

# Ringkasan Buku 1

Nama Buku: Tensorflow in Actions

Team: Wilhelmina Arlene, Luthfiah Maulidya, Fadhilah Dwi Istiani

## Chapter 11 — Sequence-to-Sequence Learning

### Tujuan Utama Chapter 11

Chapter 11 memperkenalkan **Sequence-to-Sequence (Seq2Seq) learning**, sebuah paradigma penting dalam NLP yang memungkinkan model:

**mengubah satu sequence menjadi sequence lain dengan panjang berbeda**

Contoh nyata:

- Machine Translation (English → German)
- Text Summarization
- Speech-to-Text
- Chatbot generasi awal

Chapter ini fokus pada **seq2seq klasik berbasis RNN**, *tanpa attention*.

### GAMBARAN BESAR SEQ2SEQ

#### Masalah yang Tidak Bisa Diselesaikan Model Sebelumnya

Model sebelumnya (Chapter 9–10):

- input → satu output
- atau input → prediksi kata berikutnya


Masalah baru:

- input dan output **sama-sama sequence**
- panjang bisa berbeda
- alignment tidak eksplisit

Contoh:

Input (EN): I love machine learning

Output (DE): Ich liebe maschinelles Lernen

 Dibutuhkan arsitektur baru.

### IDE INTI SEQ2SEQ

Seq2Seq terdiri dari **dua komponen utama**:

1. **Encoder**
2. **Decoder**

## Encoder

- membaca seluruh input sequence
- mengompresnya menjadi **context vector**
- context vector = ringkasan makna input

## Decoder

- menghasilkan output sequence
- satu token demi satu
- menggunakan context vector sebagai "ingatan"

## STRUKTUR BESAR CHAPTER 11

Chapter 11 dibagi menjadi **4 bagian utama**:

1. Memahami data machine translation
2. Membangun seq2seq translator (EN → DE)
3. Training & evaluasi model
4. Inference (menghasilkan terjemahan)

### 11.1 Understanding the Machine Translation Data

#### Karakteristik Data Translation

Setiap sample berisi:

- kalimat bahasa sumber (English)
- kalimat bahasa target (German)

Tantangan:

- vocabulary berbeda
- struktur grammar berbeda
- panjang kalimat berbeda

#### Preprocessing Penting

Menekankan beberapa hal krusial:

- lowercase
- tokenization terpisah untuk EN & DE
- penambahan token khusus:
  - <start>
  - <end>

Token ini membantu decoder:

- tahu kapan mulai
- tahu kapan berhenti

### 11.2 Writing an English–German Seq2Seq Translator

#### 11.2.1 TextVectorization untuk Seq2Seq

Berbeda dari sentiment analysis:

- **dua TextVectorization layer**

- satu untuk EN
- satu untuk DE

Alasan:

- vocabulary berbeda
- mapping integer berbeda

### **11.2.2 Defining the Encoder**

#### **Struktur Encoder**

Encoder biasanya berupa:

- Embedding layer
- RNN (LSTM atau GRU)

Alur:

Input tokens

→ Embedding

→ RNN

→ Final hidden state

Hidden state terakhir = **context vector**.

#### **Insight Penting Encoder**

- Seluruh kalimat dikompresi ke satu vektor
- Ini adalah **bottleneck besar**
- Menjadi kelemahan utama seq2seq klasik

(Ini akan "diperbaiki" di Chapter 12)

### **11.2.3 Defining the Decoder**

#### **Struktur Decoder**

Decoder mirip encoder, tapi:

- input: token sebelumnya
- initial state: context vector dari encoder

Alur:

Previous token

→ Embedding

→ RNN (initialized with encoder state)

→ Dense (softmax)

#### **Teacher Forcing**

##### **Apa itu Teacher Forcing?**

Selama training:

- decoder **tidak memakai prediksi sendiri**
- tapi memakai **ground truth token sebelumnya**

Tujuan:

- training lebih stabil
- konvergensi lebih cepat

⚠ Konsekuensi:

- training  $\neq$  inference
- perlu strategi khusus saat inference

### 11.3 Training and Evaluating the Model

#### Training Objective

Model dilatih untuk:

- memaksimalkan probabilitas kalimat target
- token demi token

Loss:

- categorical cross-entropy

#### Evaluasi Model

Evaluasi dilakukan dengan:

- loss
- kualitas terjemahan (manual inspection)

**Translation quality tidak bisa dinilai hanya dengan loss**

#### Kelemahan Seq2Seq Klasik yang Terlihat

- kalimat panjang  $\rightarrow$  kualitas turun
- informasi awal sering "hilang"
- konteks panjang sulit dipertahankan

➡ Ini mengarah langsung ke Chapter 12 (Attention).

### 11.4 From Training to Inference

#### Kenapa Inference Berbeda?

Saat inference:

- **tidak ada ground truth**
- decoder harus memakai output sendiri

#### Inference Pipeline

1. Encode input sentence
2. Ambil context vector
3. Mulai decoder dengan <start>
4. Prediksi token berikutnya
5. Ulangi sampai <end> atau max length

### Tantangan Inference

- error bisa terakumulasi
- kesalahan awal → fatal

Ini adalah kelemahan mendasar seq2seq tanpa attention.



### Filosofi Besar Chapter 11

1. Seq2Seq memungkinkan NLP generatif kompleks
2. Encoder–decoder adalah ide revolusioner
3. Context vector adalah kekuatan sekaligus kelemahan
4. Bottleneck ini mendorong lahirnya **attention**