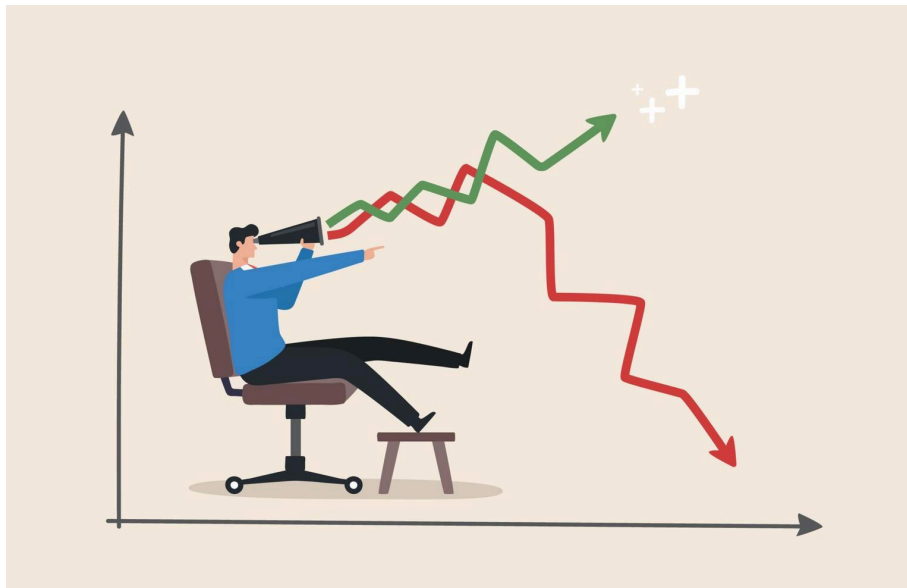


Praktikum 3 IF3270 Pembelajaran Mesin

Recurrent Neural Network

Dipersiapkan Oleh Tim Asisten IF3270 2024/2025

Versi: 1.0 07/05/2025



Deadline 1: Kamis, 8 Mei 2025 11.00 WIB

Deadline 2: Kamis, 8 Mei 2025 23:00 WIB

Tujuan

Praktikum pada kuliah IF3270 Pembelajaran Mesin bertujuan untuk memberikan pengalaman langsung kepada peserta kuliah dalam menerapkan Recurrent Neural Network.

Spesifikasi

Permintaan energi listrik merupakan komponen krusial dalam pengelolaan sistem tenaga listrik modern. Hal ini penting karena penyediaan energi listrik harus tepat dengan permintaan tenaga listrik. Supply yang berlebihan akan menyebabkan terbuangnya energi dan supply yang kekurangan akan menyebabkan masalah.

Data beban listrik biasanya ditangkap sebagai data deret waktu (time series) karena nilainya bergantung pada waktu dan memiliki pola yang berulang, baik secara harian, mingguan, maupun musiman. Dalam praktik industri, data ini digunakan untuk memperkirakan kebutuhan listrik pada masa mendatang, mengantisipasi lonjakan konsumsi, dan merancang sistem kelistrikan yang stabil serta efisien.

Dataset yang disediakan ini dikategorikan sebagai data deret waktu (time series) dan akan digunakan sebagai dasar untuk membangun sistem prediksi beban listrik otomatis (**h1**) berbasis deep learning. Pada praktikum ini, Anda akan diminta untuk membangun model prediksi harga saham menggunakan Recurrent Neural Network (RNN) dengan arsitektur Long Short-Term Memory (LSTM) yang dapat mempelajari pola pergerakan harga dari waktu ke waktu.

Pada praktikum ini, Anda akan diminta untuk membangun dan melatih model prediksi yang dapat memperkirakan **h1** pada hari berikutnya berdasarkan data **h1** dari beberapa hari sebelumnya. Berikut adalah deskripsi dari setiap fitur pada dataset ini:

1. **Date**
2. **h1**

Berikut merupakan beberapa hal yang harus dilakukan oleh praktikan:

EDA (Exploratory Data Analysis)

Exploratory Data Analysis (EDA) adalah langkah krusial dalam proses analisis data yang melibatkan pemeriksaan dan visualisasi dataset untuk mengungkap pola, tren, anomali, dan *insight*. Ini merupakan langkah awal sebelum menerapkan teknik statistik dan machine learning yang lebih lanjut. EDA membantu Anda memahami data secara mendalam, memungkinkan Anda membuat keputusan yang lebih terarah dan dapat merumuskan hipotesis untuk analisis lebih lanjut.

Pada tahap ini, Anda hanya diminta untuk melakukan **pemeriksaan awal sederhana** terhadap dataset. Fokuskan pada pemahaman struktur data dan ringkasan statistik dasar untuk mendapatkan gambaran umum mengenai dataset yang digunakan. Gunakan metode seperti `.head()` dan `.describe()` untuk membantu memperoleh gambaran umum mengenai dataset yang digunakan.

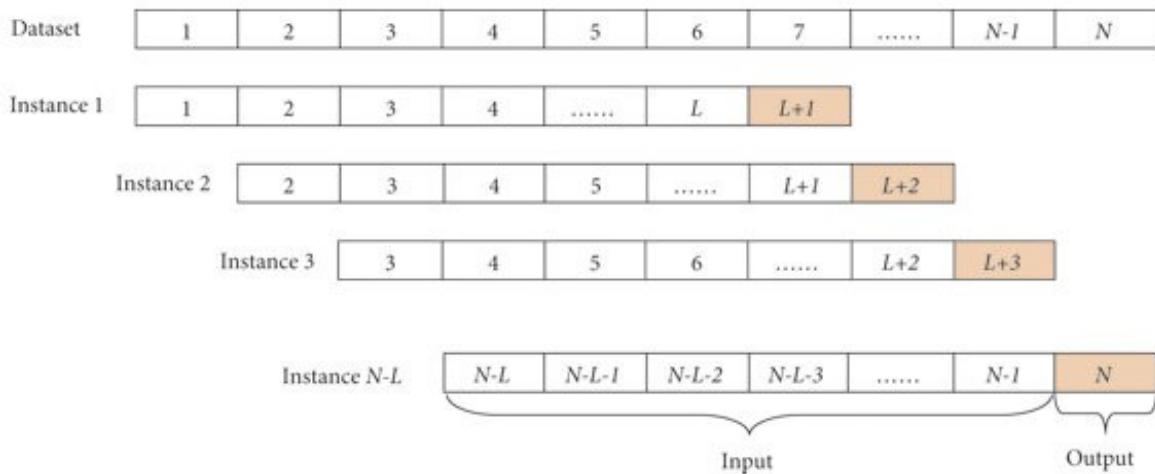
Data Cleaning and Preprocessing

Langkah ini adalah hal pertama yang dilakukan setelah seorang Data Scientist memiliki pemahaman umum tentang data. Data mentah jarang siap untuk training, sehingga perlu dilakukan langkah-langkah untuk membersihkan dan memformat data agar dapat diinterpretasikan oleh model machine learning.

Dengan melakukan data cleaning dan preprocessing, Anda memastikan bahwa dataset siap untuk training model, yang akan menghasilkan hasil machine learning yang lebih akurat dan andal. Langkah-langkah ini sangat penting untuk mentransformasi data mentah menjadi format yang dapat dipelajari secara efektif oleh algoritma machine learning dan digunakan untuk membuat prediksi.

Pada tahap ini, Anda diminta untuk melakukan proses-proses yang sekiranya tepat digunakan untuk dataset yang diberikan. Sertakan penjelasan mengapa Anda memilih untuk melakukan proses tersebut. Preprocessing yang diperlukan adalah **Scaling** dan membuat **Sliding Window**.

Sliding Window:



Gambar 1. Ilustrasi Sliding Window

Dapat dilihat di Gambar 1, Sebuah dataset dipecah menjadi panjang L ,

- Dimana window 1 sampai L digunakan untuk prediksi $L+1$
- Dimana window 2 sampai $L+1$ digunakan untuk prediksi $L+2$
- Dimana window 3 sampai $L+2$ digunakan untuk prediksi $L+3$
- Dst sampai prediksi N

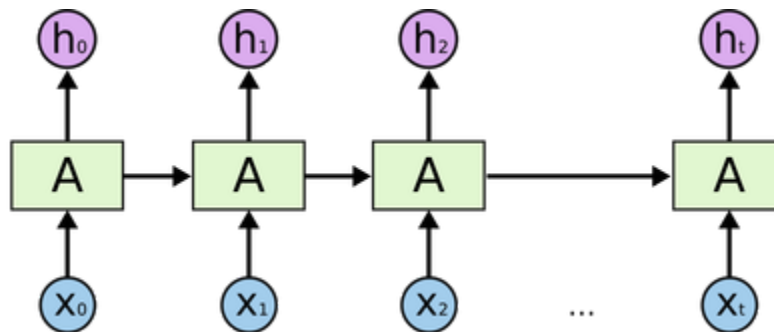
Berikut adalah pseudocode untuk implementasi sliding window:

```
- If input_columns is a string, convert to a list.
- Initialize empty list X and (if target_column provided) y.
- Loop from i = 0 to (number of rows in df - time_steps):
    - Extract a sequence from the input_columns for rows i to i+time_steps.
    - Append this sequence to X.
    - If target_column is provided,
        append the value from row (i+time_steps) in target_column to y.
// The Result of X is a 2D array with such as
[[X1,X2,X3], [X2,X3,X4], [X3,X4,X5], ... [XN-2, XN-1,XN]]
- Convert X (and y) to numpy arrays and return.
```

Modeling and Validation

Modeling adalah proses membangun model machine learning untuk menyelesaikan masalah tertentu, atau dalam konteks tugas ini, memprediksi (regression) nilai beban listrik dengan dua model: RNN dan LSTM.

Recurrent Neural Network (RNN) adalah sebuah arsitektur Neural Network yang dapat melakukan pemrosesan atas data sekuensial (berurutan, posisi penting). RNN dapat melakukan ini karena ia menerima input dan “hasil dari pemrosesan sequence sebelumnya”.



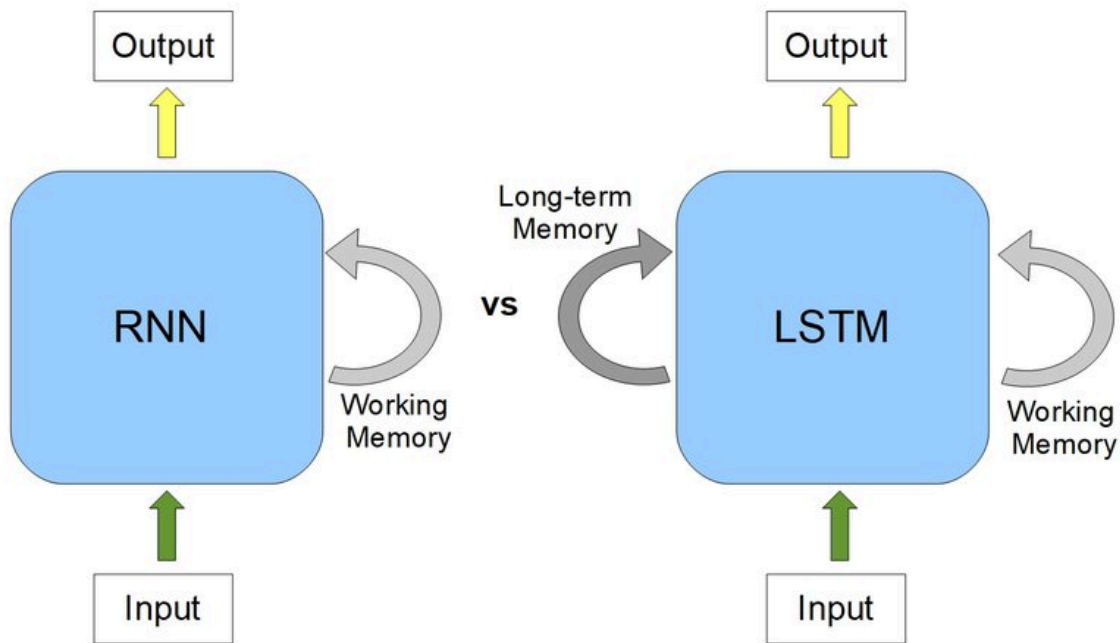
Gambar 2. Ilustrasi Pemrosesan RNN

Dapat dilihat di ilustrasi:

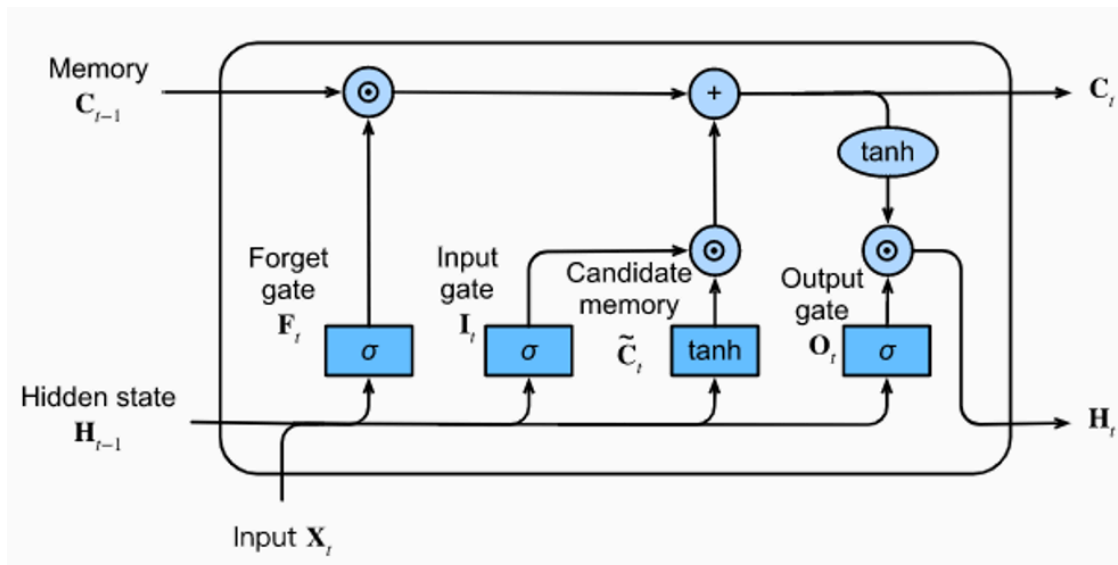
1. h_0 dihasilkan dari X_0
2. h_1 dihasilkan dari X_1 dan hasil X_0
3. h_2 dihasilkan dari X_2 dan hasil X_1
4. dst.

Arsitektur tersebut memungkinkan RNN untuk mengolah data sekuensial. Namun, sebagai arsitektur yang cukup dasar, RNN memiliki kelemahan utama berupa vanishing gradient yang menyebabkan kesulitan dalam mengingat informasi dari sequence yang jauh ke belakang (short-term memory).

Long Short Term Memory (LSTM) adalah arsitektur turunan RNN yang lebih kompleks, dirancang untuk mengatasi masalah vanishing gradient dengan menggunakan mekanisme gates yang mengatur informasi mana yang disimpan, dibuang, dan diproses dalam long-term memory (cell state).



Gambar 3. Perbandingan RNN dan LSTM



Gambar 4. Arsitektur LSTM (wawasan)

Praktikan diperlukan untuk mengimplementasikan dan membandingkan dua model:

1. Menyusun model RNN dan LSTM menggunakan PyTorch/Tensorflow
2. Menggunakan dataset beban listrik, melakukan training dan melakukan prediction beban listrik melalui regression.
3. Melakukan Evaluasi dan perbandingan atas kinerja kedua model tersebut.

Validation adalah proses mengevaluasi model yang telah dilatih menggunakan validation set atau metode cross-validation dan memberikan metrik yang dapat membantu Anda menentukan langkah yang perlu dilakukan pada iterasi pengembangan berikutnya.

Untuk validasi, metrik yang digunakan adalah **RMSE**. Hasil validasi yang harus tercantum di notebook adalah **hasil dari pemodelan yang wajib diimplementasikan** dan **hasil dari pemodelan submisi final di kaggle**.

Catatan: Disarankan untuk menggunakan resource GPU yang disediakan oleh Google Colab ataupun Kaggle untuk melatih model.

Error Analysis

Error analysis adalah proses penting dalam evaluasi model machine learning yang bertujuan untuk memahami sejauh mana dan di mana model melakukan kesalahan dalam melakukan prediksi. Dengan menganalisis kesalahan ini, kita dapat mengidentifikasi pola-pola tertentu, seperti data historis pada periode tertentu yang sering menghasilkan prediksi yang meleset, atau kondisi pasar yang tidak berhasil ditangkap oleh model.


Pada tahap ini, **bandingkan hasil prediksi dengan nilai aktual** dan sajikan dalam bentuk visualisasi, seperti grafik garis yang menampilkan kurva prediksi dan realisasi dalam satu sumbu waktu. Visualisasi ini membantu mengamati pola kesalahan, deviasi prediksi, serta potensi bias secara lebih intuitif, dan dapat menjadi dasar untuk perbaikan model kedepannya.

Insights

Untuk setiap tahap yang dilakukan, sebutkan insight yang didapatkan oleh praktikan selama pengerjaan praktikum. Insight yang dimaksud adalah analisis dan juga argumen pendukung terhadap setiap tahap yang dilakukan oleh praktikan.

Submisi Kaggle

Pada tugas ini, Anda diminta untuk mengikuti [Kaggle Competition](#) berikut dan mengumpulkan hasil prediksi Anda ke kompetisi tersebut. Dataset yang akan digunakan tersedia pada tautan Kaggle Competition yang diberikan. Skor yang Anda raih di kompetisi tersebut akan masuk ke dalam penilaian. Skor yang diambil untuk penilaian adalah skor yang diraih di private leaderboard. Submisi ke Kaggle dibatasi sebanyak **20 kali**. Submisi final yang dikumpulkan ke kaggle harus **reproducible** (dapat dihasilkan hasil yang sama dengan notebook yang dikumpulkan). Submisi yang **tidak reproducible** akan dikenakan **penalti berupa pengurangan nilai**. Format penamaan kelompok di Kaggle adalah sebagai berikut: **Kelas_NomorKelompok_NamaKelompok** (Contoh: K1_99_gAIB21, nama kelompok opsional)

Asisten telah menyiapkan *notebook template* untuk Anda gunakan dengan tautan berikut  **Kit Praktikum** .

Kelompok

Pembagian kelompok ditentukan sendiri oleh mahasiswa dengan mengisi [sheets kelompok](#) berikut ini dengan 1 kelompok terdiri dari dengan **maksimal anggota sebanyak 2 orang**.

QnA

Pertanyaan dapat ditanyakan pada [link QnA](#) berikut. Pastikan pertanyaan yang ditanyakan tidak berulang.

Aturan

Terdapat beberapa hal yang harus diperhatikan dalam pengerjaan tugas ini, yakni:

1. Jika terdapat hal yang tidak dimengerti, silahkan ajukan pertanyaan kepada asisten melalui **link QnA** yang telah diberikan di atas. Pertanyaan yang diajukan secara personal ke asisten **tidak akan dijawab** untuk menghindari perbedaan informasi yang didapatkan oleh peserta kuliah.
2. Dilarang melakukan **plagiarisme, menggunakan AI dalam bentuk apapun secara tidak bertanggungjawab, dan melakukan kerjasama antar kelompok**. Pelanggaran pada poin ini akan menyebabkan pemberian **nilai E** pada setiap anggota kelompok.
3. Tidak ada batasan untuk *library* yang boleh digunakan.

4. Dilarang untuk menggunakan **sample_submission.csv** sebagai **submisi utama**. Submisi utama harus menggunakan hasil prediksi dengan model sendiri. Penggunaan SampleSubmission.csv sebagai submisi final akan diberikan penalti berupa **pemberian nilai 0 untuk praktikum**.

Deliverables

- **Deadline 1 (Kamis, 8 Mei 2025 11.00 WIB)**
 - Link ke notebook pengerjaan dengan format penamaan file NIM1_NIM2_Kelas_Deadline1.ipynb (Contoh: 13521998_13521999_K1_Deadline1.ipynb)
 - Minimal mengumpulkan **1 kali submisi** ke kaggle, dibuktikan dengan *screenshot* leaderboard yang menunjukkan kelompok Anda sudah tercantum pada leaderboard.
- **Deadline 2 (Kamis, 8 Mei 23:00 WIB)**
 - Link ke notebook pengerjaan dengan format penamaan file NIM1_NIM2_Kelas_Deadline2.ipynb (Contoh: 13521998_13521999_K1_Deadline2.ipynb). Jika terdapat revisi atau tambahan dari notebook pertama, silakan tambah markdown cell di notebook pengerjaan yang berisi daftar perubahan yang dilakukan.
 - Notebook yang dikumpulkan harus sudah lengkap untuk seluruh bagian beserta dengan analisisnya:
 - EDA
 - Data Preprocessing
 - Modeling and Validation
 - Error Analysis
 - Insights setiap tahap
- Notebook pengerjaan yang dikumpulkan harus memiliki akses **Editor** untuk **Anyone with the link**.
- Pastikan notebook yang dikumpulkan untuk Deadline 1 dan Deadline 2 merupakan **notebook yang berbeda**.
- Pengumpulan dilakukan melalui form dengan tautan sebagai [berikut](#).
- Tugas yang terlambat dikumpulkan tidak akan diterima.
- Pengumpulan dilakukan oleh NIM terkecil.

Referensi

1. https://www.tensorflow.org/api_docs/python/tf/keras/layers/RNN
2. <https://www.geeksforgeeks.org/training-of-recurrent-neural-networks-rnn-in-tensorflow/>
3. https://www.tensorflow.org/api_docs/python/tf/keras/layers/LSTM
4. <https://www.geeksforgeeks.org/long-short-term-memory-lstm-rnn-in-tensorflow/>