Desain Proyek: Penerapan Model Promoter pada Sequence DNA Bakteri *E. coli*

K1 - Kelompok 6
Brian Kheng - 13521049
Denise Felicia Tiowanni - 13522013
Erdianti Wiga Putri Andini - 13522053

1. Latar Belakang dan Formulasi Masalah

1.1. Latar Belakang Biologis

Promoter adalah urutan DNA penting yang terletak di bagian hulu (*upstream*) gen yang berfungsi sebagai tempat pengikatan RNA polimerase dan faktor transkripsi. Promoter memainkan peran fundamental dalam regulasi ekspresi gen dengan mengontrol kapan dan seberapa banyak suatu gen ditranskripsikan. Pada prokariota seperti *E. coli*, promoter umumnya mengandung elemen-elemen konservatif, termasuk daerah -35 dan -10 (dinamai berdasarkan posisinya relatif terhadap tempat awal transkripsi), yang dikenali oleh faktor sigma dari RNA polimerase.

Pemahaman dan identifikasi promoter sangat penting karena menentukan gen mana yang diekspresikan dan kapan diekspresikan, mempengaruhi kemampuan adaptasi bakteri terhadap perubahan lingkungan, serta dapat menjadi target pengembangan antimikroba. Selain itu, informasi ini juga esensial untuk berbagai aplikasi dalam bidang biologi sintetik.

1.2. Latar Belakang Komputasi

Metode laboratorium tradisional untuk identifikasi promoter memakan waktu dan biaya mahal. Selain itu, dalam bidang bioinformatika, analisis sekuens DNA untuk mengidentifikasi elemen regulatori seperti promoter juga merupakan suatu tantangan besar karena kompleksitas data biologis. Pendekatan tradisional berbasis motif atau aturan juga tidak selalu dapat mendeteksi promoter secara akurat, terutama jika motif tidak konservatif atau terlalu bervariasi. Oleh karena itu, pendekatan machine learning, khususnya deep learning, mulai digunakan untuk menangani permasalahan ini. Model seperti DNABERT, varian dari model transformer BERT yang dilatih khusus untuk sekuens DNA, telah menunjukkan performa tinggi dalam berbagai tugas klasifikasi biologis.

1.3. Rumusan Masalah

"Bagaimana merancang dan membangun model machine learning berbasis arsitektur transformer (DNABERT) untuk memprediksi apakah suatu sekuens DNA pada *E. coli* merupakan promoter atau bukan, dengan mempertimbangkan akurasi dan kemampuan generalisasi model?"

1.4. Tujuan Proyek

1. Mengembangkan model klasifikasi DNA (DNABERT) untuk prediksi promoter.
2. Mengevaluasi performa model menggunakan metrik klasifikasi seperti akurasi, precision, recall, dan F1-score.
3. Menganalisis potensi penggunaan model dalam bidang bioinformatika dan terapi berbasis genetik.

2. Metode atau Algoritma

Pendekatan yang digunakan adalah machine learning, lebih spesifik lagi *deep learning* berbasis transformer. Algoritma utama adalah DNABERT, yakni varian dari BERT yang telah di-*pre-trained* menggunakan sekuens DNA (k-mer encoding).

2.1. Langkah-langkah Utama

1. *Preprocressing* → DNA sequence dikonversi menjadi k-mer (misal 6-mer).
2. Fine-tuning Model DNABERT → Model akan dilatih ulang (*fine-tuned*) dengan data promoter dari *E. coli*.
3. Prediksi → Model memetakan sekuens DNA menjadi label biner (promoter (1) / non-promoter (0)).
4. Evaluasi → Hasil prediksi dievaluasi menggunakan metrik yang relevan.

2.2. Alasan Pemilihan DNABERT

1. Sudah dilatih khusus untuk data sekuens DNA.
2. Telah terbukti efektif dalam berbagai tugas klasifikasi biologis.
3. Kemampuan untuk menangkap konteks sekuens secara lebih mendalam dibandingkan model konvensional.

3. Dataset

3.1. Sumber Data

Dataset yang digunakan berasal dari platform Kaggle dengan judul "Promoter Gene Sequences". Tautan dataset tersebut dapat diakses pada: https://www.kaggle.com/datasets/rodolfomendes/promoter-gene-sequences.

3.2. Deskripsi Dataset

Dataset terdiri dari 106 sekuens DNA (53 promoter, 53 non-promoter), masing-masing sepanjang 57–60 bp, disertai informasi arah baca (strand) serta nama atau ID. Ukurannya yang kecil cocok untuk uji awal dan fine-tuning model pralatih. Namun, karena itu pula, data menjadi *overfitting*, sehingga diperlukan strategi seperti ***5-fold cross-validation*** dan ***data augmentation (reverse complement, minor mutation).***

3.3. Contoh Struktur Data pada Dataset

Dataset terdiri dari tiga kolom utama:

Kolom	**Deskripsi**
Strand	Tanda orientasi strand DNA: '+' (sense) atau '-' (antisense)
Promoter_Name	Nama atau ID promotor, misalnya S10, DEOP2, LEU1_TRNA, dll
Sequence	Urutan DNA promoter, berupa string nukleotida (A, T, C, G), sepanjang ±57 nukleotida, merepresentasikan daerah upstream gen (wilayah promoter atau non-promoter)

3.4. Distribusi Kelas

Berikut adalah distribusi kelas pada dataset:

Label	Jumlah	Keterangan
Promoter (1)	53	Mengandung elemen regulasi seperti -10 dan -35 box
Non-Promoter (0)	53	Wilayah DNA acak yang bukan promoter

3.5. Ciri-Ciri Biologis Dataset

- Promoter sequences mengandung elemen khas seperti:
  - -10 box: motif TATAAT atau varian mirip
  - -35 box: motif TTGACA atau serupa
  - Spasi antar kotak biasanya 16–18 bp
- Wilayah non-promoter dipilih dari bagian genom yang diketahui tidak aktif secara transkripsi.
- Sebagian sekuens berada di strand negatif (-), mengindikasikan bahwa promotor berada pada rantai komplementer.

3.6. Ukuran & Preprocessing

Berikut adalah ukuran dan format dataset:

Aspek	Keterangan
Total baris	106 baris (53 positif + 53 negatif)
Panjang Sekuens	Rata-rata 57–60 nukleotida
Format file	CSV (Comma-Separated Values)
Format encoding DNA	Huruf kapital (A, T, C, G) tanpa simbol lain

Sebelum digunakan dalam pelatihan model, data akan mengalami langkah-langkah preprocessing sebagai berikut:

1. **Standarisasi urutan**: Pastikan semua huruf kapital dan panjang disesuaikan.
2. **Tokenisasi k-mer**: Setiap sekuens akan dikonversi menjadi unit k-mer (contoh: 6-mer) agar sesuai dengan input DNABERT.
3. **Penyandian label**: Promoter = 1, Non-promoter = 0
4. **Strand handling**: Jika diperlukan, sekuens strand negatif akan dikonversi ke reverse complement untuk menjaga konteks biologi

Dengan demikian, akan dihasilkan format seperti berikut:

Sequence	Label
tactagcaatacgcttgcgttcggtggttaagtatgtataatgcgcgggcttgtcgt	1
caatggcctctaaacgggtcttgaggggttttttgctgaaaggaggaactatatgcg	0

4. Evaluation & Testing

4.1. Strategi Pengujian Sistem

Model prediksi promoter akan divalidasi dan dievaluasi untuk mengukur performa dan generalisasinya, melalui strategi sebagai berikut:

1. ***Splitting Dataset***

   Dataset akan dibagi menjadi dua bagian utama:

   - *Training set* (80%): Digunakan untuk melatih model.
   - *Test set* (20%): Digunakan untuk mengevaluasi performa akhir model.

2. ***Cross-Validation (K-Fold)***

   Untuk menghindari *overfitting* dan mengukur stabilitas model terhadap variasi data, digunakan teknik *K-Fold Cross Validation* (K=5):

   - Dataset dibagi menjadi 5 subset (*folds*).
   - Model dilatih pada 4 subset dan diuji pada 1 subset, proses diulang sebanyak 5 kali.
   - Rata-rata skor dari seluruh *fold* digunakan sebagai evaluasi final.
   - Cocok untuk dataset kecil seperti dalam proyek ini.

4.2. Metrik Evaluasi

Evaluasi performa model dilakukan menggunakan metrik-metrik berikut:

Metrik	**Definisi**
Akurasi	Persentase total prediksi yang benar dari seluruh data (TP + TN) / (TP + TN + FP + FN)
Precision	Seberapa banyak dari hasil prediksi promoter yang benar-benar promoter → TP / (TP + FP)
Recall	Seberapa banyak promoter yang berhasil dideteksi dari semua promoter yang ada → TP / (TP + FN)
F1-Score	Rata-rata harmonik dari precision dan recall → 2 * (Precision * Recall) / (Precision + Recall)
Confusion Matrix	Matriks yang menunjukkan jumlah TP, TN, FP, FN secara eksplisit. Berguna untuk interpretasi visual model.

4.3. Rencana Pengujian

Berikut adalah langkah-langkah rencana pengujian:

1. ***Preprocessing* data** → Menstandarisasi panjang sekuens, mengonversi ke format k-mer, encoding label.
2. **Pembagian data (*Train-Test Split*)** → Memisahkan data untuk pelatihan dan pengujian akhir.
3. ***Fine-tuning model* DNABERT** → Melatih model menggunakan dataset training yang telah diproses.
4. ***K-Fold Cross Validation* (K=5)** → Menilai kestabilan dan kemampuan generalisasi model.

5. **Evaluasi performa menggunakan metrik** → Menghitung nilai akurasi, *precision*, *recall*, F1-*score*.
6. **Analisis *error*** → Melihat sekuens mana dan alasan sering salah diklasifikasikan.
7. **Interpretasi biologis hasil** → Bandingkan hasil prediksi dengan elemen motif promoter biologis nyata (TATAAT, TTGACA, dll).

4.4. Verifikasi Biologis

Untuk memastikan bahwa model tidak hanya melakukan klasifikasi statistik, tetapi juga menangkap ***insight* biologis**, dilakukan pendekatan berikut:

1. ***Motive Analysis***
   - Menganalisis sekuens yang diklasifikasikan sebagai promoter oleh model.
   - Dicocokkan dengan motif-motif khas promotor bakteri seperti:
     - -10 box: TATAAT atau variannya
     - -35 box: TTGACA atau variannya
   - Melihat apakah model cenderung mendeteksi sekuens dengan pola-pola regulasi ini.
2. ***Comparison with Known Regulatory Sites***
   Untuk sekuens dengan label positif (misalnya TRP, LEXA, RPOB), dilakukan perbandingan terhadap anotasi promoter dari terverifikasi database **RegulonDB** atau **EcoCyc** untuk melihat apakah hasil prediksi sejalan dengan posisi promoter nyata secara biologis.
3. ***False Positive / Negative Inspection***
   Jika model memberikan false positive atau false negative, sekuens diperiksa secara manual dan menggunakan software motif finder (seperti **MEME Suite**) untuk melihat apakah kegagalan model dapat dijustifikasi secara biologis.

4.5. Analisis Hasil yang Diharapkan

Model yang baik diharapkan menghasilkan:

- Akurasi: > 85% meskipun pada dataset kecil.
- F1-Score: > 0.85 dengan keseimbangan baik antara precision dan recall.
- AUC-ROC: > 0.90 menunjukkan kemampuan diskriminatif model yang tinggi.

Indikator keberhasilan biologis:

- Kesesuaian motif: > 80% sekuens yang diprediksi sebagai promoter mengandung motif -10 dan -35 yang teridentifikasi.
- Spesifisitas sekuens: Model mampu membedakan promoter dengan sekuens non-koding lain yang memiliki komposisi basa serupa.
- Sensitivitas terhadap variasi: Model dapat mengenali variasi alami dari motif promoter yang masih berfungsi secara biologis.

Jika hasil menunjukkan bias, langkah-langkah perbaikan seperti:

- Augmentasi data (dengan reverse complement)
- Penambahan data baru dari database publik
- Tuning hyperparameter model

akan dilakukan sebagai bagian dari iterasi model.