TensorFlow Modelleri İçin Kullanılacak Metinlerin Hazırlanması

Bir makine öğrenimi modelinde kullanılacak metni hazırlamak için belirli adımlar şu şekildedir:

1. Model için sayısal değerler elde etmek için kelimelerin tokenize edilmesi
2. Cümlelerin sayısal dizilerinin oluşturulması
3. Dizilerin aynı uzunlukta olacak şekilde ayarlanması

Bu colab dosyasında, dizilerin hepsinin aynı uzunlukta olmasını sağlamak için dolgu (`padding`) kullanmayı öğreneceğiz.

Gerekli Sınıfların İçeri Aktarılması

In [1]:

```
# İçeri Aktar : Tokenizer and pad_sequences
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
```

Birkaç Cümlenin Yazılması

In [2]:

```
sentences = [
    'My favorite food is ice cream',
    'do you like ice cream too?',
    'My dog likes ice cream!',
    "your favorite flavor of icecream is chocolate",
    "chocolate isn't good for dogs",
    "your dog, your cat, and your parrot prefer broccoli"
]
print(sentences)
```

```
['My favorite food is ice cream', 'do you like ice cream too?', 'My dog like
s ice cream!', 'your favorite flavor of icecream is chocolate', "chocolate i
sn't good for dogs", 'your dog, your cat, and your parrot prefer broccoli']
```

`Tokenizer` Oluşturulması ve Sözcük Dağarcığı Dışındakiler İçin Belirteç Tanımlanması

Tokenizer'ı oluştururken sözlükteki maksimum kelime sayısını belirleyebilirsiniz. Ayrıca, sözlükte olmayan (OOV- Out Of the Vocabulary), başka bir deyişle sözlükte olmayan kelimeleri temsil etmek için bir belirteç belirleyebilirsiniz. Bu OOV belirteci, kelime dizininde olmayan kelimeleri içeren cümleler için diziler oluşturduğunuzda kullanılacaktır.

In [3]:

```
tokenizer = Tokenizer(num_words = 100, oov_token="<OOV>")
```

Kelimlerin Simgelenmesi (Tokenize Edilmesi)

In [4]:

```
tokenizer.fit_on_texts(sentences)
word_index = tokenizer.word_index
print(word_index)
```

```
{'<OOV>': 1, 'your': 2, 'ice': 3, 'cream': 4, 'my': 5, 'favorite': 6, 'is':
7, 'dog': 8, 'chocolate': 9, 'food': 10, 'do': 11, 'you': 12, 'like': 13, 't
oo': 14, 'likes': 15, 'flavor': 16, 'of': 17, 'icecream': 18, "isn't": 19,
'good': 20, 'for': 21, 'dogs': 22, 'cat': 23, 'and': 24, 'parrot': 25, 'pref
er': 26, 'broccoli': 27}
```

Cümlelerin Dizilere Dönüştürülmesi

Artık her kelimenin kelime dizininde benzersiz bir numarası var. Ancak, bir cümledeki kelimeler belirli bir sıradadır. Kelimeleri rastgele karıştırıp sonucun bir cümle olmasını sağlayamazsınız.

Örneğin, "chocolate isn't good for dogs" mükkemmel bir cümle olsa da, "dogs isn't for chocolate good" cümlesi bir anlam ifade etmez.

Dolayısıyla, metni makine öğrenimi programları tarafından anlamlı bir şekilde kullanılabilecek şekilde temsil etmenin bir sonraki adımı: metindeki cümleleri temsil eden sayısal diziler oluşturmaktır.

Her cümle, her kelimenin kelime dizinindeki numarasıyla değiştirildiği bir diziye dönüştürülecektir.

In [5]:

```
sequences = tokenizer.texts_to_sequences(sentences)
print (sequences)
```

```
[[5, 6, 10, 7, 3, 4], [11, 12, 13, 3, 4, 14], [5, 8, 15, 3, 4], [2, 6, 16, 1
7, 18, 7, 9], [9, 19, 20, 21, 22], [2, 8, 2, 23, 24, 2, 25, 26, 27]]
```

Tüm Dizilerin Aynı Uzunluğa Getirilmesi

Daha sonra, bir modeli eğitmek için dizileri bir sinir ağına beslediğinizde, dizilerin hepsinin aynı boyutta olması gerekir. Şu anda dizilerin uzunlukları değişkendir, bu nedenle bir sonraki adım, hepsini sıfırlarla doldurarak veya keserek hepsini aynı boyutta yapmaktır.

Hepsinin aynı uzunlukta olması için dizilere sıfır eklemek için `tf.keras.preprocessing.sequence.pad_sequences` kullanın. Varsayılan olarak dolgu, dizilerin başında yer alır. Ancak sonunda doldurmayı belirtebilirsiniz.

Aynı şekilde sekansların doldurulacağı maksimum uzunluğu isteğe bağlı olarak belirtebilirsiniz. Belirtilen maksimum uzunluktan daha uzun olan diziler kesilecektir. Varsayılan olarak, diziler dizinin başından itibaren kesilir, ancak sondan kesmeyi belirtebilirsiniz.

Maksimum uzunluğu belirtmezseniz, diziler en uzun cümlenin uzunluğuna uyacak şekilde doldurulur.

Dizileri doldurma ve kesme ile ilgili tüm seçenekler için, aşağıdaki linki kullanabilirsiniz:
https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/sequence/pad_sequences
(https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/sequence/pad_sequences)

In [6]:

```
padded = pad_sequences(sequences)
print("\nKelime İndeksleri = " , word_index)
print("\nDiziler = " , sequences)
print("\nDoldurulmuş Diziler:")
print(padded)
```

```
Kelime İndeksleri =  {'<OOV>': 1, 'your': 2, 'ice': 3, 'cream': 4, 'my': 5,
'favorite': 6, 'is': 7, 'dog': 8, 'chocolate': 9, 'food': 10, 'do': 11, 'yo
u': 12, 'like': 13, 'too': 14, 'likes': 15, 'flavor': 16, 'of': 17, 'icecrea
m': 18, "isn't": 19, 'good': 20, 'for': 21, 'dogs': 22, 'cat': 23, 'and': 2
4, 'parrot': 25, 'prefer': 26, 'broccoli': 27}

Diziler =  [[5, 6, 10, 7, 3, 4], [11, 12, 13, 3, 4, 14], [5, 8, 15, 3, 4],
[2, 6, 16, 17, 18, 7, 9], [9, 19, 20, 21, 22], [2, 8, 2, 23, 24, 2, 25, 26,
27]]

Doldurulmuş Diziler:
[[ 0  0  0  5  6 10  7  3  4]
 [ 0  0  0 11 12 13  3  4 14]
 [ 0  0  0  0  5  8 15  3  4]
 [ 0  0  2  6 16 17 18  7  9]
 [ 0  0  0  0  9 19 20 21 22]
 [ 2  8  2 23 24  2 25 26 27]]
```

In [7]:

```
# Dolgulu dizileri (pad sequence) için bir maksimum uzunluk belirtin
padded = pad_sequences(sequences, maxlen=15)
print(padded)
```

```
[[ 0  0  0  0  0  0  0  0  0  5  6 10  7  3  4]
 [ 0  0  0  0  0  0  0  0  0 11 12 13  3  4 14]
 [ 0  0  0  0  0  0  0  0  0  0  5  8 15  3  4]
 [ 0  0  0  0  0  0  0  0  2  6 16 17 18  7  9]
 [ 0  0  0  0  0  0  0  0  0  0  9 19 20 21 22]
 [ 0  0  0  0  0  0  2  8  2 23 24  2 25 26 27]]
```

In [8]:

```
# Dolguyu (sıfırları) dizilerin sonuna koyun
padded = pad_sequences(sequences, maxlen=15, padding="post")
print(padded)
```

```
[[ 5  6 10  7  3  4  0  0  0  0  0  0  0  0  0]
 [11 12 13  3  4 14  0  0  0  0  0  0  0  0  0]
 [ 5  8 15  3  4  0  0  0  0  0  0  0  0  0  0]
 [ 2  6 16 17 18  7  9  0  0  0  0  0  0  0  0]
 [ 9 19 20 21 22  0  0  0  0  0  0  0  0  0  0]
 [ 2  8  2 23 24  2 25 26 27  0  0  0  0  0  0]]
```

In [9]:

```
# Dizilerin uzunluğunu sınırlayın, bazı dizilerin kesildiğini göreceksiniz.
padded = pad_sequences(sequences, maxlen=3)
print(padded)
```

```
[[ 7  3  4]
 [ 3  4 14]
 [15  3  4]
 [18  7  9]
 [20 21 22]
 [25 26 27]]
```

Cümlelerden bazıları kelime dizininde olmayan kelimeler içeriyorsa ne olur?

Kelime dağarcığının dışında belirtecinin kullanıldığı yere geldik. Kelime dizininde olmayan kelimeleri içeren bazı cümleler için diziler oluşturmayı deneyelim.

In [10]:

```
# Kelime dizininde olmayan kelimeleri içeren cümleleri dizilere dönüştürmeyi deneyin.

test_data = [
    "my best friend's favorite ice cream flavor is strawberry",
    "my dog's best friend is a manatee"
]
print (test_data)

# Kelime dizininde hangi sayının kelime dağarcığının yetersiz olduğuna karşılık geldiğini k
print("<OOV> 'nin kelime indekslerindeki sahip olduğu sayı :", word_index['<OOV>'])

# Test cümlelerini dizilere dönüştürün
test_seq = tokenizer.texts_to_sequences(test_data)
print("\nTest Dizisi = ", test_seq)

# Yeni dizileri doldurun (padding)
padded = pad_sequences(test_seq, maxlen=10)
print("\nDoldurulmuş Test Dizisi: ")

# Kelime dizininde olmayan bir kelimenin olduğu her yerde "1"in dizide göründüğüne dikkat e
print(padded)
```

```
["my best friend's favorite ice cream flavor is strawberry", "my dog's best
friend is a manatee"]
<OOV> 'nin kelime indekslerindeki sahip olduğu sayı : 1

Test Dizisi =  [[5, 1, 1, 6, 3, 4, 16, 7, 1], [5, 1, 1, 1, 7, 1, 1]]

Doldurulmuş Test Dizisi:
[[ 0  5  1  1  6  3  4 16  7  1]
 [ 0  0  0  5  1  1  1  7  1  1]]
```

In [ ]: