

Bab 16: Natural Language Processing with RNNs and Attention

Tujuan Bab

Membahas bagaimana model deep learning, khususnya RNN, digunakan dalam Natural Language Processing (NLP). Juga memperkenalkan konsep penting seperti word embeddings, encoder-decoder, dan attention mechanism.

Konsep Utama

1. Representasi Teks

Teks (kata, kalimat, dokumen) harus diubah menjadi representasi numerik sebelum bisa diproses oleh model.

- One-hot encoding: Tidak efisien dan sangat sparse
- Word embeddings: Representasi dens berdimensi rendah

Contoh penggunaan Embedding di Keras:

```
keras.layers.Embedding(input_dim=10000, output_dim=128)
```

2. Sequence Classification

Task seperti klasifikasi sentimen bisa dikerjakan dengan RNN (LSTM/GRU).

Contoh:

```
model = keras.models.Sequential([  
    keras.layers.Embedding(vocab_size, 128),  
    keras.layers.LSTM(64),  
    keras.layers.Dense(1, activation="sigmoid")  
])
```

3. Padding dan Masking

Input teks memiliki panjang berbeda, sehingga perlu disamakan (padding), lalu ditandai agar tidak dihitung oleh model (masking).

```
keras.layers.Embedding(..., mask_zero=True)
```

4. Sequence-to-Sequence (Seq2Seq)

Digunakan untuk tugas seperti penerjemahan mesin.

- Encoder: Membaca input dan menghasilkan konteks
- Decoder: Menghasilkan output sekuens dari konteks

Contoh tugas: terjemahan dari bahasa Inggris ke Prancis.

5. Attention Mechanism

Mengatasi kelemahan encoder-decoder tradisional dengan memungkinkan decoder fokus ke bagian input tertentu saat menghasilkan setiap token output.

Konsep utamanya:

- Alih-alih satu vektor konteks, attention memungkinkan pembobotan dinamis seluruh vektor encoder.

6. Implementasi Attention (Keras)

Pendekatan dasar:

- Gunakan keras.layers.Attention atau custom layer
- Gabungkan hasil weighted sum dari encoder output berdasarkan skor perhatian

7. Word Embedding Pra-latih

Dapat menggunakan embedding seperti GloVe, Word2Vec, dll., yang sudah dilatih dari korpus besar.

Embedding dapat di-*load* dan di-*freeze* (tidak dilatih ulang), atau di-*fine-tune*.

8. Transfer Learning di NLP

Model seperti BERT, GPT, dll., merupakan pre-trained language model yang bisa digunakan kembali untuk banyak tugas NLP.

Proyek / Notebook Praktik

Isi:

- Preprocessing teks
- Membangun model LSTM untuk klasifikasi sentimen
- Membangun model encoder–decoder untuk penerjemahan
- Implementasi mekanisme perhatian (attention)
- Visualisasi skor attention

Inti Pelajaran

Konsep	Penjelasan
Embedding	Representasi dens dari kata
RNN / LSTM / GRU	Dasar pemrosesan sekuens
Seq2Seq	Encoder-decoder untuk mapping sekuens ke sekuens
Attention	Fokus dinamis pada bagian input saat decoding
Pretrained Embedding	Transfer pengetahuan dari corpus besar
NLP Task	Klasifikasi, terjemahan, ekstraksi, dll.

Kesimpulan

Bab ini menjelaskan penerapan model RNN dan perhatian (attention) untuk tugas-tugas NLP, termasuk klasifikasi teks dan penerjemahan bahasa. Attention memungkinkan model menghasilkan output yang lebih akurat dengan memfokuskan perhatian ke bagian input yang relevan.