

# Ringkasan Buku 1

Nama Buku: Tensorflow in Actions

Team: Wilhelmina Arlene, Luthfiah Maulidya, Fadhilah Dwi Istiani

## Chapter 13 — *Transformers*

### 🎯 Tujuan Utama Chapter 13

Chapter 13 membawa semua yang sudah kamu pelajari sebelumnya ke **level state-of-the-art**.

Jika:

- Chapter 11 = Seq2Seq klasik
- Chapter 12 = Seq2Seq + Attention

maka Chapter 13 menjawab:

**“Bagaimana jika kita membuang RNN sama sekali dan hanya memakai attention?”**

Inilah **Transformer modern**, fondasi:

- BERT
- GPT
- T5
- DistilBERT
- dan hampir semua model NLP mutakhir

## GAMBARAN BESAR TRANSFORMER (REKAP CEPAT)

Transformer:

- **tidak menggunakan RNN**
- **tidak memproses token secara berurutan**
- memproses **seluruh sequence sekaligus**
- mengandalkan **self-attention**

Keuntungan besar:

- paralel
- konteks panjang
- training jauh lebih cepat

## STRUKTUR BESAR CHAPTER 13

Chapter 13 dibagi menjadi **3 bagian utama**:

1. Transformer lebih detail (teori & komponen)
2. Menggunakan **pretrained BERT** untuk spam classification
3. **Question Answering** dengan Hugging Face Transformers

## **13.1 Transformers in More Detail**

### **13.1.1 Revisi Komponen Transformer**

Chapter ini mengulang (dengan lebih teknis) komponen inti Transformer:

#### **a. Embeddings**

- token embedding
- positional embedding
- dijumlahkan, bukan digabung

Positional embedding:

- memberi informasi urutan
- wajib karena attention tidak tahu posisi

#### **b. Self-Attention (Scaled Dot-Product)**

Proses inti:

1. Query (Q), Key (K), Value (V)
2. Hitung similarity  $Q \cdot K$
3. Scale ( $\sqrt{d}$ )
4. Softmax
5. Weighted sum terhadap V

Ini **lebih efisien** dari attention RNN.

#### **c. Multi-Head Attention**

Beberapa attention berjalan paralel:

- tiap head fokus aspek berbeda
- grammar
- semantic
- dependency jauh

Output semua head digabung.

#### **d. Residual Connections & Layer Normalization**

Kenapa penting?

- training model sangat dalam
- tanpa ini → gradient bermasalah

Residual:

- menjaga aliran gradient

LayerNorm:

- stabilitas training

## **13.2 Using Pretrained BERT for Spam Classification**

### **Apa itu BERT?**

BERT (Bidirectional Encoder Representations from Transformers):

- Transformer **encoder-only**
- membaca konteks **dua arah**
- pretrained pada korpus besar

Berbeda dari GPT:

- GPT → generative, autoregressive
- BERT → representational, understanding

## **Kenapa Pretrained Model Penting?**

Training Transformer dari nol:

- mahal
- butuh data besar
- butuh resource tinggi

Solusi:

## **Transfer learning dengan pretrained Transformer**

### **Pipeline Spam Classification dengan BERT**

Alur lengkap:

1. Input text
2. Tokenization khusus BERT
3. BERT encoder
4. Classification head (Dense)
5. Fine-tuning end-to-end

### **Hugging Face Tokenizer**

Tokenizer BERT:

- subword-based (WordPiece)
- menangani kata tidak dikenal
- konsisten dengan pretraining

👉 **Tokenizer sama pentingnya dengan model.**

### **Fine-Tuning BERT**

Fine-tuning berarti:

- semua parameter BERT ikut dilatih
- learning rate kecil
- training singkat

Hasil:

- performa tinggi
- jauh di atas LSTM klasik

## **13.3 Question Answering with Hugging Face Transformers**

### **Apa itu Question Answering (QA)?**

### Task QA:

- input:
  - context (paragraf)
  - question
- output:
  - jawaban (span teks)

Contoh:

Context: TensorFlow is developed by Google.

Question: Who developed TensorFlow?

Answer: Google

### Model yang Digunakan: DistilBERT

DistilBERT:

- versi ringan BERT
- lebih cepat
- ukuran lebih kecil
- performa mendekati BERT

Cocok untuk:

- eksperimen
- deployment

### Pipeline QA dengan Transformers

Langkah:

1. load dataset QA
2. tokenize context + question
3. model memprediksi:
  - start token
  - end token
4. ekstrak span jawaban

### Kenapa QA Sulit?

- model harus:
  - memahami konteks
  - memahami pertanyaan
  - menemukan jawaban **di dalam teks**

Transformer unggul karena:

- contextual representation kuat
- attention ke seluruh teks