Ringkasan Buku 1

Nama Buku: Tensorflow in Actions
Team: Wilhelmina Arlene, Luthfiah Maulidya, Fadhilah Dwi Istiani

Chapter 12 — *Sequence-to-Sequence Learning*

🎯 **Tujuan Utama Chapter 12**
Chapter 12 menjawab **masalah paling fatal** dari seq2seq klasik (Chapter 11):
"Kenapa satu context vector tidak cukup untuk mewakili seluruh kalimat?"
Solusinya:
ATTENTION MECHANISM
Attention memungkinkan model:

- *tidak lagi mengompres seluruh kalimat ke satu vektor*
- **memilih bagian input yang relevan** saat menghasilkan setiap output token

Ini adalah **loncatan konseptual terbesar** sebelum Transformer.

REKAP MASALAH DI CHAPTER 11 (KENAPA ATTENTION DIPERLUKAN)
Kelemahan Seq2Seq Klasik

1. **Information bottleneck**
    - seluruh input → 1 vector
2. **Kalimat panjang → performa turun**
3. **Alignment implicit & lemah**
4. **Makna awal kalimat sering hilang**

Contoh masalah:
Input (EN): The book that you gave me yesterday was very interesting
Output (DE): ????
Decoder sulit tahu:

- kata mana yang relevan saat ini

➡️ **Attention lah jawabannya.**

IDE INTI ATTENTION
Intuisi Sederhana Attention
Saat menerjemahkan kata ke-t:

Decoder "melihat kembali" seluruh output encoder dan memilih bagian mana yang penting saat ini

Artinya:

- decoder **tidak lagi bergantung pada satu context vector**
- context bersifat **dinamis**, berubah tiap timestep

STRUKTUR BESAR CHAPTER 12

Chapter 12 terdiri dari **2 bagian utama**:

1. Menambahkan **Attention Mechanism** ke seq2seq
2. **Visualisasi attention** untuk interpretabilitas

12.1 Improving Seq2Seq with Attention

Encoder Tetap Sama

Encoder:

- masih membaca seluruh input
- tapi sekarang:
  - **menyimpan semua hidden state**
  - bukan hanya state terakhir

Misalnya:

Input: x1 x2 x3 x4

Encoder states: h1 h2 h3 h4

Decoder dengan Attention

Untuk setiap timestep decoding t:

Decoder melakukan:

1. hitung **attention score** antara state decoder dan semua encoder states
2. normalisasi (softmax)
3. buat **context vector baru**
4. gabungkan context + decoder state
5. prediksi token berikutnya

➡️ Context vector sekarang **berubah-ubah**, bukan statis.

12.1.1 Bahdanau Attention (Additive Attention)

Kenapa Bahdanau?

Bahdanau Attention adalah:

- salah satu attention pertama
- sederhana
- sangat intuitif
- cocok untuk pembelajaran konsep

Cara Kerja Bahdanau Attention
Untuk setiap decoder step:

1️⃣ Alignment Score
Mengukur kecocokan:
score(decoder_state, encoder_state_i)
Biasanya menggunakan:

- small feed-forward network

2️⃣ Attention Weights
Semua score:

- dinormalisasi dengan softmax
- menghasilkan bobot:

α1, α2, α3, ..., αn

3️⃣ Context Vector
Context vector dihitung sebagai:
context = Σ (αi × encoder_state_i)
Artinya:

- encoder state yang relevan → bobot besar
- yang tidak relevan → diabaikan

Insight Penting
Attention membuat alignment:

- **eksplisit**
- **dapat dipelajari**
- **dapat divisualisasikan**

Ini revolusioner.

12.1.2 Defining the Final Seq2Seq Model
Arsitektur Lengkap Seq2Seq + Attention
Encoder:
Input → Embedding → RNN → {h1, h2, ..., hn}

Decoder (per timestep):
Previous token
→ Embedding
→ RNN
→ Attention(h_decoder, {h1..hn})
→ Context Vector

→ Dense → Output token

Perubahan Penting dari Chapter 11

Tanpa Attention	Dengan Attention
1 context vector	context dinamis
bottleneck	tidak ada bottleneck
sulit long sentence	lebih stabil
implicit alignment	explicit alignment

12.1.3 Training the Attention Model

Training Strategy

- masih menggunakan **teacher forcing**
- loss tetap categorical cross-entropy
- training sedikit lebih mahal
- tapi performa jauh lebih baik

Dampak Nyata Attention

- kalimat panjang diterjemahkan lebih baik
- grammar lebih konsisten
- makna lebih terjaga

12.2 Visualizing the Attention

Kenapa Visualisasi Penting?

Attention bukan cuma:

- meningkatkan performa
- tapi juga **interpretabilitas**

Dengan attention:

- kita bisa *melihat*:
    - kata input mana memengaruhi output tertentu

Attention Matrix

Biasanya divisualisasikan sebagai:

- heatmap
- sumbu X = input tokens
- sumbu Y = output tokens
- warna = attention weight