

Digital Talent Scholarship 2022

Natural Language Processing 1

Lead a sprint through the Machine Learning Track

Agenda

- Intro to NLP
- Sentiment In Text
- Tokenization
- Pad Sequences
- Word Embedding

Objektif Pembelajaran

- Memahami cara penggunaan Tokenizer
- Memahami apa itu Embedding
- Menggunakan Sequence Model

Are your students ML-ready?

Apa itu NLP?

Natural Language Processing, atau yang biasa disingkat sebagai **NLP**, adalah sebuah cabang dari kecerdasan buatan yang berhubungan dengan interaksi antara komputer dan manusia menggunakan bahasa alami.

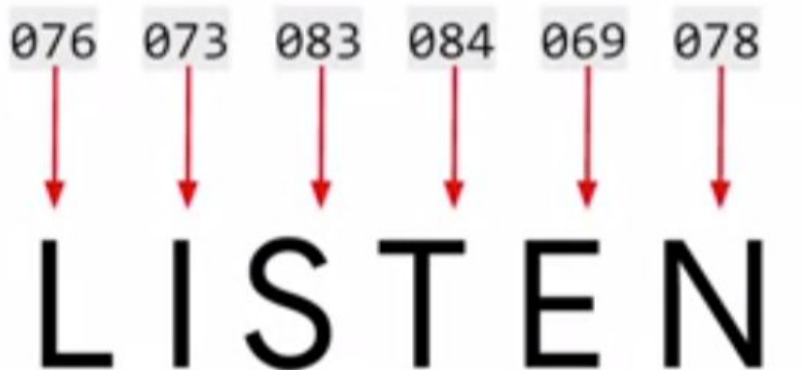
Contoh NLP

- Auto-Correct
- Checking on copyright and plagiarism violation
- Summarize
- Change words
- Search Result
- Email filters
- Language Translation

Sentiment in Text

Gimana caranya agar komputer dapat mengerti kata-kata?

Kita bisa memulai dengan menggunakan ASCII per huruf. Tapi...



A diagram illustrating the mapping of the word "LISTEN" to its corresponding ASCII values. Above each letter, its ASCII value is displayed in a grey box: 076 for 'L', 073 for 'I', 083 for 'S', 084 for 'T', 069 for 'E', and 078 for 'N'. Red arrows point from each of these boxes down to the respective letter in the word "LISTEN".

Letter	ASCII Value
L	076
I	073
S	083
T	084
E	069
N	078

Kelemahan

Kata dengan huruf yang sama namun berbeda urutan bisa dianggap sama oleh mesin.

083 073 076 069 078 084
↓ ↓ ↓ ↓ ↓ ↓

SILENT

076 073 083 084 069 078
↓ ↓ ↓ ↓ ↓ ↓

LISTEN

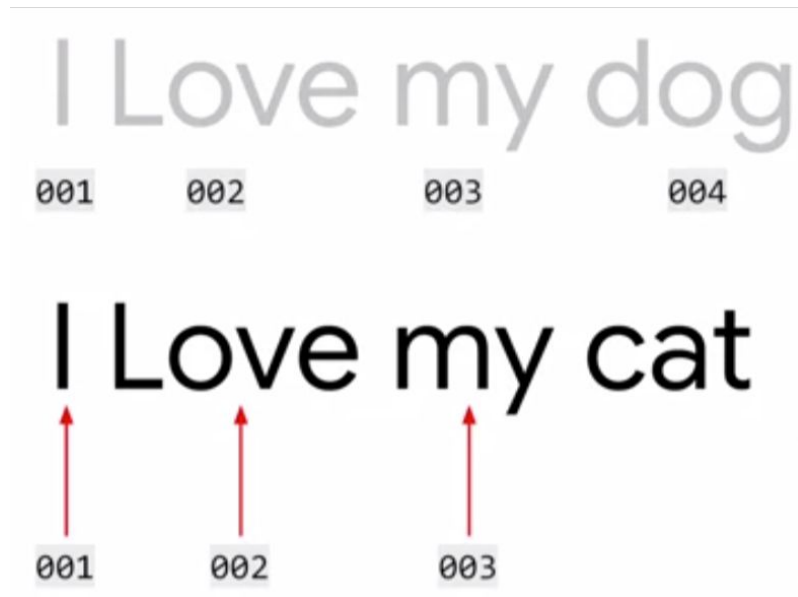
Sentiment in Text

Bagaimana kalau kita menafsirnya per kata?



Sentiment in Text

Kalau ada kata baru bagaimana?



Sentiment in Text

Apa yang dilihat Komputer

001

002

003

004

001

002

003

005

[https://www.youtube.com/
watch?v=fNxaJsNG3-s](https://www.youtube.com/watch?v=fNxaJsNG3-s)

Tokenization

Apa itu Tokenizer? Apa yg dilakukan oleh Tokenizer?

Tokenizer adalah salah satu API Keras yang berfungsi untuk memecah kalimat menjadi kata-kata. Tokenizer akan membantu dalam memahami konteks atau mengembangkan model untuk NLP. Tokenisasi membantu dalam menafsirkan makna teks dengan menganalisis urutan kata-kata.

Tokenization

Pertanyaan

1. Apa itu num_words?
2. Apa yang akan terjadi jika num_words besar ataupun kecil?

```
import tensorflow as tf
from tensorflow import keras
from keras.preprocessing.text import Tokenizer

sentences = [
    "I love my dog",
    "I love my cat"
]

tokenizer = Tokenizer(num_words = 100)
tokenizer.fit_on_texts(sentences)
word_index = tokenizer.word_index
print(word_index)
```

Tokenization

Jawaban

1. Apa itu num_words?

Besar vocabulary dalam sebuah tokenizer.

2. Apa yang akan terjadi jika num_words besar ataupun kecil?

Semakin besar num_words, semakin tinggi akurasi, namun semakin lama dalam training. Begitu juga sebaliknya.

Tokenization

Peraturan Tokenizer:

1. Tidak ada huruf kapital
2. Tidak ada tanda baca
3. Sort tergantung pada kata yang paling sering digunakan

Sequences

Sequences dengan kalimat sebelumnya

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

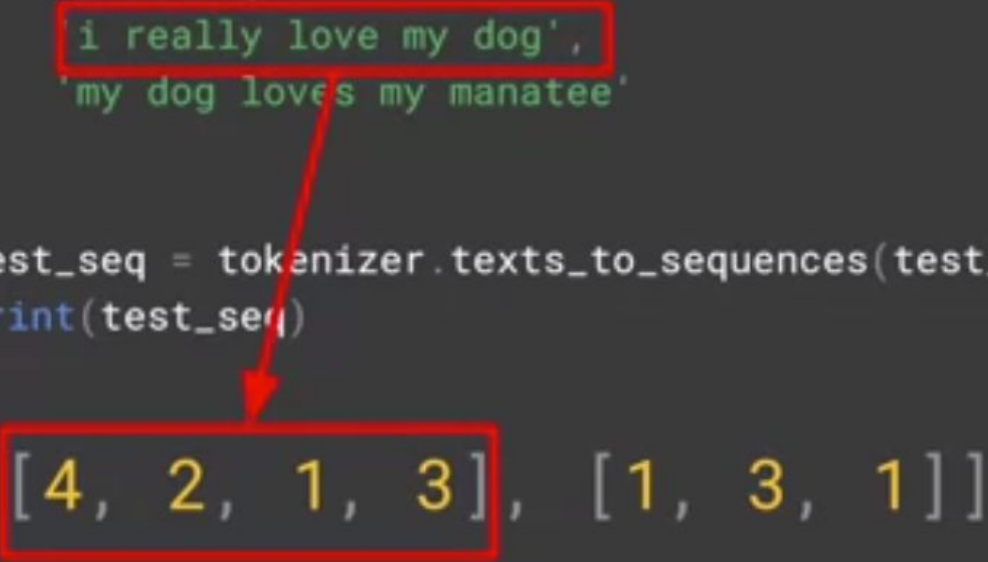
[[1, 2, 3, 4], [1, 2, 3, 5]]
```

Bagaimana kalau kita menggunakan kalimat lain untuk mencobanya?

Sequences dengan kalimat baru

Beberapa data sebelumnya hilang. Apa yang bisa kita lakukan?

```
test_data = [  
    'i really love my dog',  
    'my dog loves my manatee'  
]  
  
test_seq = tokenizer.texts_to_sequences(test_data)  
print(test_seq)
```



```
[[4, 2, 1, 3], [1, 3, 1]]
```

<OOV>

OOV (Out of Vocabulary)

```
tokenizer = Tokenizer(num_words = 100, oov_token="<OOV>")  
tokenizer.fit_on_texts(sentences)  
word_index = tokenizer.word_index
```

```
test_data = [  
    "I really love my dog",  
    "my dog loves my manatee"  
]
```

```
test_seq = tokenizer.texts_to_sequences(test_data)  
test_seq
```

```
[[2, 1, 3, 4, 5], [4, 5, 1, 4, 1]]
```

Padding and Truncating

Kenapa perlu Padding and Truncating? Karena kita butuh keseragaman dalam input_shape

```
from tensorflow.keras.preprocessing.sequence import pad_sequences  
sequences = tokenizer.texts_to_sequences(sentences)
```

```
padded = pad_sequences(sequences)  
padded
```

```
array([[ 0,  0,  0,  3,  4,  2,  5],  
       [ 0,  0,  0,  3,  4,  2,  6],  
       [ 7,  8,  9,  2,  5, 10, 11]], dtype=int32)
```

Padding and Truncating

```
padding="post"
```

```
from tensorflow.keras.preprocessing.sequence import pad_sequences  
sequences = tokenizer.texts_to_sequences(sentences)
```

```
padded = pad_sequences(sequences, padding="post")  
padded
```

```
array([[ 3,  4,  2,  5,  0,  0,  0],  
       [ 3,  4,  2,  6,  0,  0,  0],  
       [ 7,  8,  9,  2,  5, 10, 11]], dtype=int32)
```

Padding and Truncating

maxlen=5

```
from tensorflow.keras.preprocessing.sequence import pad_sequences  
sequences = tokenizer.texts_to_sequences(sentences)
```

```
padded = pad_sequences(sequences, padding="post", maxlen = 5)  
padded
```

```
array([[ 3,  4,  2,  5,  0],  
       [ 3,  4,  2,  6,  0],  
       [ 9,  2,  5, 10, 11]], dtype=int32)
```

Padding and Truncating

```
truncating="post"
```

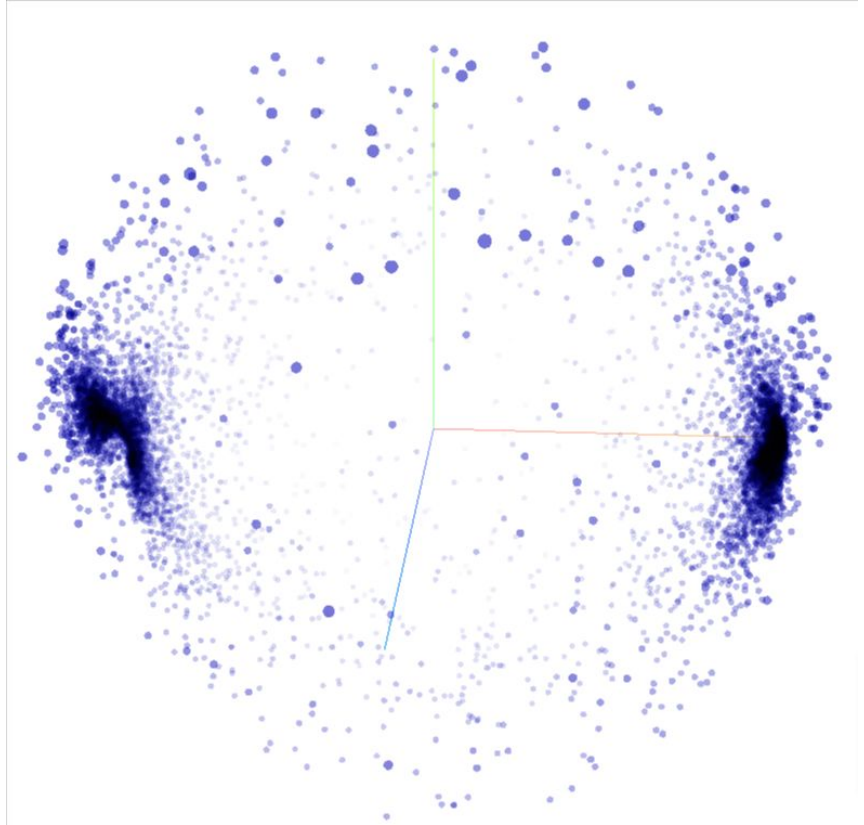
```
from tensorflow.keras.preprocessing.sequence import pad_sequences
sequences = tokenizer.texts_to_sequences(sentences)

padded = pad_sequences(sequences, padding="post", truncating="post", maxlen=5)
padded

array([[3, 4, 2, 5, 0],
       [3, 4, 2, 6, 0],
       [7, 8, 9, 2, 5]], dtype=int32)
```

Word Embedding

Embedding adalah kelas teknik di mana kata-kata individual direpresentasikan sebagai vektor bernilai nyata dalam ruang vektor yang telah ditentukan. Setiap kata dipetakan ke satu vektor dan nilai-nilai vektor dipelajari dengan cara yang menyerupai jaringan saraf.



Q & A

Thank You