

# Keras

가 (https://keras.io/)

- API(Keras Functional API)

(Compile) (Training)

compile(): 가 ,

```
#
from tensorflow.keras.layers import SimpleRNN, Embedding, Dense
from tensorflow.keras.models import Sequential
max_features = 10000

model = Sequential()
model.add(Embedding(max_features, 32))
model.add(SimpleRNN(32)) #RNN
model.add(Dense(1, activation='sigmoid'))
model.compile(optimizer='rmsprop', loss='binary_crossentropy',
metrics=['acc'])
```

, 가 ,

optimizer : . adam sgd  
. loss : (loss function) . metric :

	mean_squared_error ( )	-	-
	categorical_crossentropy ( )		10
	sparse_categorical_crossentropy		- 가
	binary_crossentropy ( )		10 IMDB ,

fit() : , ,  
(fitting) 가 fit()

```
# compile()
```

```
model.fit(X_train, y_train, epochs=10, batch_size=32)
```

```
epochs = 10, batch_size = 32,
        batch_size=None
```

```
model.fit(X_train, y_train, epochs=10, batch_size=32, verbose=0,
          validation_data(X_val, y_val))
```

```
validation_data(x_val, y_val) = (validation data)
```

(overfitting) loss가 가

```
validation_split = validation_data
```

Xtrain y\_train

## 가(Evaluation) (Prediction)

- [keras](#)
- [Deep Learning with Keras](#)
- [Keras](#)
- [Keras Documentation](#)
- [keras\\_](#)

- [LSTM](#)
- <https://tykimos.github.io/lecture/>
- <http://course.fast.ai/>
- <http://iostream.tistory.com/111>
- <https://datascienceschool.net/view-notebook/1d93b9dc6c624fbaa6af2ce9290e2479/>

- [Hyperparameter tuning](#)
- [Machine Learning](#)

From:  
<http://jace.link/> - **Various Ways**

Permanent link:  
<http://jace.link/open/keras>

Last update: **2020/07/19 11:52**

