투빅스 프로젝트 2조

ToBig's 12기 신윤종

Transformer with Tensroflow 2.0

19/11/17

Chapter 01 | Data Loading

Data Loading

- Q_token column : 질문
- A_token column : 대답

- 하나의 인스턴스는 리스트형태로 이루어짐.
- 띄어쓰기 단위에 따라서 쉼표로 구분
- 형태소 단위에 따라서 “+”로 구분

- 어떻게 토큰을 나눌까?

```
df['Q_token'][3]
```

```
['특수/NNG + 교육/NNG + 대상자/NNG',
 '선정/NNG + 에/JKB',
 '필요/NNG + 하/XSA + ㄴ/ETM',
 '서류/NNG + 신규/NNG + 로/JKB',
 '특수/NNG + 교육/NNG + 대상자/NNG + 로/JKB',
 '선정/NNG + 되/XSV + 기/ETN + 를/JKO',
 '희망/NNG + 하/XSV + ㅂ니다/EC',
 '이/MM',
 '때/NNG',
 '필요/NNG + 하/XSA + ㄴ/ETM',
 '서류/NNG + 는/JX',
 '무엇/NP + 이/VCP + ㄴ가요/EC']
```

Chapter 01 | Data Loading

Data Loading

- 굳이 텐서플로의 토크나이저를 쓰겠다면 우리의 Khaiii 토큰은 그대로 쓰기 힘듦
  - ~~직접 토크나이저 만들어서 Vocab사전 만들기 무섭다는 건 안비밀~~
- Tf tokenizer가 여러 종류가 있는데, 어느 것을 쓰든 pos정보는 없애야 했다.

Token

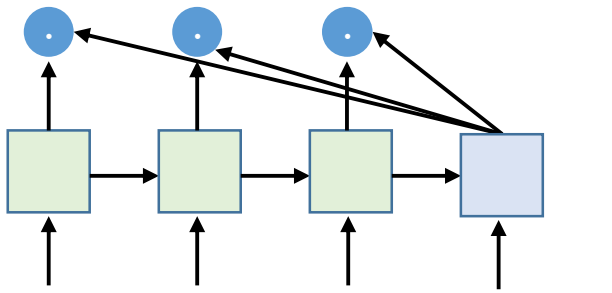
Subword

Chapter 02 | Attention

scaled_dot_product_attention(query, key, value, mask)

$$Attention(Q, K, V) = softmax_k(\frac{QK^T}{\sqrt{d_k}})V$$

Attention score = Matmul_df



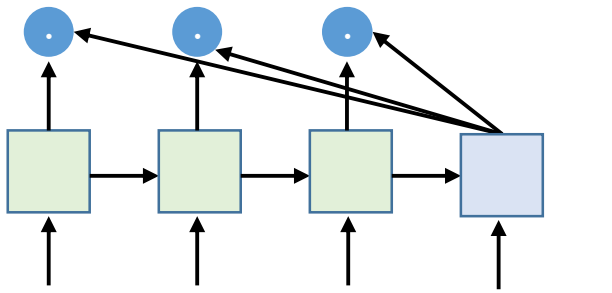
- matmul_df = tf.matmul(query, key, transpose_b=True)

Chapter 02 | Attention

scaled_dot_product_attention(query, key, value, mask)

$$Attention(Q, K, V) = softmax_k(\frac{QK^T}{\sqrt{d_k}})V$$

Attention score = Matmul_df



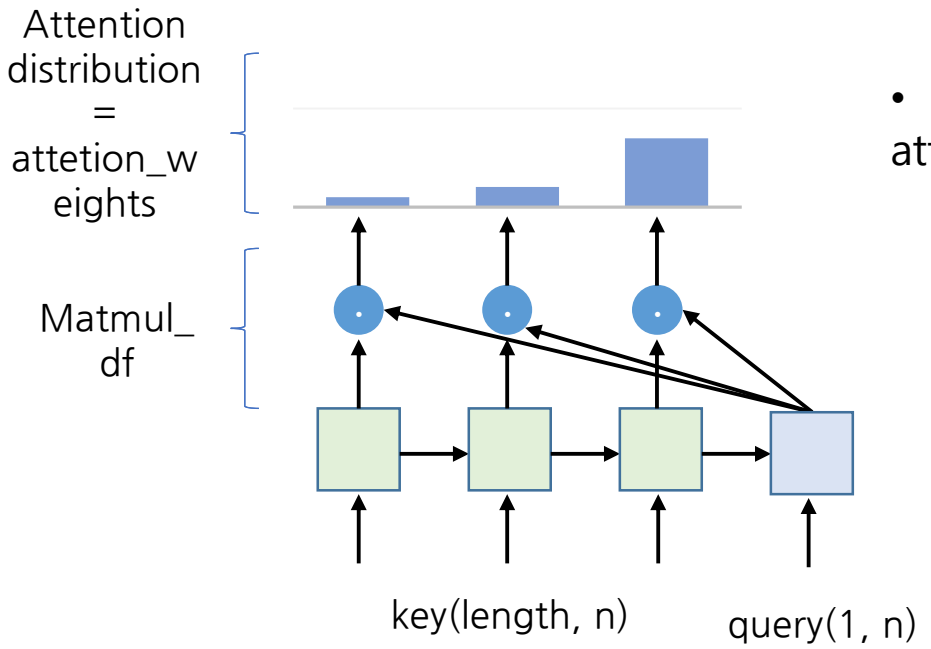
- Scale Attention Score

```
depth = tf.cast(tf.shape(key)[-1], tf.float32)
logits = matmul_qk / tf.math.sqrt(depth)
```

Chapter 02 | Attention

scaled_dot_product_attention(query, key, value, mask)

$$Attention(Q, K, V) = softmax_k(\frac{QK^T}{\sqrt{d_k}})V$$



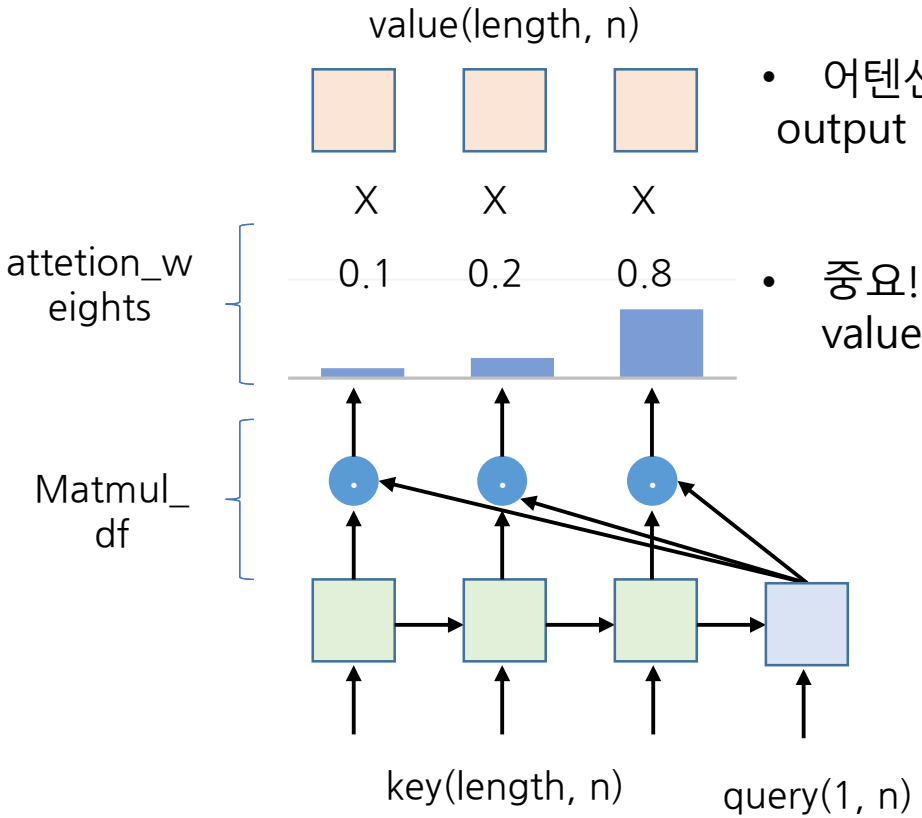
- softmax is normalized on the last axis (seq_len_k)

attention_weights = tf.nn.softmax(logits, axis=-1)

Chapter 02 | Attention

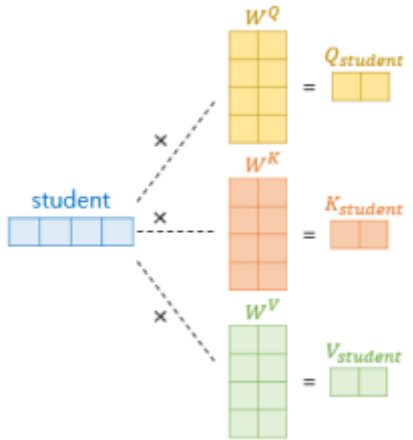
scaled_dot_product_attention(query, key, value, mask)

$$Attention(Q, K, V) = softmax_k(\frac{QK^T}{\sqrt{d_k}})V$$



- 어텐션 가중치와 value를 행렬곱 해줍니다.
output = tf.matmul(attention_weights, value)

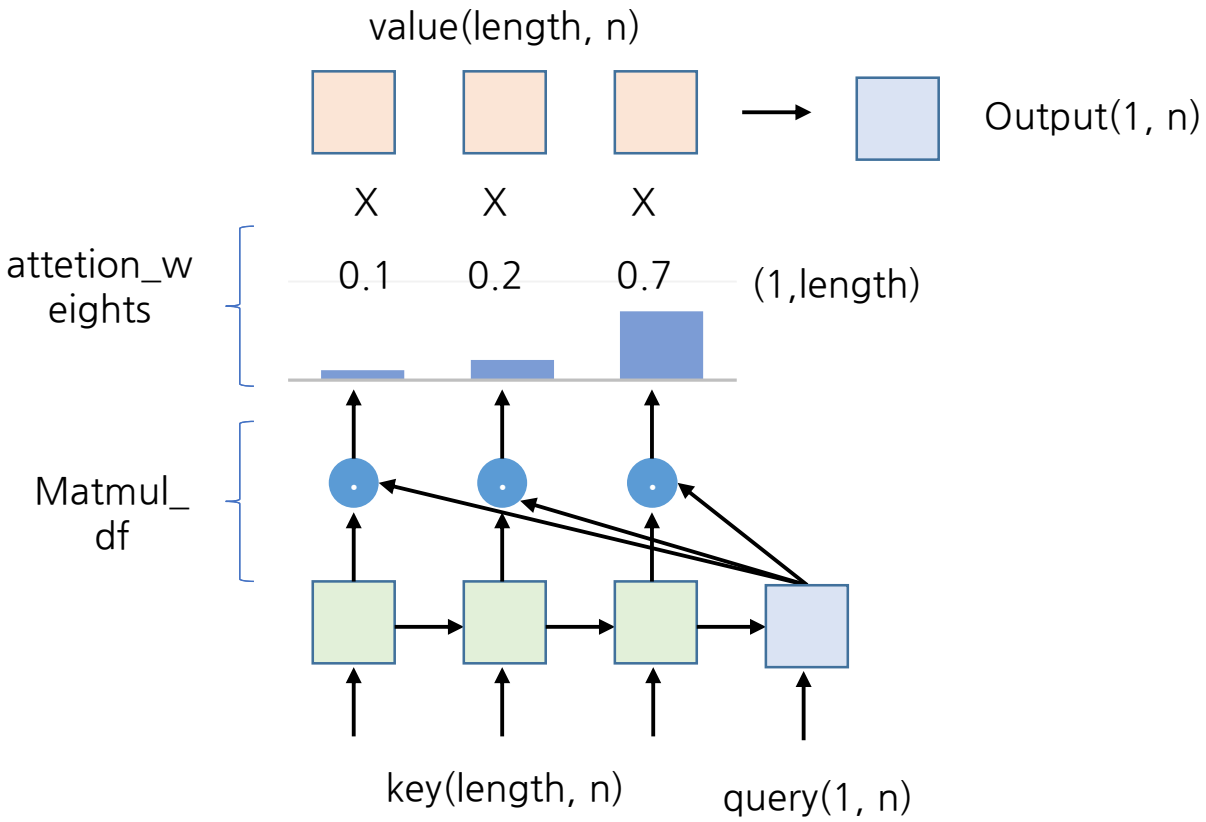
- 중요! 가중치를 다시 key와 곱하는 게 아니라 따로 준비된 value에 곱하는 겁니다!



Chapter 02 | Attention

scaled_dot_product_attention(query, key, value, mask)

$$Attention(Q, K, V) = softmax_k(\frac{QK^T}{\sqrt{d_k}})V$$



Chapter 02 | Attention

MultiHeadAttetion (d_model, num_heads)

- D_model = 토큰의 임베딩 차원
- num_heads = 헤드 개수

```
self.query_dense = tf.keras.layers.Dense(units=d_model)
self.key_dense = tf.keras.layers.Dense(units=d_model)
self.value_dense = tf.keras.layers.Dense(units=d_model)
```

- Q, K, V마다 각각 따로 학습하고 있다.

Chapter 02 | Attention

Training Result

- 기존의 Output
- 우리의 데이터와 상관없음

```
Epoch 1/20
689/689 [==============================] - 114s 165ms/step - loss: 2.1129 - accuracy: 0.0421
Epoch 2/20
689/689 [==============================] - 104s 150ms/step - loss: 1.5029 - accuracy: 0.0784
Epoch 3/20
689/689 [==============================] - 104s 150ms/step - loss: 1.3981 - accuracy: 0.0855
Epoch 4/20
689/689 [==============================] - 103s 149ms/step - loss: 1.3386 - accuracy: 0.0902
Epoch 5/20
689/689 [==============================] - 103s 150ms/step - loss: 1.2868 - accuracy: 0.0941
Epoch 6/20
689/689 [==============================] - 103s 150ms/step - loss: 1.2402 - accuracy: 0.0978
Epoch 7/20
689/689 [==============================] - 104s 151ms/step - loss: 1.1851 - accuracy: 0.1020
Epoch 8/20
689/689 [==============================] - 104s 151ms/step - loss: 1.1236 - accuracy: 0.1075
Epoch 9/20
689/689 [==============================] - 104s 151ms/step - loss: 1.0677 - accuracy: 0.1129
Epoch 10/20
689/689 [==============================] - 104s 150ms/step - loss: 1.0145 - accuracy: 0.1187
Epoch 11/20
689/689 [==============================] - 103s 150ms/step - loss: 0.9655 - accuracy: 0.1245
Epoch 12/20
689/689 [==============================] - 102s 148ms/step - loss: 0.9215 - accuracy: 0.1302
Epoch 13/20
689/689 [==============================] - 104s 151ms/step - loss: 0.8806 - accuracy: 0.1356
Epoch 14/20
689/689 [==============================] - 104s 152ms/step - loss: 0.8434 - accuracy: 0.1411
Epoch 15/20
689/689 [==============================] - 104s 151ms/step - loss: 0.8093 - accuracy: 0.1463
Epoch 16/20
689/689 [==============================] - 105s 153ms/step - loss: 0.7789 - accuracy: 0.1510
Epoch 17/20
689/689 [==============================] - 105s 152ms/step - loss: 0.7505 - accuracy: 0.1554
Epoch 18/20
689/689 [==============================] - 105s 153ms/step - loss: 0.7242 - accuracy: 0.1601
Epoch 19/20
689/689 [==============================] - 105s 152ms/step - loss: 0.7005 - accuracy: 0.1638
Epoch 20/20
689/689 [==============================] - 104s 151ms/step - loss: 0.6788 - accuracy: 0.1676
<tensorflow.python.keras.callbacks.History at 0x7f94d3a1c3c8>
```

Chapter 02 | Attention

Training Result

- 토크나이징 1
  - 첫 번째 방법으로 토크나이징한 결과

```
82/82 [==============================] - 21s 258ms/step - loss: 2.0815 - accuracy: 0.1284
Epoch 5/200
82/82 [==============================] - 21s 259ms/step - loss: 1.9955 - accuracy: 0.1363
Epoch 6/200
82/82 [==============================] - 21s 259ms/step - loss: 1.9177 - accuracy: 0.1441
Epoch 7/200
82/82 [==============================] - 21s 260ms/step - loss: 1.8433 - accuracy: 0.1527
Epoch 8/200
82/82 [==============================] - 21s 261ms/step - loss: 1.7715 - accuracy: 0.1614
Epoch 9/200
82/82 [==============================] - 21s 258ms/step - loss: 1.7035 - accuracy: 0.1698
Epoch 10/200
82/82 [==============================] - 21s 259ms/step - loss: 1.6357 - accuracy: 0.1777
Epoch 11/200
82/82 [==============================] - 21s 260ms/step - loss: 1.5689 - accuracy: 0.1852
Epoch 12/200
82/82 [==============================] - 21s 259ms/step - loss: 1.5026 - accuracy: 0.1929
Epoch 13/200
82/82 [==============================] - 21s 259ms/step - loss: 1.4425 - accuracy: 0.2004
Epoch 14/200
82/82 [==============================] - 21s 258ms/step - loss: 1.3803 - accuracy: 0.2083
Epoch 15/200
82/82 [==============================] - 21s 260ms/step - loss: 1.3170 - accuracy: 0.2160
Epoch 16/200
82/82 [==============================] - 21s 258ms/step - loss: 1.2554 - accuracy: 0.2244
Epoch 17/200
82/82 [==============================] - 21s 259ms/step - loss: 1.1961 - accuracy: 0.2327
Epoch 18/200
82/82 [==============================] - 21s 258ms/step - loss: 1.1336 - accuracy: 0.2409
Epoch 19/200
82/82 [==============================] - 21s 258ms/step - loss: 1.0742 - accuracy: 0.2497
Epoch 20/200
82/82 [==============================] - 21s 260ms/step - loss: 1.0148 - accuracy: 0.2584
```

Chapter 02 | Attention

Training Result

- 토크나이징 2
  - 첫 번째 방법으로 토크나이징한 결과
- 쪼금 낫다
- 고민거리들
  - 아직 충분한 데이터 셋 크기가 갖추어지지 않았나?
  - 하이퍼 파라미터를 조절해야 하나?
  - 토큰화 방법을 수정해야 하나?
  - Max 시퀀스 길이도 영향을 끼치나?

```
82/82 [==============================] - 21s 258ms/step - loss: 2.0815 - accuracy: 0.1284
Epoch 5/200
82/82 [==============================] - 21s 259ms/step - loss: 1.9955 - accuracy: 0.1363
Epoch 6/200
82/82 [==============================] - 21s 259ms/step - loss: 1.9177 - accuracy: 0.1441
Epoch 7/200
82/82 [==============================] - 21s 260ms/step - loss: 1.8433 - accuracy: 0.1527
Epoch 8/200
82/82 [==============================] - 21s 261ms/step - loss: 1.7715 - accuracy: 0.1614
Epoch 9/200
82/82 [==============================] - 21s 258ms/step - loss: 1.7035 - accuracy: 0.1698
Epoch 10/200
82/82 [==============================] - 21s 259ms/step - loss: 1.6357 - accuracy: 0.1777
Epoch 11/200
82/82 [==============================] - 21s 260ms/step - loss: 1.5689 - accuracy: 0.1852
Epoch 12/200
82/82 [==============================] - 21s 259ms/step - loss: 1.5026 - accuracy: 0.1929
Epoch 13/200
82/82 [==============================] - 21s 259ms/step - loss: 1.4425 - accuracy: 0.2004
Epoch 14/200
82/82 [==============================] - 21s 258ms/step - loss: 1.3803 - accuracy: 0.2083
Epoch 15/200
82/82 [==============================] - 21s 260ms/step - loss: 1.3170 - accuracy: 0.2160
Epoch 16/200
82/82 [==============================] - 21s 258ms/step - loss: 1.2554 - accuracy: 0.2244
Epoch 17/200
82/82 [==============================] - 21s 259ms/step - loss: 1.1961 - accuracy: 0.2327
Epoch 18/200
82/82 [==============================] - 21s 258ms/step - loss: 1.1336 - accuracy: 0.2409
Epoch 19/200
82/82 [==============================] - 21s 258ms/step - loss: 1.0742 - accuracy: 0.2497
Epoch 20/200
82/82 [==============================] - 21s 260ms/step - loss: 1.0148 - accuracy: 0.2584
```