0423 진행상황

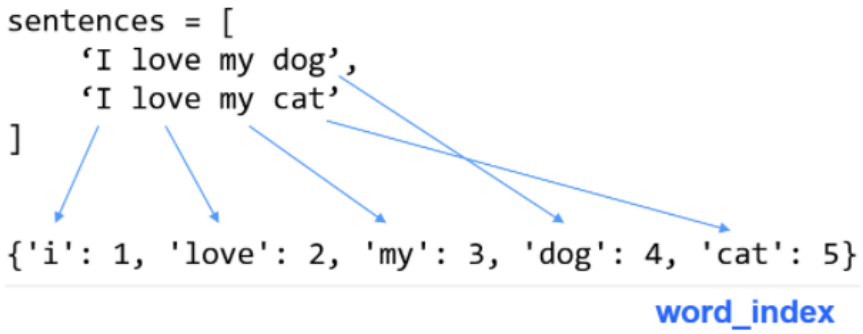
1. 영화 리뷰 데이터 진행

- 참고자료: https://codetorial.net/tensorflow/natural_language_processing_in_tensorflow_01.html

방법론

1. `from tensorflow.keras.preprocessing.text import Tokenizer`

- 단어와 숫자의 키-값 쌍을 포함하는 딕셔너리를 반환



2. `tokenizer.texts_to_sequences`

```
sentences = [
  'I love my dog',
  'I love my cat',
  'You love my dog!',
  'Do you think my dog is amazing?'
]
```

```
{'my': 1, 'love': 2, 'dog': 3, 'i': 4, 'you': 5, 'cat': 6, 'do': 7, 'think': 8, 'is': 9, 'amazing': 10}
[[4, 2, 1, 3], [4, 2, 1, 6], [5, 2, 1, 3], [7, 5, 8, 1, 3, 9, 10]]
```

- 이 때, 등장하지 않은 단어는 아예 포함되지 않음!!
- `Tokenizer( oov_token="<OOV>")` : 등장하지 않은 토큰 <OOV> 처리 : 1

3. `from tensorflow.keras.preprocessing.sequence import pad_sequences`

- 리스트 형식의 input값을 모두 같은 입력값을 가지는 array 형태의 시퀀스로 변환

```
[[5, 3, 2, 4], [5, 3, 2, 7], [6, 3, 2, 4], [8, 6, 9, 2, 4, 10, 11]]
```

sequences

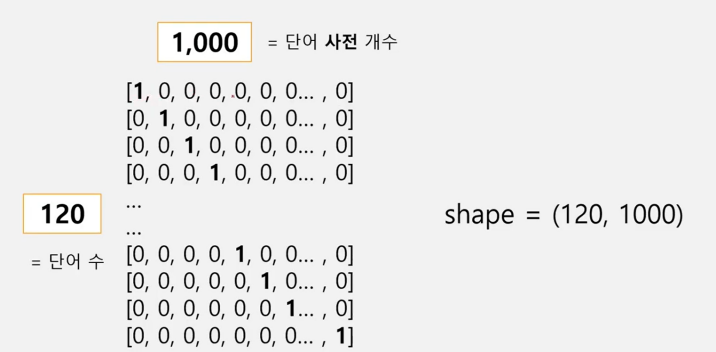
```
[[ 0  0  0  5  3  2  4]
 [ 0  0  0  5  3  2  7]
 [ 0  0  0  6  3  2  4]
 [ 8  6  9  2  4 10 11]]
```

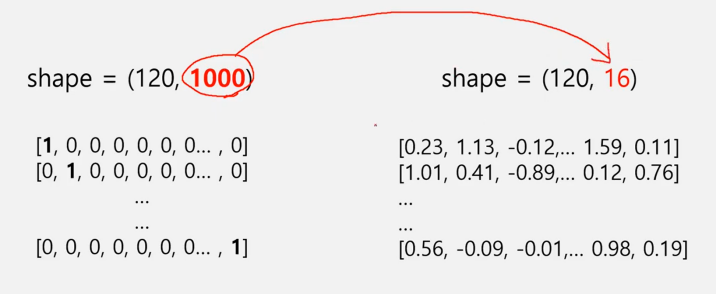
padded

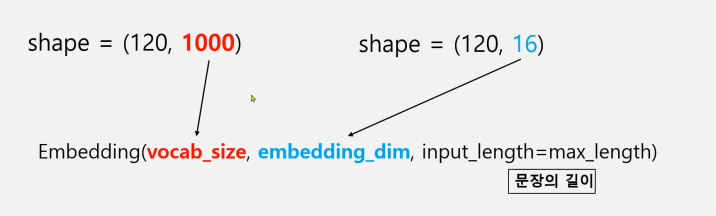
- max_len 의 파라미터를 조절해가며 대부분의 정보를 담는 파라미터를 선정!

4. Embedding layer

- `Embedding(단어 사전 개수, 임베딩 차원, input_length = 문장 길이)`







5. Modeling

- DNN, LSTM, LSTM(2층), BI-LSTM, BI-LSTM(2층),1D-CNN
- Config
  - Optimizer : Adam (빠르게 수렴할듯)
  - Loss : binary_crossentropy
  - Batch_size : 32,
  - Early stopping (10)

6. Evaluation

- 성능 비교

	형태소 분석	모델명	test 정확도
0	okt	DNN	0.546701
1	okt	LSTM	0.792896
2	okt	LSTM_2layer	0.789895
3	okt	Bi-LSTM	0.770385
4	okt	Bi-LSTM-2	0.510255
5	okt	1D-CNN	0.799400
6	okt_조사제거	DNN	0.537605
7	okt_조사제거	LSTM	0.798899
8	okt_조사제거	LSTM_2layer	0.781891
9	okt_조사제거	Bi-LSTM	0.775388
10	okt_조사제거	Bi-LSTM-2	0.781891
11	okt_조사제거	1D-CNN	0.798399
12	mecab	DNN	0.565037
13	mecab	LSTM	0.806403
14	mecab	LSTM_2layer	0.789895
15	mecab	Bi-LSTM	0.795398
16	mecab	Bi-LSTM-2	0.786893
17	mecab	1D-CNN	0.797899
18	mecab_조사제거	DNN	0.549084
19	mecab_조사제거	LSTM	0.793897
20	mecab_조사제거	LSTM_2layer	0.800400
21	mecab_조사제거	Bi-LSTM	0.784392
22	mecab_조사제거	Bi-LSTM-2	0.750375
23	mecab_조사제거	1D-CNN	0.798899

악플 리뷰

- **이진분류 성능 개선에 집중하려고함**
- baseline

	모델명	test 정확도
0	DNN	0.642614
1	LSTM	0.829670
2	LSTM_2layer	0.640797
3	Bi-LSTM	0.640797
4	Bi-LSTM-2	0.640797
5	1D-CNN	0.843407

추후 진행상황

- 텍스트 분야에서 data argumentation
- word2vec feature로 성능 비교