PENGGUNAAN HIDDEN MARKOV MODEL (HMM) UNTUK MENGIDENTIFIKASI RNA FAMILY

Toto Haryanto [1] Agus Buono [2] Taufik Djatna [3]

[1] Departemen Ilmu Komputer FMIPA IPB
Jl. Meranti Wing 20 Lv.5 Kampus IPB Darmaga-Bogor 16680 Indonesia
email : haryanto.toto@gmail.com
[2] Departemen Ilmu Komputer FMIPA IPB
Jl. Meranti Wing 20 Lv.5 Kampus IPB Darmaga-Bogor 16680 Indonesia
email : pudesha@yahoo.co.id
[3] Departemen Teknologi Industri Pertanian IPB
Kampus IPB Darmaga-Bogor 16680 Indonesia
email: taufikdjatna@ipb.ac.id

ABSTRACT

Pada awalnya, untuk mengklasifikasikan sekuens baru dari Asam Ribonukelat (RNA) dilakukan pensejajaran dua sekuens. Namun hal ini mengalami kendala apabila terdapat fragmen dari sekuens tersebut yang tidak lengkap. Hidden Markov Model (HMM) merupakan model probabilistik yang banyak diaplikasikan untuk permasalahan deret waktu atau sekuens linear. Di sisi lain, non-coding RNA memiliki banyak family yang dapat diidentifikasi dengan melihat untaian sekuensnya. Penelitian pendekatan model HMM yang digunakan memiliki jumlah state sebanyak 2 hidden state dan 3 hidden state.

Data yang digunakan pada penelitian ini adalah data sekuens non-coding RNA (ncRNA) dari Genome Research Institute yang telah terlabeli sebanyak 5066 yang terbagi menjadi 7 kelas. Dari data tersebut 50% digunakan sebagai data pelatihan dan sisanya digunakan sebagai pengujian.

Hasil pengujian menunjukkan bahwa penggunaan 3 hidden state akan menghasilkan identifikasi yang secara umum lebih tinggi dibandingkan dengan 2 state. Rata-rata akurasi terbaik terdapat pada identifikasi kelas ke-7 dengan 3 hidden state yaitu sebesar 77.08%. Akan tetapi, terjadi kondisi yang kontradiksi pada identifikasi kelas pertama dengan penurunan rata-rata tingkat akurasi yang sangat signifikan menjadi 5.44 % dengan 3 hidden state.

Key words

Asam Ribonukleat (RNA), Hidden Markov Model (HMM), non-coding RNA

1. Pendahuluan

Protein, RNA dan berbagai fitur dalam genome biasanya dapat diklasifikasikan menjadi satu keluarga atau tertentu sesuai dengan sekuens atau strukturnya [1].

Non-coding RNA (ncRNA) molekul adalah RNA yang tidak menyandikan protein, akan tetapi melayani beberapa fungsi lain di dalam sel. ncRNA memiliki berbagai peran kritis di semua kerajaan hidup dan memiliki fungsi penting untuk menentukan fungsi tiga dimensi dari fungsi molekul [2]. ncRNA ini memiliki banyak dengan tugas yang berbeda. Perkembangan non-coding RNA demikian pesat sehingga komputasi dalam bidang ini selalu memiliki tantangan baru. RNA memerlukan protein untuk pendamping dalam proses pelipatan atau *folding*, namun untuk sebagian besar pada akhir struktur tiga dimensi, ditentukan oleh struktur sekundernya [2]. Ini menunjukkan bahwa pengembangan tools berdasarkan struktur sekunder RNA sangat penting untuk penemuan Non-coding RNA dan mengklasifikasikan peran fungsional dari RNA tersebut.

Berbagai metode komputasi dapat digunakan untuk mencari dan mengkategorikan ncRNA berdasarkan sekuensnya. Salah satunya adalah Hidden Markov Model (HMM). Menurut Eddy[3], HMM merupakan suatu kelas dari model probabilistik yang secara umum dapat diaplikasikan untuk permasalahan deret waktu atau sekuens yang bersifat linear. Sejalan dengan itu, HMM merupakan metode yang dianggap memiliki kesuksesan dalam menyelesaikan permasalahan di dalam analisis sekuens meskipun dari sisi kompleksitas masih sulit untuk ditentukan secara manual [4].

Di antara beberapa permasalahan yang terdapat di dalam HMM adalah masih terbatasnya model untuk dijadikan

acuan dalam memprediksi non-coding RNA (ncRNA). Oleh karena itu, pada penelitian ini akan diusulkan pembuatan model ncRNA dengan menggunakan HMM untuk melakukan klasifikasi terhadap Keluarga RNA. Adapun data yang digunakan adalah data yang berasal dari koleksi data RNA *family*. Data tersebut dapat diunduh di alamat *website http://rfam.sanger.ac.uk* yang merupakan Genome Research Institute.

Pada dasarnya, di dalam melakukan prediksi struktur protein dapat terbagi menjadi dua [5], yaitu:

- Membandingkan model yang telah ada dengan struktur yang akan diprediksi
- *de novo* yaitu apabila tidak terdapat model yang tersedia untuk dibandingkan dengan struktur yang akan diklasifikasikan.

Pada penelitian ini yang akan dilakukan adalah membuat suatu model untuk mengklasifikasikan RNA tersebut. Model tersebut akan dibuat dengan menggunakan Hidden Markov Model (HMM) yang telah secara luas digunakan untuk menyelesaikan permasalahan dalam analisis sekuens.

2. Hidden Markov Model untuk Identifikasi RNA

Hidden Markov Model (HMM) merupakan model probabilistik yang dapat diaplikasikan untuk menganalisa model deret waktu atau sekuens linear. Pada era tahun 1990, untuk membandingkan dua buah sekuens data biologi baik DNA atau RNA digunakanlah perbandingan pasangan antara dua sekuen yang akan disamakan. Namun, terdapat kendala yang ada apabila dua sekuens tersebut tidak lengkap fragmennya di samping kesulitan apabila adanya sekuens baru [6]. HMM adalah salah satu pendekatan yang digunakan untuk memodelkan kumpulan sekuens tersebut. HMM telah banyak dikembangkan pada banyak permasalahan diantarnya adalah *speech recognition* [7]. Adapun HMM mulai diperkenalkan pada bidang Komputasi Biologi sekitar tahun 1980 [8]. Pada Hidden Markov Model (HMM) sekuens dari simbol (seperti A,C,G,T/U) dalam sekuens RNA merupakan peubah yang secara langsung dapat diobservasi. Pada kasus anaslisis sekuens dari data biologi, state sekuens akan berasosiasi dengan label biologis yang bermakna (seperti: struktur pada posisi lokus 42) [3].

2.1 Data Set

Data set yang digunakan untuk penelitian ini adalah data segmen RNA Family dengan format FASTA (*file* berekstensi .fasta). Format FASTA merupakan format *file* berbasis teks yang digunakan untuk merepresentasikan sekuens nukleotida atau sekuens asam amino yang dikodekan dalam bentuk karakter huruf tunggal. *File* dengan format FASTA dapat merepresentasikan sekuens tunggal atau *multiple* sekuens dalam satu *file*. *File* dengan format FASTA dimulai dengan tanda "<" yang diikuti dengan spesifikasi dari sekuensnya. Berikut adalah contoh data RNA family dengan format FASTA.

```
<BA000002.3/627912-
27792RF00001;5S_rRNA;
CGGCCCGGCCAUAGCGGCCGGGUAACACCCGGACUCAUU
UCGAACCCGGAAGUUAAGCCGGCCGCGUUGGAGGCUCCA
GUGGGGUCCGAGAGGCCCUGCAGGGGCCUCCAAGCCGG
GGCCG;
```

Ket: A=Adenin, C= Cytosine, G= Guanin, U=Urasil

Informasi yang didapatkan dari data tersebut adalah bahwa RF0001;5s_rRNA menunjukkan kelas target. Sedangkan urutan sekuens tersebut adalah nilai dari penciri yang dimiliki oleh setiap kelasnya. Pada penelitian ini, RNA akan diklasifikasikan menjadi 7 kelas sebagaimana dapat dilihat pada Tabel I.

Tabel 1. Kelas RNA

No	Kode Kelas	Deskripsi
1	RF00001;5S_rRNA;	RNA ribosom
2	RF00003;U1;	spliceosomal RNA
3	RF00005;tRNA;	transfer RNA
4	RF00634;SAM-IV;	riboswitches
5	RF00655;mir-28;	microRNAs
6	RF01299;snR39B;	Small nucleolar RNAs
7	RF01313;HHBV_epsilon;	HBV RNA encapsidation signal epsilon

2.2 Praproses

Tahap ini dilakukan untuk mengubah format data menjadi format data yang sesuai untuk dijadikan masukan dalam HMMs. Praproses perlu dilakukan karena format data dalam *file* berekstensi .fasta harus diubah dan dilakukan encoding terlebih dahulu. Oleh karena itu, dilakukan parsing terlebih dahulu.

2.3 Pelatihan

Proses pelatihan bertujuan untuk mendapatkan model atau profil dari setiap kelas RNA. Pada proses pelatihan ini akan dihasilkan 7 model yang merepresentasikan setiap kelasnya. Model tersebut dapat dinotasikan $\lambda = (A, B, \pi)$ sebagai model parameter di mana A menunjukkan matriks peluang transisi, B menunjukkan matriks peluang observasi dan π menunjukkan matriks peluang suatu kejadian pada tahap awal. Model tersebut akan diinisialisiasi terlebih dahulu.

Dengan proses pelatihan ini, akan didapatkan nilai model baru yang telah terlatih. Terdapat beberapa algoritma yang digunakan untuk melakukan pelatihan. Pada penelitian ini yang akan digunakan adalah Algoritma Baum-Welch. Pada Algoritma Baum-Welch, mendukung pelatihan dengan *Multiple Observation Sequence* sehingga sangat sesuai dengan permasalahan yang dihadapi. Berikut adalah langkah-langkah algoritman Baum-Welch [7].

Inisialisas i: set nilai $\lambda = (A, B, \pi)$. Algoritma ini akan memperbaiki nilai λ secara iteratif sampai konvergen.

prosedur forward : definisikan $\alpha_t(i) = p(O_1 = O_1, O_2, ...O_t, i_t = i \mid, \lambda)$ sebagai peluang obsevasi parsial dari sekuens $O_1, O_2, ...O_t$ sampai dengan state ke-i pada saat t. Secara rekursif, $\alpha_t(i)$ dapat dihitung sebagai berikut :

$$\alpha_t(i) = \pi_i b_i(o_i) \quad \text{.....(1)}$$

$$\alpha_{t+1}(j) = b_j(o_{t+1}) \left[\sum_{i=1}^{N} \alpha_t(i).a_{ij} \right] \quad \text{.....(2)}$$

prosedur backward : definisikan $\beta_t(i) = P(O_{t+1}, O_{t+2}, ..., O_T \mid i_t = i, \lambda)$, adalah peluang observasi parsial sekuens dari t + 1 sampai T dengan state i pada saat t dan model λ. Secara effisien dapat dihitung :

$$\beta_T(i) = 1, \; 1 \le i \le N \quad \text{.....(3)}$$

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \quad \text{.....(4)}$$

Menghitung γ dan ξ

Dengan menggunakan α dan β, akan ditentukan dua variabel, yaitu $\gamma_t(i)$ dan $\xi_t(i, j)$ dengan persamaan sebagai berikut:

$$\gamma_t(i) = \left[\frac{\alpha_t(i) \beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i) \beta_t(i)} \right] \quad \text{.....(5)}$$

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} \beta_{t+1}(j) b_j(O_{t+1})}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} \beta_{t+1}(j) b_j(O_{t+1})} \quad \text{.....(6)}$$

Mengupdate paramater model

Dengan mengasumsikan model saat inisialisasi adalah $\lambda = (A, B, \pi)$, maka, *update* nilai baru untuk mereestimasi parameter adalah:

$$\overline{\pi_i} = \gamma_1(i), \qquad 1 \le i \le N \quad \text{.....(7)}$$

$$\overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}, \quad 1 \le i \le N, \; 1 \le j \le N \quad \text{.....(8)}$$

$$\overline{b_j}(k) = \frac{\sum_{\substack{t=1 \\ O_t = vk}}^{T} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)}, \; 1 \le j \le N, \; 1 \le k \le M \quad \text{.....(9)}$$

dengan

λ adalah model HMM
A adalah matrik peluang transisi,
B adalah matrik peluang emisi dan
π adalah matrik peluang awal / matrik priority
$O = O_1, O_2, ...O_T$ adalah variabel observasi

3. Hasil Percobaan

3.1 Praproses Data Sekuens

Data yang tersedia pada awalnya merupakan data format fasta. Untuk dapat dijadikan model, formatnya akan diubah menjadi vektor sekuens untuk setiap instance-nya dengan melakukan proses parsing. Selain itu, pada data ncRNA ini semua kelas masih berada pada satu *file* sehingga akan dikelompokkan dan dipisahkan perkelas untuk dijadikan data latih dan data uji.

Pada praproses juga dilakukan *encoding* karakter 'A','C','G','U' menjadi '1','2','3','4'. Proses *encoding* ini ialah hanya untuk memudahkan dalam proses perhitungan yang dilakukan. Contoh hasil praproses dari data sekuens adalah suatu vektor = [1 2 3 4 2 3 2 3 3 2 3 4 4 4 4 3].

Pada penelitian ini, jumlah data total yang digunakan adalah 5066 *record* data yang terbagi ke dalam tujuh kelas. Dari setiap kelas tersebut 50 persen digunakan sebagai data pelatihan dan sisanya untuk pengujian. Implementasi dari pengembangan model HMM ini dengan mengunakan bahasa pemrograman PHP untuk proses *parsing* dan *library* Matlab 7.0 dari Mathworks untuk proses pelatihan serta pengujiannya.

3.2 Pelatihan Baum-Welch

Salah satu algoritma yang populer dalam melakukan proses pelatihan pada HMMs adalah Baum-Welch. Proses pelatihan ini dilakukan untuk melakukan estimasi parameter model. Pada penelitian ini proses pelatihan dilakukan dengan menggunakan 2 hidden state dan 3 hidden state. Pada proses pelatihan ini iterasi maksimum yang dilakukan sebanyak 300 iterasi dengan toleransi kekonvergenan 0.001. Hasil dari proses pelatihan ini adalah matriks peluang transisi dan matriks peluang emisi. Proses inisialiasi matriks peluang transisi adalah dengan random dengan dengan jumlah baris sama dengan 1 sebagai peluang total perpindahan dari hidden state satu ke *hidden state*. Adapun inisialisasi matriks emisi adalah dengan memberikan nilai masing masing 0.25 yang mengasumsikan bahwa peluang setiap karakter 'A', 'C','G' dan 'U' memiliki peluang yang sama pada suatu state. Adapun matriks peluang prioritas diasumsikan bernilai 1 sebagai peluang pertama kali kemunculan state. Berikut ini adalah matriks peluang transisi dan matriks peluang emisi proses pelatihan dengan dua hidden state.

Kelas 1

$$A = \begin{bmatrix} 0.9646 & 0.0354 \\ 0.3250 & 0.6750 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.2427 & 0.2588 & 0.2812 & 0.2174 \\ 0.2453 & 0.2571 & 0.2835 & 0.2141 \end{bmatrix}$$

Kelas 2

$$A = \begin{bmatrix} 0.9307 & 0.0693 \\ 0.0148 & 0.9852 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.4149 & 0.0913 & 0.0297 & 0.4642 \\ 0.1791 & 0.2681 & 0.3089 & 0.2439 \end{bmatrix}$$

Kelas 3

$$A = \begin{bmatrix} 0.7921 & 0.2079 \\ 0.4440 & 0.5560 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0145 & 0.4581 & 0.3894 & 0.1380 \\ 0.4512 & 0.0308 & 0.3053 & 0.2127 \end{bmatrix}$$

Kelas 4

$$A = \begin{bmatrix} 0.7794 & 0.2206 \\ 0.4379 & 0.5621 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0527 & 0.3052 & 0.4404 & 0.2018 \\ 0.3916 & 0.4044 & 0.1947 & 0.0093 \end{bmatrix}$$

Kelas 5

$$A = \begin{bmatrix} 0.9131 & 0.0869 \\ 0.0714 & 0.9286 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.2608 & 0.3396 & 0.1607 & 0.2390 \\ 0.2249 & 0.1734 & 0.3212 & 0.2806 \end{bmatrix}$$

Kelas 6

$$A = \begin{bmatrix} 0.2433 & 0.1407 \\ 0.1951 & 0.8049 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.2489 & 0.1923 & 0.3020 & 0.2568 \\ 0.4770 & 0.2666 & 0.0131 & 0.2433 \end{bmatrix}$$

Kelas 7

$$A = \begin{bmatrix} 0.6531 & 0.3469 \\ 0.1591 & 0.8409 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0945 & 0.5288 & 0.0337 & 0.3431 \\ 0.2918 & 0.1524 & 0.3804 & 0.1754 \end{bmatrix}$$

Sedangkan matriks peluang transisi dan matriks peluang emisi proses pelatihan dengan tiga hidden state sebagai berikut:

Kelas 1

$$A = \begin{bmatrix} 0.7525 & 0.0653 & 0.1822 \\ 0.5980 & 0.3582 & 0.0438 \\ 0.0585 & 0.2007 & 0.7408 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.2413 & 0.2597 & 0.2797 & 0.2193 \\ 0.2447 & 0.2575 & 0.2833 & 0.2144 \\ 0.2442 & 0.2577 & 0.2829 & 0.2152 \end{bmatrix}$$

Kelas 2

$$A = \begin{bmatrix} 0.9208 & 0.0027 & 0.0766 \\ 0.1485 & 0.0000 & 0.85158 \\ 0.0000 & 0.1270 & 0.8730 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.4075 & 0.0865 & 0.0227 & 0.4833 \\ 0.8930 & 0.0000 & 0.1070 & 0.0000 \\ 0.0931 & 0.3017 & 0.3337 & 0.2715 \end{bmatrix}$$

Kelas 3

$$A = \begin{bmatrix} 0.7389 & 0.0233 & 0.2378 \\ 0.1756 & 0.3949 & 0.4295 \\ 0.0001 & 0.1591 & 0.8408 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0000 & 0.3827 & 0.6170 & 0.0003 \\ 0.0000 & 0.9195 & 0.0004 & 0.0800 \\ 0.2268 & 0.1505 & 0.4035 & 0.2192 \end{bmatrix}$$

Kelas 4

$$A = \begin{bmatrix} 0.7659 & 0.0060 & 0.2280 \\ 0.5411 & 0.4575 & 0.0015 \\ 0.0006 & 0.0717 & 0.9277 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0000 & 0.2177 & 0.5768 & 0.2055 \\ 0.0426 & 0.8459 & 0.0317 & 0.0798 \\ 0.2389 & 0.3120 & 0.3271 & 0.1219 \end{bmatrix}$$

Kelas 5

$$A = \begin{bmatrix} 0.4500 & 0.0045 & 0.5455 \\ 0.5107 & 0.2970 & 0.1923 \\ 0.2230 & 0.6590 & 0.1180 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.2871 & 0.6216 & 0.0162 & 0.0751 \\ 0.0043 & 0.0223 & 0.7274 & 0.2459 \\ 0.4062 & 0.0007 & 0.0833 & 0.5098 \end{bmatrix}$$

Kelas 6

$$A = \begin{bmatrix} 0.7111 & 0.0100 & 0.2789 \\ 0.2371 & 0.7456 & 0.0173 \\ 0.0134 & 0.1135 & 0.8732 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.2196 & 0.0524 & 0.2939 & 0.4341 \\ 0.2020 & 0.3549 & 0.4093 & 0.0338 \\ 0.4515 & 0.2509 & 0.0486 & 0.2489 \end{bmatrix}$$

Kelas 7

$$A = \begin{bmatrix} 0.5448 & 0.0215 & 0.4337 \\ 0.3425 & 0.3074 & 0.3501 \\ 0.3583 & 0.6256 & 0.0161 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.2503 & 0.5360 & 0.0141 & 0.1996 \\ 0.0009 & 0.0554 & 0.9345 & 0.0092 \\ 0.4131 & 0.0797 & 0.0277 & 0.4794 \end{bmatrix}$$

3.3 Skenario Pengujian

Pengujian dilakukan dengan melakukan perbandingan hasil akurasi dari setiap kelas dengan dua hidden state dan tiga hidden state. Secara umum pelatihan dengan tiga hidden state memiliki nilai akurasi yang lebih baik bila dibandingkan dengan dua hidden state.

3.4 Hasil Akurasi

Nilai akurasi dihitung dengan melakukan pengujian kepada data uji setiap kelas. Data uji suatu kelas akan dihitung peluangnya dan dibandingkan dengan setiap model. Sekuens dengan nilai peluang terbesar akan dikelompokkan menjadi kelasnya. Adapun hasil akurasi untuk kelas 1 sampai dengan kelas 7 dapat dilihat pada Tabel 1 sampai dengan Tabel 7

Tabel 2 Hasil Klasifikasi Data Uji Kelas 1

diidentifikasi sebagai kelas							# data	# state	akurasi
1	2	3	4	5	6	7			
865	24	0	0	267	24	70	1250	2	69,20
68	8	1	0	1002	1	170	1250	3	5,44

Tabel 3 Hasil Klasifikasi Data Uji Kelas 2

diidentifikasi sebagai kelas							# data	# state	akurasi
1	2	3	4	5	6	7			
9	99	1	0	21	1	11	142	2	69,72
12	100	1	0	7	0	22	142	3	70,42

Tabel 4 Hasil Klasifikasi Data Uji Kelas 3

diidentifikasi sebagai kelas							# data	# state	akurasi
1	2	3	4	5	6	7			
123	0	240	47	9	0	84	503	2	47,71
138	1	299	45	7	0	13	503	3	59,44

Tabel 6 Hasil Klasifikasi Data Uji Kelas 5

diidentifikasi sebagai kelas							# data	# state	akurasi
1	2	3	4	5	6	7			
52	13	0	17	143	40	49	314	2	45,54
30	9	5	8	145	18	100	314	3	55,81

Tabel 5 Hasil Klasifikasi Data Uji Kelas 4

diidentifikasi sebagai kelas							# data	# state	akurasi
1	2	3	4	5	6	7			
17	0	9	17	0	0	0	43	2	39,53
16	0	3	24	0	0	0	43	3	55,81

Tabel 6 Hasil Klasifikasi Data Uji Kelas 5

diidentifikasi sebagai kelas							# data	# state	akurasi
1	2	3	4	5	6	7			
52	13	0	17	143	40	49	314	2	45,54
30	9	5	8	145	18	100	314	3	55,81

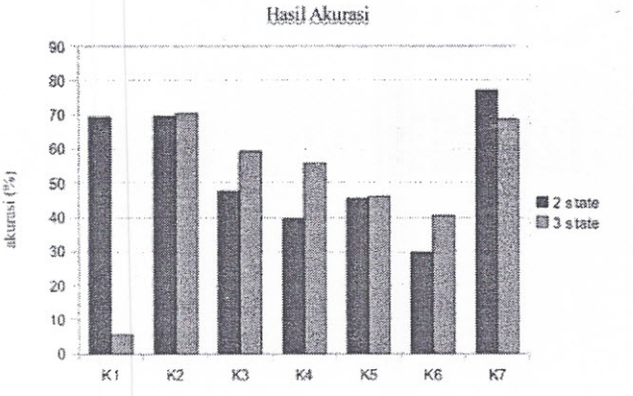
Tabel 7 Hasil Klasifikasi Data Uji Kelas 6

diidentifikasi sebagai kelas							# data	# state	akurasi
1	2	3	4	5	6	7			
13	47	14	10	21	55	25	185	2	29,72
26	44	8	18	11	75	3	185	3	40,54

Tabel 8 Hasil Klasifikasi Data Uji Kelas 7

diidentifikasi sebagai kelas							# data	# state	akurasi
1	2	3	4	5	6	7			
0	2	0	2	16	2	74	96	2	77,08
1	1	2	2	24	0	66	96	3	68,75

Perbandingan akurasi secara visual dapat dilihat pada Gambar 1



Gambar 1. Perbandingan tingkat akurasi dengan dua hidden state dan tiga hidden state

4. Kesimpulan

Paper ini menerapkan Hidden Markov Model dalam membuat model suatu *classifier*. Namun demikian, berdasar hasil yang didapatkan ternyata akurasinya masih sangat rendah. Itu terlihat dari nilai akurasi untuk setiap kelasnya yang secara rata-rata masih di bawah 60 persen pada penggunaan dua hidden state. Bahkan ada yang hanya mencapai 29.72 persen. Beberapa kasus juga didapati bahwa masih banyak data uji yang banyak terkelaskan di luar kelas yang sebenarnya.

Akurasi tertinggi yang dihasilksan pada klasifikasi ini terdapat pada identifikasi kelas 7 dengan rata-rata 77,08 persen pada penggunaan dua hidden state.

Penggunaan tiga hidden state secara umum bisa meningkatkan akurasi klasifikasi namun tidak terlalu signifikan. Kondisi kontradiksi justru terjadi pada klasifikasi kelas 1 yang mengalami penurunan akurasi sangat tajam.

REFERENSI

[1] S. Henikoff, E. A. Greene, S. Pietrokovski, P. Bork, T. K. Attwood, and L. Hood, "Gene families: The taxonomy of protein paralogs and chimeras," *Science*, vol. 278, pp. 609–614, 1997.

[2] Y. Karklin, "Classification of non coding rna using graph representations of secondary structure," 2004.

[3] S. R. Eddy, "Profile hidden markov models," *Bioinformatics*, vol. 14, pp.755–763, 1998.

[4] K.-J. Won, T. Hamelryck, A. Prugel-Bennett, and A. Krogh, "Evolutionary method for learning hmm structure:predictions of secondary structure," *BMC Bioinformatics*, 2007.

[5] J. Martin, J.-F. Gibrat, and F. Rodolphe, "Hidden markov model for protein secondary structure," 2005.

[6] P. Baldi and S. Brunak, Bioinformatics: The Machine Learning Approach, 2nd ed., . Second Edition. MIT Press, 2007, ch. 9.

[7] L. R. Rabiner, "A tutorial on hidden markov model and selected applications in speech recognitions," *IEEE*, 1989.

[8] G. A. Chruchill, "Stochastic model for heterogeneous DNA sequence," *Bull.Math*, vol. 51, pp. 79–94, 1979.

Toto Haryanto, memperoleh gelar Sarjana dari Departemen Ilmu Komputer Institut Pertanian Bogor pada tahun 2006. Saat ini sedang menempuh pendidikan Master Ilmu Komputer di Institut Pertanian Bogor.

Agus Buono, memperoleh gelar Insinyur dari Institut Pertanian Bogor pada tahun 1992. Gelar Master dan Doktor di bidang Ilmu Komputer diperoleh dari Universitas Indonesia pada tahun 2000 dan tahun 2009. Penulis juga memperoleh gelar Master Statistika dari Institut Pertanian Bogor pada tahun 1997. Saat ini sebagai staf pengajar di Departemen Ilmu Komputer FMIPA IPB.

Taufik Djatna, memperoleh gelar Sarjana dan Master dari Institut Pertanian Bogor memperoleh gelar Doktor dari Universitas Hirosima, Jepang pada tahun 2008. Saat ini sebagai staf pengajar di Departemen Teknologi Industri Pertanian IPB.