

# Aplikasi Hidden Markov Model dalam Pengenalan Ucapan

Made Prisha Wulansari 13518049  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia  
13518049@std.stei.itb.ac.id

**Abstrak**—Pengenalan ucapan merupakan sebuah teknologi yang sering digunakan pada saat ini, salah satunya yang ada di hampir setiap gawai yang digunakan yaitu salah satu mesin pencari dengan menggunakan pengenalan ucapan. Pada proses pengenalan ucapan, akan dilakukan pemodelan terlebih dahulu sebelum dilakukan pengenalan ucapan masukan. Teknologi ini menggunakan metode Markov Model Tersembunyi dalam pemodelan yang dilakukan. Markov Model Tersembunyi merupakan salah satu pengembangan dari graf.

**Kata Kunci**—Markov Model Tersembunyi, *Speech Recognition*, Graf

## I. PENDAHULUAN

Di era perkembangan teknologi seperti saat ini, banyak teknologi yang semakin memudahkan kita sebagai pengguna teknologi dalam melakukan suatu hal, salah satunya dalam hal mencari informasi. Mencari informasi pada sebuah mesin pencari informasi dapat dilakukan dengan beberapa cara seperti mengetikkan langsung informasi apa yang ingin kita dapatkan pada mesin pencari informasi atau bisa juga dengan cara mengatakan informasi apa yang ingin kita cari pada mesin pencari informasi yang dapat memproses ucapan. Untuk memproses kata-kata yang diucapkan, dibutuhkan teknologi yaitu pengenalan ucapan atau yang lebih sering diketahui dengan *speech recognition*. Penggunaan *speech recognition* tidak hanya pada mesin pencari informasi saja, *speech recognition* saat ini juga digunakan untuk menerima ucapan yang diubah menjadi kata-kata untuk diterjemahkan ke bahasa lain atau digunakan pada aplikasi untuk mengenali judul dari sebuah lagu yang sedang diputarkan.

Pada *speech recognition*, ada dua proses utama yaitu pemodelan dan pengenalan. Pemodelan yaitu ketika menerima informasi berupa ucapan agar. Pemodelan yang digunakan pada *speech recognition* ini adalah Markov Model Tersembunyi. Markov Model Tersembunyi merupakan salah satu pengembangan dari graf.

## II. LANDASAN TEORI

### A. Graf

Graf merupakan pasangan himpunan  $(V, E)$ . Dalam hal ini:

$V$  = himpunan simpul-simpul (*vertices* atau *node*) yang tidak kosong

$E$  = himpunan sisi (*edges* atau *arcs*) yang menghubungkan sepasang simpul

$V = \{v_1, v_2, \dots, v_n\}$

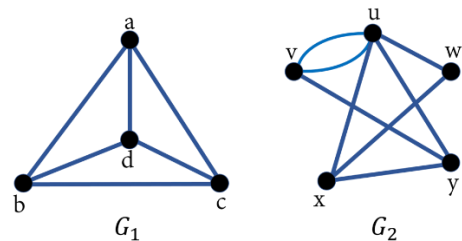
$E = \{e_1, e_2, \dots, e_n\}$

Sehingga graf  $G$  dapat ditulis dengan notasi:

$G = (V, E)$

Sisi merupakan penghubung antar sepasang simpul. Misalkan simpul  $v_1$  dan simpul  $v_2$  dihubungkan oleh sisi  $e$ . Maka dapat dituliskan menjadi:

$e = (v_1, v_2)$



Gambar 2.1 Contoh Graf

(Sumber: <https://mathcyber1997.com/soal-latihan-dan-penyelesaian-teori-dasar-graf-graph-basic-theory/>)

Pada Gambar 2.1, terdapat dua contoh graf yaitu  $G_1$  dan  $G_2$ .  $G_1$  adalah sebuah graf dengan himpunan simpul  $V$  dan himpunan sisi  $E$  yaitu:

$V = \{a, b, c, d\}$

$E = \{(a, b), (a, d), (a, c), (b, c), (b, d), (c, d)\}$

Sedangkan pada  $G_2$ .  $G_2$  adalah sebuah graf dengan himpunan simpul  $V$  dan himpunan sisi  $E$  yaitu:

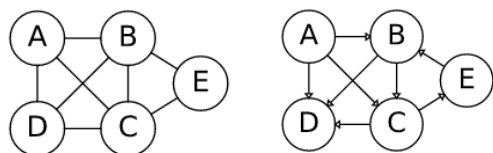
$V = \{u, v, w, x, y\}$

$E = \{(u, v), (u, v), (u, w), (u, x), (u, y), (v, y), (x, y)\}$

Graf dikelompokkan menjadi beberapa jenis tergantung cara mengelompokkannya. Beberapa cara mengelompokkan jenis graf yaitu berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, jumlah simpul pada suatu graf, dan ada atau tidaknya orientasi arah pada graf,

Dilihat dari ada tidaknya gelang atau sisi-ganda pada suatu graf, graf dapat digolongkan menjadi dua jenis yaitu graf sederhana dan graf tak-sederhana. Gelang atau kalang (*loop*) merupakan sisi yang berawal dan berakhir pada simpul yang sama. Sedangkan sisi-ganda (*multiple edges* atau *parallel edges*) yaitu sisi-sisi yang menghubungkan dua buah simpul yang sama. Pada Gambar 2.1,  $G_1$  memiliki sisi-ganda yaitu dua buah sisi  $(u, v)$ . Graf sederhana (*simple graph*) yaitu graf yang tidak

memiliki gelang maupun sisi-ganda. Contoh dari graf sederhana dapat dilihat pada  $G_1$  pada Gambar 2.1. Graf tak-sederhana (*unsimple graph*) yaitu graf yang mengandung gelang atau sisi ganda. Graf tak-sederhana terbagi menjadi dua jenis yaitu graf ganda dan graf semu. Sesuai dengan namanya, graf ganda (*multigraph*) adalah graf yang memiliki sisi ganda. Contoh dari graf ganda dapat dilihat pada  $G_2$  pada Gambar 2.1. Sedangkan graf semu (*pseudograph*) adalah graf yang mengandung gelang, meskipun memiliki sisi ganda juga, graf tersebut merupakan graf semu.

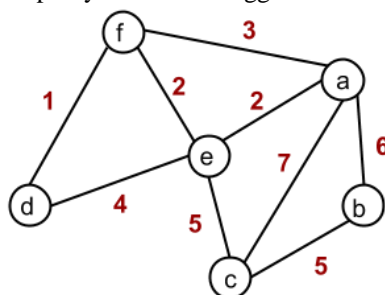


Gambar 2.2 Undirect Graph dan Directed Graph

(Sumber: [https://frnsys.com/ai\\_notes/foundations/graphs.html](https://frnsys.com/ai_notes/foundations/graphs.html))

Jika dilihat dari orientasi arah pada sisi, graf dibedakan menjadi dua jenis yaitu graf tak-berarah dan graf berarah. Graf tak-berarah (*undirect graph*) adalah graf yang sisinya tidak memiliki orientasi arah sehingga  $(v_i, v_j) = (v_j, v_i)$  adalah sisi yang sama, urutan dari pasangan simpul yang terhubung tidak diperhatikan pada jenis graf ini. Sedangkan pada graf berarah (*directed graph* atau *digraph*) memiliki orientasi arah sehingga  $(v_i, v_j) \neq (v_j, v_i)$  karena kedua sisi tersebut berbeda. Pada  $(v_i, v_j)$ ,  $v_i$  merupakan simpul asal (*initial vertex*) dan  $v_j$  merupakan simpul terminal (*terminal vertex*).

Jika dilihat dari jumlah simpul yang ada pada suatu graf, maka terdapat dua jenis graf yaitu graf berhingga dan graf tak-berhingga. Sesuai dengan namanya, graf berhingga (*limited graph*) adalah graf yang jumlah simpulnya berhingga. Sedangkan graf tak-berhingga (*unlimited graph*) adalah graf yang jumlah simpulnya tidak berhingga.

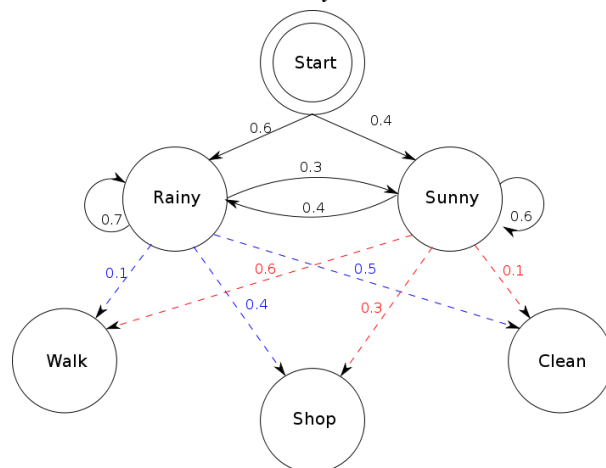


Gambar 2.3 Weighted Graph

(Sumber: <https://www.chegg.com/homework-help/questions-and-answers/undirected-weighted-graph-g-given-use-prim-s-algorithm-compute-minimum-spanning-tree-weig-q21203421>)

Sisi pada sebuah graf ada yang memiliki harga dan ada yang tidak memiliki harga. Jenis graf yang setiap sisinya memiliki harga atau bobot adalah graf berbobot (*weighted graph*) seperti pada Gambar 2.3. Bobot setiap sisi pada graf dapat berupa jarak antara dua kota, biaya perjalanan antara dua tempat, peluang antar *state*, atau lainnya. Lintasan yang diambil pada graf akan mempengaruhi total bobot yang akan dihasilkan.

## B. Markov Model Tersembunyi



Gambar 2.4 Contoh Hidden Markov Model

(Sumber: <https://medium.com/@kangeugine/hidden-markov-model-7681c22f5b9>)

Markov Model Tersembunyi atau *Hidden Markov Model* (HMM) didasari dari rantai Markov. Rantai Markov sendiri merupakan pemodelan mengenai probabilitas dari urutan variable acak atau status. Rantai Markov sangat berguna ketika diperlukan untuk menghitung probabilitas peristiwa yang terjadi secara berurutan. Pada beberapa kasus merupakan peristiwa yang belum terjadi, sehingga peristiwa tersebut masih tersembunyi. Pemodelan yang digunakan pada peristiwa yang belum diketahui inilah yang merupakan *hidden Markov model*. Markov model tersembunyi berupa graf berarah yang memiliki bobot, dapat dilihat pada Gambar 2.4. Pada Gambar 2.4 bobot yang dimaksud merupakan peluang antar *state*.

Markov model menggunakan asumsi Markov tentang probabilitas dari suatu urutan yaitu ketika memprediksi masa depan, masa lalu tidak berpengaruh, tetapi masa kini yang mempengaruhi. Asumsi Markov yaitu:

$$P(q_i = a | q_1 \dots q_{i-1}) = P(q_i = a | q_{i-1})$$

$P$  merupakan probabilitas

Dengan HMM, kita dapat menggunakan peristiwa yang diamati dan peristiwa tersembunyi sebagai faktor penyebab dari model probabilitas.

Komponen dari HMM yaitu

$$Q = q_1 q_2 \dots q_N$$

$Q$  merupakan set dari *state*

$$A = a_{11} a_{12} \dots a_{n1} \dots a_{nn}$$

$A$  merupakan matriks probabilitas transisi

$a_{ij}$  merupakan probabilitas perpindahan *state* dari  $i$  ke  $j$

$$O = o_1 o_2 \dots o_T$$

$O$  merupakan urutan dari observasi  $T$

$$B = b_i(o_t)$$

$B$  merupakan probabilitas emisi

$$\pi = \pi_1, \pi_2, \dots, \pi_N$$

$\pi$  merupakan distribusi probabilitas awal

$\pi_i$  merupakan probabilitas rantai Markov akan dimulai dari *state*  $i$ .

*Hidden Markov Model* orde pertama memberikan dua asumsi. Asumsi pertama yaitu probabilitas sebuah *state* hanya bergantung pada *state* sebelumnya yaitu pada Asumsi Markov.

Asumsi kedua yaitu probabilitas dari observasi *output*  $o_i$  bergantung hanya pada *state* yang menghasilkan observasi  $q_i$ , bukan observasi lain. *Output independence*:

$$P(o_i|q_1 \dots q_i \dots q_T, o_1 \dots o_i \dots o_T) = P(o_i|q_i)$$

### C. Pengenalan Ucapan (*Speech Recognition*)

*Speech Recognition* merupakan sebuah sistem yang memungkinkan perangkat teknologi untuk menerima masukan berupa kata-kata yang diucapkan. Dengan teknologi ini, kata-kata yang diucapkan akan dikenali dan dipahami oleh perangkat melalui digitalisasi kata kemudian mencocokkan sinyal digital dengan suatu pola yang telah tersimpan.

Pengenalan ucapan dapat dikelompokkan menjadi lima jenis berdasarkan kemampuan dalam mengenal kata yang diucapkan. Pertama yaitu pengenalan kata-kata yang terisolasi. Pengenalan kata-kata yang terisolasi merupakan proses untuk mengenal kata-kata hanya jika pengucapan kata-kata tersebut memiliki jeda antar kata. Pengenalan kata lainnya juga yaitu pengenalan kata-kata yang berhubungan yaitu proses pengidentifikasian kata seperti kata-kata yang terisolasi tetapi dengan jeda yang lebih singkat daripada kata-kata yang terisolasi. Ada juga pengenalan kata-kata yang berkelanjutan yaitu proses pengenalan kata-kata yang memiliki jeda antar tiap kata sangat singkat atau bahkan tanpa jeda waktu, sehingga pengguna dapat mengucapkan kata-kata yang dimaksud secara natural. Tetapi proses pengenalan kata-kata berkelanjutan ini sangat rumit karena system yang digunakan harus membeda-bedakan kata tanpa jeda waktu. Pengenalan ucapan lainnya dapat mengenal kata-kata spontan yaitu kata-kata yang diucapkan tanpa jeda waktu antar kata. Pengenalan ucapan juga dapat mengidentifikasi atau memverifikasi suara yang memungkinkan tidak hanya mengetahui kata-kata apa yang dimaksud dari sebuah ucapan, tetapi juga mengetahui siapa yang mengucapkan kata-kata tersebut.

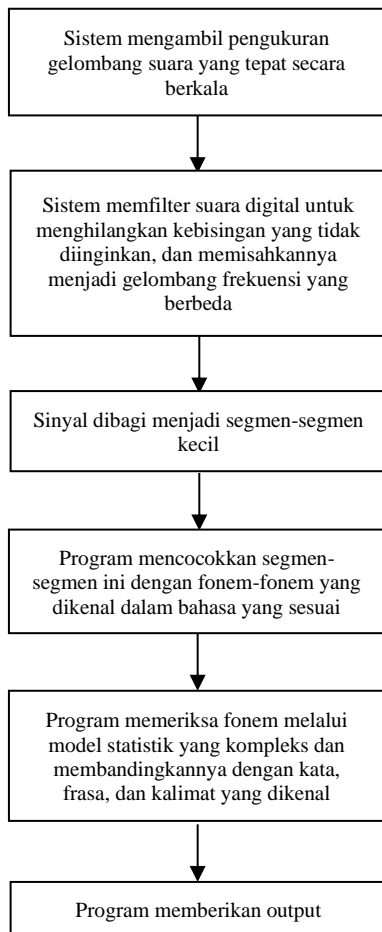
Pengenalan ucapan tentu bukan sebuah system yang benar-benar sempurna. Para *developer* dari sistem ini terus melakukan pengembangan untuk menyempurnakan kekurangan-kekurangan yang dimiliki system ini. Kekurangan pada sistem ini akan mempengaruhi akurasi hasil yang didapatkan dari pengenalan ucapan. Ada beberapa faktor yang mempengaruhi kurangnya akurasi hasil dari sistem ini diantaranya adalah rendahnya rasio sinyal terhadap *noise*, ucapan yang *overlapping*, penggunaan daya komputer, juga kata-kata homonim. Pada kasus rendahnya rasio sinyal terhadap *noise*, ketika pengguna mengucapkan kata-kata yang dimaksud, ada kemungkinan adanya *noise* atau suara tambahan yang mengganggu program untuk menerima kata-kata yang dimaksud. Maka dari itu, pengguna harus berada di ruangan yang tenang dan memposisikan mikrofon sedekat mungkin dengan mulut agar mengurangi *noise* yang diterima sistem. Sedangkan pada kasus ucapan yang *overlapping*, sistem pengenalan ucapan saat ini masih kesulitan untuk mengidentifikasi kata-kata jika ada ucapan lain yang masuk dalam sistemnya, misalnya ketika pengguna menggunakan sistem ini ketika ada percakapan lain di dekatnya sehingga kata-kata tersebut juga masuk ke dalam sistem dan ikut diproses, akurasi dari hasil yang didapat akan sangat buruk karena hal ini. Kemudian ada faktor penggunaan daya komputer untuk

menjalankan sistem pengenalan ucapan ini. Dalam menjalankan sistem pengenalan ucapan, diperlukan *processor* untuk melakukan pekerjaan yang berat. Dalam pengenalan ucapan, sistem perlu melakukan tahap pencarian setiap kata untuk menghasilkan kata-kata yang sesuai dengan yang dimaksud pengguna, terutama pada perintah atau frasa yang rumit. Kata-kata yang ada di dalam perpustakaan sistem pengenalan ucapan ada sangat banyak, ini juga menghabiskan banyak ruang pada *hard drive*. Faktor lainnya yang dapat mempengaruhi akurasi hasil dari pengenalan ucapan adalah homonim. Homonim merupakan kata-kata yang memiliki ejaan sama tetapi maknanya berbeda. Sistem mengenali kata-kata yang diberikan pengguna berdasarkan pelafalannya sehingga kata-kata homonim akan diidentifikasi sebagai kata-kata yang sama oleh sistem. Satu-satunya cara untuk mengidentifikasi kata-kata yang benar adalah dengan melihat konteks dari kalimat yang diterima pada kata tersebut.

*Speech recognition* telah diterapkan ke beberapa bidang. Dalam bidang komunikasi, *speech recognition* telah digunakan pada program komputer yang menerima masukan suara sebagai perintah untuk melakukan sesuatu hal. Dalam bidang komunikasi juga *speech recognition* digunakan untuk melakukan pendiktean pada proses pembuatan laporan. Dalam bidang lainnya yaitu dalam bidang kesehatan, *speech recognition* bermanfaat salah satunya bagi para penyandang disabilitas misalkan dalam pengendalian saklar lampu sehingga pengguna cukup mengucapkan perintah untuk mengontrol nyala lampu. Pada bidang militer, *speech recognition* digunakan pada pengatur lalu lintas udara yaitu *Air Traffic Controllers* (ATC).

### III. PROSES PENGENALAN UCAPAN DENGAN MARKOV MODEL TERSEMBUNYI

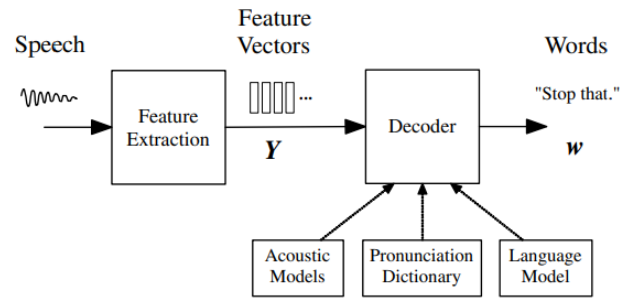
Ada beberapa tahapan kompleks yang dilewati dalam memproses dari kata-kata yang diucapkan pengguna hingga kata-kata tersebut teridentifikasi.



Gambar 2.5 Cara Kerja *Speech Recognition*

Pada Gambar 2.5 merupakan proses dari teknologi *speech recognition* yang menerima informasi berupa suara yang diberikan pengguna hingga program memberikan output baik berupa teks atau mengeluarkan perintah pada perangkat yang digunakan.

Teknologi *speech recognition* menggunakan sistem pemodelan yang rumit yaitu menggunakan fungsi probabilitas dan matematika untuk menentukan hasil dengan tingkat akurasi tertinggi. Pemodelan yang banyak digunakan pada sistem *speech recognition* ini adalah *Hidden Markov Model*. *Hidden Markov Model* banyak dipilih karena lebih sederhana untuk pemodelan dengan variasi waktu. Sehingga sistem pengenalan ucapan atau suara kontinu saat ini banyak yang didasari *Hidden Markov Model*.



Gambar 2.6 Arsitektur Pengenalan Berbasis *Hidden Markov Model*

(Sumber: [https://mi.eng.cam.ac.uk/~mjfg/mjfg\\_NOW.pdf](https://mi.eng.cam.ac.uk/~mjfg/mjfg_NOW.pdf))

Pada Gambar 2.6 merupakan proses pada pengenalan ucapan berbasis *Hidden Markov Model*. Ucapan yang diterima oleh sistem akan melalui proses *feature extraction* yaitu pemrosesan yang melakukan transformasi pada data masukan menjadi kumpulan data dengan berkurangnya dimensi dari data agar lebih mudah untuk diproses. Hasil dari kumpulan data ini merupakan informasi yang relevan dan merupakan representasi dari data masukan.

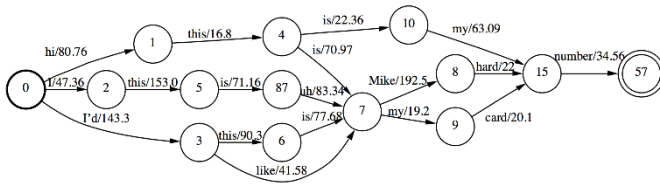
Setelah proses *feature extraction*, kemudian *decoder* akan mengenali kata-kata masukan dari hasil *feature extraction*. *Decoder* tidak langsung mengenali setiap kata, tetapi *decoder* mencari keadaan yang memaksimalkan pengamatan.

$$\hat{w} = \arg \max_w \{P(w|Y)\}.$$

Karena  $P(w|Y)$  sulit untuk dimodelkan secara langsung, digunakan *Bayes' Rule* untuk mentransformasikannya menjadi

$$\hat{w} = \arg \max_w \{p(Y|w)P(w)\}.$$

$p(Y|w)$  ditentukan oleh *acoustic models* yang sesuai dan  $P(w)$  ditentukan oleh *language models*. Pada setiap  $w$ , *acoustic models* disintesiskan melalui konkatensi dengan *phone models* untuk membentuk kata-kata yang ada pada *pronunciation dictionary*. Parameter dari model ini berasal dari data-data *training* yang berisi gelombang-gelombang suara dan transkripsi ortografinya. Cara kerja *decoder* yaitu dengan mencari semua urutan kata yang mungkin dengan penghapusan hipotesis yang tidak memungkinkan. Setelah semua proses selesai, urutan kata yang paling mungkin merupakan *output*. Ada juga *decoder* yang dapat menghasilkan kisi yang berisi representasi *output* yang paling mungkin. Contoh dari graf yang berisi kisi-kisi kata ada pada Gambar 2.7



Gambar 2.7 Contoh Graf Kisi Kata

(Sumber: <https://medium.com/@jichangchunhan/microsoft-sr-2016-paper-overview-f4d925ee046d>)

#### IV. KESIMPULAN

Dalam pengenalan ucapan yang berawal dari ucapan pengguna, ada proses rumit dan kompleks di dalamnya. Salah satu proses tersebut adalah proses pemodelan. Proses pemodelan dari pesan suara tersebut dapat menggunakan *Hidden Markov Model*. *Hidden Markov Model* merupakan implementasi dari Rantai Markov. Rantai Markov merupakan salah satu pengembangan dari graf berbobot dan berarah. *Hidden Markov Model* sangat membantu dalam pengenalan ucapan karena sistem yang ditawarkan lebih sederhana untuk memprosesnya.

#### V. UCAPAN TERIMA KASIH

Terima kasih dan syukur penulis sampaikan kepada Tuhan Yang Maha Esa yang telah memberikan kesempatan untuk membuat makalah ini juga atas kelancaran dalam pembuatan makalah ini. Terima kasih juga kepada seluruh dosen pengampu mata kuliah IF2120 Matematika Diskrit dan juga kepada semua pihak yang telah membantu dalam keberjalanan pembuatan makalah ini.

#### REFERENSI

- [1] Munir, Rinaldi. 2006. *Diktat Kuliah IF2120 Matematika Diskrit (Edisi Keempat)*. Bandung: Institut Teknologi Bandung.
- [2] <https://mti.binus.ac.id/2018/03/05/teori-graph-sejarah-dan-manfaatnya/> diakses pada 6 Desember 2019 pukul 06.28 WIB
- [3] <https://electronics.howstuffworks.com/gadgets/high-tech-gadgets/speech-recognition2.htm> diakses pada 6 Desember 2019 pukul 06.45 WIB
- [4] <https://web.stanford.edu/~jurafsky/slp3/A.pdf> diakses pada 6 Desember 2019 pukul 09.03 WIB
- [5] [https://mi.eng.cam.ac.uk/~mjfg/mjfg\\_NOW.pdf](https://mi.eng.cam.ac.uk/~mjfg/mjfg_NOW.pdf) diakses pada 6 Desember 2019 pukul 09.41 WIB
- [6] [https://medium.com/@jonathan\\_hui/speech-recognition-gmm-hmm-8bb5eff8b196](https://medium.com/@jonathan_hui/speech-recognition-gmm-hmm-8bb5eff8b196) diakses pada 6 Desember 2019 pukul 09.47 WIB

#### PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 6 Desember 2019

Made Prisha Wulansari 13518049