PENERAPAN HIDDEN MARKOV MODEL DALAM CLUSTERING SEQUENCE PROTEIN GLOBIN

Sri Mulyana [1)] , Afiahayati [2)], Wijaya Adhi Surya [3)]
Program Studi Ilmu Komputer, FMIPA, Universitas Gadjah Mada[1,2,3)]
Gedung Selatan FMIPA UGM, Sekip Unit III, Bulaksumur, Yogyakarta
smulyana@ugm.ac.id [1)], af1a@mail.ugm.ac.id [2)], fauzan.sa@gmail.com [3)]

Abstrak

Mudah dan murahnya proses pengumpulan data biologi molekuler saat ini menyebabkan ukuran basis data genetika meningkat dengan pesat. Hal ini meningkatkan kebutuhan akan alat bantu komputasi untuk menganalisa data tersebut. Salah satu task dasar dalam menganalisa data biologi molekuler adalah pengelompokkan dari kumpulan sequence protein.

Metode komputasi yang banyak dikaji dalam bioinformatika saat ini adalah hidden markov model (HMM). HMM menggunakan algoritma pembelajaran Baum-Welch untuk mengestimasi parameter – parameter untuk menemukan model terbaik yang mendeskripsikan kumpulan sequence (training set).

Telah dilakukan penelitian penerapan metode HMM dalam melakukan pengelompokkan (clustering) dari kumpulan sequence protein globin. Protein globin merupakan protein yang terkandung di dalam darah. Sequence protein globin yang digunakan dalam penelitian ini berasal dari basisdata UNIPROT. Sistem yang dibangun dalam penelitian ini memanfaatkan library biojava. Hasil eksperimen menunjukkan bahwa metode HMM dapat digunakan untuk melakukan pengelompokkan (clustering) sequence protein.

Kata Kunci : Hidden Markov Model, Clustering, Sequence Protein Globin,UNIPROT.

PENDAHULUAN

Jumlah data biologi molekuler yang semakin meningkat *pasca genome project* membutuhkan pengelompokkan data ke dalam suatu kelompok subfamili berdasarkan tingkat kesamaan data tersebut. Pengelompokan dan penentuan subfamili dari kumpulan *sequence* protein merupakan salah satu *task* penting dalam biologi. *Task* ini hampir dikatakan tidak bisa dilakukan secara manual sehingga membutuhkan alat bantu komputasi.

Hidden Markov Model merupakan salah satu metode komputasi yang dapat digunakan untuk mengenali pola *sequence* protein yang dapat dikembangkan untuk mengelompokkan dan menentukan subfamili suatu *sequence* protein.

Pada penelitian ini telah diterapkan, dirancang dan diimplementasikan metode *hidden markov model* untuk melakukan pengelompokkan (*clustering*) dan penentuan subfamili dari kumpulan *sequence* protein. *Sequence* protein yang dikelompokkan adalah *sequence* protein globin yang berasal dari basisdata UNIPROT.

Tujuan utama yang ingin dicapai dalam penelitian ini adalah mengimplementasikan dan membangun prototipe perangkat lunak yang menggunakan *Hidden Markov Model* untuk mengelompokkan dan menentukan subfamili dari kumpulan *sequence* protein globin sehingga dapat digunakan oleh praktisi biomolekuler.

Penelitian yang membahas *Hidden Markov Model* cukup banyak. Rabiner (1989) memperkenalkan penerapan metode *Hidden Markov Model* dalam *speech recognition*. *Pasca genome project*, penelitian *hidden markov model* dalam menganalisis data biomolekuler semakin dikembangkan.

Krogh, et al. (1992) memperkenalkan untuk pertama kali, penerapan *hidden markov model* dalam bioinformatika. Karena dinilai metode *hidden markov model* merupakan metode yang cukup baik diterapkan dalam bioinformatika maka metode tersebut dikembangkan terus sampai saat ini. Pada tahun 1996, Krogh dan Hughey melakukan penelitian penerapan *hidden markov model* dalam *analysis sequence*. Gupta (2004) melakukan penelitian tentang percepatan hardware untuk penerapan *hidden markov model* pada aplikasi bioinformatika. Afiahayati (2008) melakukan penelitian tentang *Multiple Sequence Alignment* menggunakan *Hidden Markov Model.*

METODOLOGI PENELITIAN

Metodologi yang digunakan dalam penelitian ini adalah sebagai berikut:

- Studi Literatur dan Pustaka
  Studi pustaka dan literatur terhadap Hidden Markov Model dan protein globin, macam – macam protein globin.
- Analisa sistem
  Kegiatan analisa sistem meliputi analisa spesifikasi sistem, analisa fungsionalitas, dan analisa kelas yang dibutuhkan.
- Perancangan Sistem
  Perancangan sistem meliputi perancangan kelas dan perancangan antarmuka.
- Implementasi Sistem
  Implementasi sistem menggunakan bahasa pemrograman Java dan library Biojava.
- Pengujian Sistem
  Pengujian dilakukan untuk menemukan dan memperbaiki bug-bug yang ada dan menguji kinerja sistem dalam melakukan clustering protein globin.

DASAR TEORI

Hidden Markov Model

Sebuah HMM menggabungkan dua atau lebih rantai Markov dengan hanya satu rantai yang terdiri dari state yang dapat diobservasi dan rantai lainnya membentuk state yang tidak dapat diobservasi (hidden), yang mempengaruhi hasil dari state yang dapat diobservasi. Probabilitas dari satu state ke state lainnya dinamakan transition probability. Setiap state mungkin dibentuk oleh sejumlah elemen atau simbol. Untuk sequence asam amino, terdapat dua puluh buah simbol. Nilai probabilitas yang berasosiasi dengan setiap simbol dalam setiap state disebut emission probability. Untuk menghitung probabilitas total dari suatu jalur dalam model, baik transition probability maupun emission probability yang menghubungkan semua hidden state dan state yang dapat diobservasi harus dimasukkan dalam perhitungan (Gupta, 2004).

Sebuah Hidden Markov Model dikarakteristikkan dengan parameter berikut (Rabiner, 1989):

- N, jumlah state dalam model.
- M, jumlah simbol pengamatan yang dimiliki setiap state.
- A = {aij}, aij = P(qt+1 = Sj | qt = Si), himpunan distribusi kemungkinan perpindahan state (transition probability) .
- B = {bj(k)}, bj(k) = P(vk pada t | qt = Sj), himpunan distribusi kemungkinan simbol pengamatan pada state j (emission probability).
- π = {πi}, πi = P(q1 = Si), himpunan distribusi kemungkinan state awal.

Bentuk ringkas dari HMM adalah

$$\lambda = (A, B, \pi) \quad (1)$$

Persoalan Utama dalam HMM

Terdapat tiga persoalan utama yang harus dipecahkan agar HMM dapat digunakan dalam suatu aplikasi nyata. Persoalan tersebut adalah (Rabiner, 1989):

1. Diberikan model λ = (A, B, π), bagaimana menghitung P(O | λ), yaitu kemungkinan ditemuinya rangkaian pengamatan O = O1, O2, ..., OT.
2. Diberikan model λ = (A, B, π), bagaimana memilih rangkaian state I = i1, i2,...,iT sehingga P(O, I | λ), kemungkinan gabungan rangkaian pengamatan O = O1, O2, ..., OT dan rangkaian state jika diberikan model, maksimal.
3. Bagaimana mengubah parameter HMM, λ = (A, B, π) sehingga P(O | λ) maksimal.

Persoalan 1 dan 2 dapat dilihat sebagai persoalan analisis sedangkan persoalan 3 sebagai persoalan sintesis (atau disebut identifikasi model atau pelatihan).

Solusi Persoalan 1

Persoalan 1 dapat diselesaikan dengan menggunakan algoritma yang dinamakan prosedur maju-mundur (forward-backward procedure) (Rabiner, 1989). Pertama, dijelaskan prosedur forward, diasumsikan variabel maju αt(i) didefinisikan sebagai:

αt(i) = P(O1,O2,...,Ot, qt = Si | λ) (2.2)

yaitu kemungkinan rangkaian pengamatan parsial hingga waktu t dan berada pada state Si pada waktu t, jika diberikan model λ. Maka αt(i) dapat dihitung dengan induksi sebagai berikut:

- Inisialisasi

$$\alpha t(i) = \pi i b i(O1),\ 1 \le i \le N \quad (2)$$

- Induksi

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] b_j(O_{t+1}) \qquad 1 \le t \le T-1 \qquad 1 \le j \le N \quad (4)$$

- Terminasi

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \quad (5)$$

Langkah pertama merupakan inisialisasi, langkah induksi merupakan langkah yang paling utama pada prosedur forward. P(O|λ) dapat dicari dengan menjumlahkan variabel maju dengan t=T dari semua state. P(O|λ) merupakan probabilitas model menghasilkan rangkaian pengamatan O = O1, O2, ..., OT.

Dengan cara yang sejenis dapat didefinisikan variabel mundur βt(i) sebagai berikut:

$$\beta t(i) = P(Ot+1, Ot+2, ..., OT \mid qt = Si, \lambda) \quad (6)$$

yaitu kemungkinan rangkaian pengamatan dari t+1 hingga T jika diberikan state Si pada waktu t dan model λ. βt(i) dapat diselesaikan sebagaimana αt(i) .

- Inisialisasi

$$\beta T(i) = 1,\ 1 \le i \le N \quad (7)$$

- Induksi

$$\beta_t(i)=\sum_{j=1}^{N} a_{ij} b_j(O_{t+1})\beta_{t+1}(j) \quad t=T-1,T-2,\ldots,1 \quad 1\leq i\leq N \tag{8}$$

Variabel mundur akan digunakan pada persoalan 3, melakukan estimasi nilai parameter – parameter HMM.

Solusi Persoalan 2

Pada persoalan ini akan dicari rangkaian state I = i1, i2, ..., iT sedemikian hingga kemungkinan kemunculan rangkaian pengamatan O=O1,O2,...,OT menjadi maksimal. Atau dengan kata lain mencari I yang memaksimalkan P(O, I | λ). Salah satu solusi penyelesaian persoalan ini adalah dengan menggunakan algoritma Viterbi (Rabiner, 1989). Pertama-tama didefinisikan:

$$\delta_{t+1}(j)=\max_{q_1 q_2 \ldots q_{t-1}} P[q_1 q_2 \ldots q_t=i, O_1 O_2 \ldots O_t|\lambda] \tag{9}$$

sebagai probabilitas tertinggi pada waktu t dan berakhir di state Si.

Algoritma Viterbi sebagai berikut :

- Inisialisasi

$$\delta_1(i)=\pi_i b_i(O_1) \quad \psi_1(i)=0 \quad 1\leq i\leq N \tag{10}$$

- Rekursi

$$\delta_t(j)=\max_{1\leq i\leq N}[\delta_{t-1}(i)a_{ij}]b_j(O_t) \quad 2\leq t\leq T \quad 1\leq j\leq N \tag{11}$$

$$\Psi_t(j)=\arg\max_{1\leq i\leq N}[\delta_{t-1}(i)a_{ij}] \quad 2\leq t\leq T \quad 1\leq j\leq N \tag{12}$$

- Terminasi

$$P^*=\max_{1\leq i\leq N}[\delta_T(i)] \tag{13}$$

- Penelusuran balik rangkaian state yang optimal

$$\vec{q_t}=\psi_{t+1}(\vec{q_{t+1}}) \quad t=T-1,T-2,\ldots,1 \tag{14}$$

Solusi Persoalan 3

Persoalan 3 merupakan persoalan yang paling sulit, bisa dikatakan sebagai persoalan pelatihan yang digunakan untuk menghasilkan parameter model A, B, dan π optimal sehingga dapat dengan baik merepresentasikan rangkaian observasi yang terjadi. Kriteria optimal adalah memaksimalkan probabilitas rangkaian pengamatan, P(O | λ), dengan diberikan model, λ(A, B, π). Tidak ada pendekatan analitik untuk permasalahan ini, namun terdapat prosedur iteratif seperti metode Baum-Welch yang dapat digunakan (Rabiner, 1989). Untuk mendeskripsikan formula pelatihan secara matematis, diasumsikan ξt(i, j) sebagai probabilitas berada pada state i pada waktu t, dan state j pada waktu t+1, diberikan model dan rangkaian pengamatan:

$$\xi t\ (i, j) = P(qt = Si\ ,qt+1 = Sj \mid O,\lambda) \tag{16}$$

Dengan menggunakan definisi variabel maju dan mundur, persamaan di atas dapat ditulis dalam bentuk:

$$\xi_t(i,j)=\frac{(\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j))}{(P(O|\lambda))} \tag{17}$$

$$\xi_t(i,j)=\frac{(\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j))}{(\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j))} \tag{18}$$

P(O|λ) merupakan nilai probabilitas model λ memberikan sequence O. Biasanya P(O|λ) sering diberi nilai 1, nilai harapan model λ memberikan sequence O. Menjumlahkan ξt (i, j) pada $1 \leq t \leq T - 1$ menghasilkan jumlah perpindahan dari i ke j yang diharapkan. Untuk kebutuhan pelatihan, didefinisikan probabilitas berada di state i pada waktu t, diberikan rangkaian pengamatan dan model sebagai γt(i)

$$\gamma_t(i)=\frac{(\alpha_t(i)\beta_t(i))}{(P(O|\lambda))}=\frac{(\alpha_t(i)\beta_t(i))}{(\sum_{i=1}^{N}\alpha_t(i)\beta_t(i))} \tag{19}$$

Selanjutnya, γt(i) dan ξt (i, j) dapat dihubungkan dengan menjumlahkan ξt (i, j) untuk semua j:

$$\gamma_t(i)=\sum_{j=1}^{N}\xi_t(i,j) \tag{20}$$

Menjumlahkan , γt(i) sepanjang waktu memberikan jumlah state i dikunjungi. Jika waktu T tidak dimasukkan, dengan kata lain menjumlahkan sepanjang $1 \leq t \leq T - 1$, ini memberikan jumlah perpindahan dari state i.

$$\sum_{t=1}^{T-1}\gamma_t(i) \quad = \text{jumlah perpindahan yang diharapkan dari } S_i \tag{21}$$

$$\sum_{j=1}^{N}\xi_t(i,j) \quad = \text{jumlah perpindahan yang diharapkan dari } S_i \text{ ke } S_j \tag{22}$$

Menggunakan formula di atas, dapat didefinisikan formula untuk melakukan estimasi terhadap nilai – nilai parameter HMM

$$\overline{a_{ij}}=\frac{(\textit{jumlah perpindahan yang diharapkan dari Si ke Sj})}{(\textit{jumlah perpindahan yang diharapkan dari Si})}$$

$$\overline{a_{ij}}=\frac{(\sum_{t=1}^{N}\xi_t(i,j))}{(\sum_{t=1}^{T-1}\gamma_t(i))} \tag{23}$$

$$\overline{b_j(k)}=\frac{(\textit{jumlah frekuensi yang diharapkan pada state j dan menghasilkan simbol Vk})}{(\textit{jumlah frekuensi pada state j})}$$

$$\overline{b_j(k)}=\frac{(\sum_{\substack{t=1\\ s.t.O_t=v_k}}^{T-1}\gamma_t(i))}{(\sum_{t=1}^{T-1}\gamma_t(i))} \tag{24}$$

Algoritma Baum Welch

Parameter-parameter pada HMM seperti transition probability dan distribusi asam amino (emission probability) dapat dipilih secara manual berdasarkan alignment sequence protein yang ada, atau dari informasi struktur tiga dimensi protein. Parameter – parameter HMM diestimasi untuk menemukan model

yang paling baik dalam mendeskripsikan suatu kumpulan sequence(data training).
Pada penelitian ini, parameter-parameter pada HMM akan ditentukan secara otomatis melalui proses pembelajaran data sequence protein yang belum ter-align menggunakan algoritma Baum-Welch. Inti dari pendekatan ini adalah menemukan model yang paling baik untuk mendeskripsikan suatu kumpulan sequence (training set).
Langkah – langkah dalam prosedur forward – backward untuk mengestimasi parameter-parameter dari suatu HMM adalah sebagai berikut (Krogh et al, 1992):

- Menentukan model awal, menentukan panjang model, menentukan struktur model, memberikan suatu nilai awal (acak) untuk setiap *transition probability* dan *emission probability* dari HMM.
- Menghitung probabilitas maju (*forward probability*) setiap *state* dari semua kemungkinan jalur perpindahan antar-*state* dengan menggunakan prosedur *forward*.
- Menghitung probabilitas mundur (*backward probability*) setiap *state* dari semua kemungkinan jalur perpindahan antar-*state* dengan menggunakan prosedur *backward*.
- Untuk mengestimasi *transition probability* dari *n* buah jalur yang berasal dari suatu *state*, gunakan persamaan 2.17 untuk menghitung dari masing-masing jalur tersebut untuk setiap kemungkinan jalur perpindahan antar-*state* pada langkah 2. Untuk mengestimasi *transition probability* dari sebuah jalur, gunakan persamaan 2.23 dengan pembilang berupa penjumlahandari jalur yang sama (yang sedang diestimasi) dan pembagi berupa penjumlahan dari *n* buah jalur tersebut untuk setiap kemungkinan jalur perpindahan antar-*state* pada langkah 2.
- Untuk mengestimasi *emission probability* dari suatu *state* tertentu, gunakan persamaan 2.19 untuk menghitung $\gamma_t(i)$ dari setiap simbol yang dapat dihasilkan oleh *state* tersebut. Untuk mengestimasi *emission probability* dari sebuah simbol, gunakan persamaan 2.24 dengan pembilang berupa penjumlahan $\gamma_t(i)$ untuk simbol yang sama (yang sedang diestimasi) dan pembagi berupa penjumlahan seluruh $\gamma_t(i)$ dari *state* yang bersangkutan.
- Memperbaharui *transition probability* untuk setiap jalur perpindahan dan *emission probability* untuk setiap *state* dari HMM dengan menggunakan estimasi *transition probability* dan *emission probability* yang diperoleh dari langkah 4 dan 5.
- Melakukan iterasi langkah 2 sampai 6 sedemikian sehingga parameter-parameter dari HMM konvergen (berubah secara tidak signifikan).

HASIL DAN PERANCANGAN

Secara singkat algoritma dalam pengelompokkan (clustering) protein globin menggunakan HMM dapat dijabarkan sebagai berikut:

1. Menentukan jumlah maksimal subfamili atau jumlah maksimal *cluster* (n).
2. Membangun arsitektur awal HMM sebanyak jumlah maksimal *cluster* (n), dengan panjang HMM sama dengan rata – rata panjang sequence protein.
3. Melakukan proses pembelajaran Baum Welch untuk setiap arsitektur HMM dan untuk setiap *sequence* protein yang akan dikelompokkan.
4. Menghitung NLL *score* untuk setiap *sequence* protein terhadap arsitektur HMM hasil pembelajaran. NLL *score* adalah nilai negatif dari nilai HMM menghasilkan *sequence* tersebut atau dapat dikatakan NLL score merupakan nilai negatif dari nilai yang dihasilkan oleh persamaan 2.5.
5. Suatu *sequence* protein dikelompokkan ke dalam suatu *cluster* tertentu berdasarkan NLL *score* terendah terhadap salah satu arsitektur HMM tertentu. Misalkan *Sequence* 1 memiliki NLL *score* terendah terhadap arsiterktur HMM 3, maka *sequence* 1 termasuk ke dalam *cluster* 3.

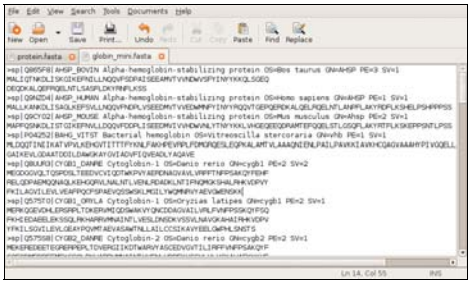
Sistem ini dibangun menggunakan library biojava dan bahasa pemrograman java. Input sistem berupa file berformat fasta (*.fasta) yang diambil dari *database* UNIPROT. *Sequence* protein pada file dari *database* UNIPROT sudah diidentifikasi jenis protein dan subfamily dari protein tersebut. File fasta dari database UNIPROT (uniprot_sprot.fasta) dengan ukuran 176.8 MB, berisi tidak hanya *sequence* protein globin. Sistem ini akan diuji menggunakan protein globin, maka dipilih *sequence* protein globin dari file uniprot_sprot.fasta dan disimpan di file globin.fasta. File globin.fasta berisi lebih dari seribu sequence protein globin. Karena metode *Hidden Markov Model* memerlukan biaya yang cukup mahal, maka untuk menguji sistem ini dipilih secara random 30 *sequence* dari file globin.fasta dan disimpan dalam file globin_mini.fasta. Tiga puluh *sequence* tersebut akan di-*cluster*. *Sequence* protein pada file dari database UNIPROT sudah diidentifikasi jenis protein dan *subfamily* dari protein tersebut. File *input* dapat dilihat pada Gambar 1. Dari file tersebut, misal untuk *sequence* protein pertama

Tabel 1. Sequence Protein Globin

```
>sp|Q865F8|AHSP_BOVIN      Alpha-
hemoglobin-stabilizing    protein
OS=Bos taurus GN=AHSP PE=3 SV=1

MALIQTNKDLISKGIKEFNILLNQQVFSDPAI
SEEAMVTVVNDWVSFYINYYKKQLSGEQ
DEQDKALQEFRQELNTLSASFLDKYRNFLKSS
```

Pada *sequence* protein tersebut, 3 baris utama merupakan keterangan *sequence* protein, pada sequence protein di atas adalah *sequence* Alpha-hemoglobin, sedangkan 3 baris selanjutnya merupakan *sequence* protein yang berisi *sequence* atau urutan asam amino.

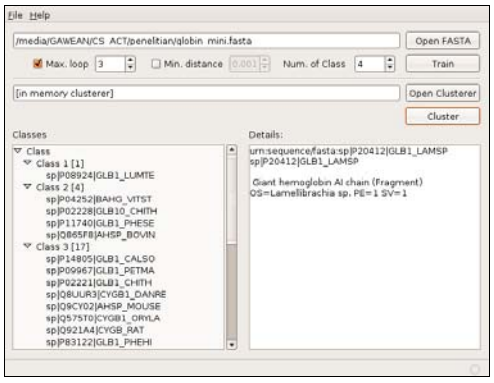


Gambar 1. globin_mini.fasta

Sesuai dengan rancangan algoritma di atas, sistem ini membutuhkan input file fasta, jumlah maksimal *cluster*, dan *stopping criteria*. *Stopping criteria* adalah parameter berhenti untuk melakukan proses pembelajaran Baum Welch bagi HMM yang dibangun. Terdapat dua parameter berhenti yang disediakan yaitu jumlah maksimal *loop* dan *maximal distance* (jarak maksimal). Jarak didapatkan dengan hasil selisih nilai Baum Welch pada iterasi saat ini dikurangi nilai Baum Welch pada iterasi sebelumnya.

Sistem ini juga menyediakan fasilitas untuk menyimpan aristektur HMM yang telah di*training*, yang telah dilakukan proses pembelajaran Baum Welch. Jika arsitektur telah disimpan, maka user dapat menggunakan arsitektur tersbut untuk meng-*cluster* file *sequence* protein globin lainnya.

Untuk pengujian sistem, seperti diterangkan sebelumnya, menggunakan file globin_mini.fasta, dengan parameter maksimal *cluster* sejumlah 4, dan menggunakan *stopping criteria* maksimal loop sebanyak 3. Hasil arsitektur HMM yang telah dikenai pembelajaran disimpan dengan nama file ArsiHMM.psc Hasil *clustering* dapat dilihat pada Gambar 2.



Gambar 2. Hasil CLustering

Tiga puluh *sequence* pada globin_mini.fasta di *cluster* menjadi 4 class.

Tabel 2. Hasil *Clustering*

Class	Jumlah (sequence protein)	Komposisi SubFamily
Class 1	1	1 globin
Class 2	4	1 alpha, 3 globin
Class 3	17	8 globin, 7 cytoglobin, 2 alpha
Class 4	8	2 hemoglobin, 6 globin

Jika dilihat dari hasil *clustering* di atas, sistem ini belum bisa meng-*cluster* secara optimal, tetapi sudah bisa memisahkan *subfamily* hemoglobin dan alpha walaupun masih bercampur dengan sequence protein *subfamily* globin.

Hasil *cluster* yang belum optimal disebabkan arsitektur HMM yang juga belum optimal. Pada sistem ini, belum dilakukan pengoptimalan HMM, belum memperhatikan dan mengatasi masalah pada prosedur *training* Baum-Welch, seperti *local minima* dan *overfitting.*

KESIMPULAN

Berdasarkan hasil penelitian ini, *Hidden Markov Model* dapat digunakan sebagai salah satu metode alternatif dalam melakukan *clustering sequence* protein globin dan *sequence* protein lainnya. Pada penelitian ini, *clustering sequence* protein globin belum optimal, disebabkan arsitektur HMM belum optimal, belum memperhatikan dan mengatasi masalah pada prosedur training Baum-Welch, seperti *local minima* dan *overfitting*.

Clustering sequence protein dapat dijadikan sebagai salah satu cara untuk membangun pohon *pilogenetik* secara *top – down*. Pada penelitian selanjutnya, diharapkan dapat mengatasi masalah pada pengoptimalan HMM sehingga hasil *clustering sequence* protein menjadi optimal dan dapat dibangun aplikasi pembangun pohon *pilogenetik* dari *sequence* protein yang tidak hanya *sequence* protein globin.

DAFTAR PUSTAKA

[1] Afiahayati, 2008, Multiple Sequence Alignment Menggunakan Hidden Markov Model, Skripsi S1, UGM.

[2] Birney, E., 2001, Hidden Markov Models in Biological Sequence Analysis, Volume 45, IBM Journal of Research and Development.

[3] Booch,G., Rumbaugh,J., Jacobson,I., 2005, The Unified Modeling Language

[4] User Guide. Addison Wesley Professional.

[5] Colton, S., 2007, Introduction to Bioinformatics, Genetics Background, Course 341 Lecture Slide.

Department of Computing Imperial College, London.

[6] Gupta, S., 2004, Hardware Acceleration of Hidden Markov Models for Bioinformatics Applications, Boise State University.

[7] Hughey, R., Krogh, A. , 1996, Hidden Markov Models for Sequence Analysis : Extension and Analysis of The Basic Method, University of California, Santa Cruz.

[8] Koski, T., 2001, Hidden Markov Models for Bioinformatics, Kluwer Academic Publishers,Netherlands.

[9] Krogh, A., Brown, M., Mian, I.S., Sjölander, K., Haussler, D., 1992, Hidden Markov Models in Computational Biology : Application to Protein Modeling, University of California, Santa Cruz.

[10] Rabiner, L.R., 1989, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proceedings of the IEEE, Vol. 77, No. 2, pp. 257-286.

[11] Schmidt,J.W., Matthes,F.,Niederée,C., 1999, Object-Oriented Analysis and Design Course Lecture Slide, TU Hamburg, Harburg.

[12] Thompson, J.D., Frederic, P., Olivier, P., 1999, A Comprehensive Comparison of Multiple Sequence Alignment Programs, Nucleic Acid Research, Vol. 27, No. 13

[13] Xiong, J., 2006. Essential Bioinformatics., Cambridge University Press, Cambridge