

Ringkasan Chapter 1

Nama Buku: Tensorflow in Actions

Team: Wilhelmina Arlene, Luthfiah Maulidya, Fadhilah Dwi Istiani

Chapter 1 — *The Amazing World of TensorFlow*

Tujuan Utama Chapter 1

Chapter ini berfungsi sebagai fondasi konseptual. Penulis tidak langsung masuk ke coding, tetapi membangun pemahaman mental tentang:

- apa itu TensorFlow,
- mengapa TensorFlow penting,
- kapan sebaiknya (dan tidak sebaiknya) digunakan,
- serta bagaimana TensorFlow cocok dalam *machine learning lifecycle* secara end-to-end.

1.1 Apa itu Machine Learning & Peran TensorFlow

Definisi Machine Learning

Machine learning adalah proses melatih model komputasi agar:

- menerima input data
- menghasilkan prediksi/output
- dengan cara belajar dari data, bukan aturan statis.

Tahapan umum ML yang dijelaskan:

1. Exploratory Data Analysis (EDA) – memahami data
2. Data cleaning – membersihkan noise & kesalahan
3. Feature engineering – membentuk fitur yang bermakna
4. Modeling – melatih model
5. Evaluation – menguji performa pada data baru
6. Deployment/UI – membuat model bisa digunakan user

Penekanan penting:

Proses ML tidak linear, tapi iteratif. Bisa bolak-balik antar tahap.

Apa itu TensorFlow?

TensorFlow adalah:

- end-to-end machine learning framework
- dikembangkan oleh Google
- dirancang agar cepat dan efisien di hardware khusus seperti GPU & TPU

TensorFlow tidak hanya untuk training model, tapi mencakup:

- pengolahan data

- training
- monitoring
- debugging
- deployment
- serving model ke user

Dengan kata lain, TensorFlow mengelola seluruh siklus hidup model ML.

1.1.1 Ekosistem TensorFlow

TensorFlow mendukung banyak use case, di antaranya:

- Deep learning
- Probabilistic ML (TensorFlow Probability)
- Computer graphics
- Pretrained models (TensorFlow Hub)
- Visualisasi & debugging (TensorBoard)

Penulis menekankan bahwa kekuatan TensorFlow bukan hanya model, tapi ekosistem terpadu.

1.1.2 Komponen Utama TensorFlow (Big Picture)

Alur kerja TensorFlow yang dijelaskan:

1 Data Pipeline

Karena data deep learning sangat besar dan tidak muat di memori:

- TensorFlow menggunakan batch processing
- Data dialirkan secara bertahap

Tools utama:

- tf.data API → membuat pipeline data efisien & scalable
- tensorflow-datasets → dataset siap pakai
- Keras Data Generators → khusus image, text, dll

2 Model Building

Empat tipe utama deep learning model:

- Fully Connected Networks
- Convolutional Neural Networks (CNN)
- Recurrent Neural Networks (RNN)
- Transformers

Pendekatan API:

- Low-level TensorFlow → fleksibel tapi kompleks
- Keras (high-level API) → lebih ringkas & aman

Keras menyediakan:

- layer siap pakai
- pretrained models

- beberapa gaya pembuatan model:
 - Sequential
 - Functional
 - Subclassing

3 Training & Monitoring

Training model deep learning:

- mahal secara komputasi
- butuh monitoring terus-menerus

Solusi:

- TensorBoard
 - visualisasi loss & metric
 - deteksi overfitting
 - profiling performa

4 Saving & Deployment

Model harus disimpan agar:

- tidak hilang
- bisa dilanjutkan training
- bisa digunakan ulang

Format penyimpanan:

- HDF5
- SavedModel (standar TensorFlow)

Deployment:

- TensorFlow Serving
- menyediakan API untuk inference di production

1.2 GPU vs CPU vs TPU

Analogi Transportasi

- CPU = mobil
 - cepat untuk tugas kompleks
 - sedikit core
 - low latency
- GPU = bus
 - banyak core
 - cocok untuk komputasi paralel
 - high throughput
- TPU = bus khusus
 - lebih terbatas
 - sangat efisien untuk ML

Kenapa GPU/TPU penting?

Deep learning didominasi oleh:

- matrix multiplication
- operasi numerik besar & paralel

GPU & TPU:

- jauh lebih cepat dari CPU
- memungkinkan training model besar

TPU:

- dikembangkan Google
- menggunakan bfloat16
- sangat efisien, tapi kurang fleksibel

1.3 Kapan Menggunakan TensorFlow

✅ Kapan TensorFlow Tepat Digunakan

1. Prototyping deep learning models
 - CNN, RNN, LSTM, GRU, Transformer
2. Training di hardware teroptimasi
 - GPU / TPU
3. Production & deployment
 - tidak perlu pindah framework
4. Monitoring training
 - TensorBoard
5. Data pipeline skala besar
 - streaming data
 - preprocessing kompleks

❌ Kapan Tidak Disarankan

1. Traditional ML
 - linear regression, SVM, decision tree
 - lebih efisien pakai scikit-learn
2. Data kecil & structured
 - pandas & NumPy lebih cepat
3. NLP preprocessing kompleks
 - stemming, lemmatization → spaCy lebih cocok

Penulis menekankan:

TensorFlow bukan "silver bullet".

1.4 Apa yang Akan Dipelajari dari Buku Ini

1 TensorFlow Fundamentals

- execution model

- tf.Variable, tf.Tensor, tf.Operation
- Keras APIs
- data ingestion
- 2** Deep Learning Algorithms
 - FCN, CNN, RNN
 - aplikasinya di:
 - image classification
 - segmentation
 - sentiment analysis
 - machine translation
 - Transformer sebagai model modern

3 Monitoring & Optimization

- TensorBoard
- model explainability
- performance optimization
- training acceleration

1.5 Target Pembaca

Buku ini cocok untuk:

- ML practitioner pemula–menengah
- data scientist / ML engineer
- mahasiswa ML

Prasyarat:

- Python (OOP dasar)
- NumPy / pandas
- linear algebra dasar
- konsep DL secara umum

1.6 Kenapa Python & TensorFlow 2

Kenapa Python?

- ekosistem data science sangat kaya
- mudah eksperimen & visualisasi
- dominan di ML community

Kenapa TensorFlow?

- stabil & mature
- ekosistem lengkap
- performa unggul di data besar
- skalabilitas tinggi

Perbandingan menunjukkan:

- NumPy → cepat di data kecil
- TensorFlow → jauh lebih efisien saat data besar

🔑 Inti Chapter 1 (Kesimpulan)

- TensorFlow adalah framework ML end-to-end
- Sangat kuat untuk deep learning skala besar
- Tidak cocok untuk semua kasus
- Buku ini fokus pada menulis solusi TensorFlow yang efektif, bukan sekadar “berjalan”