In [ ]:
```
# Install TensorFlow
# !pip install -q tensorflow-gpu==2.0.0-beta1

try:
  %tensorflow_version 2.x  # Colab only.
except Exception:
  pass

import tensorflow as tf
print(tf.__version__)
```

```
`%tensorflow_version` only switches the major version: `1.x` or `2.x`.
You set: `2.x  # Colab only.`. This will be interpreted as: `2.x`.


TensorFlow 2.x selected.
2.0.0-beta1
```

In [ ]:
```
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
```

In [ ]:
```
# Just a simple test
sentences = [
    "I like eggs and ham.",
    "I love chocolate and bunnies.",
    "I hate onions."
]
```

In [ ]:
```
MAX_VOCAB_SIZE = 20000
tokenizer = Tokenizer(num_words=MAX_VOCAB_SIZE)
tokenizer.fit_on_texts(sentences)
sequences = tokenizer.texts_to_sequences(sentences)
```

In [ ]:
```
print(sequences)
```

```
[[1, 3, 4, 2, 5], [1, 6, 7, 2, 8], [1, 9, 10]]
```

In [ ]:
```
# How to get the word to index mapping?
tokenizer.word_index
```

Out[ ]:
```
{'and': 2,
 'bunnies': 8,
 'chocolate': 7,
 'eggs': 4,
 'ham': 5,
 'hate': 9,
 'i': 1,
 'like': 3,
 'love': 6,
 'onions': 10}
```

In [ ]:
```
# use the defaults
data = pad_sequences(sequences)
```

```
print(data)
```

```
[[ 1  3  4  2  5]
 [ 1  6  7  2  8]
 [ 0  0  1  9 10]]
```

In [ ]:

```
MAX_SEQUENCE_LENGTH = 5
data = pad_sequences(sequences, maxlen=MAX_SEQUENCE_LENGTH)
print(data)
```

```
[[ 1  3  4  2  5]
 [ 1  6  7  2  8]
 [ 0  0  1  9 10]]
```

In [ ]:

```
data = pad_sequences(sequences, maxlen=MAX_SEQUENCE_LENGTH, padding='post')
print(data)
```

```
[[ 1  3  4  2  5]
 [ 1  6  7  2  8]
 [ 1  9 10  0  0]]
```

In [ ]:

```
# too much padding
data = pad_sequences(sequences, maxlen=6)
print(data)
```

```
[[ 0  1  3  4  2  5]
 [ 0  1  6  7  2  8]
 [ 0  0  0  1  9 10]]
```

In [ ]:

```
# truncation
data = pad_sequences(sequences, maxlen=4)
print(data)
```

```
[[ 3  4  2  5]
 [ 6  7  2  8]
 [ 0  1  9 10]]
```

In [ ]:

```
data = pad_sequences(sequences, maxlen=4, truncating='post')
print(data)
```

```
[[ 1  3  4  2]
 [ 1  6  7  2]
 [ 0  1  9 10]]
```