

Embedding

[\[source\]](#)

```
keras.layers.embeddings.Embedding(input_dim, output_dim, embeddings_initializer='uniform', e
```

正の整数（インデックス）を固定次元の密ベクトルに変換します。 例) `[[4],[20]] -> [[0.25, 0.1], [0.6, -0.2]]`

このレイヤーはモデルの最初のレイヤーとしてのみ利用できます。

例

```
model = Sequential()
model.add(Embedding(1000, 64, input_length=10))
# the model will take as input an integer matrix of size (batch, input_length).
# the largest integer (i.e. word index) in the input should be no larger than 999 (vocabulary size)
# now model.output_shape == (None, 10, 64), where None is the batch dimension.

input_array = np.random.randint(1000, size=(32, 10))

model.compile('rmsprop', 'mse')
output_array = model.predict(input_array)
assert output_array.shape == (32, 10, 64)
```

引数

- **input_dim**: 正の整数. 語彙数. 入力データの最大インデックス + 1.
- **output_dim**: 0以上の整数. 密なembeddingsの次元数.
- **embeddings_initializer**: `embeddings` 行列の **Initializers**.
- **embeddings_regularizer**: `embeddings` 行列に適用する **Regularizers**.
- **embeddings_constraint**: `embeddings` 行列に適用する **Constraints**.
- **mask_zero**: 真理値. 入力の0をパディングのための特別値として扱うかどうか. これは入力の系列長が可変長となりうる変数を入力にもつ **Recurrentレイヤー** に対して有効です. この引数が `True` のとき, 以降のレイヤーは全てこのマスクをサポートする必要があり, そうしなければ, 例外が起きます. `mask_zero` が `True` のとき, `index 0` は語彙の中で使えません (`input_dim` は `語彙数+1` と等しくなるべきです) .
- **input_length**: 入力の系列長 (定数) . この引数はこのレイヤーの後に `Flatten` から `Dense` レイヤーへ接続する際に必要です (これがないと, `dense` の出力の `shape` を計算できません).

入力のshape

`shape` が `(batch_size, sequence_length)` の2階テンソル.

出力のshape

shapeが `(batch_size, sequence_length, output_dim)` の3階テンソル.

参考文献

- [A Theoretically Grounded Application of Dropout in Recurrent Neural Networks](#)