

/ Trabajo Final: NLP

Francisco Vives

```
"""
    Operator to the selected object
    """
    def __init__(self, mirror_mirror_x):
        self.mirror_mirror_x = mirror_mirror_x
```

```
def __call__(self, context):
    """
    """
    if context.active_object is not None:
```

```
def __call__(self, context):
    """
    """
    if context.active_object is not None:
```

```
def __call__(self, context):
    """
    """
    if context.active_object is not None:
```

```
def __call__(self, context):
    """
    """
    if context.active_object is not None:
```

```
def __call__(self, context):
    """
    """
    if context.active_object is not None:
```

> Se entrega el código realizado en
<https://github.com/franciscovives/itba-nlp.git>

> Completados, nada por mostrar

> R2 = 0.6811023779012132

> Callbacks: ModelCheckpoint, EarlyStopping, ReduceLROnPlateau

```
model = Sequential()
model.add(Embedding(vocab_size + 1, ...
model.add(Bidirectional(LSTM(lstm... activation='tanh'))))
model.add(Dense(dense_size))
model.add(BatchNormalization())
model.add(Dropout(drops))
model.add(Dense(dense_size))
model.add(BatchNormalization())
model.add(Dropout(drops))
model.add(Activation('relu'))
model.add(Dense(1, activation='linear'))
```

> R2 = 0.6811023779012132

> Callbacks: ModelCheckpoint,
EarlyStopping, ReduceLROnPlateau

```
Layer (type)                 Output Shape              Param #   Connected to
=====
input x (InputLayer)         [(None, 227)]             0         []
embedding (Embedding)        (None, 227, 128)          5504      ['input x[0][0]']
conv1d (Conv1D)              (None, 227, 128)          49280     ['embedding[0][0]']
'''
conv1d_8 (Conv1D)            (None, 227, 128)          82048     ['embedding[0][0]']
max_pooling1d (MaxPooling1D) (None, 4, 128)            0         ['conv1d[0][0]']
...
max_pooling1d_8 (MaxPooling1D) (None, 4, 128)            0         ['conv1d_8[0][0]']
concatenate (Concatenate)    (None, 4, 1152)           0         ['max_pooling1d_8[0][0]']
flatten (Flatten)            (None, 4608)              0         ['concatenate[0][0]']
dense (Dense)                (None, 100)               460900    ['flatten[0][0]']
dense_1 (Dense)              (None, 100)               10100     ['dense[0][0]']
dense_2 (Dense)              (None, 1)                 101       ['dense_1[0][0]']
=====
Total params: 1,067,581
Trainable params: 1,067,581
```

> **Archivo** = 005-dataaug-lstm_Para_investigar.ipynb

> **Conclusión:** No mejoró, le llevó mas iteraciones llegar a valores de R2 similares, la loss fue más alta.

> **Archivo** = 005-tsne_Para_investigar.ipynb. Datos en path ./005

> **Conclusión:** El isótopo 1: se encuentran cercanos en ambos modelos, podría indicar que las moléculas requieren poca cantidad de neutrones

Enlaces triples (#): se encuentran muy distantes.

La composición de los enlaces dobles (/) también están distantes

Se probó TSNE con varios parámetros, imágenes en directorio ./005

> El trabajo no se completó por el error obtenido en el ejercicio 4, donde el método `_get_latent` no existía.

>> Probé alterar el método

>> Usar versión más vieja del `stable_baseline` pero otras librerías ya no compilan

> **Conclusión:** Se entrega igualmente el código del 1, 2, 3, y 4. Respuestas de la plataforma hechas.



Francisco Vives