PENERAPAN *MULTITASK LEARNING* DAN BERT UNTUK MEMPREDIKSI *SPLICE SITES* PADA SEKUENS DNA EUKARIOT

PROPOSAL TESIS

Karya tulis sebagai salah satu syarat
kelulusan MK IF5099 Metodologi Penelitian/Tesis 1

Oleh
MUHAMMAD ANWARI LEKSONO
NIM: 23520050
(Program Studi Magister Informatika)

INSTITUT TEKNOLOGI BANDUNG

12 2021

PENERAPAN *MULTITASK LEARNING* DAN BERT UNTUK MEMPREDIKSI *SPLICE SITES* PADA SEKUENS DNA EUKARIOT

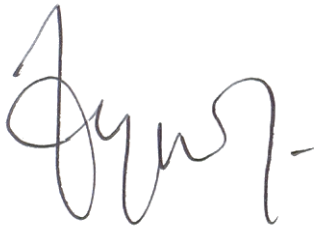
Oleh
MUHAMMAD ANWARI LEKSONO
NIM: 23520050
(Program Studi Magister Informatika)

Institut Teknologi Bandung

Menyetujui
Calon Tim Pembimbing

Tanggal 20 Desember 2021

Calon Dosen Pembimbing



(Dr. Eng. Ayu Purwarianti, S. T., M. T.)

DAFTAR ISI

DAFTAR SINGKATAN DAN LAMBANG

Singkatan	Nama
A	*Adenine*
C	*Cytosine*
G	*Guanine*
T	*Thymine*
BERT	*Bidirectional Encoder Representations from Transformers*
SS	*Splice sites*
TSS	*Transcription Start Site*
TBS	*Transcription Binding Site*
PAS	*Polyadenylation Signal*
DNA	*Deoxyribonucleic acid*
RNA	*Ribonucleic acid*
MTL	*Multitask Learning*
TL	*Transfer Learning*
FC	*Fully connected* (*layer*)

DAFTAR ISTILAH

Istilah	Arti
k-mer	Bagian dari sekuens dengan panjang *k*.
gen	Unit genetik yang berada pada lokasi tertentu pada kromosom tertentu.
genom	Kumpulan informasi genetik yang lengkap.
basa nitrogen	Molekul organik yang bersifat basa dan memiliki atom nitrogen
DNA/RNA sequencing	Proses membaca sekuens DNA/RNA dari suatu sampel organisme
eukariot	Jenis dari sel yang memiliki area inti sel yang batas yang jelas
prokariot	Jenis dari sel yang tidak memiliki batas-batas daerah inti sel yang jelas.
DNA	Material yang membawa informasi mengenai aspek struktural dan fungsional organisme dan terdiri dari dua helai yang berpasangan (*double strand*)
RNA	Meterial yang membawa informasi mengenai aspek struktural dan fungsional organisme tertentu dan hanya terdiri dari satu helai (*single strand*)
mRNA	RNA hasil transkripsi DNA dalam proses pembentukan protein.
tRNA	RNA yang berfungsi membawa asam amino untuk dirangkai menjadi protein sesuai dengan urutan codon yang dibawa oleh mRNA.
promoter	Daerah pada DNA yang berada di awal gen dan

Istilah	Arti
	bertindak sebagai tempat RNA polimerase berikatan untuk proses transkripsi.
splice sites	Daerah yang menjadi titik pemisah antara ekson dan intron pada DNA.
splicing	Proses pemotongan atau pemisahan ekson dan intro yang dilakukan untuk membentuk mRNA.
poly-A	Bagian akhir dari gen yang memiliki motif A berulang dan sebagai penanda akhir dari gen.
ekson	Bagian di antara promoter dan poly-A yang dibaca menjadi mRNA pada proses transkripsi
intron	Bagian yang berada di antara ekson.
transcription binding site	Titik pada sekuens DNA yang menjadi tempat menempelnya enzim RNA polimerase 2 yang bertujuan untuk membuat mRNA.
transcription start site	(lihat *transcription binding site*)
codon	Kombinasi dari tiga basa nitrogen yang mengkodekan satu asam amino.
start codon	Kombinasi tiga basa nitrogen yang menandakan awal dari proses translasi.
stop codon	Kombinasi tiga basa nitrogen yang menandakan akhir dari proses translasi.
FGENESH	Perangkat lunak berbasis *hidden Markov model* yang digunakan untuk anotasi gen eukariot.
FGENESB	Perangkat lunak berbasis *hidden Markov model* yang digunakan untuk anotasi gen prokariot.

Istilah	Arti
protein	Senyawa yang dibentuk dari rantai asam amino yang berguna untuk pertumbuhan sel.
RNA polimerase	Enzim pembentuk RNA.
transkripsi	Proses pembacaan DNA oleh RNA polimerasi 2 untuk membentuk mRNA.
translasi	Proses pembacaan mRNA oleh tRNA dalam rangka menyusun asam amino menjadi rantai asam amino untuk membentuk protein.
regulasi	Proses pengaturan bilamana sebuah gen dapat berekspresi menjadi protein.

DAFTAR GAMBAR

Bab I Pendahuluan

I.1. Latar Belakang

Genom adalah kunci kehidupan semua organisme di Bumi. Pada genom terdapat informasi yang lengkap mengenai sebuah makhluk hidup. Genom tersusun dari sejumlah sekuens genetik yang dibentuk dari rantai basa nitrogen yang dikodekan dengan karakter A, T/U, G, dan C. Seiring dengan berkembangnya teknologi DNA/RNA *sequencing*, sekuens genetik dapat dibaca dari jaringan sampel dengan akurasi tinggi dan dalam jumlah besar. Hal ini mengakibatkan fokus penelitian genomik bergeser dari metode *sequencing* ke metode analisis sekuens genetik (Ejigu dan Jung, 2020).

Analisis sekuens genetik dilakukan untuk mendapatkan karakteristik struktural dan fungsional sebuah sekuens. Hal ini dapat dilakukan dengan mencocokkan sekuens pada basis data gen (*homology-based*) atau menganalisis sekuens tersebut secara statistik secara keseluruhan (*ab initio*) (Xiong, 2006). Analisis sekuens secara *ab initio* telah dilakukan menggunakan *machine learning* dan seiring dengan meningkatkan popularitas *neural network*, analisis juga dilakukan menggunakan *deep learning*. Salah satu bentuk analisis sekuens dasar adalah anotasi gen. Anotasi gen dilakukan untuk memprediksi keberadaan gen pada sekuens dan memberikan label terhadap struktur gen tersebut. Dari analisis ini dapat diprediksi ekspresi protein dari sekuens tersebut dan kemudian dapat dilanjutkan ke dalam analisis metabolisme.

Penggunaan machine learning untuk anotasi gen telah banyak dilakukan. Penggunaan Hidden Markov Model (HMM) untuk anotasi DNA eukariot (Solovyev et. al., 2006) dan prokariot (Solovyev dan Salamov, 2011) telah berhasil dilakukan dan diimplementasikan sebagai *webservice* Softberry yang dapat digunakan oleh umum. Penerapan *deep learning* juga dilakukan untuk memprediksi keberadaan promoter dan *splice site* yang memisahkan komponen intron dan ekson pada DNA eukariot (Albaradei et. al., 2020; Oubounyt et. al., 2019; Umarov dan Solovyev, 2017; Umarov et. al., 2019).

Banyaknya variasi genetik mendorong penelitian pada representation learning untuk memperoleh fitur-fitur dari sekuens yang dapat digunakan pada berbagai analisis. Dengan memandang sekuens biologis sebagai teks bahasa alami yang menyimpan informasi makhluk hidup, metode *representation learning* pada *natural language processing* (NLP) kemudian diterapkan untuk analisis sekuens biologis (Iuchi et. al., 2021).

BERT (Devlin et. al., 2018) telah menjadi metode state-of-the-art untuk menghasilkan *distributed representation* dari teks dengan memperhitungkan konteks. Dalam analisis sekuens biologis, konteks menjadi penting karena seringkali ditemukan suatu sekuens yang sama pada suatu DNA tetapi membentuk ekspresi yang berbeda karena sekuens tersebut terletak pada posisi yang berbeda (Iuchi et. al., 2021). Oleh karena itu, arsitektur BERT diadaptasi untuk persoalan analisis sekuens genetik dalam DNABERT untuk menghasilkan reprensentasi DNA yang cukup generik sehingga dapat digunakan berbagai analisis prediksi. Dengan melakukan *fine-tuning* menggunakan data yang spesifik, DNABERT mampu melakukan prediksi promoter, identifikasi *transcription binding sites*, dan identifikasi *splice sites* (Ji et. al., 2021).

Fleksibilitas BERT yang mampu diadaptasi ke berbagai persoalan NLP menjadikannya model yang populer. Liu et. al. (2019) berargumen bahwa dengan menggunakan multi-task learning (Caruana et. al., 1997) pada BERT, representasi yang dihasilkan bisa jauh lebih baik untuk berbagai persoalan NLP yang saling terkait. Hal ini dibuktikan dengan penerapan BERT pada Multi-task Deep Neural Network (MT-DNN) (Liu et. al., 2015) terhadap empat persoalan (natural language understanding) NLU mampu menghasilkan model baru yang dapat meraih skor *state-of-the-art* yang baru (Liu et. al., 2019).

Anotasi gen secara *ab initio* terdiri dari berbagai task. Beberapa diantaranya adalah prediksi promoter, intron, ekson, *transcription start site* atau *transcription start site* (TSS), *splice site* (SS), dan poly-A (Xiong, 2006). Dengan menganalogikan sekuens genetik sebagai teks bahasa alami (Iuchi et. al., 2021), penggunaan *multitask learning* dapat dilakukan untuk membentuk model representation learning dalam lingkup anotasi gen. Keberhasilan *multitask*

learning dalam meningkatkan kemampuan BERT untuk menghasilkan representasi bahasa yang lebih baik membuka peluang untuk meningkatkan kualitas representasi yang dihasilkan DNABERT dengan *multitask learning* yang dapat digunakan untuk prediksi task anotasi gen.

Sampai saat ini belum ditemukan penelitian yang memanfaatkan metode *multitask learning* untuk analisis genetik. Oleh karena itu, penelitian ini diharapkan dapat memberikan kontribusi baru terhadap penerapan metode tersebut dan perbaikan dari implementasi DNABERT (Ji et. al., 2021) pada analisis sekuens genetik.

I.2. Masalah Penelitian

Masalah yang diangkat pada penelitian ini prediksi *splice site* dengan rincian masalah sebagai berikut.

1. Pada satu sekuens bisa terdapat lebih dari satu titik potong (*splice site*) tetapi model *deep learning* dari berbagai penelitian yang ada hanya dapat mendeteksi satu *splice site* saja.
2. Model-model tersebut belajar dari sekuens yang hanya memiliki satu *splice site* dan tidak belajar dari sekuens utuh.

I.3. Tujuan

Untuk menjawab masalah di atas, penelitian ini bertujuan untuk menghasilkan model dengan kemampuan sebagai berikut.

1. Model yang dapat memprediksi semua *splice site* yang terdapat pada sebuah sekuens utuh
2. Model dapat digunakan pada berbagai spesies.

I.4. Hipotesis

Hipotesis pada penelitian ini dibangun berdasarkan argumen-argumen berikut.

1. BERT mampu menghasilkan *language model* dari data nirlabel dan berhasil menjadi model *state-of-the-art* pada persoalan NLU (Devlin et. al., 2018).

2. Penggunaan *multitask learning* dan BERT pada model MT-DNN mampu menghasilkan performa model yang lebih baik pada NLU dibandingkan dengan BERT dan menjadi model *state-of-the-art* yang baru (Liu. et. al., 2019).
3. DNABERT dapat digunakan membuat *language model* dari genom manusia yang cukup universal sehingga dapat dipakai pada beberapa task analisis genetik dari spesies lain dengan melakukan fine-tuning pada DNABERT (Ji et. al., 2021).

Dari empat argumen di atas, dibentuk dua hipotesis sebagai berikut.

1. Penggunaan *pretrained* DNABERT dapat menghasilkan model yang mampu memprediksi *splice site* pada berbagai spesies.
2. Penggunaan informasi keberadaan promoter, poly-A, dan motif *splice site* dapat meningkatkan kemampuan model dalam mencari *splice site*.

I.5. Batasan Masalah

Penelitian ini dibatasi pada ruang lingkup berikut.

1. Penelitian dikerjakan di atas *pretrained* DNABERT yang telah dilatih pada data genom manusia (Ji et. al., 2021)
2. Sekuens DNA yang digunakan pada penelitian ini adalah sekuens DNA eukariot.

Bab II Tinjauan Pustaka

II.1. Multitask Learning

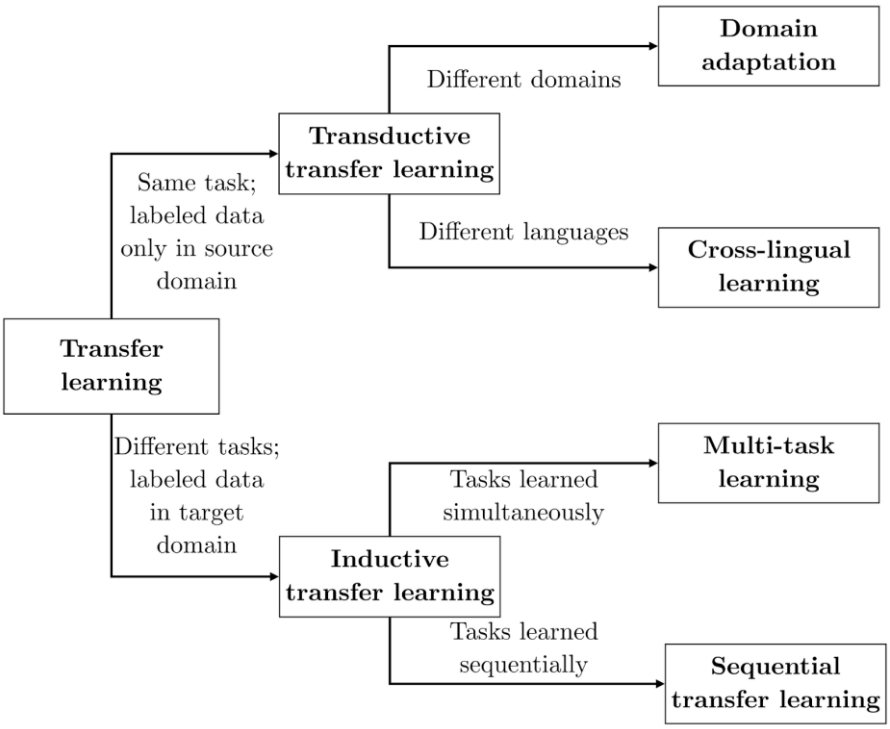
Multi-task learning (MTL) adalah pendekatan *inductive transfer* yang digunakan untuk meningkatkan generalisasi dari model dengan menggunakan informasi dari pelatihan berbagai task yang saling berkaitan sebagai *inductive bias*. MTL melakukan learning secara paralel untuk berbagai task terkait dengan membagi hasil learning tersebut dalam bentuk *shared representation* di antara sesama *task* sehingga proses learning dapat dilakukan lebih baik (Caruana, 1997).

MTL dapat meningkatkan generalisasi melalui beberapa kemungkinan. Cara pertama adalah dengan menambah sinyal *backpropagation* dari berbagai task yang tidak terkait pada *shared representation*. Ketika sinyal dari task yang tidak terkait masuk pada *shared representation*, task yang menjadi tujuan utama learning dapat mengenali hal tersebut sebagai noise. Dengan demikian, task utama dapat mengetahui sinyal-sinyal yang tidak penting untuk dirinya. Kemungkinan lainnya adalah ukuran shared representation yang semakin kecil karena digunakan oleh banyak task mengakibatkan hanya fitur-fitur penting saja yang dipelajari oleh model sehingga model mampu melakukan generalisasi lebih baik.

Ruder (2019) menempatkan MTL sebagai bagian dari Transfer Learning (Gambar II.1). Transfer Learning adalah metode *learning* yang melibatkan penggunaan pengetahuan dari suatu domain pada *task* dari domain lain yang terkait. *Transfer learning* dilakukan ketika pengumpulan data latih untuk sebuah *task* sulit atau mahal untuk dilakukan (Weiss et. al., 2016). TL secara umum dapat dibagi berdasarkan tiga aspek, yaitu kesamaan *task* antar domain, karakteristik dari domain sumber dan domain target, dan urutan pengerjaan task (Ruder, 2019). MTL merupakan transfer learning dengan karakteristik *task* domain sumber berbeda dengan *task* domain target dan berbagai *task* dipelajari secara simultan.

MTL memiliki dua arsitektur shared representation, yaitu hard parameter sharing dan soft parameter sharing. Pada arsitektur hard parameter sharing, hidden layer pada model digunakan secara bersama-sama oleh semua task. Hal yang berbeda pada arsitektur soft parameter sharing. Pada arsitektur ini tiap-tiap task memiliki

hidden layer masing-masing. Namun demikian, di antara hidden layer tersebut terjadi komunikasi dengan tujuan meningkatkan kemiripan nilai di antara hidden layer tersebut.

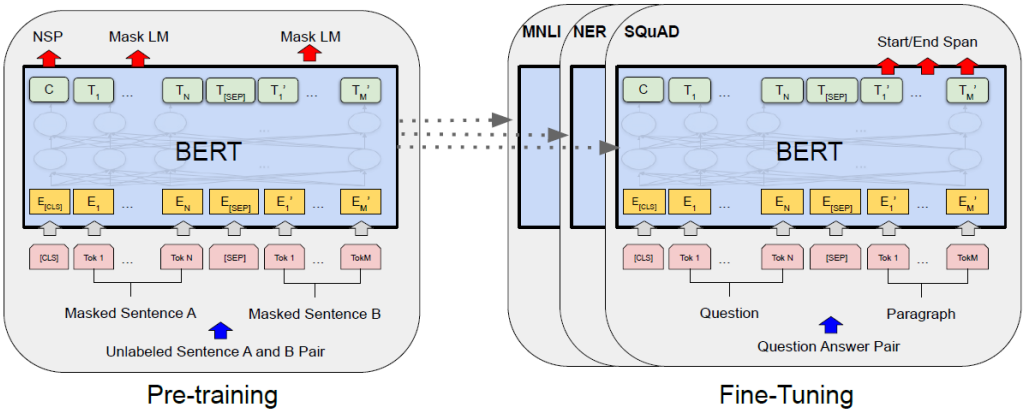


Gambar II.1 Pengelompokan *Transfer Learning* dari Ruder (2019)

II.2. Bidirectional Transformers (BERT)

Salah satu bentuk *transfer learning* adalah *language modelling* (LM). LM adalah model yang menghasilkan representasi numerik sebuah kata berdasarkan semantik dari kata tersebut pada sebuah kalimat dalam konteks tertentu. Setelah dilatih, LM dapat digunakan untuk berbagai task. Salah satu LM yang berhasil menjadi state-of-the-art adalah BERT (Devlin et. al., 2018).

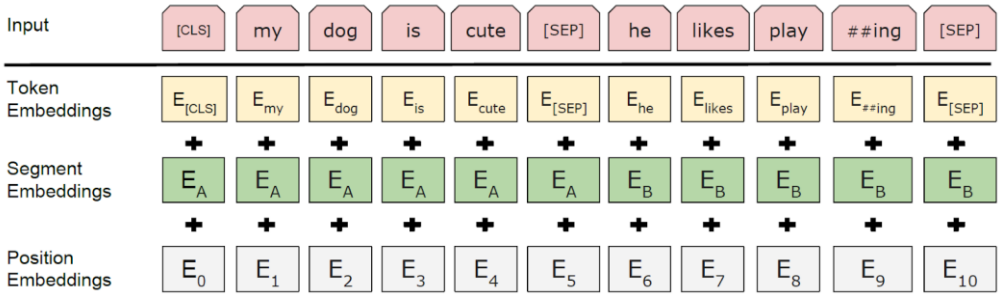
BERT dikembangkan dengan menggunakan Transformer (Vaswani et. al., 2017). Implementasi BERT dibagi ke dalam dua tahap, yaitu *pretraining* dan *fine-tuning*. Pada tahap pretraining, model dilatih dengan menggunakan data nirlabel pada berbagai task dan pada tahap *fine-tuning*, model diinisiasi dengan nilai parameter hasil pretraining dan parameter-parameter tersebut diperbaiki (*fine-tune*) dengan data berlabel untuk *task* tertentu.



Gambar II.2 Arsitektur *Pretraining* dan *Fine-Tuning* BERT (Devlin et. al., 2018)

Proses pretraining BERT diawali dengan menyiapkan sebuah *sequence*. *Sequence* ini dapat berupa satu kalimat atau dua kalimat yang dijadikan satu. Tiap *sequence* diawali dengan token khusus yang disebut dengan classification token atau [CLS]. Tiap token diberikan *embedding* dengan WordPiece (Wu et. al., 2016). Kemudian pada sequence yang terdiri dari dua kalimat, dua kalimat ini dibedakan dengan dua cara. Cara pertama adalah dengan menempatkan token [SEP] di antara dua kalimat tersebut dan cara kedua adalah memberikan embedding pada setiap token yang menandakan kepemilikan token tersebut pada masing-masing kalimat (*segment embeddings*). Setelah itu, tiap-tiap token juga diberikan *embedding* yang menyatakan posisi token tersebut dalam input (*position embedding*). Ilustrasi persiapan input BERT dapat dilihat pada Gambar II.3.

Proses pretraining dilakukan dengan melatih model untuk melakukan dua tugas, yaitu Masked LM (MLM) dan Next Sentence Prediction (NSP). MLM adalah task memprediksi kata tertentu yang disembunyikan pada kalimat (*masked* LM). Input yang diberikan adalah sepasang kalimat nirlabel yang dipisah dengan token khusus. Dari masing-masing kalimat salah satu token dipilih secara acak untuk ditutup (masking) dan model diminta untuk memprediksi token tersebut. Pada NSP, model diminta untuk memprediksi apakah pada input sequence kalimat kedua merupakan kelanjutan dari kalimat pertama. Arsitektur proses *fine-tuning* hanya berbeda dengan proses *pretraining* pada bagian output layer. Output layer pada proses *fine-tuning* disesuaikan dengan *task* tujuan.

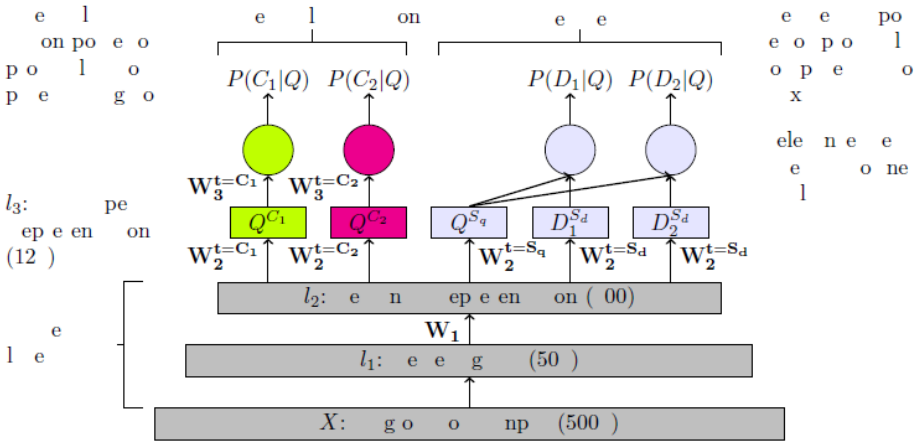


Gambar II.3 Representasi Input BERT (Devlin et. al., 2018)

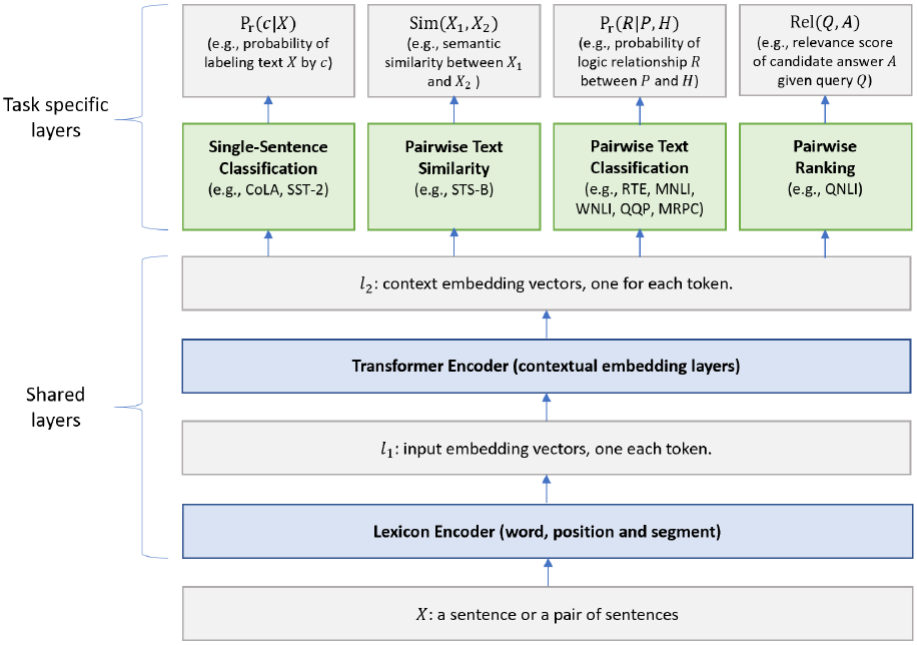
II.3. Multi-Task Deep Learning Neural Network (MT-DNN)

Liu et. al. (2015) pertama kali mengenalkan MT-DNN untuk menghasilkan representasi semantik dari teks (*representation learning*) yang dikhususkan pada persoalan *semantic classification* dan *semantic information retrieval*. Untuk menghasilkan representasi ini, model dilatih secara paralel untuk menjawab persoalan klasifikasi dan ranking. MT-DNN terdiri dari empat bagian, yaitu *input layer* (X), *word hash lay*er (l1), *semantic representation layer* (l2), dan *task-specific representation* (l3). *Input layer*, *word hash layer*, dan *semantic representation layer* adalah tiga bagian yang digunakan oleh semua *task* secara bersamaan (*shared parameter*). Arsitektur MT-DNN dapat dilihat pada Gambar II.4.

Liu et. al. (2019) melanjutkan pengembangan MT-DNN (Liu et. al., 2015) untuk memecahkan masalah natural language understanding (NLU). Liu et. al. (2019) menggunakan pendekatan gabungan antara MTL dan BERT dalam membuat model MT-DNN. MT-DNN dibuat dengan menggunakan empat *task*, yaitu *single text classification*, *pairwise text classification*, *text similarity scoring*, dan *relevance ranking*. Model dilatih terhadap keempat task ini untuk menghasilkan representasi yang lebih baik untuk persoalan NLU. Arsitektur MT-DNN dengan BERT dapat dilihat pada Gambar II.5.



Gambar II.4 Arsitektur MT-DNN (Liu et. al., 2015)



Gambar II.5 Arsitektur MT-DNN dengan BERT (Liu et. al., 2019)

Karena MT-DNN menggunakan BERT sebagai *shared parameter* untuk MTL, input layer (X) model disesuaikan dengan BERT untuk menerima input berupa sequence yang terdiri dari satu atau dua kalimat. Lexicon Encoder melakukan embedding pada input X sesuai dengan perlakukan input pada BERT. Transformer Encoder terdiri dari beberapa lapisan Transformer dua arah (bidirectional) (Vaswani et. al., 2017). Transformer Encoder ini merupakan

shared parameter yang digunakan pada empat *task*. Pada model ini BERT akan dilatih menggunakan berbagai task setelah sebelumnya dilatih dengan unsupervised learning task.

II.4. Analisis Sekuens Genetik

Analisis sekuens genetik adalah proses identifikasi karakter struktural dan fungsional pada sekuens genetik (DNA dan RNA). Data yang digunakan dalam analisis sekuens genetik adalah data genetik berupa teks yang tersusun dari karakter basa nitrogen, yaitu A (adenin), T (Thymine), G (guanine), dan C (cytosine). Empat karakter tersebut ditemukan pada DNA. Hal yang sama juga ditemukan pada RNA kecuali karakter T yang digantikan dengan U (urasil). Selain itu, analisis sekuens genetik juga dilakukan pada protein. Panjang sekuens genetik dinyatakan dalam satuan karakter base-pair (bp). Bentuk lain dari sekuens genetik adalah protein. Protein merupakan sekuens dari asam amino. Berbeda dengan DNA/RNA, sekuens protein terdiri dari karakter-karakter asam amino. Sekuens protein memiliki kaitan erat dengan DNA/RNA karena sekuens asam amino merupakan hasil dari translasi DNA/RNA (Xiong, 2006).

Analisis sekuens genetik dapat dilakukan dengan berbagai metode untuk berbagai tujuan. Salah satu bentuk analisis sekuens genetik adalah prediksi gen dan promoter. Prediksi gen dan promoter adalah analisis terhadap sekuens untuk memperkirakan struktur gen dari sekuens tersebut. Prediksi gen adalah tahap awal dari proses anotasi gen dan genom secara keseluruhan (Xiong, 2006). Dengan memprediksi anotasi gen, ekspresi gen tersebut dapat diperoleh. Dengan memetakan fungsi dan struktur gen dari keseluruhan genom, mekanisme metabolisme organisme dapat diketahui dengan baik. Pengetahuan mengenai metabolisme dapat dimanfaatkan untuk berbagai hal khususnya hal-hal terkait bidang medis.

Struktur gen sel eukariot memiliki struktur seperti pada Gambar II.6. DNA eukariot terbagi menjadi daerah antargen atau *intergenic*, promoter, TSS, daerah ekson dan intron, dan poly-A. Daerah intergenic adalah deretan basa yang memisahkan satu gen dengan gen lainnya. Promoter merupakan daerah yang menjadi tempat berikatan faktor-faktor transkripsi seperti RNA polimerase II

untuk proses pembacaan DNA menjadi mRNA (transkripsi). TSS adalah tempat awal terjadinya proses transkripsi. Kumpulan ekson dan intron adalah bagian dari gen yang memiliki ekspresi seperti protein atau enzim. Poly-A adalah daerah yang menandakan akhir dari gen.

Gambar II.6 Ilustrasi Struktur Gen pada Sel Eukariot

Informasi protein pada DNA eukariot terletak di antara area promoter dan TSS, dan poly-A. Daerah ini terdiri dari kumpulan ekson dan intron. Ketika DNA eukariot akan diekspresikan menjadi protein, bagian intron dari DNA ini harus dikenali dan dipotong pada titik tertentu (intron splicing) sehingga hanya tersisa ekson saja. Ekson-ekson ini kemudian digabung dan diterjemahkan menjadi mRNA untuk kemudian ditranslasikan menjadi protein oleh tRNA. Dari kedua karakter DNA ini, dapat disimpulkan prediksi gen pada DNA eukariot lebih kompleks dibandingkan pada DNA prokariot.

Metode prediksi gen secara umum terbagi dua, yaitu *ab initio* dan *homology based*. Prediksi *ab initio* dilakukan berdasarkan karakter/fitur yang terkandung dari sekuens yang diprediksi. Ab initio bergantung pada dua hal. Hal pertama adalah gene signal yang terdiri dari start dan stop codon, *intron splice*, *transcription binding* sites, *ribosomal binding sites*, dan *polyadenylation sites*. Hal kedua adalah konten dari sekuens itu sendiri. Pendekatan *homology-based* merupakan pendekatan yang membandingkan sekuens dengan basis data sekuens. Dari perbandingan ini dapat diketahui karakteristik dari sekuens yang dianalisis (Xiong, 2006).

Variasi dan jumlah data genetik yang besar melahirkan kebutuhan akan kemampuan untuk memproses data genetik dengan komputer. Untuk meningkatkan kemampuan analisis, metode machine learning diterapkan untuk analisis data genetik. Seiring dengan perkembangan *deep learning*, penelitian implementasi *deep learning* untuk analisis data genetik pun semakin meningkat.

Dalam konteks anotasi gen, deep learning telah dieksplorasi untuk mendeteksi motif-motif tertentu yang menandakan fitur dari sekuens. Berikut ini akan dijelaskan secara ringkas beberapa task terkait anotasi gen dan model *deep learning* yang dirancang untuk menyelesaikan task tersebut.

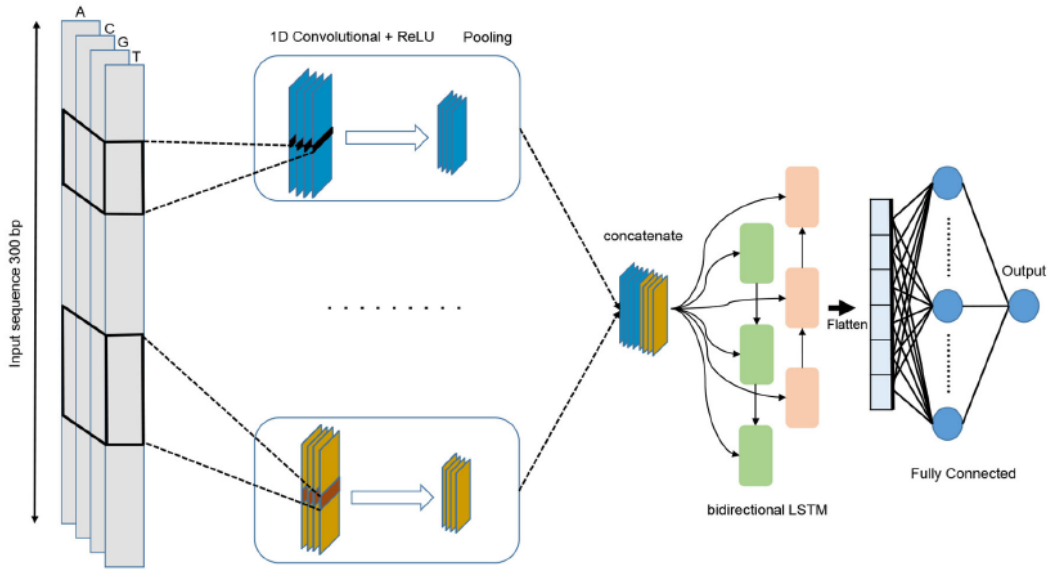
II.5. Prediksi Promoter

Prediksi promoter adalah tahap awal dari prediksi gen. Promoter adalah bagian dari DNA yang terletak sebelum area *transcription start site* (TSS) dan berfungsi sebagai tempat RNA polimerase dan *transcription factor* berikatan. Dengan demikian, kedua hal ini dapat dikatakan sebagai regulator dari ekspresi gen. Pada organisme eukariot umumnya ditemukan promoter bernama TATA-box yang terletak pada area tiga puluh basa sebelum titik TSS dan memiliki motif TATA(A/T) A(A/T) (Xiong, 2006).

DeePromoter merupakan model *deep learning* yang dikembangkan untuk mendeteksi keberadaan promoter TATA-box pada DNA eukariot (Oubounyt et. al., 2019). Promoter TATA-box adalah bagian dari sekuens DNA dengan karakter basa A dan T yang berulang dan menandakan awal proses transkripsi (Baker et. al., 2003). Sebagai penanda awal transkripsi, keberadaan promoter menjadi aspek penting dari analisis ekspresi gen dan jaringan regulasi gen.

DeePromoter dilatih menggunakan dataset dari Eukaryotic Promoter Database (EPDnew) (Dreos et. al., 2012). Dataset terbagi menjadi dua label, yaitu TATA dan non-TATA. Model dilatih untuk mengenali manusia dan tikus. Dari kombinasi dua spesies dan dua label ini terbentuk empat dataset, yaitu Human TATA, Human non-TATA, Mouse, dan Mouse non-TATA. Tiap sekuens memiliki ukuran 300 bp. Dari empat dataset ini, Oubounyt et. al. (2019) membentuk empat dataset negatif. Masing-masing dari keempat label di atas diperlakukan sebagai label positif. Untuk membuat dataset negatif, Oubounyt et. al. (2019) membuat sekuens negatif yang merupakan pasangan dari sekuens positif. Sekuens negatif ini dibuat dengan cara membagi sekuens positif pasangannya ke dalam dua puluh bagian. Posisi dua belas bagian tersebut diacak sementara posisi delapan bagian sisanya tidak diubah.

Oubounyt et. al. (2019) menguji DeePromoter dengan beberapa arsitektur, yaitu CNN, LSTM, BiLSTM, dan kombinasi CNN-LSTM. Arsitektur yang dipilih adalah CNN-BiLSTM. Pada arsitektur ini, sekuens input dikonversi menjadi vektor dua dimensi dengan metode one-hot encoding. Metode ini mengubah masing-masing karakter basa nitrogen menjadi vektor satu dimensi dengan berukuran empat. Asosiasi karakter basa nitrogen dengan vektor konversi yang digunakan adalah (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), dan (0, 0, 0, 1) untuk karakter A, C, G, dan T secara berurutan.



Gambar II.7 Arsitektur DeePromoter (Oubounyt et. al., 2019)

Terlihat pada Gambar II.7 bahwa DeePromoter menggunakan konfigurasi arsitektur yang berbeda untuk memprediksi label TATA dan non-TATA. DeePromoter menggunakan dua lapisan *convolutional layer* dengan *window size* 27 dan 14 untuk data berlabel TATA dan tiga lapisan *convolutional layer* dengan window size 27, 14, dan 7. Semua *convolutional layer* diikuti dengan fungsi aktivasi ReLU (Glorot et. al., 2011), *max pooling layer* dengan window size 6. Hasil dari layer ini digabung secara sekuensial dan diteruskan pada lapisan BiLSTM yang terdiri dari 32 node untuk menangkap fitur ketergantungan antara karakter basa. Setelah itu, proses dilanjutkan pada dua lapis *fully connected layer* yang terdiri dari 128 node di lapis pertama dengan fungsi aktivasi ReLU dan satu

node pada lapis kedua sebagai *classification layer* dengan fungsi aktivasi sigmoid. Performa DeePromoter diukur dengan metrik *precision*, *recall*, dan *mcc*.

Performa DeePromoter diukur dan dibandingkan dengan CNNProm (Umarov dan Solovyev, 2017) sebagai model *state-of-the-art* sebelumnya. Pengujian dilakukan dengan membuat data latih dan data uji dengan metode DeePromoter dan CNNProm. CNNProm membentuk dataset negatif dengan menggunakan dataset non promoter. Hasil pengujian menunjukkan DeePromoter berhasil mencapai skor yang lebih baik untuk tiga metrik di atas dibandingkan dengan model state-of-the-art CNNProm dengan skor rata-rata *precision* dan *recall* masing-masing sebesar 90% dan skor rata-rata *mcc* sebesar 87%.

II.6. Prediksi *Splice Sites*

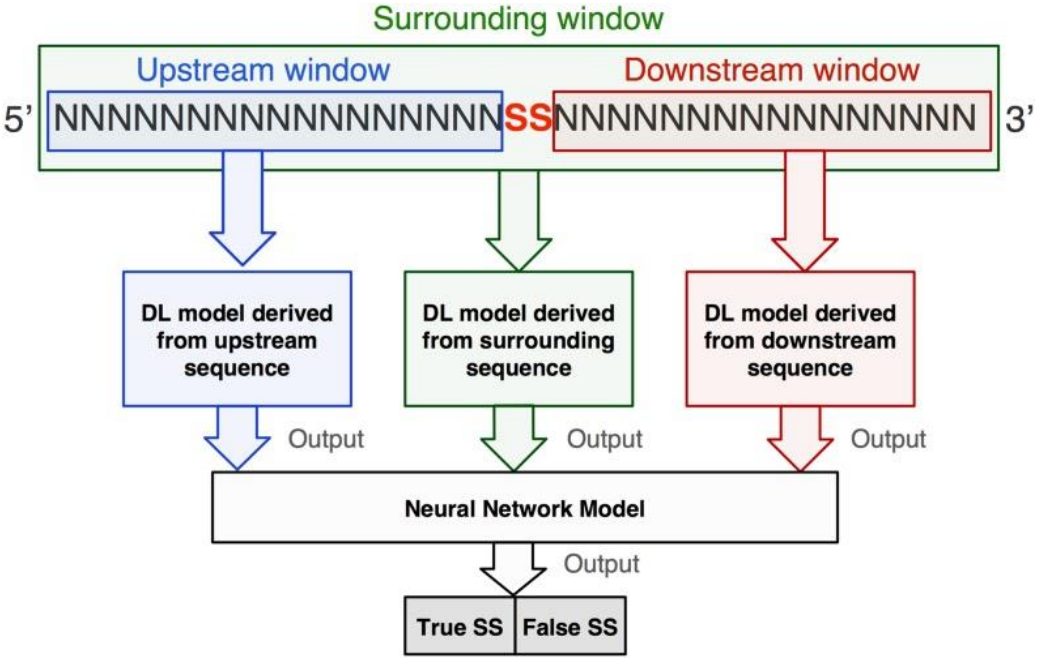
DNA pada sel eukariot memiliki komponen intron dan ekson (Xiang, 2006). Pada proses pembentukan protein, transkripsi dilakukan pada DNA untuk membentuk mRNA dan mRNA tidak mengandung intron. Oleh karena itu pada saat transkripsi, komponen intron akan dipotong dan komponen ekson akan langsung digabung. Splicing adalah operasi pemotongan intron. Dengan memprediksi lokasi splicing, posisi dan bentuk ekson dapat diprediksi dan sekuens mRNA pun dapat diperkirakan. Dari sekuens mRNA dapat diprediksi sekuens asam amino pembentuk protein. Variasi genetik yang tinggi memungkinkan terjadinya alternative splicing yang mengakibatkan proses transkripsi pada satu sekuens DNA dapat menghasilkan berbagai mRNA. *Splice site* (SS) terbagi menjadi dua, yaitu Donor (DoSS) dan Acceptor (AcSS). DoSS adalah *splice site* yang berada setelah ekson dan sebelum intron. AcSS adalah splice site yang berada setelah intron dan sebelum ekson.

Splice2Deep adalah model deep learning yang dilatih untuk memperkirakan lokasi *splice site* pada DNA eukariot (Albaradei et. al., 2020). Splice2Deep terdiri dari lima model yang masing-masing dilatih dengan data genetik organisme yang berbeda. Lima organisme tersebut adalah *H. sapiens*, *A. thaliana*, *Oryza sativa japonica*, *Drosophila melanogaster*, and *C. elegans*. Untuk menghasilkan model yang mampu menggeneralisasi dengan baik, masing-masing model divalidasi

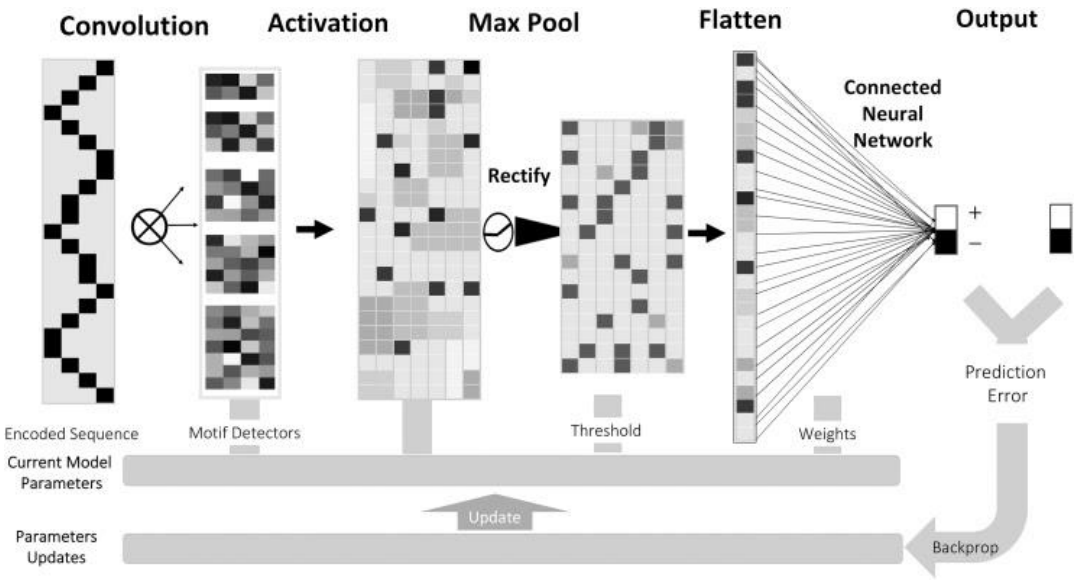
dengan menggunakan data organisme yang berbeda (*cross-organism validation*). Untuk masing-masing model, dibentuk dua model untuk memprediksi DoSS dan AcSS. Dataset yang digunakan untuk *H. sapiens*, *A. thaliana*, *Oryza sativa japonica*, *Drosophila melanogaster*, and *C. elegans* secara berurutan adalah GRCh38.p12 (Zerbino et. al., 2018), TAIR10 (Cheng et. al., 2016), IRGSP-1.0 (Sakai et. al., 2013), BDGP6.22 (Thurmond et. al., 2019), dan WBcel235 (Lee et. al., 2017).

Model dilatih dengan menggunakan data DNA. Sebuah sekuens input (*surrounding window*) dibagi menjadi tiga bagian, yaitu *upstream window*, SS, dan *downstream window*. Pada kasus DoSS, sekuens pada *upstream window* akan direpresentasikan sebagai vektor berukuran 64 yang menggambarkan kombinasi dari tiga karakter basa atau vektor trinukleotida. Contoh representasi trinukleotida adalah AAA = [1, 0, …, 0] dan TTT = [0, 0, …, 1]. Karakter basa pada bagian *downstream window* direpresentasikan dalam vektor dengan ukuran panjang empat (vektor mononukleotida) untuk mewakili A, C, G, dan T. Vektor untuk masing-masing basa secara berurutan adalah [1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], dan [0 ,0, 0, 1]. Pada kasus AcSS, bagian *upstream window* direpresentasikan dengan vektor mononukleotida dan karakter-karakter pada bagian downstream window direpresentasikan dalam vektor trinukleotida. Sekuens surrounding window direpresentasikan dengan vektor mononukleotida. Gambaran umum model dan sekuens input dapat dilihat pada Gambar II.8 dan arsitektur model dapat dilihat pada Gambar II.9.

Splice2Deep menerima input dalam bentuk sekuens karakter DNA, melakukan *feature extraction*, dan feature selection dari *upstream window*, *downstream window*, dan *surrounding window* tersebut di atas. Proses ini dilakukan dengan CNN yang terdiri dari enam bagian, yaitu sequence encoding, convolutional layer (CONV), ReLU, pooling layer (POOL), *fully connected layer* , dan *softmax* layer. Hasil dari model CNN ini kemudian digunakan sebagai input untuk model neural network untuk melakukan klasifikasi biner.



Gambar II.8 Gambaran Umum Splice2Deep (Albaradei et. al., 2020)



Gambar II.9 Arsitektur Model Splice2Deep (Albaradei et. al., 2020)

Setelah masing-masing model dilatih dengan dataset tersebut di atas, masing-masing model divalidasi dengan *cross-organism validation*, yaitu validasi menggunakan data organisme yang bukan organisme data latih model tersebut. Untuk beberapa kasus, model dapat melakukan prediksi lintas organisme dan memberikan hasil yang lebih baik dibandingkan dengan model yang memang

dilatih untuk organisme tersebut. Sebagai contoh, model *C. elegans* memperoleh skor akurasi 94.07% ketika memprediksi SS *D. melanogaster* sedangkan model *D. melanogaster* hanya memperoleh skor akurasi 88.69% ketika memprediksi data dari spesies tersebut. Dari kasus-kasus unik ini, Splice2Deep menunjukkan kemampuan adaptasi untuk memprediksi SS dari data genetik yang tidak dikenal atau data genetik yang baru dan berasal dari spesies yang berbeda.

II.7. Prediksi Poly-A

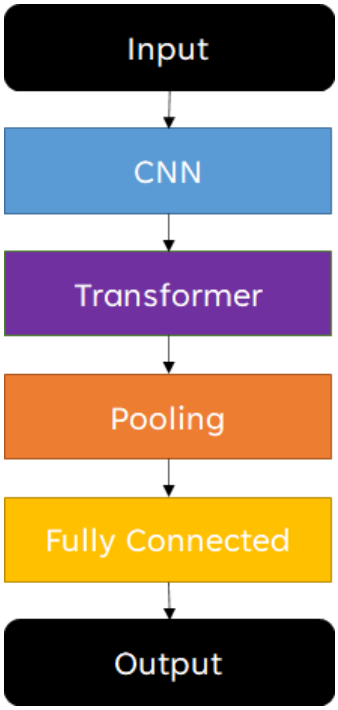
Poly-A atau polyadenylation adalah sekuens yang menandakan akhir dari sebuah gen. Prediksi terhadap lokasi Poly-A dapat memberikan dugaan area gen berada. Pada manusia, sekuens Poly-A umumnya dapat dikenali dengan perulangan basa nitrogen Adenin (A). Motif Poly-A beragam di antara spesies. Sebagai contoh, manusia motif Poly-A yang umum ditemukan adalah AAUAA (Beaudoing et. al., 2000) tetapi motif ini tidak ditemukan pada spesies jamur dan tanaman (Shen et. al., 2008).

SANPolyA (Yu dan Dai, 2020) adalah model deep learning yang dikembangkan menggunakan Transformer (Vaswani et. al., 2017) dan CNN untuk mendeteksi motif Poly-A pada genom manusia dan tikus. Model ini adalah model generik yang tidak dilatih secara khusus untuk mendeteksi motif tertentu sehingga dapat mendeteksi berbagai motif Poly-A yang umum.

Dalam penelitiannya Yu dan Dai (2020) menggunakan dataset Poly-A yang telah digunakan pada berbagai penelitian yang pernah ada. Beberapa dataset tersebut adalah dragon-human (Kalkatawi et. al., 2019), omni-human (Magana-Mora et. al., 2017), C57BL/6J (BL) (Xia et. al., 2018), dan SPRET/EiJ (Xia et. al., 2018). Dataset dragon-human dan omni-human adalah dataset Poly-A yang memiliki dua belas varian motif Poly-A yang umum ditemukan pada genom manusia. Dataset C57BL/6J (BL) dan SPRET/EiJ adalah dataset Poly-A dari genom tikus dan memiliki tiga belas varian motif Poly-A.

Arsitektur umum model SANPolyA dapat dilihat pada Gambar II.10. SANPolyA memiliki empat komponen, yaitu *convolution layer* 1D, Transformer, *pooling layer*, dan *fully connected layer*. Model menerima input berupa sekuens DNA

yang telah dikenakan operasi *mononucleotide encoding*. *Mononucleotide encoding* ini adalah mengubah karakter A, T, C and G menjadi vektor (1,0,0,0), (0,1,0,0), (0,0,1,0) and (0,0,0,1).



Gambar II.10 Arsitektur Model SANPolyA (Yu dan Dai, 2020)

Layer berikutnya adalah *convolutional layer* satu dimensi yang terdiri dari enam belas filter dengan kernel berukuran sepuluh. Untuk memberikan informasi urutan karakter pada sekuens, hasil dari *convolutional layer* diberikan *position embedding*. Setelah diberikan *position embedding*, output tersebut diteruskan pada blok Transformer.

Output dari Transformer diteruskan pada pooling layer dengan window berukuran sepuluh. Untuk mencegah overfitting, output dilewatkan pada dropout layer dan kemudian diteruskan pada *fully connected layer*. *Fully connected layer* terdiri dari 64 hidden unit atau node dengan fungsi aktivasi ELU. Setelah itu klasifikasi dilakukan pada dua node berikutnya dengan fungsi aktivasi sigmoid.

Performa SANPoly-A dibandingkan dengan model deep learning dengan task sejenis. Model deep learning pembanding adalah DPA (Kalkatawi et. al., 2012),

HMM-SVM (Xie et. al., 2013), DeeReCT-PolyA (Xia et. al., 2018), dan Omni-PolyA (Margana-Mora et. al., 2017). Perbandingan dilakukan dengan menghitung skor *error rate* terhadap prediksi tiap-tiap varian motif poly-A dari masing-masing dataset. Untuk semua motif yang ditemukan dari semua dataset yang digunakan, SANPoly-A memiliki error rate yang lebih rendah dibandingkan dengan model deep learning yang dikembangkan untuk masing-masing dataset tersebut.

II.8. Representation Learning

Representation learning atau feature learning adalah metode untuk membentuk sebuah representasi fitur dari data mentah secara otomatis dengan bantuan komputer. Representasi fitur ini kemudian dapat digunakan oleh komputer untuk menyelesaikan berbagai task. Representation learning dapat dibagi menjadi dua jenis, yaitu *supervised representation learning* dan *unsupervised representation learning*. Supervised *representation learning* menghasilkan representasi fitur data mentah yang spesifik untuk persoalan tertentu. Lain halnya dengan pendekatan *supervised, unsupervised representation learning* menghasilkan representasi fitur yang lebih universal dan dapat diadaptasi untuk berbagai persoalan. Contoh dari *representation learning* pada *natural language processing* (NLP) adalah BERT (Devlin et. al., 2018), word2vec (Mikolov et. al., 2013), dan GloVe (Pennington et. al., 2014).

Pada analisis sekuens genetik, *representation learning* dilakukan untuk menggali fitur dari data sekuens untuk kepentingan analisis. Dengan memandang sekuens genetik sebagai teks, metode *representation learning* pada NLP dapat diterapkan pada data sekuens genetik. Beberapa contoh penerapan tersebut adalah sebagai berikut. Prinsip model word2vec (Mikolov et. al., 2013) diterapkan pada ProtVec untuk menghasilkan representasi data sekuens asam amino untuk keperluan klasifikasi famili protein. Model BERT (Devlin et. al., 2018), sebagai salah satu model state-of-the-art di berbagai persoalan NLP, diterapkan pada DNABERT untuk mengekstrak fitur dari data genom manusia dengan pendekatan *unsupervised learning*. Representasi yang dihasilkan dapat digunakan untuk

persoalan klasifikasi lintas spesies dengan melakukan adaptasi atau *fine-tuning* terlebih dahulu (Ji et. al., 2021).

Pada analisis sekuens genetik, BERT (Devlin et. al., 2018) diadaptasi dalam model DNABERT. DNABERT dilatih dengan data genom manusia untuk menghasilkan language model dari genom manusia tersebut. *Language model* ini kemudian digunakan untuk persoalan spesifik (*downstream task*) seperti prediksi promoter, identifikasi varian genetik, dan prediksi *splice site* (Ji et. al., 2021).

DNABERT dilatih dengan data genom manusia dengan metode *unsupervised learning* yang sama dengan BERT (Devlin et. al., 2018). Representasi input pada BERT adalah urutan token yang dikenai *embedding*. Pada DNABERT, token dianalogikan dengan k-mer. K-mer adalah bagian dari sekuens dengan panjang k. Sebagai contoh, sekuens "ATGGCT" dapat dikonversi menjadi empat 3-mer, yaitu ATG, TGC, GGC, dan GCT. Jika semua k-mer digabungkan maka sekuens asal dapat terbentuk kembali. Seperti halnya BERT, DNABERT juga menggunakan token khusus seperti [CLS], [SEP], dan [MASK]. Untuk keperluan pretraining dengan data DNA, DNABERT menambahkan dua token khusus, yaitu unk*nown to*ken [UNK] dan *padding token* [PAD]. Eksperimen dilakukan dengan melatih DNABERT untuk empat nilai k, yaitu 3, 4, 5, dan 6.

Pretrained DNABERT diujikan pada beberapa analisis sekuens genetik, yaitu prediksi area promoter, identifikasi transcription factor binding sites, identifikasi splice sites, dan identifikasi variasi gen. Pada task identifikasi promoter, DNABERT-Prom-300 dibentuk dengan melakukan *fine-tuning* dengan data promoter manusia dari Eukaryotic Promoter Database (EPD) (Dreos et. al., 2013). Input fine-tuning berupa sekuens sepanjang 300 karakter bp. *Fine-tuning* dilakukan terhadap empat label, yaitu TATA positif, TATA negatif, non-TATA positif, dan non-TATA negatif. Data TATA positif dan non-TATA negatif diambil langsung dari data sekuens TATA dan non-TATA. Masing-masing data TATA negatif dan non-TATA negatif dibentuk dengan cara mengambil sekuens sepanjang 300 bp secara acak dari sekuens TATA dan non-TATA.

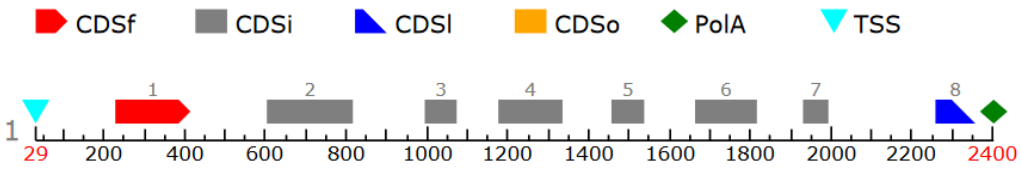
Prediksi promoter dilakukan pada dataset TATA dan non-TATA. Pada masing-masing dataset, DNABERT-Prom-300 digunakan untuk memprediksi label positif dan negatif. Pada dataset TATA DNABERT-Prom-300 berhasil mengungguli DeePromoter (Oubounyt et. al., 2019) dengan selisih skor akurasi dan MCC masing-masing sebesar 0.335 dan 0.554. Pada dataset non-TATA, model tidak menunjukkan peningkatan yang signifikan, yaitu 0.014 (1.4%) dan 0.027 (2.7%) pada skor akurasi dan *mcc*.

Bab III Analisis Masalah dan Perancangan

III.1. Analisis Masalah

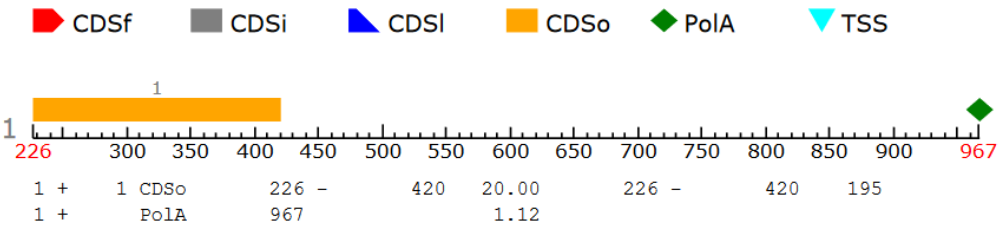
Supervised Learning pada Anotasi Gen. Karakter dari suatu sekuens bergantung pada komposisi basa nitrogen dan karakter dari suatu basa nitrogen ditentukan oleh karakter-karakter pendahulunya. Oleh karena itu, analisis sekuens genetik amat bergantung pada kelimpahan data sekuens. Analisis akan mudah dilakukan jika data berlimpah dan sebaliknya sulit dilakukan jika data tersebut sedikit.

Model yang dihasilkan dengan supervised learning adalah model yang spesifik. Hal ini menyatakan bahwa model tersebut hanya dapat memproses data dengan label tertentu untuk persoalan tertentu. Ilustrasi hal ini dapat ditemukan pada perangkat FGENESH (Solovyev et. al., 2006) dan FGENESB (Solovyev et. al., 2011) yang disediakan oleh Softberry untuk antoasi sekuens gen eukariot dan prokariot. Kedua perangkat ini dibedakan karena karakteristik dari gen eukariot dan prokariot yang berbeda. DNA eukariot terdapat ekson dan intron sedangkan DNA prokariot tidak memiliki kedua hal tersebut. Masing-masing perangkat dibagi berdasarkan spesies yang akan diprediksi. Hal ini disebabkan tiap spesies memiliki DNA yang berbeda baik dari aspek ukuran maupun komposisi basa nitrogen. Berikut ini adalah contoh prediksi gen pada genom A. thaliana (GenBank ID LR782542.1) pada FGENESH *A. thaliana* (Gambar III.1) dan FGENESH *H. sapiens* (Gambar III.2).



Gambar III.1 Prediksi DNA *A. thaliana* pada FGENESH *A. Thaliana*

Hasil prediksi di atas menunjukkan bahwa sekuens genom A. thaliana diprediksi memiliki sebuah gen oleh FGENESH *A. thaliana* tetapi diprediksi tidak memiliki gen oleh FGENESH *H. sapiens*. Hal ini membuktikan adanya karakteristik tertentu pada A. thaliana yang tidak terdeteksi oleh FGENESH H. sapiens.



Gambar III.2 Prediksi DNA *A. thaliana* pada FGENESH *H. sapiens*

Hal ini juga cukup menggambarkan bahwa model *supervised learning* hanya mampu memprediksi data dan label yang sesuai dengan data latihnya. Sampai saat ini, FGENESH telah mampu memprediksi 506 spesies eukariot. Jumlah ini tergolong kecil jika dibandingkan dengan jumlah organisme yang secara keseluruhan.

Anotasi gen dari sekuens DNA secara umum dapat digambarkan seperti Gambar III.1 dan Gambar III.2 di atas. Mesin diberikan sekuens DNA dan kemudian memproses sekuens tersebut untuk mencari bagian-bagian dari gen yang terdiri dari transcription start site (TSS), start codon (CDSf), exon (CDSi), intron, stop codon (CDSl), dan poly-A. Pada umumnya implementasi model *deep learning* untuk anotasi gen masih membahas masing-masing bagian gen tersebut di atas secara terpisah. Bahasan DeePromoter (Oubounyt et. al., 2019) dan model berbasis CNN (Umarov dan Solovyev, 2017; Umarov et. al., 2019) masih terbatas pada deteksi promoter pada gen manusia, tumbuhan, dan bakteri. Hal-hal terkait ekson dan intron dibahas pada penelitian *splice sites* yang terpisah dengan promoter (Albaradei et. al., 2020; Du et. al., 2018). Begitu pula halnya dengan deteksi poly-A (Yu dan Dai, 2020).

Proses prediksi anotasi gen tidak dapat dilakukan secara terpisah. Prediksi lokasi promoter akan menentukan lokasi *transcription start site* (TSS) pada DNA. Lokasi poly-A yang ditemukan setelah TSS dapat diduga sebagai titik akhir dari gen dan terdapat lokasi stop codon (CDSl) di area sebelum poly-A tersebut. Prediksi splice sites dilakukan di antara TSS dan poly-A karena pada lokasi tersebut terdapat ekson dan intron. Oleh karena itu, persoalan anotasi gen adalah persoalan yang memiliki kaitan dengan persoalan prediksi lainnya dan untuk

memecahkan masalah anotasi gen, model harus mampu untuk mengenali bagian-bagian dari gen tersebut di atas.

Dari beberapa permasalahan prediksi di atas, penelitian ini mengangkat masalah prediksi *splice site*. *Splice site* merupakan titik pemisah antara ekson dan intron. Model *deep learning* untuk prediksi *splice site* umumnya merupakan model yang memprediksi keberadaan *splice site* pada posisi tertentuk dari sebuah sekuens. Sebagai contoh, Splice2Deep (Albaradei et. al., 2020) memprediksi keberadaan *splice site* pada posisi tengah sekuens. Prediksi ini dinilai tidak mewakili sekuens yang nyata karena pada beberapa sekuens gen dapat ditemukan lebih dari satu titik *splice site* dan posisi titik tersebut tidak berada di tengah. Oleh karena itu, definisi prediksi *splice site* dapat diperluas menjadi persoalan *sequential labelling* yang tidak hanya melakukan *binary classification* terhadap keberadaan *splice site* tetapi juga dapat menemukan semua lokasi *splice site* pada sebuah gen. *Sequential labelling* dilakukan dengan memberikan label “ekson” atau “intron” pada bagian tertentu dari sekues dan daerah perbatasan antara “ekson” dan “intron” disebut sebagai *splice site*. Untuk dapat melakukan *sequential labelling* model harus dapat membedakan motif atau pola dari promoter, TSS, lokasi titik *splice site*, dan poly A. Keberadaan informasi bertujuan agar model tidak salah dalam mengenali bagian-bagian dari sekuens sehingga dapat memberikan label “ekson” atau “intron” pada bagian yang tepat.

Model *deep learning* yang dihasilkan dengan *supervised learning* adalah model yang dihasilkan adalah model khusus persoalan tertentu. Pada kasus analisis sekuens, model cenderung pada satu spesies tertentu dan tidak mampu melakukan prediksi pada sekuens dari spesies lain. Dengan kata lain, model tidak mampu mengenali fitur-fitur umum dari sekuens spesies-spesies yang mirip sehingga tidak dapat digunakan untuk persoalan prediksi pada spesies lain.

DNABERT (Ji et. al., 2021) merupakan model *deep learning* yang dilatih dengan *unsupervised learning* dengan tujuan mengenali fitur-fitur umum sekuens genetik sehingga dapat menghasilkan representasi sekuens yang universal. Kelebihan dari hal ini adalah representasi DNABERT dapat digunakan untuk berbagai task dengan *fine-tuning* menggunakan dataset yang spesifik untuk *task* tujuan.

Kelemahan pendekatan ini adalah ketika tidak ada data yang tersedia untuk fine-tuning. Selain itu fine-tuning membuat representasi DNABERT memiliki kecenderungan terhadap task tertentu sehingga tidak dapat digunakan untuk menyelesaikan task yang terkait jika tidak dilakukan *fine-tuning* dengan data yang berbeda.

III.2. Analisis Solusi

Masalah utama yang diangkat adalah identifikasi splice site yang hanya mampu menemukan satu titik splice site sedangkan pada sekuens boleh jadi terdapat lebih dari satu splice site. Oleh karena itu, identifikasi *splice site* adalah persoalan *sequential labelling*. Pada persoalan ini, sebuah model memberikan label pada bagian-bagian dari sekuens input berdasarkan karakteristik tertentu.

Untuk menemukan berbagai titik *splice site* pada sebuah sekuens perlu diketahui terlebih dahulu area yang menjadi lokasi ekson dan intron, dan pola daerah yang menjadi titik *splice site*. Dengan demikian, dibutuhkan tiga informasi untuk dapat melakukan sequential labelling pada sekuens DNA. Tiga informasi tersebut adalah promoter sebagai awal area ekson-intron, poly-A sebagai akhir area ekson-intron, dan pola atau motif sekuens yang menandakan titik splice site. Untuk menghasilkan model yang memiliki ketiga informasi ini, multask learning diusulkan untuk dilakukan pada model.

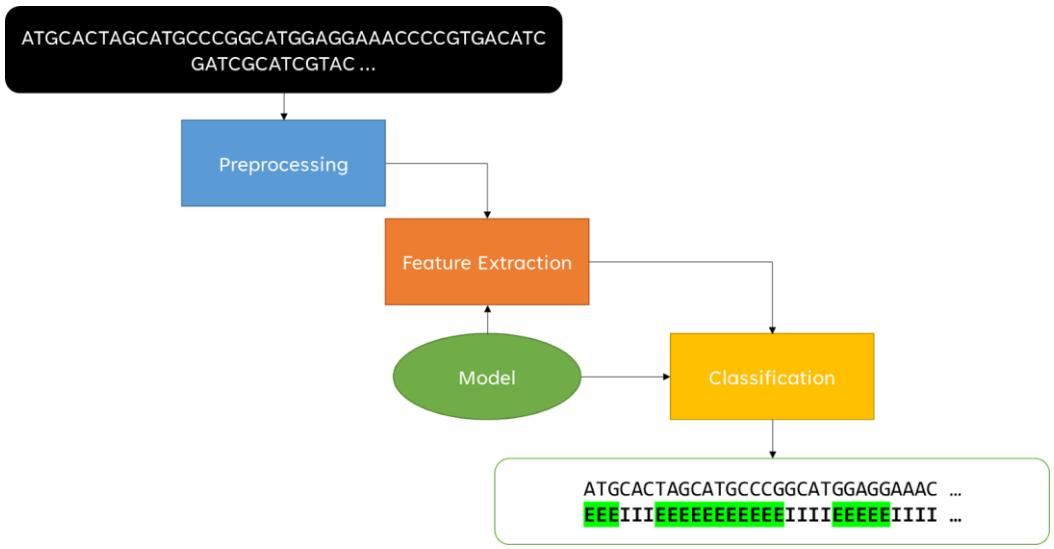
Pada *multitask learning* model dilatih secara simultan untuk beberapa persoalan tertentu dengan harapan dapat menyelesaikan persoalan utama yang terkait. Dalam proses identifikasi *splice site* pada gen terdapat beberapa task yang perlu dilakukan untuk menemukan bagian dari gen yang akan mengekspresikan protein atau ekson. Dengan mengetahui pola area promoter, poly-A, dan titik *splice site*, model diperkirakan dapat memberikan label pada sekuens pada daerah-daerah yang menjadi ekson atau intron dan tidak memberikan label ekson atau intron pada daerah promoter atau poly-A.

Model dilatih secara simultan untuk task prediksi promoter, prediksi poly-A, dan prediksi *splice site* dengan harapan model dapat mengenali pola untuk masing-masing *prediction task*. Model kemudian diminta untuk melakukan sequential

labelling pada sekues dengan label “ekson” dan “intron”. Lokasi di antara dua label ini disebut sebagai *splice site*.

III.3. Rancangan Solusi

Arsitektur umum sistem solusi yang diusulkan dapat dilihat pada Gambar III.3 berikut. Sistem menerima input berupa data sekuens DNA dan melakukan pelabelan terhadap sekuens dengan cara *sequential labelling* dengan label “ekson” atau “intron”.

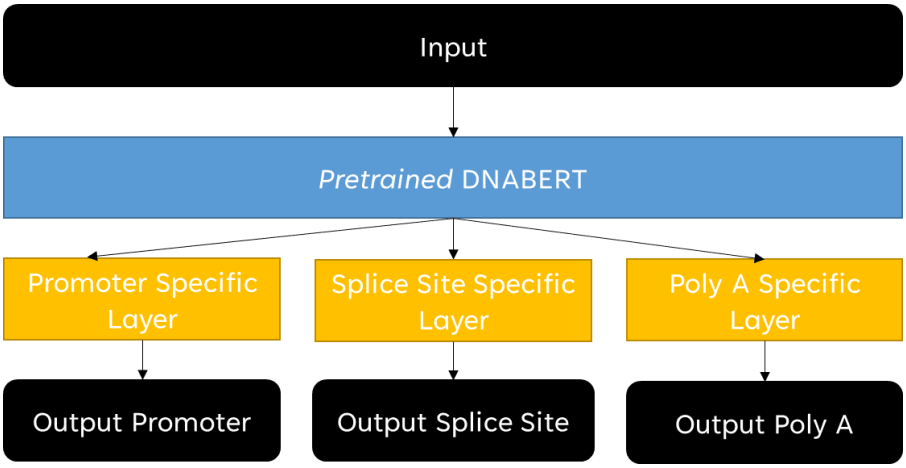


Gambar III.3 Arsitektur Umum Sistem

Sekuens DNA adalah teks yang terdiri dari karakter A, C, G, dan T yang mewakili empat basa nitrogen Adenine, Cytosine, Guanine, dan Thymine. Pada sekues ini kemudian dilakukan *preprocessing* untuk memecah sekuens menjadi daftar token atau k-mer. Pada domain pemrosesan bahasa alami, token secara umum dapat diartikan sebagai kata. Pada pemrosesan data genetik, token adalah k-mer. Istilah k-mer dapat diartikan sebagai bagian dari sekuens berukuran k. Sebagai contoh, jika terdapat sekuens genetik “ATGCGACGAA” maka 3-mer atau k-mer berukuran 3 dapat diperoleh dengan membaca sekuens tersebut secara tiga karakter per tiga karakter. Token list atau k-mer list yang diperoleh adalah ATG, TGC, GCG, CGA, GAC, ACG, CGA, dan GAA.

Setelah daftar k-mer berhasil dibentuk, tiap-tiap k-mer kemudian dilakukan *feature extraction* untuk mendapatkan vektor yang merepresentasikan k-mer tersebut. Pembentukan vektor dilakukan dengan mengubah tiap-tiap karakter basa nitrogen menjadi vektor mononucleotide. Vektor ini berukuran empat. Karakter A, C, G, dan T secara berturut-turut akan diubah menjadi vektor (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), dan (0, 0, 0, 1). Setelah itu vektor dari tiap-tiap k-mer digabung menjadi satu dan diklasifikasikan terhadap label dari *task* tertentu dengan menggunakan *classification model*. Model ini diberi nama DNABERT-MT-SeqSplice.

Ilustrasi arsitektur umum model dapat dilihat pada Gambar III.4. Arsitektur DNABERT-MT-SeqSplice secara umum terdiri dari tiga bagian, yaitu *input layer*, *pretrained* DNABERT sebagai *shared representation layer*, dan *task specific layer*. Karena model dirancang untuk melakukan DNABERT sebagai *shared representation layer*, *input layer* melakukan pemrosesan terhadap sekuens input yang berupa data sekuens untuk menghasilkan *embedding* yang dapat diproses oleh DNABERT.



Gambar III.4 Arsitektur Umum Model DNABERT-MT-SeqSplice

Pada sistem ini DNABERT digunakan sebagai perangkat untuk mengekstrak fitur dari sekuens. Hasil dari *feature extraction* ini adalah sebuah representasi input. Representasi input ini kemudian digunakan untuk berbagai kebutuhan persoalan. Pada *multitask learning* representasi ini digunakan untuk prediksi tiga hal, yaitu

promoter, *splice site*, dan polyA. Masing-masing persoalan menggunakan arsitektur *task specific layer* yang berbeda. Untuk ketiga persoalan ini, arsitektur *task specific layer* yang digunakan mengacu pada beberapa model yang telah dikembangkan untuk persoalan yang sama.

Task specific layer adalah lapisan yang dirancang khusus untuk suatu persoalan prediksi tertentu. DeePromoter menggunakan arsitektur CNN, bidirectional LSTM (biLSTM) dan *fully connected layer* (FC) untuk memprediksi promoter TATA pada manusia dan tikus (Oubounyt et. al., 2019). CNN dan biLSTM digunakan untuk mengekstrak fitur multidimensi dan urutan dari sekuens DNA untuk kemudian dihitung pada FC layer untuk klasifikasi. SANPolyA (Yu dan Dai, 2020) menggunakan arsitektur CNN, Transformer (Vaswani et. al., 2017), dan *fully connected layer* untuk mendeteksi keberadaan Poly-A pada sekuens DNA. CNN digunakan untuk mengekstrak dan dari karakter sekuens dan Transformer digunakan untuk mengambil informasi terkait hubungan dan urutan antara karakter input. Dengan kata lain, penelitian-penelitian sebelumnya memiliki perbedaan pada arsitektur *feature extraction* dan konfigurasi *fully connected layer* untuk klasifikasi. Pada penelitian ini, tiap-tiap task akan menggunakan DNABERT sebagai lapisan *feature extraction*. Dengan demikian, lapisan yang bisa diadaptasi adalah *fully connected layer*.

Arsitektur *fully connected layer* dari model DeePromoter (Oubounyt et. al., 2019) diadopsi sebagai task specific layer untuk persoalan prediksi promoter. *Fully connected layer* untuk prediksi promoter terdiri dari dua lapisan. Lapisan pertama terdiri dari 128 node dengan ReLU sebagai fungsi aktivasi dan lapisan kedua terdiri dari dua node sebagai classification layer. Pada persoalan prediksi poly-A juga digunakan fully connected layer dari SANPolyA (Yu dan Dai, 2020). Layer ini terdiri dari dua lapis. Lapis pertama 64 node dengan ELU sebagai fungsi aktivasi dan lapis kedua terdiri dua node dengan fungsi aktivasi sigmoid sebagai classification layer. Konfigurasi ini diadopsi untuk task specific layer persoalan prediksi poly-A. Persoalan prediksi *splice site* menggunakan *task specific layer* berupa *fully connected layer* yang terdiri dari dua lapisan. Lapisan pertama terdiri 512 node dan lapisan kedua terdiri dari dua node untuk *binary classification*.

Setelah proses *multitask learning* dilakukan, model kemudian diadaptasi untuk melakukan *sequential labelling*. Arsitektur DNABERT-MT-SeqSplice diubah pada bagian *output layer* agar dapat menghasilkan keluaran berupa sekuens label.

DAFTAR PUSTAKA

Albaradei, S., Magana-Mora, A., Thafar, M., Uludag, M., Bajic, V. B., Gojobori, T., Essack, M., & Jankovic, B. R. (2020). Splice2Deep: An ensemble of deep convolutional neural networks for improved splice site prediction in genomic DNA. Gene: X, 5, 100035. https://doi.org/10.1016/j.gene.2020.100035

Baker, T. A., Watson, J. D., Bell, S. P., Gann, A., Losick, M., and Levine, R. (2003). Molecular Biology of the Gene (San Francisco, CA:). Benjamin-Cummings Publishing Company

Caruana, R. (1997). Multitask Learning. Machine Learning 28, 41–75. https://doi.org/10.1023/A:1007379606734

Dreos, R., Ambrosini, G., Cavin Perier, R. and Bucher, P. (2013) EPD and EPDnew, high-quality promoter resources in the next-generation sequencing era. Nucleic Acids Res, 41, D157-164

Du, X., Yao, Y., Diao, Y., Zhu, H., Zhang, Y., & Li, S. (2018). DeepSS: Exploring Splice Site Motif Through Convolutional Neural Network Directly From DNA Sequence. IEEE Access, 6, 32958–32978. doi:10.1109/access.2018.2848847

Ejigu, G. F., & Jung, J. (2020). Review on the Computational Genome Annotation of Sequences Obtained by Next-Generation Sequencing. Biology, 9(9), 295. https://doi.org/10.3390/biology9090295

Ji, Y., Zhou, Z., Liu, H., & Davuluri, R. V. (2021). DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. Bioinformatics, 37(15), 2112–2120. https://doi.org/10.1093/bioinformatics/btab083

Kalkatawi, M., et al. (2012) Dragon PolyA Spotter: predictor of poly(A) motifs within human genomic DNA sequences, Bioinformatics, 28, 127-129.

Liu, X., Gao, J., He, X., Deng, L., Duh, K., & Wang, Y. Y. (2015). Representation learning using multi-task deep neural networks for semantic classification

and information retrieval. NAACL HLT 2015 - 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 912–921. https://doi.org/10.3115/v1/n15-1092

Liu, X., He, P., Chen, W., & Gao, J. (2019). Multi-Task Deep Neural Networks for Natural Language Understanding. http://arxiv.org/abs/1901.11504

Magana-Mora, A., Kalkatawi, M. and Bajic, V.B. (2017) Omni-PolyA: a method and tool for accurate recognition of Poly(A) signals in human genomic DNA, BMC Genomics, 18, 620.

Mikolov, T., Chen, K., Corrado, G., & rey Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. Nips (2013), 1–9. DOI: h p. Dx. Doi. Org/10.1162/Jmlr, 4–5.

Oubounyt, M., Louadi, Z., Tayara, H., & To Chong, K. (2019). Deepromoter: Robust promoter predictor using deep learning. Frontiers in Genetics, 10(APR), 286. https://doi.org/10.3389/fgene.2019.00286

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 1532–1543. https://doi.org/10.3115/v1/d14-1162

Quang, D., & Xie, X. (2016). DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. Nucleic acids research, 44(11), e107. https://doi.org/10.1093/nar/gkw226

Solovyev, V., Kosarev, P., Seledsov, I. et al. Automatic annotation of eukaryotic genes, pseudogenes and promoters. Genome Biol 7, S10 (2006). https://doi.org/10.1186/gb-2006-7-s1-s10

Solovyev, V., Salamov, A. (2011). Automatic Annotation of Microbial Genomes and Metagenomic Sequences. In Metagenomics and its Applications in Agriculture, Biomedicine and Environmental Studies (Ed. R.W. Li), Nova Science Publishers, p. 61-78

Umarov, R. K., Solovyev, V. V. (2017). Recognition of prokaryotic and eukaryotic promoters using convolutional deep learning neural networks. PLoS ONE 12(2): e0171410. https://doi.org/10.1371/journal.pone.0171410

Umarov, R., Kuwahara, H., Li, Y., Gao, X., & Solovyev, V. (2019). Promoter analysis and prediction in the human genome using sequence-based deep learning models. https://doi.org/10.1093/bioinformatics/bty1068

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 2017-Decem, 5999–6009

Xia, Z., et al. (2018) DeeReCT-PolyA: a robust and generic deep learning method for PAS identification

Xiong, J. (2006). Essential Bioinformatics. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511806087

Yu, H., & Dai, Z. (2020). SANPolyA: a deep learning method for identifying Poly(A) signals. Bioinformatics. doi:10.1093/bioinformatics/btz970

Zhou, J., & Troyanskaya, O. (2015) Predicting effects of noncoding variants with deep learning–based sequence model. Nat Methods 12, 931–934. https://doi.org/10.1038/nmeth.3547