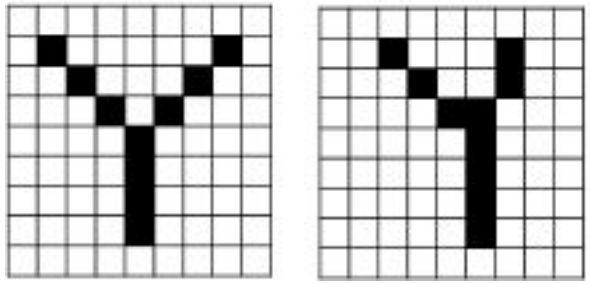
11장 NLP를 위한 합성곱 신경망

Convolution Neural Network

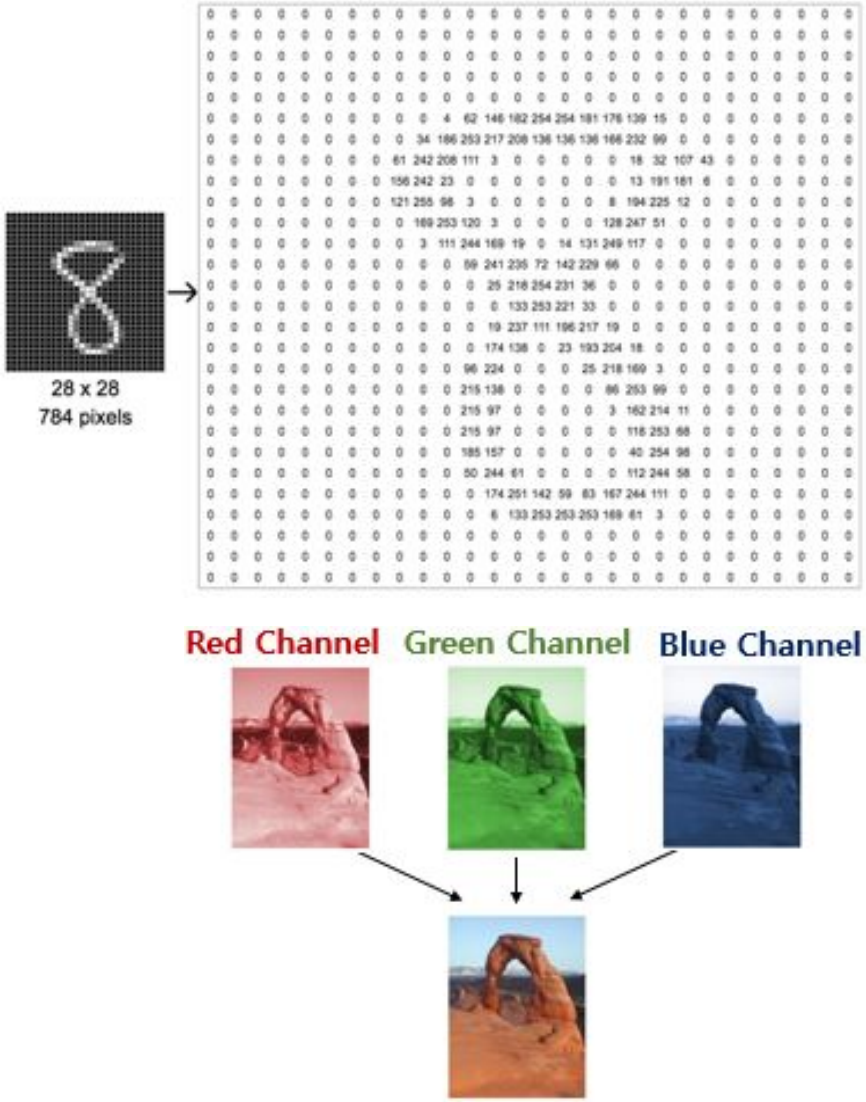
엄호용

목차 Table of Contents

01 합성곱 신경망 Convolution Neural Network

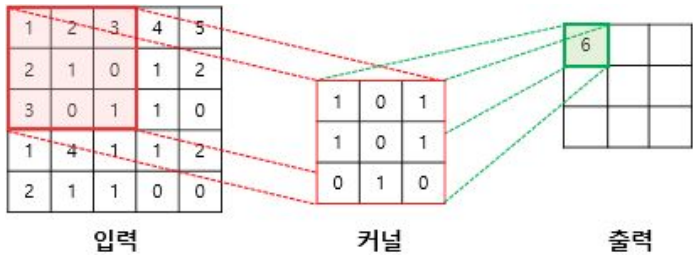


(높이, 너비, 채널
(=색상))

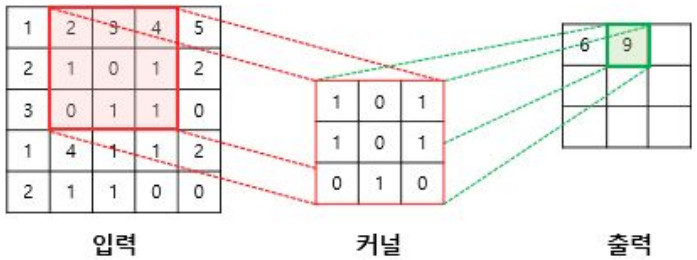


01 합성곱 신경망 Convolution Neural Network

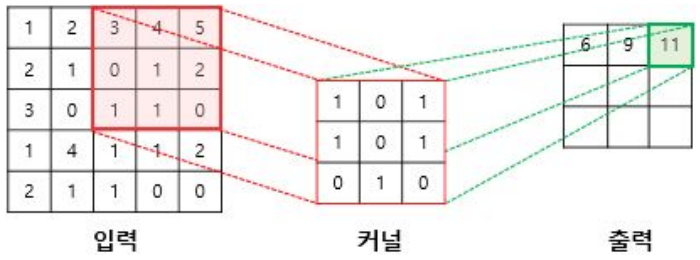
첫번째 스텝



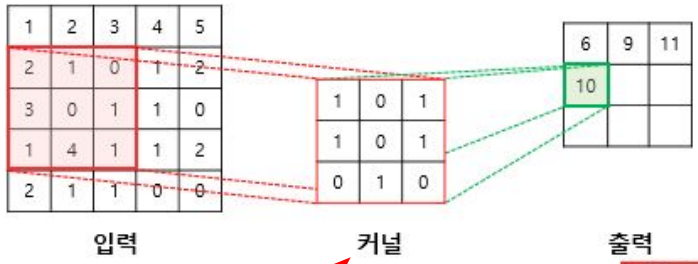
두번째 스텝



세번째 스텝

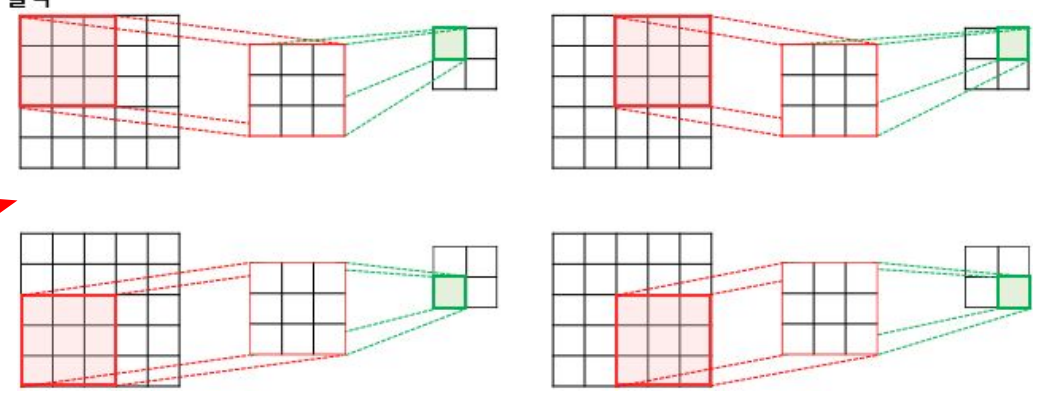


네번째 스텝



stride = 1

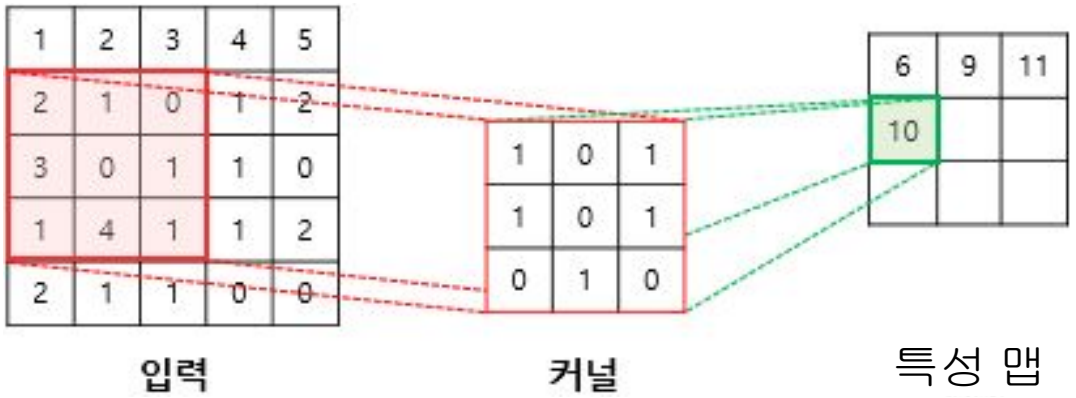
stride = 2

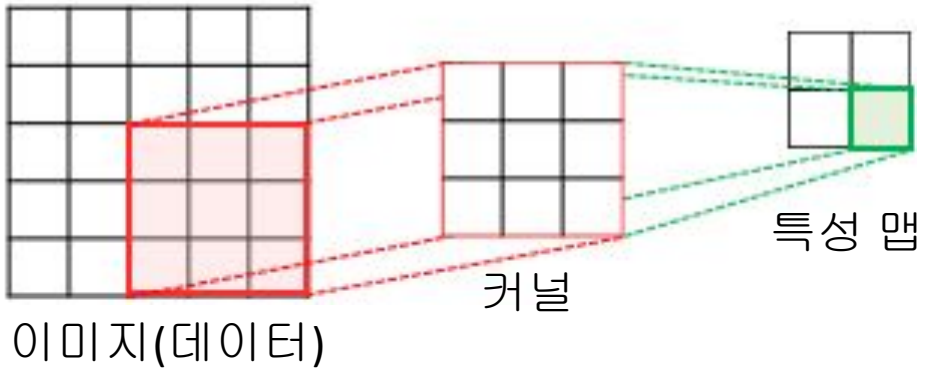


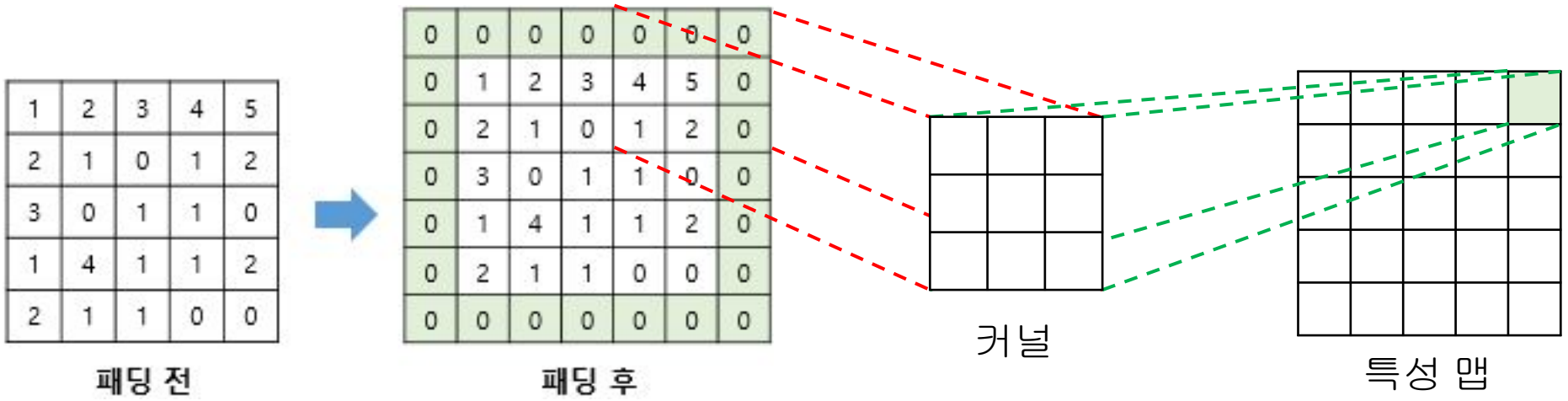
6	9	11
10	4	4
7	7	4

특성 맵(feature map)

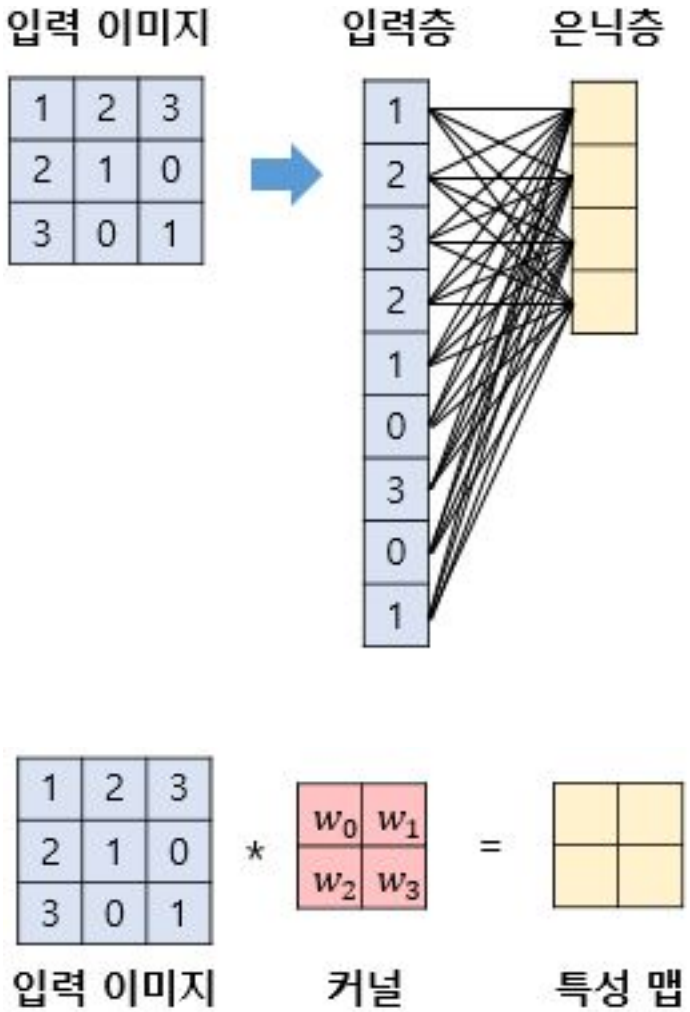
01 합성곱 신경망 Convolution Neural Network

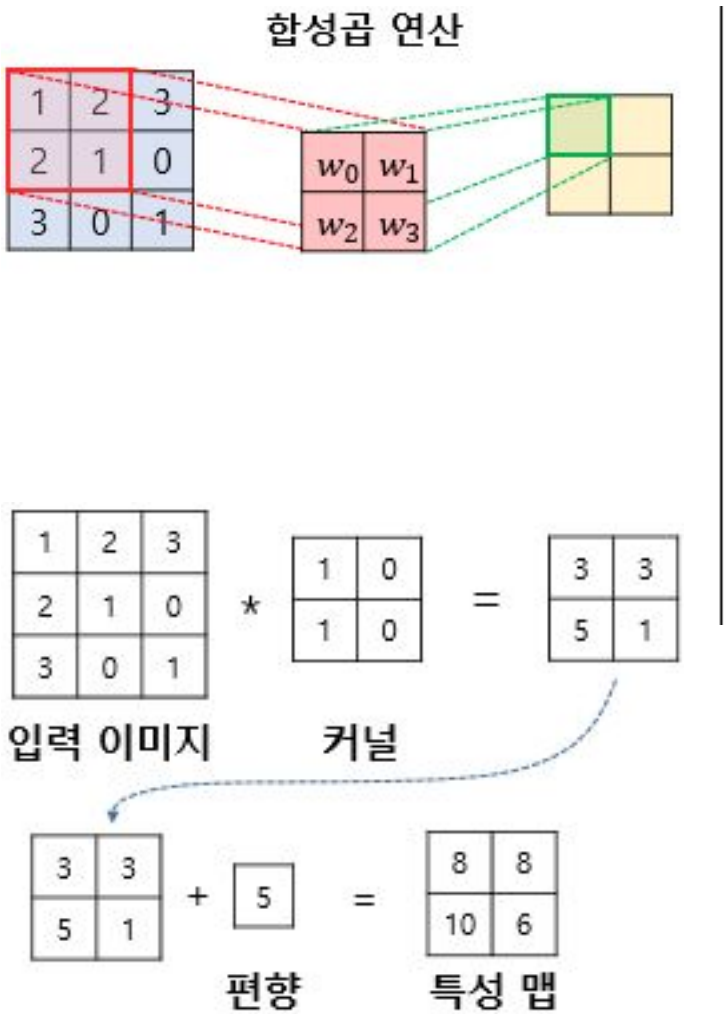


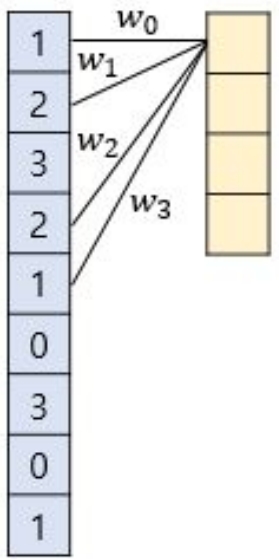




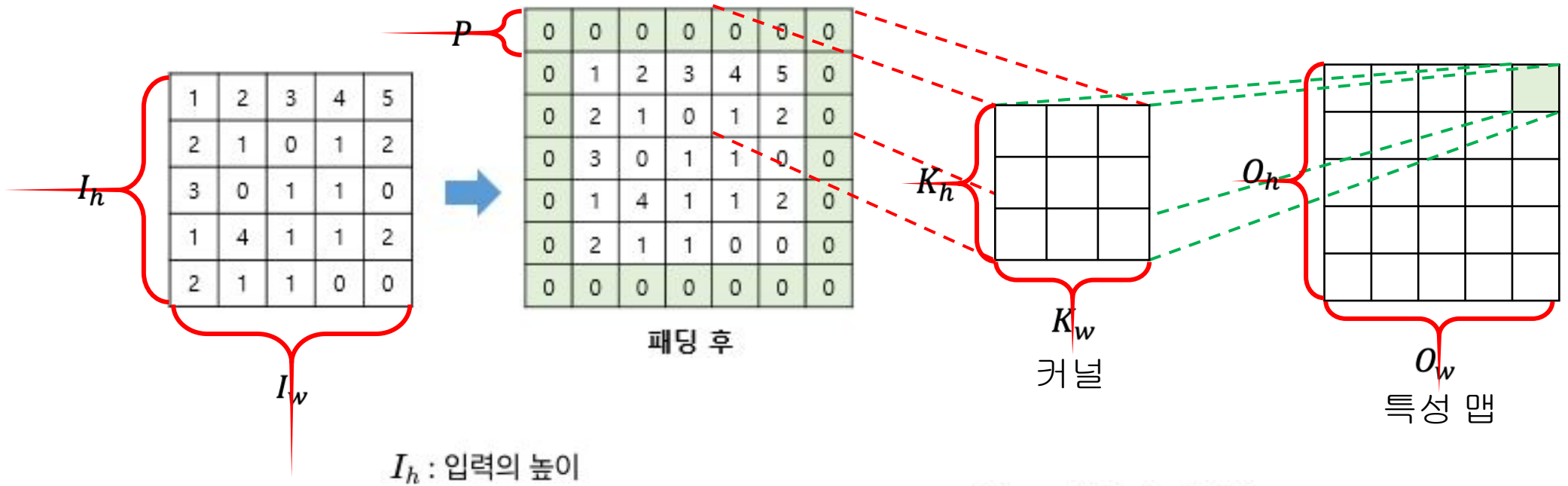
01 합성곱 신경망 Convolution Neural Network







01 합성곱 신경망 Convolution Neural Network

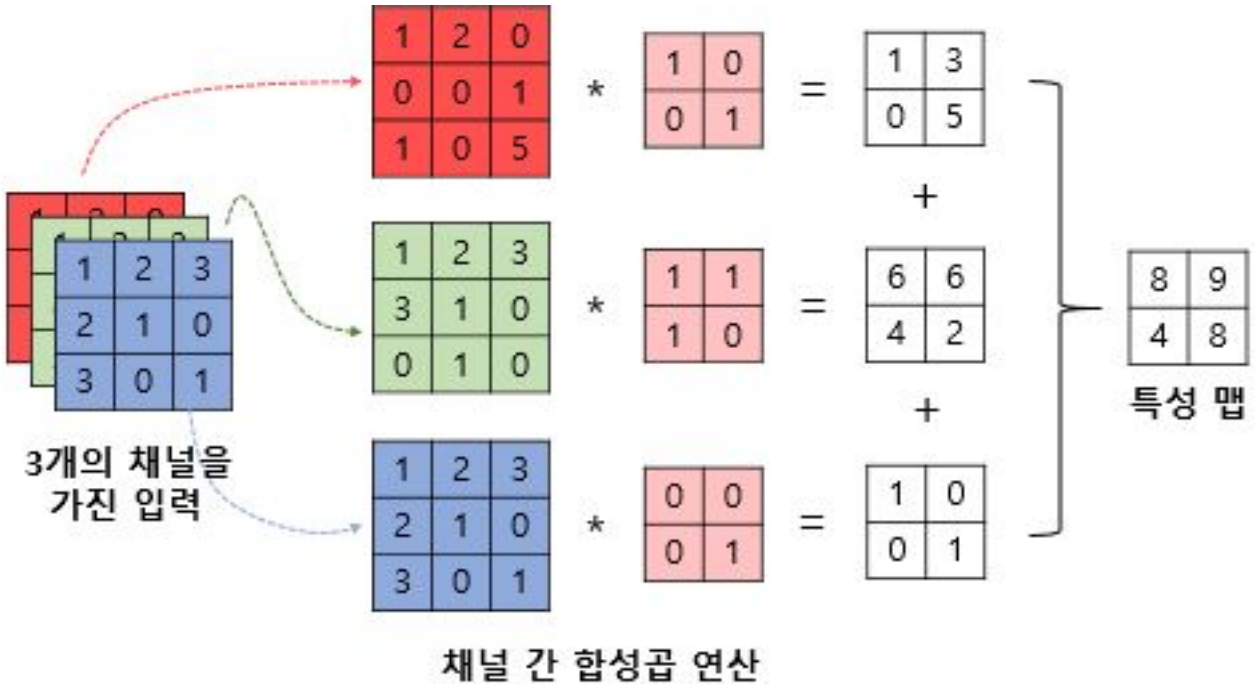


I_h : 입력의 높이
I_w : 입력의 너비
K_h : 커널의 높이
K_w : 커널의 너비
S : 스트라이드
O_h : 특성 맵의 높이
O_w : 특성 맵의 너비

$$O_h = floor(\frac{I_h - K_h + 2P}{S} + 1)$$

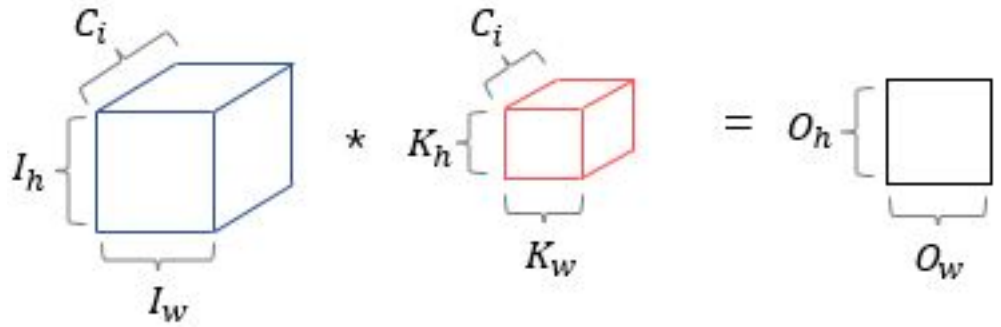
$$O_w = floor(\frac{I_w - K_w + 2P}{S} + 1)$$

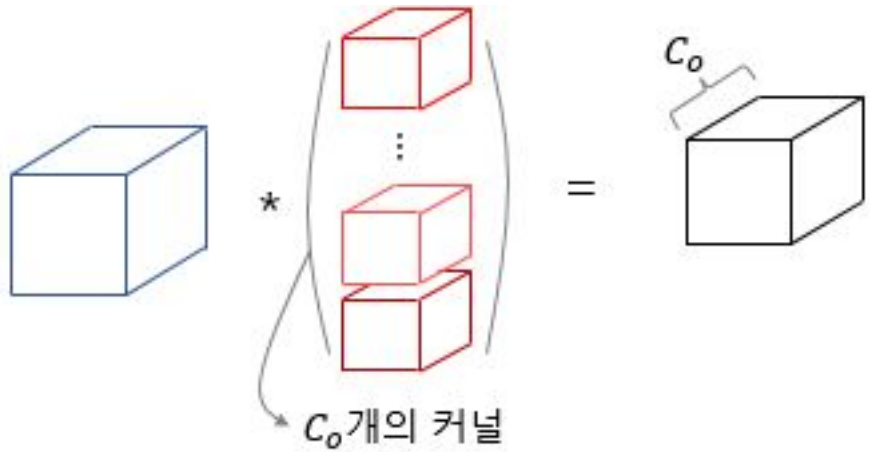
01 합성곱 신경망 Convolution Neural Network



01 합성곱 신경망 Convolution Neural Network

I_h : 입력의 높이
I_w : 입력의 너비
K_h : 커널의 높이
K_w : 커널의 너비
O_h : 특성 맵의 높이
O_w : 특성 맵의 너비
C_i : 입력 데이터의 채널

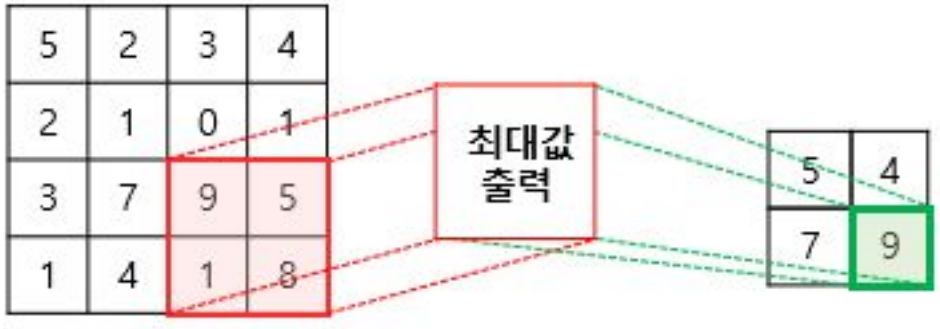




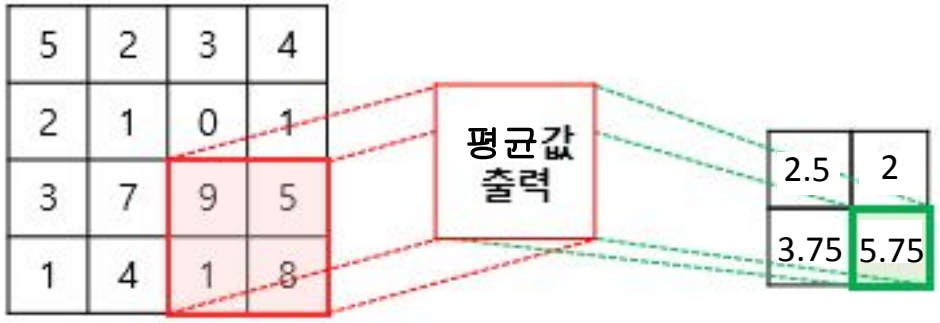
01 합성곱 신경망 Convolution Neural Network

풀링(Pooling)

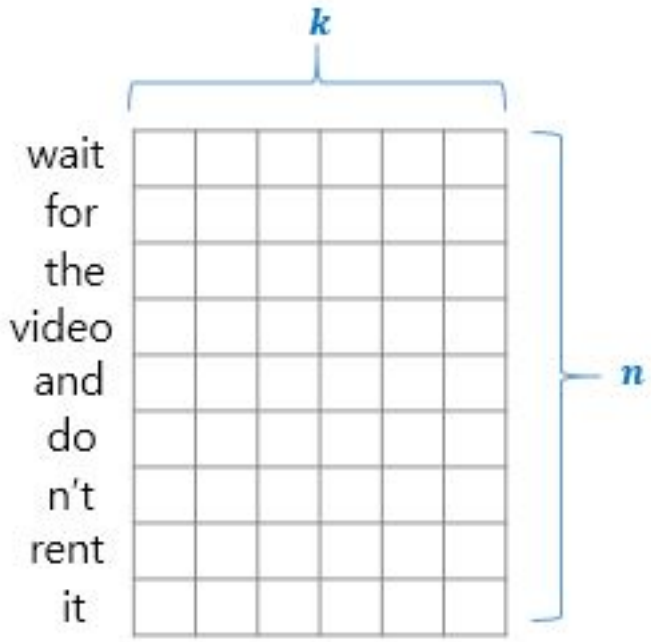
- 최대풀링(Max Pooling)

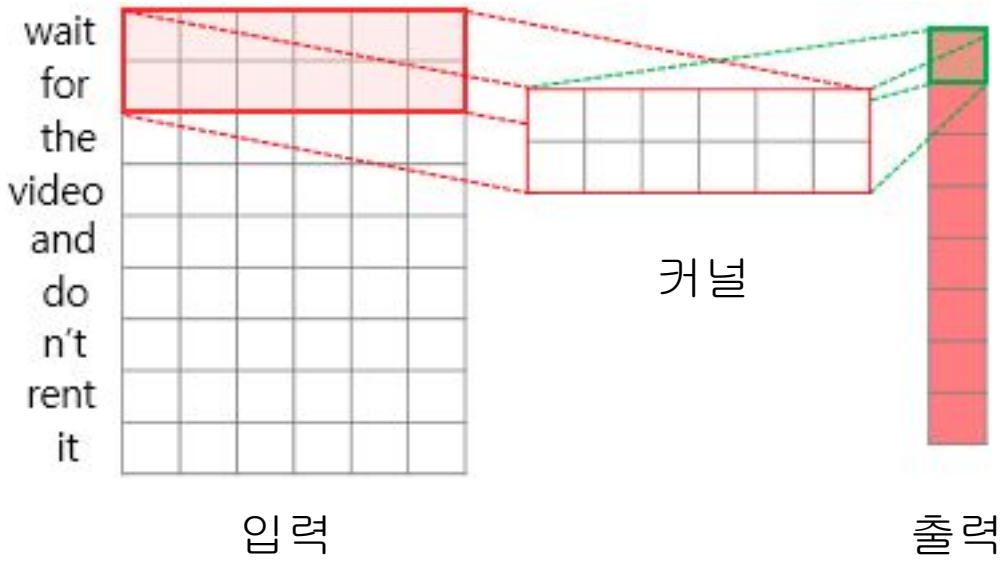


- 평균풀링(Average Pooling)

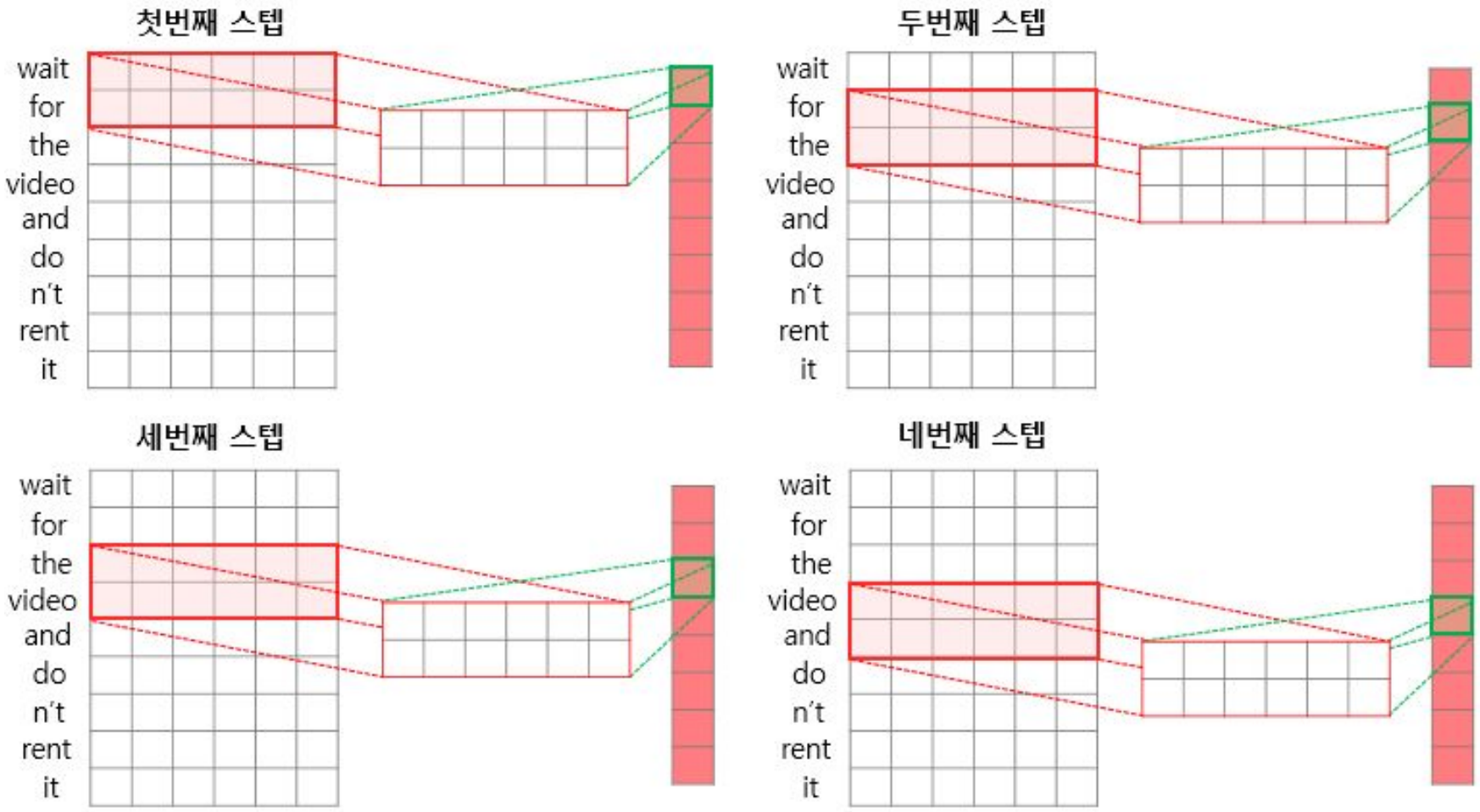


02 자연어 처리를 위한 1D CNN 1D CNN

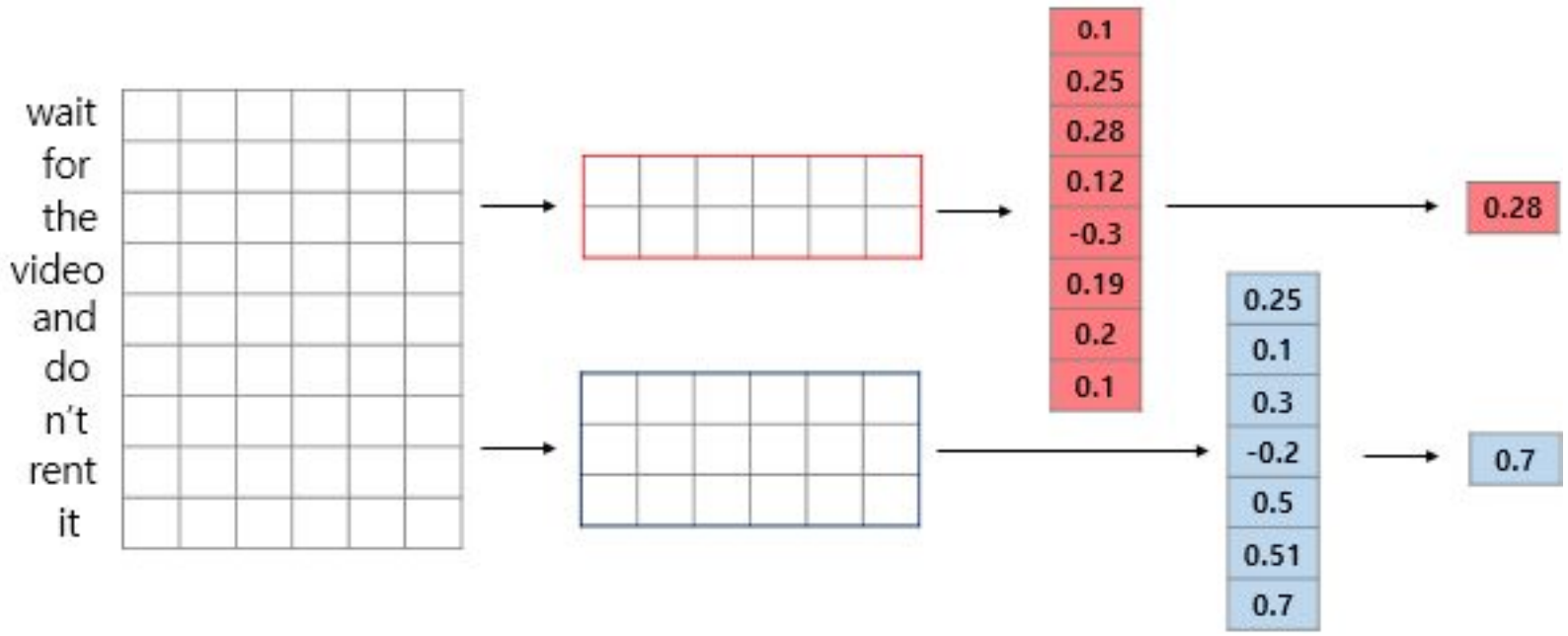




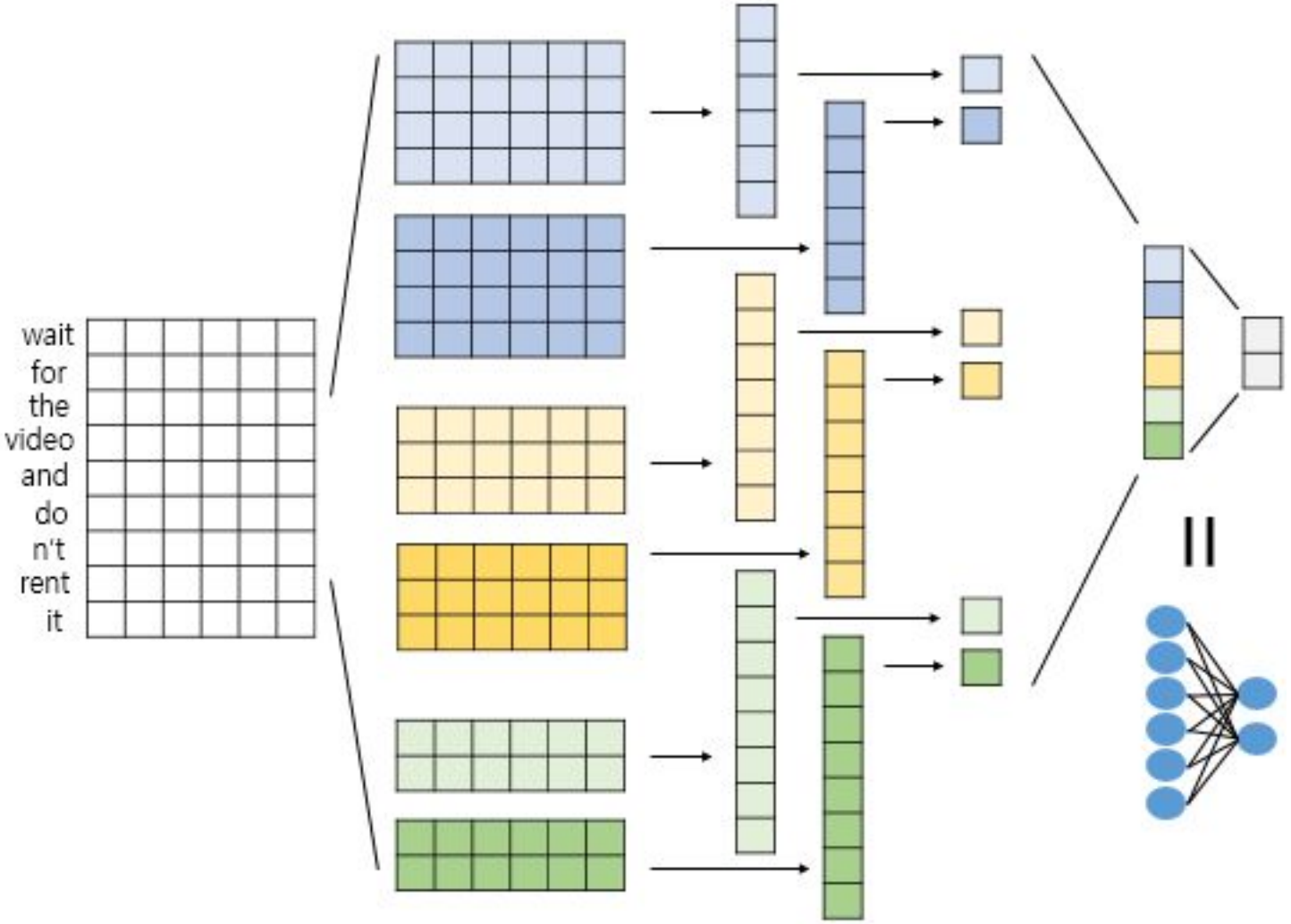
02 자연어 처리를 위한 1D CNN 1D CNN



02 자연어 처리를 위한 1D CNN 1D CNN



02 자연어 처리를 위한 1D CNN 1D CNN



02 자연어 처리를 위한 1D CNN 1D CNN

- Keras를 사용한 CNN 구현

```
from tensorflow.keras.layers import Conv1D, GlobalMaxPooling1D

model = Sequential()
model.add(Conv1D(num_filters, kernel_size, padding='valid', activation='relu'))
```

```
model = Sequential()
model.add(Conv1D(num_filters, kernel_size, padding='valid', activation='relu'))
model.add(GlobalMaxPooling1D())
```

num_filters = 커널의 개수.

kernel_size = 커널의 크기.

padding = 패딩 방법.

- valid : 패딩 없음. 제로 패딩없이 유효한(valid) 값만을 사용한다는 의미에서 valid.
- same : 합성곱 연산 후에 출력이 입력과 동일한 차원을 가지도록 왼쪽과 오른쪽(또는 위, 아래)에 제로 패딩을 추가.

activation = 활성화 함수.

03 1D CNN으로 IMDB 리뷰 분류하기
04 1D CNN으로 스팸 메일 분류하기

1D CNN으로 IMDB 리뷰 [긍정/부정] 분류하기

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Embedding, Dropout, Conv1D, GlobalMaxPooling1D, Dense
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
from tensorflow.keras.models import load_model

embedding_dim = 256 # 임베딩 벡터의 차원
dropout_ratio = 0.3 # 드롭아웃 비율
num_filters = 256 # 커널의 수
kernel_size = 3 # 커널의 크기
hidden_units = 128 # 뉴런의 수

model = Sequential()
model.add(Embedding(vocab_size, embedding_dim))
model.add(Dropout(dropout_ratio))
model.add(Conv1D(num_filters, kernel_size, padding='valid', activation='relu'))
model.add(GlobalMaxPooling1D())
model.add(Dense(hidden_units, activation='relu'))
model.add(Dropout(dropout_ratio))
model.add(Dense(1, activation='sigmoid'))

es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=3)
mc = ModelCheckpoint('best_model.h5', monitor='val_acc', mode='max', verbose=1, save_best_only=True)

model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['acc'])
history = model.fit(X_train, y_train, epochs=20, validation_data=(X_test, y_test), callbacks=[es, m
c])
```

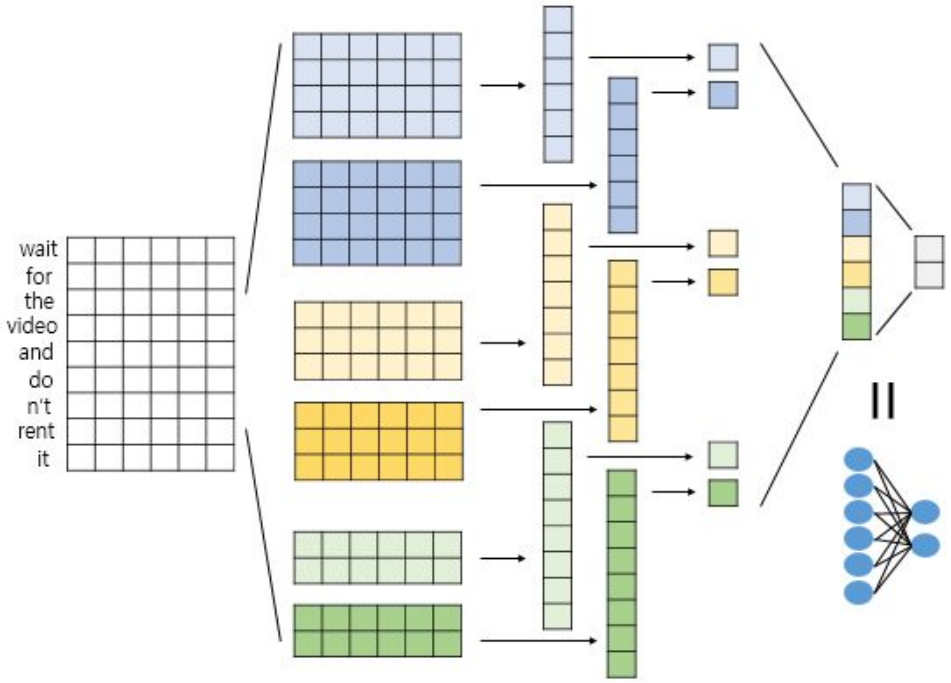
```
25000/25000 [==============================] - 3s 3ms/step - loss: 0.5373 - acc: 0.8873
테스트 정확도: 0.8873
```

1D CNN으로 스팸 메일 여부 분류하기

```
from tensorflow.keras.layers import Dense, Conv1D, GlobalMaxPooling1D, Embedding, Dropout, MaxPoolin
g1D
from tensorflow.keras.models import Sequential
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint

embedding_dim = 32
dropout_ratio = 0.3
num_filters = 32
kernel_size = 5

model = Sequential()
model.add(Embedding(vocab_size, embedding_dim))
model.add(Dropout(dropout_ratio))
model.add(Conv1D(num_filters, kernel_size, padding='valid', activation='relu'))
model.add(GlobalMaxPooling1D())
model.add(Dropout(dropout_ratio))
model.add(Dense(1, activation='sigmoid'))
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['acc'])

es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=3)
mc = ModelCheckpoint('best_model.h5', monitor = 'val_acc', mode='max', verbose=1, save_best_only=Tru
e)

history = model.fit(X_train_padded, y_train, epochs=10, batch_size=64, validation_split=0.2, callbac
ks=[es, mc])
```

```
테스트 정확도: 0.9797
```

05 Multi-Kernel 1D CNN으로 네이버 영화 리뷰 분류하기




```
from tensorflow.keras.models import Sequential, Model
from tensorflow.keras.layers import Embedding, Dropout, Conv1D, GlobalMaxPooling1D, Dense, Input, Fl
atten, Concatenate
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
from tensorflow.keras.models import load_model

embedding_dim = 128
dropout_ratio = (0.5, 0.8)
num_filters = 128
hidden_units = 128

model_input = Input(shape = (max_len,))
z = Embedding(vocab_size, embedding_dim, input_length = max_len, name="embedding")(model_input)
z = Dropout(dropout_ratio[0])(z)

conv_blocks = []

for sz in [3, 4, 5]:
    conv = Conv1D(filters = num_filters,
                          kernel_size = sz,
                          padding = "valid",
                          activation = "relu",
                          strides = 1)(z)
    conv = GlobalMaxPooling1D()(conv)
    conv_blocks.append(conv)
z = Concatenate()(conv_blocks) if len(conv_blocks) > 1 else conv_blocks[0]
z = Dropout(dropout_ratio[1])(z)
z = Dense(hidden_units, activation="relu")(z)
model_output = Dense(1, activation="sigmoid")(z)

model = Model(model_input, model_output)
model.compile(loss="binary_crossentropy", optimizer="adam", metrics=["acc"])

es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=4)
mc = ModelCheckpoint('CNN_model.h5', monitor='val_acc', mode='max', verbose=1, save_best_only=True)

model.fit(X_train, y_train, batch_size=64, epochs=10, validation_split=0.2, verbose=2, callbacks=[e
s, mc])

loaded_model = load_model('CNN_model.h5')
print("\n 테스트 정확도: %.4f" % (loaded_model.evaluate(X_test, y_test)[1]))
```

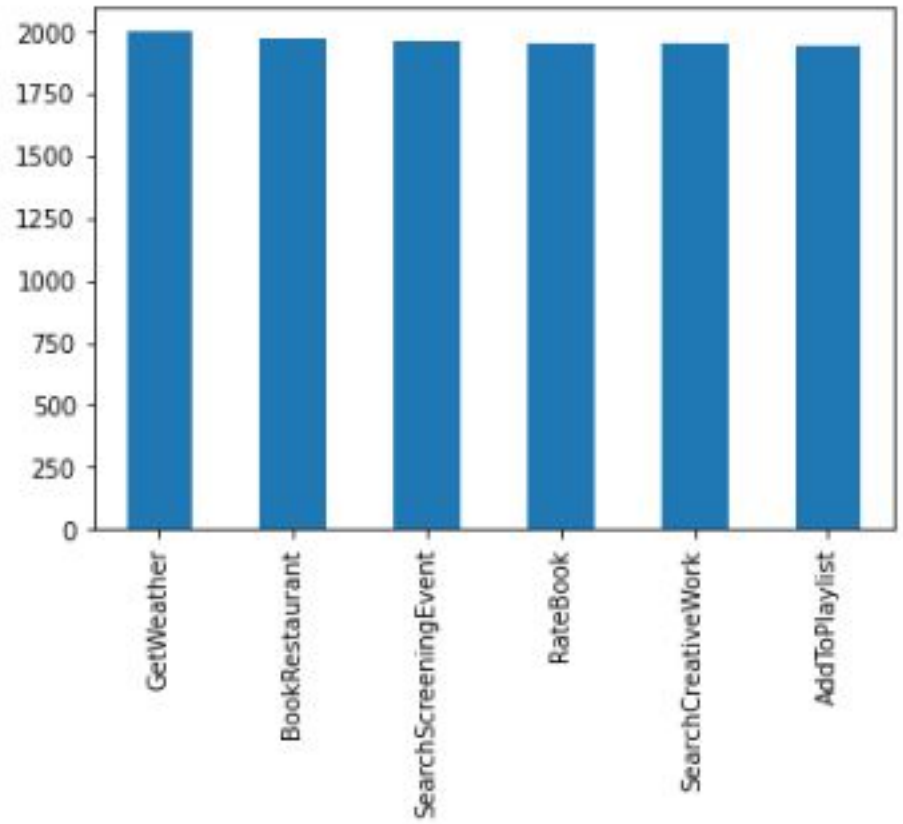
```
테스트 정확도: 0.8430
```

06 사전 훈련된 워드 임베딩을 이용한 의도 분류

```
train_data
```

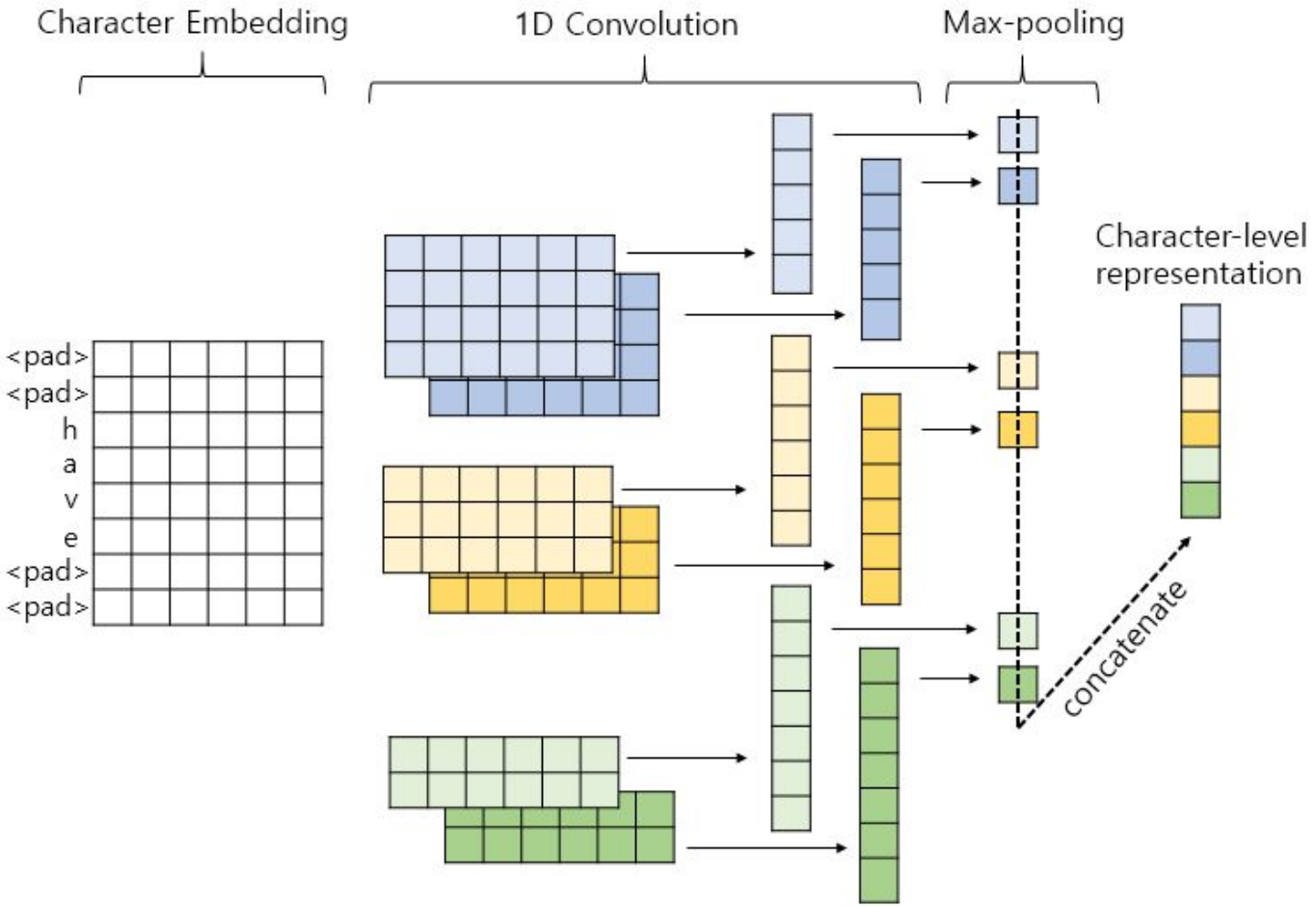
	intent	label
0	add another song to the cita rom ntica playlist	AddToPlaylist
1	add clem burke in my playlist pre party r b jams	AddToPlaylist
2	add live from aragon ballroom to trapeo	AddToPlaylist
3	add unite and win to my night out	AddToPlaylist
4	add track to my digster future hits	AddToPlaylist
...	...	...
11779	can a i get the movie schedule for sympathy fo...	SearchScreeningEvent
11780	find movie schedules for animated movies aroun...	SearchScreeningEvent
11781	what time is bordertown trail showing	SearchScreeningEvent
11782	in the neighbourhood find movies with movie times	SearchScreeningEvent
11783	what cinema has the closest movies	SearchScreeningEvent

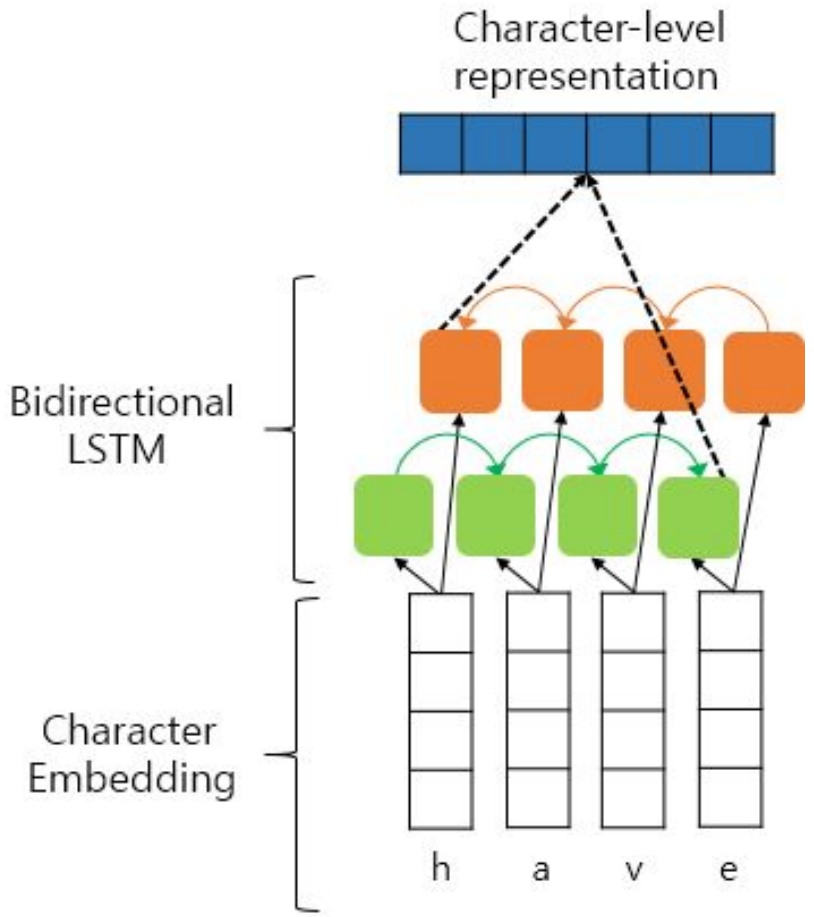
11784 rows × 2 columns




```
정확도(Accuracy) : 0.99
```

07 문자 임베딩 Character Embedding





감사합니다

Q&A