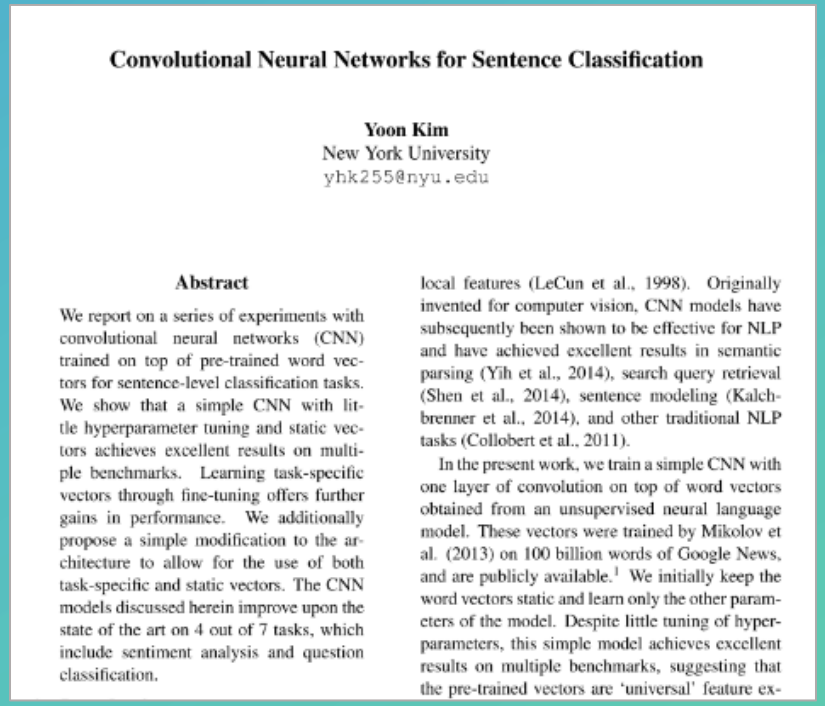
CNN 감성분석 논문 구현

PROJECT3

CNN for Sentence Classification

03

ABOUT PROJECT



Convolutional Neural Networks for Sentence Classification

Yoon Kim
New York University
yhk255@nyu.edu

Abstract

We report on a series of experiments with convolutional neural networks (CNN) trained on top of pre-trained word vectors for sentence-level classification tasks. We show that a simple CNN with little hyperparameter tuning and static vectors achieves excellent results on multiple benchmarks. Learning task-specific vectors through fine-tuning offers further gains in performance. We additionally propose a simple modification to the architecture to allow for the use of both task-specific and static vectors. The CNN models discussed herein improve upon the state of the art on 4 out of 7 tasks, which include sentiment analysis and question classification.

local features (LeCun et al., 1998). Originally invented for computer vision, CNN models have subsequently been shown to be effective for NLP and have achieved excellent results in semantic parsing (Yih et al., 2014), search query retrieval (Shen et al., 2014), sentence modeling (Kalchbrenner et al., 2014), and other traditional NLP tasks (Collobert et al., 2011).

In the present work, we train a simple CNN with one layer of convolution on top of word vectors obtained from an unsupervised neural language model. These vectors were trained by Mikolov et al. (2013) on 100 billion words of Google News, and are publicly available.[1] We initially keep the word vectors static and learn only the other parameters of the model. Despite little tuning of hyperparameters, this simple model achieves excellent results on multiple benchmarks, suggesting that the pre-trained vectors are 'universal' feature ex-

CNN 모델을 활용한 영화 리뷰 감성 분석

2020.08.05 ~ 2020.08.11

CNN 감성분석 논문 구현

개요

Convolution Neural Network for Sentence Classification 논문을 참고하여
CNN 모델을 구현하고 한국어 영화 리뷰를 감성분석한 프로젝트

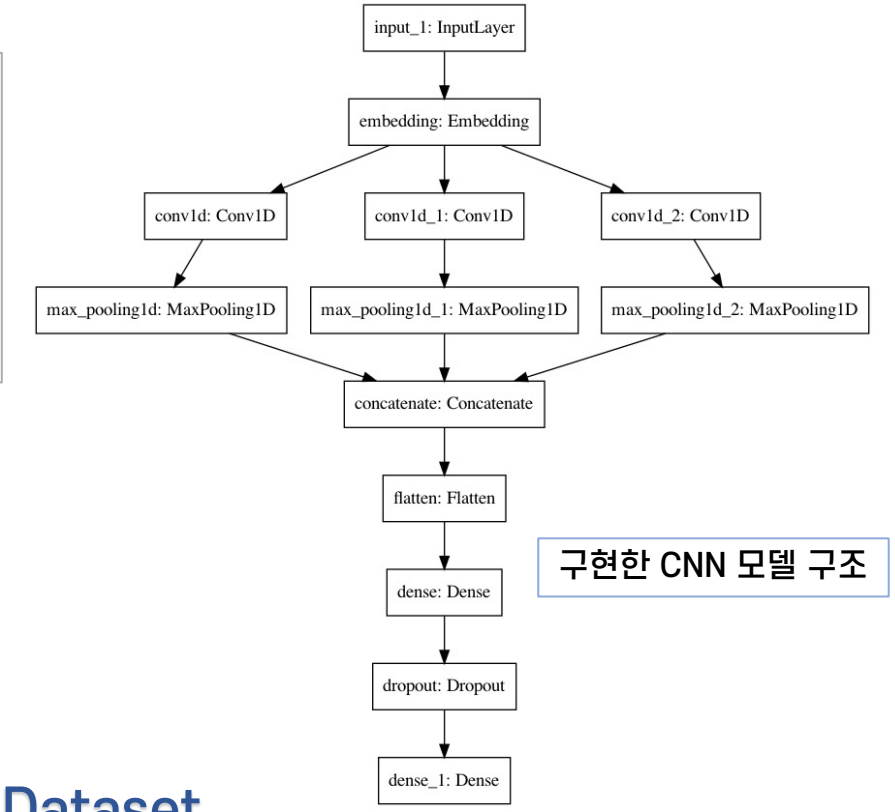
Github 👉 https://github.com/jyshin0926/CNN-for-sentence-classification

한국어 데이터셋에 맞추어 학습시킨 word embedding을 CNN 모델에 적용하여 영화 포털 사이트의 리뷰를 감성분석한 프로젝트입니다.

✓ Yoon Kim(2014), 'Convolutional Neural Networks for Sentence Classification' 의 CNN 구현

✓ subword 파악이 어려운 word2vec의 특징을 고려하여 일부 은어·신조어는 표준화하고 불용어 처리

✓ 전처리, Negative Sampling 및 L2 regularization과 early stopping을 통해 일반화된 성능 고려



구현한 CNN 모델 구조

수행 사항

NLP 교육과정에서 개인별 competition 으로 진행되었고,
이때, word embedding은 랜덤으로 배정되었습니다.
word2vec의 특성과 학습 환경(CPU)의 한계를 극복하기 위해 노력하며 **성능 평가에서 1위**를 하였습니다.

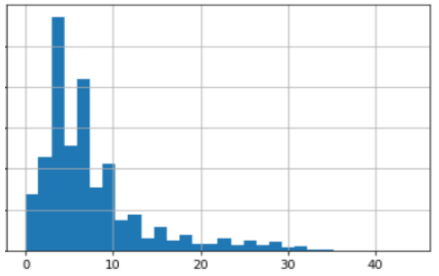
✓ **특징 ① : 전처리 및 word2vec 학습**
중복데이터 · null 값 · 특수 문자 제거 및 불용어 처리, negative sampling, skip gram으로 단어 예측, mecab으로 형태소 분석하여 연산 속도 효율성 증대

✓ **특징 ② : CNN 모델링 및 학습**
✓ **문장 길이 분포 기반으로 max length 설정 후 padding**
✓ **word2vec embedding 가중치 적용**
✓ **callback 함수 · Ridge 적용하여 과적합 방지**

SKILLs/IDE

python · TensorFlow · Keras · scikit-learn · NumPy · pandas · jupyter · CO

데이터 문장 길이
(문장 당 토큰 수)
분포 확인 결과
→ 4~6개의 토큰이
가장 많은 것으로
나타남



Dataset

✓ **NSMC (Naver Sentiment Movie Corpus)**

출처 : https://github.com/e9t/nsmc
특징 : 네이버 영화 페이지의 리뷰, 한국어
형태 : 총 분량 200K (test : 50K, train : 150K)

CNN 감성분석 논문 구현

Convolution Neural Network for Sentence Classification 논문을 참고하여
CNN 모델을 구현하고 한국어 영화 리뷰를 감성분석한 프로젝트

Github 👉 https://github.com/jyshin0926/CNN-for-sentence-classification

INFERENCE

임의 리뷰	완전 추천함	아 겁나 지루했음 개비추..	내 인생 영화	에라이 때려쳐 이게 영화라고?	조금 기대 이하이긴 했는데 그래도 만족함
점수 (negative 0 ~ positive 1)	0.99	0.05	0.76	0.02	0.65

✓ 임의의 문장에 대한 추론 결과, 긍정적인 글('완전 추천함', '내 인생 영화')에 대해서는 1에 가까운 점수로 나타남

✓ 부정적인 글('아 겁나 지루했음 개비추..', '에라이 때려쳐 이게 영화라고?')에 대해서는 0에 가까운 점수로 나타남

✓ '조금 기대 이하이긴 했는데 그래도 만족함' 처럼 모호한 표현에 대해서도 분석이 잘 이루어진 것으로 나타남

✓ 전처리를 word2vec embedding 모델과 학습 모델에 적절히 해주었을 경우 val_accuracy가 크게 높아짐

✓ 왓챠 platform의 영화 리뷰 데이터에 대해 accuracy 성능 평가

👉 추가 task : bi-LSTM 모델을 구현 및 학습하여 성능 비교
✓ 보통 nlp task에서 좋은 성능을 보이는 bi-LSTM 모델의 성능이 더 낮게 나온 이유는 평가 리뷰 길이의 영향으로 예상 (리뷰 문장의 토큰 수는 4~60이 가장 많은 것으로 나타남)

RESULT

모델	전처리 이전 val_accuracy	전처리 이후 val_accuracy	평가 accuracy
CNN	0.7993	0.8539	0.86
bi-LSTM	0.7872	0.8527	0.84

읽어주셔서 감사합니다.