

Laporan Chapter 7 - NLP dengan Hugging Face

Main NLP Task

Nama Rey Rizqi Anugerah

Kelas TK45 01

NIM: 1103210146

Tujuan Chapter

1. Mengenal berbagai tugas utama dalam Natural Language Processing (NLP).
2. Memahami penerapan tugas-tugas tersebut menggunakan model dari library Hugging Face Transformers.
3. Mengetahui pipeline NLP yang tersedia untuk mempermudah implementasi tugas-tugas NLP.

1. Tugas Utama NLP

Natural Language Processing mencakup berbagai tugas yang berfokus pada pemahaman dan pengolahan teks dalam bahasa alami. Berikut adalah tugas utama NLP yang dibahas:

1.1 Sentiment Analysis

- **Tujuan:** Menentukan sentimen dari sebuah teks (positif, negatif, atau netral).
- **Contoh:** Input: *"I love this product!"*
Output: Sentimen: Positif

Implementasi:

```
from transformers import pipeline  
  
sentiment_analyzer = pipeline("sentiment-analysis")  
  
result = sentiment_analyzer("I love Hugging Face!")  
  
print(result)
```

1.2 Text Classification

- **Tujuan:** Mengkategorikan teks ke dalam kelas yang telah ditentukan.
- **Contoh:** Input: *"This article talks about machine learning."*
Output: Kategori: Teknologi

Implementasi:

```
classifier = pipeline("text-classification")  
result = classifier("This is an article about AI.")  
print(result)
```

1.3 Named Entity Recognition (NER)

- **Tujuan:** Mengidentifikasi entitas penting dalam teks (contoh: nama, tempat, organisasi).
- **Contoh:** Input: "Hugging Face is located in New York."
Output: Hugging Face: ORGANIZATION, New York: LOCATION

Implementasi:

```
ner = pipeline("ner", grouped_entities=True)  
result = ner("Hugging Face is based in New York City.")  
print(result)
```

1.4 Question Answering

- **Tujuan:** Menjawab pertanyaan berdasarkan teks yang diberikan.
- **Contoh:** Context: "Hugging Face was founded in 2016 in New York."
Pertanyaan: "Where was Hugging Face founded?"
Jawaban: "New York"

Implementasi:

```
qa = pipeline("question-answering")  
result = qa({  
    "question": "Where was Hugging Face founded?",  
    "context": "Hugging Face was founded in 2016 in New York."  
})  
print(result)
```

1.5 Text Generation

- **Tujuan:** Menghasilkan teks baru berdasarkan input.
- **Contoh:** Input: *"Once upon a time, there was a robot who"*
Output: *"loved to learn about humans and their languages."*

Implementasi:

```
text_generator = pipeline("text-generation", model="gpt2")  
  
result = text_generator("Once upon a time, there was a robot who", max_length=50)  
  
print(result)
```

1.6 Summarization

- **Tujuan:** Membuat ringkasan dari teks panjang.
- **Contoh:** Input: *"Hugging Face is an open-source AI company specializing in NLP. They provide tools for researchers and developers to build AI applications."*
Output: *"Hugging Face specializes in NLP and provides AI tools."*

Implementasi:

```
summarizer = pipeline("summarization")  
  
result = summarizer("Hugging Face is an open-source AI company specializing in NLP. They provide tools for researchers and developers to build AI applications.")  
  
print(result)
```

1.7 Translation

- **Tujuan:** Menerjemahkan teks dari satu bahasa ke bahasa lain.
- **Contoh:** Input: *"Bonjour, comment ça va?"*
Output: *"Hello, how are you?"*

Implementasi:

```
translator = pipeline("translation_fr_to_en")  
  
result = translator("Bonjour, comment ça va?")  
  
print(result)
```

1.8 Text-to-Speech (TTS)

- **Tujuan:** Mengubah teks menjadi audio.
- **Catatan:** Tugas ini tidak didukung langsung dalam pipeline **Transformers** tetapi memerlukan library tambahan seperti **SpeechBrain** atau **Coqui TTS**.

1.9 Zero-Shot Classification

- **Tujuan:** Mengklasifikasikan teks tanpa pelatihan awal pada label tertentu.
- **Contoh:** Input: *"This is a tutorial on NLP."*
Labels: ["education", "politics", "entertainment"]
Output: *"education"*

Implementasi:

```
zero_shot_classifier = pipeline("zero-shot-classification")

result = zero_shot_classifier(
    "This is a tutorial on NLP.",
    candidate_labels=["education", "politics", "entertainment"]
)

print(result)
```

2. Pipeline untuk Tugas NLP

Library Hugging Face menyediakan **pipeline** untuk menyederhanakan implementasi tugas-tugas NLP. Pipeline memungkinkan Anda untuk memanfaatkan model pra-trlatih dengan minimal kode.

Contoh Pipeline:

```
from transformers import pipeline

# Pipeline untuk sentiment analysis

sentiment_pipeline = pipeline("sentiment-analysis")

result = sentiment_pipeline("I love this course!")

print(result)
```

3. Model dan Dataset

- Setiap pipeline menggunakan model dan dataset bawaan yang sesuai dengan tugasnya.
- Anda dapat mengganti model dengan model kustom dari Hub:
- `pipeline("task", model="model_name")`

Kesimpulan

Chapter ini memberikan gambaran lengkap tentang tugas-tugas utama dalam NLP dan bagaimana memanfaatkannya dengan **Transformers Library**. Pipeline memungkinkan implementasi yang mudah, sedangkan fleksibilitas model Hugging Face membuka peluang besar untuk eksperimen.