

# Algoritma Viterbi dalam Metode Hidden Markov Models pada Teknologi Speech Recognition

Angela Irfani<sup>1</sup>, Ratih Amelia<sup>2</sup>, Dyah Saptanti P<sup>3</sup>

Laboratorium Ilmu dan Rekayasa Komputasi  
Departemen Teknik Informatika, Institut Teknologi Bandung  
Jl. Ganेशha 10, Bandung

E-mail : [if14011@students.if.itb.ac.id](mailto:if14011@students.if.itb.ac.id)<sup>1</sup>, [if14012@students.if.itb.ac.id](mailto:if14012@students.if.itb.ac.id)<sup>2</sup>,  
[if14017@students.if.itb.ac.id](mailto:if14017@students.if.itb.ac.id)<sup>3</sup>

## Abstrak

Perkembangan teknologi khususnya di bidang teknologi informasi yang semakin cepat telah memungkinkan segala sesuatu terjadi dengan perangkat-perangkat yang semakin canggih. Perangkat-perangkat ini menjadi sangat membantu dalam kehidupan manusia sekarang ini namun, tidak dapat dipungkiri segala sesuatu memiliki dampak positif dan negatif. Perkembangan teknologi speech recognition adalah salah satu bentuk perkembangan teknologi di abad ke 20 yang memanfaatkan suara sebagai masukan, suara tersebut merupakan metode alternatif bagi manusia untuk berinteraksi dengan komputer. Komputer akan mengenali suara tersebut sebagai perintah dan melakukan reaksi terhadap perintah tersebut. Teknologi speech recognition yang dikembangkan menggunakan algoritma viterbi sebagai implementasi dynamic programming. Sistem pengenalan suara modern secara umum berdasarkan pada Hidden Markov Models (HMMs). Dari HMMs diperoleh sinyal suara yang dapat dikarakteristikan sebagai proses parameter acak, serta parameter dari proses stokastik yang dapat ditentukan dengan tepat. Model statistik ini kemudian diolah dengan menggunakan algoritma Viterbi. Algoritma Viterbi adalah algoritma *dynamic programming* untuk menemukan kemungkinan rangkaian status yang tersembunyi (biasa disebut Viterbi path) yang dihasilkan pada rangkaian pengamatan kejadian, terutama dalam lingkup HMM. Dengan algoritma viterbi maka pengolahan status-status dalam sistem pengenalan suara dapat lebih optimal.

**Kata kunci:** teknologi, algoritma, dynamic programming

## 1. Pendahuluan

Teknologi pengenalan suara adalah teknologi yang menggunakan peralatan dengan sumber masukannya adalah suara, seperti mikrofon untuk menginterpretasikan suara manusia untuk transkripsi atau sebagai metode alternatif interaksi dengan komputer. Teknologi pengenalan suara tidak sama dengan teknologi *voice recognition* yang hanya mengenali suara sebagai identifikasi keamanan. Sistem komersial untuk pengenalan suara telah ada sejak 1990. Walaupun kesuksesan teknologi ini nyata, hanya sedikit orang yang menggunakan sistem pengenalan suara pada komputer. Hal yang muncul pada kebanyakan pengguna komputer dalam membuat dan mengedit dokumen serta berinteraksi dengan komputernya lebih cepat dan nyaman dengan menggunakan peralatan-peralatan masukan konvensional yaitu keyboard dan mouse, walaupun secara fakta dengan menggunakan teknologi pengenalan suara memungkinkan pengguna untuk berbicara secara langsung dan cepat serta efisien daripada harus mengetikkan suatu perintah dengan menggunakan *keyboard*.

Suatu lingkungan perkantoran dengan tingkat kebisingan yang tinggi merupakan salah satu lingkungan yang merugikan untuk teknologi

pengenalan suara karena dengan begitu suara yang terdengar pada sistem tidak jelas sehingga sistem pengenalan suara tidak dapat bekerja dengan akurat. Pengenalan suara hanya dapat diterima dengan sistem speaker yang independent 80%-90% untuk lingkungan yang nyaman dan tidak banyak suara. Sistem pengenalan suara telah menemukan hal tentang kecepatan masukan teks yang diminta agar dapat bertambah cepat. Sistem pengenalan suara dapat membantu orang-orang yang mengalami kesulitan berinteraksi dengan komputer melalui keyboard contohnya orang yang memiliki *carpal tunnel syndrome*, serta orang-orang yang memiliki cacat fisik.

Teknologi pengenalan suara lebih banyak digunakan untuk aplikasi telepon seperti pemesanan dan informasi travel, informasi laporan keuangan, rute pelayanan pelanggan melalui telepon, dsb. Riset dan pengembangan di dalam teknologi pengenalan suara telah tumbuh dan berkembang juga meningkatkan kegunaan dan efisiensi dari teknologi ini. Lebih jauh lagi, pengenalan suara dapat memunculkan otomatisasi dari aplikasi tertentu yang bukan merupakan aplikasi non-otomata yang menggunakan tombol sistem IVR (Interactive Voice Response). Pengenalan suara juga digunakan di

dalam instruksi bahasa dan evaluasi kelancaran suara.

## 2. Ruang Lingkup

Makalah ini membahas tentang:

1. Markov Models
2. Hidden Markov Models
3. Algoritma Viterbi

## 3. Pembahasan

### 3.1. Markov Models

Misalkan sebuah sistem dideskripsikan pada setiap waktu sebagai salah satu dari  $N$  status  $\{1, 2, \dots, N\}$ . Sistem mungkin mengalami perubahan status dalam satuan waktu diskrit (mungkin kembali ke status semula) tergantung kepada aturan probabilitas setiap status. Misalkan waktu disimbolkan sebagai  $t$ , status pada setiap waktu disimbolkan sebagai  $q_t$ . Deskripsi keseluruhan probabilitas dari sistem secara umum membutuhkan spesifikasi status saat ini dan status-status sebelumnya. Untuk kasus khusus dengan spesifikasi waktu diskrit, *first order*, dan *Markov chain*, dapat rumus ketergantungan probabilitas suatu status terhadap status-status sebelumnya disederhanakan hanya tergantung pada tepat satu status sebelumnya, yaitu,

$$P[q_t = j | q_{t-1} = i, q_{t-2} = k, \dots] = P[q_t = j | q_{t-1} = i]$$

Penyederhanaan lebih lanjut dengan hanya memperhitungkan proses di bagian kanan, bebas terhadap waktu, maka pendekatan aturan probabilitas transisi status

$$a_{ij} = P[q_t = j | q_{t-1} = i]$$

dengan sifat

$$a_{ij} \geq 0, \text{ untuk setiap } j, i$$

$$\sum a_{ij} = 1, \text{ untuk setiap } i, i=1 \text{ sampai } N$$

Karena mematuhi batasan *stochastic* standar.

Contoh permasalahan:

Cuaca dalam satu hari dimodelkan ke dalam tiga status: rainy (1), cloudy (2) dan sunny (3). Aturan probabilitas dari setiap transisi status dideskripsikan sebagai berikut

$$A = \{a_{ij}\} = \begin{matrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{matrix}$$

Probailitas cuaca untuk 8 hari berturut-turut "sun-sun-sun-rain-rain-sun-cludy-sun" adalah

Misalkan  $O$  adalah tahap pengamatan

$$O = \{\text{sunny, sunny, sunny, rainy, rainy, sunny, cloudy, sunny}\} \\ = \{3, 3, 3, 1, 1, 3, 2, 3\}$$

$$P(O|Model) = P[3, 3, 3, 1, 1, 3, 2, 3 | Model] \\ = P[3] P[3]^2 P[1] P[1] P[3] P[2] P[3] \\ = \Pi_3 \cdot (a_{33})^2 a_{31} a_{11} a_{13} a_{32} a_{23} \\ = (1.0)(0.8)^2(0.1)(0.4)(0.3) \\ (0.1)(0.2) \\ = 1.536 \times 10^{-4}$$

Dengan,

$$\Pi_3 = P[q_t = i], 1 \leq i \leq N$$

### 3.2. Hidden Markov Models

Pada pembahasan di atas Markov Model diimplementasikan pada kejadian yang bisa diobservasi dan keluaran dari setiap status tidak acak. Model tersebut terlalu terbatas untuk diaplikasikan pada permasalahan yang lebih beragam. Oleh karena itu, konsep Markov Model diperluas menjadi Hidden Markov Model dimana obeservasi merupakan fungsi probabilitas dari status. Model ini dapat diaplikasikan pada kasus yang proses tidak dapat diobservasi secara langsung (tersembunyi) tetapi bisa diobservasi hanya melalui kumpulan proses stokastik yang lain yang menghasilkan tahapan observasi, misalnya pada persolana *coin-toss* (pelemparan koin) dan *speech recognition* (pengenalan suara).

### 3.3. Pengenalan Suara Dengan Hidden Markov Models

Sistem pengenalan suara modern secara umum berdasarkan pada Hidden Markov Models (HMMs). Hidden Markov Models merupakan model statistik dimana mempunyai keluaran rangkaian simbol dan kuantitas. Dengan memiliki sebuah model yang memberikan kemungkinan dari rangkaian akustik data yang telah diobservasi dari sebuah atau banyak kata (rangkaiannya kata) akan dapat menyebabkan sistem bekerja dengan rangkaian kata tersebut sesuai dengan aplikasi aturan Bayes's.

$$Pr(\text{word} | \text{acoustics}) = \frac{Pr(\text{acoustics} | \text{word}) Pr(\text{word})}{Pr(\text{acoustics})}$$

Dari rangkaian data akustik yang ada, Pr (acoustics) adalah konstan dan tidak dapat diabaikan. Pr (word) adalah merupakan kemungkinan terbesar dari suatu kata. Pr (acoustics | word) masa yang paling terlibat di dalam persamaan dan diperoleh dari Hidden Markov Models.

Untuk dapat digunakan sebagai aplikasi pada dunia nyata, tiga masalah yang mendasar harus dapat diselesaikan. Diberikan:

Rangkaian pengamatan:

$$O = (o_1 o_2 \dots o_T),$$

$$\text{Model} : \lambda = (A, B, \Pi),$$

Tiga masalah dasar tersebut antara lain:

1. Masalah pertama adalah bagaimana memperhitungkan kemungkinan rangkaian pengamatan  $P(O|\lambda)$  secara efisien dari model. Jika ada beberapa model, maka dapat dipilih model yang cocok dengan pengamatan. Dengan melakukan kombinasi maka jumlah komputasi yang dibutuhkan sangatlah besar. Dan ini menyebabkan munculnya *forward procedure* untuk memenuhi kebutuhan akan prosedur yang lebih efektif.
2. Masalah kedua adalah bagaimana memilih rangkaian status yang cocok:  $q = (q_1 q_2 \dots q_T)$  yang optimal pada beberapa tafsiran. Permasalahannya adalah bagaimana menemukan rangkaian status yang cocok. Karena itu standar yang optimal digunakan untuk menyelesaikan masalah ini dengan sebaik mungkin. Meskipun dapat memaksimumkan jumlah status yang benar, rangkaian status hasil masih menjadi suatu masalah. Misalnya ketika HMM mempunyai transisi status dengan kemungkinan bernilai 0, maka rangkaian status optimal dapat menjadi rangkaian status yang tidak valid. Hal ini terjadi karena sebelumnya solusi hanya menentukan status-status yang mungkin cocok tanpa memperhitungkan rangkaian status. Satu cara yang mungkin dilakukan untuk menyelesaikan masalah ini adalah dengan mengubah standar optimal. Misalnya menyelesaikan masalah rangkaian status dengan memaksimalkan jumlah pasangan yang cocok dari dua status ( $q_t, q_{t+1}$ ) atau tiga status ( $q_t, q_{t+1}, q_{t+2}$ ), dan seterusnya. Meskipun standar ini dapat digunakan untuk beberapa

aplikasi, namun standar yang biasa digunakan adalah dengan menemukan satu buah rangkaian status terbaik dengan memaksimalkan  $P(q|O, \lambda)$ , yang ekuivalen dengan memaksimalkan  $P(q, O|\lambda)$ . Teknik untuk menemukan satu buah rangkaian status terbaik adalah dengan berdasarkan metode *dynamic programming* yang disebut algoritma Viterbi. Dan algoritma inilah yang akan dibahas pada makalah ini.

3. Masalah ketiga adalah bagaimana menyesuaikan parameter model  $\lambda$  untuk memaksimalkan  $P(O|\lambda)$ .

### 3.4. Algoritma Viterbi

Algoritma Viterbi adalah algoritma *dynamic programming* untuk menemukan kemungkinan rangkaian status yang tersembunyi (biasa disebut Viterbi path) yang dihasilkan pada rangkaian pengamatan kejadian, terutama dalam lingkup HMM.

Untuk menemukan sebuah rangkaian status terbaik,  $q = (q_1 q_2 \dots q_T)$ , untuk rangkaian observasi  $O = (o_1 o_2 \dots o_T)$ , perlu didefinisikan kuantitas:

$$(1) \delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P[q_1 q_2 \dots q_{t-1}, q_t = i, o_1 o_2 \dots o_t | \lambda]$$

$\delta_t(i)$  adalah rangkaian terbaik, yaitu dengan kemungkinan terbesar, pada waktu  $t$  dimana perhitungan untuk pengamatan  $t$  pertama dan berakhir pada status  $i$ . Dengan menginduksi, didapat:

$$(2) \delta_{t+1}(j) = \max_i [\delta_t(i) a_{ij}] \cdot b_j(o_{t+1})$$

Untuk mendapatkan kembali rangkaian status, perlu adanya penyimpanan hasil yang memaksimalkan persamaan (2), untuk tiap  $i$  dan  $j$ , dengan menggunakan tabel  $A_t(j)$ . Prosedur lengkap untuk menemukan kumpulan status-status terbaik bisa dirumuskan sebagai:

1. Inisialisasi

$$\delta_1(i) = \Pi_i b_i(o_1), \quad 1 \leq i \leq N$$

$$A_1(1) = 0.$$

2. Rekursif

$$\delta_t(i) = \max_{1 \leq l \leq N} [\delta_{t-1}(l) a_{li}] b_i(o_t)$$

$$2 \leq t \leq T, 1 \leq j \leq N$$

$$A_t(j) = \arg \max [\delta_{t-1}(i) a_{ij}]$$

$$1 \leq i \leq N$$

$$2 \leq t \leq T, 1 \leq j \leq N$$

### 3. Terminasi

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

$$q_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)]$$

### 4. Lintasan status

$$q_t^* = \text{Ar}(t+1)(q_{t+1}^*)$$

$$t = T-1, T-2, \dots, 1.$$

Contoh implementasi algoritma viterbi:

Diberikan sebuah model dari percobaan koin dengan probabilitas

	State 1	State 2
P(H)	0.5	0.75
P(T)	0.5	0.25

dan dengan probabilitas kemungkinan = 1/3 serta probabilitas inisial = 1/3 untuk observasi rangkaian

**O = (H H H H T H T T T T)**

Observasi dari koin diatas dimisalkan sebagai kata dalam penerapannya pada teknologi pengenalan suara.

Maka dengan menggunakan viterbi lintasan yang dibentuk adalah

$$a_{ij} = 1/3$$

$$\delta_1(1) = 0.5, \delta_1(2) = 0.75, \delta_1(3) = 0.25$$

Proses rekursif untuk  $\delta_t(j)$   $2 \leq t \leq 10$

$$\delta_2(1) = (0.75)(0.5)$$

$$\delta_3(1) = (0.75)^2(0.5)$$

$$\delta_4(1) = (0.75)^3(0.5)$$

$$\delta_5(1) = (0.75)^4(0.5)$$

$$\delta_6(1) = (0.75)^5(0.5)$$

$$\delta_7(1) = (0.75)^6(0.5)$$

$$\delta_8(1) = (0.75)^7(0.5)$$

$$\delta_9(1) = (0.75)^8(0.5)$$

$$\delta_{10}(1) = (0.75)^9(0.5)$$

$$\delta_2(2) = (0.75)^2$$

$$\delta_3(2) = (0.75)^3$$

$$\delta_4(2) = (0.75)^4$$

$$\delta_5(2) = (0.75)^4(0.25)$$

$$\delta_6(2) = (0.75)^6$$

$$\delta_7(2) = (0.75)^6(0.25)$$

$$\delta_8(2) = (0.75)^7(0.25)$$

$$\delta_9(2) = (0.75)^8(0.25)$$

$$\delta_{10}(2) = (0.75)^9(0.25)$$

$$\delta_2(3) = (0.75)(0.25)$$

$$\delta_3(3) = (0.75)^2(0.25)$$

$$\delta_4(3) = (0.75)^3(0.25)$$

$$\delta_5(3) = (0.75)^5$$

$$\delta_6(3) = (0.75)^5(0.25)$$

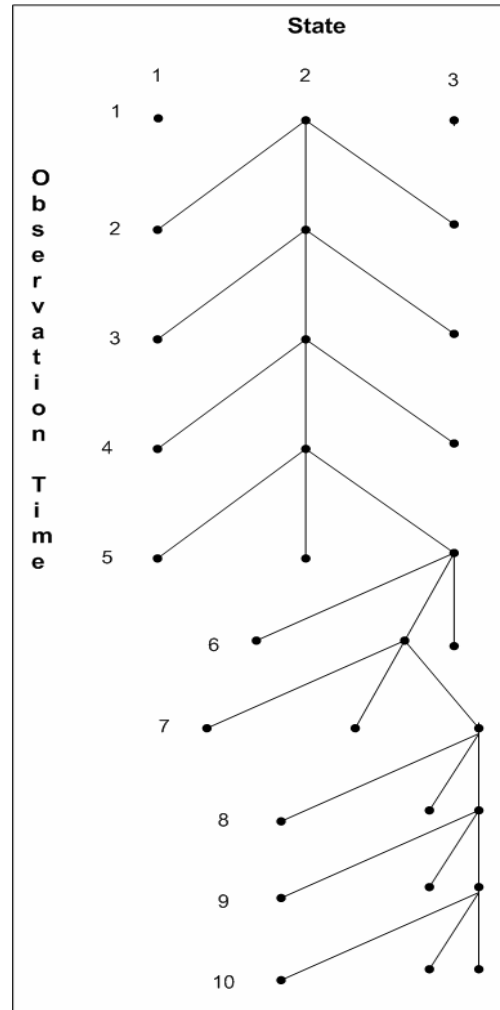
$$\delta_7(3) = (0.75)^7$$

$$\delta_8(3) = (0.75)^8$$

$$\delta_9(3) = (0.75)^9$$

$$\delta_{10}(3) = (0.75)^{10}$$

Dari penjabaran di atas maka dapat diperoleh diagram seperti ini:



Karena itu, rangkaian status yang mungkin cocok adalah {2,2,2,2,3,2,3,3,3,3}.

## 4. Kesimpulan

Dalam teknologi pengenalan suara digunakan metode Hidden Markov Models untuk mengenali suara. HMM menggunakan konsep statistik dan probabilitas. Namun, dalam penerapannya HMM memiliki 3 permasalahan utama yaitu bagaimana memperhitungkan kemungkinan rangkaian pengamatan secara efisien dari model, bagaimana memilih rangkaian status yang cocok dan optimal pada beberapa tafsiran, dan bagaimana menyesuaikan

parameter model  $\lambda$  yang maksimal. Permasalahan kedua dapat ditangani dengan menggunakan algoritma Viterbi. Algoritma ini menggunakan konsep *dynamic programming* dalam menentukan rangkaian staus yang digunakan yaitu nilai maksimal probabilitas status pada setiap tahap.

## 5. Daftar Pustaka

1. <http://en.wikipedia.org>
2. Munir, Rinaldi.2006. "*Strategi Algoritmik*".Departemen Teknik Informatika, Institut Teknologi Bandung
3. R. Lawrence and B.H. Juang, *Fundamental of Speech Recognition*, PTR Prentice Hall, 1993.