



**TESIS TE142599**

**SISTEM IDENTIFIKASI PERINTAH BAHASA INDONESIA  
PADA PROTOTIPE ROBOT PELAYAN DAN KEAMANAN**

**Karisma Trinanda Putra  
2212204013**

**DOSEN PEMBIMBING  
Dr. Ir. Djoko Purwanto, M.Eng.  
Dr. Ronny Mardiyanto, ST., MT.**

**PROGRAM MAGISTER  
BIDANG KEAHLIAN ELEKTRONIKA  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2015**



**TESIS TE142599**

**INDONESIAN COMMAND IDENTIFICATION SYSTEM FOR  
SERVICE AND SECURITY ROBOT PROTOTYPE**

**Karisma Trinanda Putra  
2212204013**

**DOSEN PEMBIMBING  
Dr. Ir. Djoko Purwanto, M.Eng.  
Dr. Ronny Mardiyanto, ST., MT.**

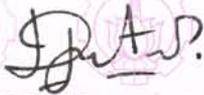
**PROGRAM MAGISTER  
BIDANG KEAHLIAN ELEKTRONIKA  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2015**

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar  
Magister Teknik (M.T)  
di  
Institut Teknologi Sepuluh Nopember

Oleh:  
Karisma Trinanda Putra  
Nrp. 2212204013

Tanggal Ujian : 22 Juni 2015  
Periode Wisuda : September 2015

Disetujui oleh:



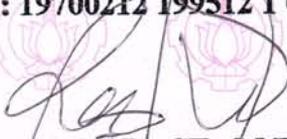
1. Ir. Djoko Purwanto, M.Eng., Ph.D. (Pembimbing I)  
NIP: 19651211 199002 1 002



2. Ronny Mardiyanto, ST., MT., Ph.D. (Pembimbing II)  
NIP: 19810118 200312 1 003



3. Dr. Tri Arief Sardjono, ST., MT. (Penguji)  
NIP: 19700212 199512 1 001



4. Achmad Arifin, ST., M.Eng., Ph.D. (Penguji)  
NIP: 19710314 199702 1 001



5. Dr. Muhammad Rivai, ST., MT. (Penguji)  
NIP: 19690426 199403 1 003



Direktur Program Pascasarjana,

Prof. Dr. Ir. Adi Soeprijanto, MT.  
NIP: 19640405 199002 1 001

# INDONESIAN COMMAND IDENTIFICATION SYSTEMS FOR SERVICE AND SECURITY ROBOT PROTOTYPE

Student Name : Karisma Trinanda Putra  
Student Identity Number : 2212204013  
Supervisor : Djoko Purwanto  
Ronny Mardiyanto

## ABSTRACT

Human-machine interaction has been growing with the discovery of artificial intelligence technology. The development of human-machine interaction leads to a more natural interaction. In daily interactions, human uses speech, more dominant than the other way such as gestures and eye contact. Speech is the vocalized form of human communication which is closely related to language system. The problem is meaning, ambiguity, and the language that is not according to the rules of syntax, causing the command translation become more complex. We need certain rules of syntax and semantics to understand the meaning of a command. An artificial intelligence technology developed in this research to understand the Indonesian speech commands for applications in the field of robotics, which is realized on the service and security robot prototype. The purpose of this research is to translate voice command into the robots action, to generate human-machine interaction more natural. The voice command will be extracted using bark-frequency cepstral coefficients. Cepstrals identified into words using neural networks. Words in a complete sentences will be processed using natural language processing so that, the meaning can be translated into robot action. Speech recognition experiments with 28 sets of speech signal obtain 82 % accuracy, while natural language processing experiments obtain 93 % accuracy with 100 sets of Indonesian vocabulary.

**Keywords:** bark-frequency cepstral coefficients, neural network, natural language processing.

*(Halaman ini sengaja dikosongkan)*

# **SISTEM IDENTIFIKASI PERINTAH BAHASA INDONESIA PADA PROTOTYPE ROBOT PELAYAN DAN KEAMANAN**

Nama Mahasiswa : Karisma Trinanda Putra  
NRP : 2212204013  
Pembimbing : Djoko Purwanto  
Ronny Mardiyanto

## **ABSTRAK**

Dewasa ini, perkembangan interaksi manusia-mesin semakin mengarah pada sistem interaksi yang lebih natural dan mendekati sistem interaksi manusia. Sistem interaksi manusia menggunakan wicara lebih dominan daripada sarana komunikasi yang lain seperti kontak mata dan bahasa tubuh atau isyarat. Wicara adalah bentuk vokal sarana komunikasi manusia yang lebih dekat dengan sistem bahasa. Permasalahan yang muncul adalah pemaknaan, ambiguitas dan bahasa yang tidak sesuai aturan sintaksis menyebabkan penerjemahan perintah lisan menjadi lebih kompleks. Pada penelitian ini, dikembangkan teknologi kecerdasan buatan yang dapat memahami perintah lisan Bahasa Indonesia untuk aplikasi di bidang robotika, yang direalisasikan pada prototipe robot pelayan dan keamanan. Tujuan dari penelitian ini adalah untuk menerjemahkan perintah suara menjadi perintah bagi robot sehingga robot dapat mengerjakan perintah yang dimaksud dan menghasilkan sistem interaksi yang lebih natural. Sinyal suara akan diekstrak menggunakan *bark-frequency cepstral coefficients*. *Cepstral* akan diidentifikasi sebagai kata-kata dengan menggunakan *neural network*. Kata-kata dalam kalimat lengkap akan diproses menggunakan *natural language processing* sehingga maksud yang diucapkan dapat diterjemahkan menjadi aksi bagi robot. Percobaan pengenalan wicara dengan 28 set sinyal wicara memperoleh akurasi 82 %, sedangkan percobaan pemrosesan bahasa natural menghasilkan akurasi 93 % dengan 100 set kosakata bahasa Indonesia.

**Kata kunci:** *bark-frequency cepstral coefficients, neural network, natural language processing.*

*(Halaman ini sengaja dikosongkan)*

## KATA PENGANTAR

Alhamdulillah, puji dan syukur penulis panjatkan kepada Allah SWT, karena atas segala nikmat-Nya lah tesis ini dapat diselesaikan. Tesis berjudul “*Sistem Identifikasi Perintah Bahasa Indonesia pada Prototipe Robot Pelayan dan Keamanan*” ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Magister Teknik (MT) pada Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember.

Penulis menyadari bahwa dalam penyusunan tesis ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, dengan ketulusan dan kerendahan hati penulis menyampaikan terima kasih kepada:

1. Ibunda dan Ayahanda atas segala dukungan dan doanya hingga sampai terselesaikannya tesis ini.
2. Bapak Ir. Djoko Purwanto, M.Eng., Ph.D selaku dosen pembimbing dan koordinator program Pasca Sarjana jurusan Teknik elektro.
3. Bapak Ronny Mardiyanto, ST, MT., Ph.D selaku dosen pembimbing yang telah banyak memberikan saran dan bantuan.
4. Bapak Dr. Muhammad Rivai, ST, MT selaku dosen penguji yang telah banyak memberikan saran.
5. Bapak Achmad Arifin, ST., M.Eng., Ph.D selaku koordinator Bidang Studi Elektronika.
6. Bapak Dr. Tri Arief Sardjono, ST., MT selaku ketua jurusan Teknik Elektro.
7. Seluruh dosen pengajar Jurusan Teknik Elektro yang telah banyak memberikan ilmu selama penulis menempuh kuliah.
8. Rekan–rekan seperjuangan angkatan 2012, Mbak Vita, Mas Syahri, Mas Riza, Mas Riki, Pak Dadang, Pak Edi, Pak Dwi, Mas Bayu, Pak Tukadi, Mas Hepi, Mas Himawan, Suhu Fauzan, Mas Gusdur, Pak Komandan Ribus, Mas Oki serta teman-teman mahasiswa bidang studi elektronika atas segala bantuan dan sumbangan pikiran dan tenaga dalam menyelesaikan tesis ini.

Pada akhirnya, penulis menyadari bahwa tesis ini masih belum sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun. Penulis berharap semoga tesis ini dapat bermanfaat bagi pengembangan ilmu pengetahuan dan teknologi serta bagi masyarakat.

Surabaya, 7 Juni 2015

Penulis

## DAFTAR ISI

LEMBAR PENGESAHAN .....	i
KATA PENGANTAR .....	iii
ABSTRAK .....	v
ABSTRACT .....	vii
DAFTAR ISI .....	ix
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL .....	xv
BAB 1 PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Perumusan Masalah .....	2
1.3. Tujuan Dan Manfaat Penelitian .....	3
BAB 2 KAJIAN PUSTAKA DAN DASAR TEORI .....	5
2.1. Kajian Pustaka .....	5
2.2. Dasar Teori .....	6
2.2.1. Voice Activity Detection .....	6
2.2.2. Bark-Frequency Cepstral Coefficients .....	7
2.2.3. Neural Network .....	13
2.2.4. Recursive Neural Network .....	15
BAB 3 METODOLOGI PENELITIAN .....	17
3.1. Speech Recognition .....	20
3.1.1. Voice Activity Detection .....	21
3.1.2. Pre-emphasis Filtering .....	22
3.1.3. Frame Blocking .....	23
3.1.4. Windowing .....	24
3.1.5. Fast Fourier Transform .....	24
3.1.6. Bark-Frequency Wrapping .....	26
3.1.7. Discrete Cosinus Transform .....	28
3.1.8. Pattern Identification .....	29
3.2. Natural language processing .....	32

3.2.1.	Vector space model .....	32
3.2.2.	Parsing Sentence with Recursive Neural Network.....	35
3.2.3.	Decoding.....	37
3.3.	Keluaran.....	38
BAB 4 PENGUJIAN DAN ANALISA SISTEM .....		43
4.1.	Pengujian Voice Activity Detection. ....	43
4.1.1.	Prosedur Pengujian.....	43
4.1.2.	Hasil Pengujian dan Analisa.....	44
4.2.	Pengujian Ekstraksi Ciri Sinyal Suara. ....	47
4.2.1.	Prosedur Pengujian.....	47
4.2.2.	Hasil Pengujian dan Analisa.....	48
4.3.	Pengujian Sistem Identifikasi Pola Dengan Neural Network. ....	51
4.3.1.	Prosedur Pengujian.....	51
4.3.2.	Hasil Pengujian dan Analisa.....	52
4.4.	Pengujian Variasi Kata Dan <i>Noise</i> Dalam Proses Pengenalan Pola Data Sinyal Wicara.....	53
4.4.1.	Prosedur Pengujian.....	53
4.4.2.	Hasil Pengujian dan Analisa.....	54
4.5.	Pengujian Variasi Pewicara Dalam Proses Pengenalan Pola Data Sinyal Wicara.....	55
4.5.1.	Prosedur Pengujian.....	55
4.5.2.	Hasil Pengujian dan Analisa.....	56
4.6.	Pengujian Pemetaan Kosakata Ke Dalam Model Representasi Distribusi. ....	57
4.6.1.	Prosedur Pengujian.....	57
4.6.2.	Hasil Pengujian dan Analisa.....	57
4.7.	Pengujian Penguraian dan Penerjemahan Perintah Berdasarkan Struktur Pohon Sintaksisnya. ....	58
4.7.1.	Prosedur Pengujian.....	58
4.7.2.	Hasil Pengujian dan Analisa.....	59
BAB 5 PENUTUP.....		63
5.1.	Kesimpulan .....	63

5.2. Saran .....	64
DAFTAR PUSTAKA .....	65
LAMPIRAN .....	67
RIWAYAT HIDUP PENULIS .....	69

*(Halaman ini sengaja dikosongkan)*

## DAFTAR TABEL

Tabel 4.1. Akurasi voice activity detection dengan variasi SNR.....	45
---------------------------------------------------------------------	----

*(Halaman ini sengaja dikosongkan)*

## DAFTAR GAMBAR

Gambar 2.1. Voice activity detection.....	7
Gambar 2.2. Blok diagram perhitungan cepstrum .....	9
Gambar 2.3. Diagram kupu-kupu FFT radix-2 DIT11 .....	11
Gambar 2.4. Bark-Frequency Filter Bank.....	12
Gambar 2.5. Struktur dasar neuron .....	14
Gambar 2.6. Struktur pohon sintaksis kalimat ‘tolong-jaga-kamar-tidur’ .....	15
Gambar 3.1. Pemetaan kalimat perintah pada robot pelayan dan keamanan.....	17
Gambar 3.2. Bagan sistem identifikasi perintah wicara.....	18
Gambar 3.3. Rancangan perangkat keras prototipe robot pelayan dan keamanan	19
Gambar 3.4. Keluaran sistem speech recognition.....	20
Gambar 3.5. Ilustrasi proses frame blocking .....	23
Gambar 3.6. Topologi neural network untuk identifikasi pola .....	31
Gambar 3.7. Ilustrasi pemetaan obyek pada vector space model .....	33
Gambar 3.8. Pemetaan kelas kata dan frase dalam vector space model .....	34
Gambar 3.9. Topologi neural network untuk pemetaan vector space model.....	34
Gambar 3.10. Topologi recursive neural network untuk penguraian struktur sintaksis kalimat.....	36
Gambar 3.11. Contoh proses penguraian struktur kalimat.....	37
Gambar 3.12. Proses decoding pada pemrosesan bahasa natural .....	38
Gambar 3.13. Proses penyusunan database ciri kata.....	39
Gambar 3.14. Proses pelatihan dan pengujian sinyal wicara .....	39
Gambar 3.15. Proses pelatihan dan pengujian bahasa natural .....	40
Gambar 3.16. Prototipe robot pelayan dan keamanan .....	40
Gambar 4.1. VAD pada sinyal wicara ‘jarvis-tolong-maju-lima-langkah’ pada (a) kondisi ideal (b) kondisi pengucapan terlalu cepat .....	44
Gambar 4.2. Sinyal noise dan pengambilan keputusan VAD pada sinyal wicara ‘jarvis-tolong-maju-lima-langkah’ dengan (a) SNR 56, (b) SNR 25 dan (c) 21 dB .....	46

Gambar 4.3. Sinyal mentah hasil pemotongan VAD .....	48
Gambar 4.4. Keluaran sinyal setelah melalui pre-emphasis filtering.....	49
Gambar 4.5. Keluaran sinyal setelah melalui proses transformasi fourier cepat ..	49
Gambar 4.6. Ciri sinyal wicara setelah melalui bark-frequency wrapping .....	50
Gambar 4.7. Ciri sinyal setelah melalui proses DCT .....	50
Gambar 4.8. Ciri sinyal wicara untuk kata 'jarvis' dalam dua kali percobaan.....	51
Gambar 4.9. Pengaruh variasi variabel neural network terhadap kompleksitas perhitungan .....	52
Gambar 4.10. Pengaruh variasi minimum error terhadap akurasi pengenalan pola .....	53
Gambar 4.11. Variasi pengaruh SNR terhadap akurasi pengenalan kata .....	54
Gambar 4.12. Pengaruh jumlah pewicara terhadap akurasi pengenalan pewicara	56
Gambar 4.13. Pemetaan 100 kosakata menggunakan VSM.....	58
Gambar 4.14. Proses pengenalan struktur sintaksis pada kalimat 'jarvis-tolong- bersihkan-ruang-tamu' .....	60
Gambar 4.15. Keluaran struktur sintaksis pada aplikasi yang dibuat.....	60
Gambar 4.16. Keluaran terjemahan dalam bahasa mesin sesuai dengan perintah yang diberikan .....	61
Gambar 4.17. Pengaruh variasi data learning dan testing terhadap akurasi pengenalan struktur kalimat.....	62

# BAB 1

## PENDAHULUAN

### 1.1. Latar Belakang

Robot adalah mesin yang dapat mengerjakan tugas fisik menggunakan pengontrolan dan pengendalian oleh manusia ataupun dengan kecerdasan buatan. Sama halnya dengan interaksi sosial, manusia juga akan perlu menjalin interaksi dengan robot misalkan memberikan perintah bagi mesin tersebut. Manusia memiliki beberapa cara untuk berinteraksi seperti dengan kontak mata, bahasa tubuh dan wicara. Cara yang paling sering digunakan oleh manusia adalah interaksi dengan wicara. Manusia memiliki aset kompleks untuk berbicara, meliputi organ reproduksi suara, organ pendengaran dan otak sebagai organ pemroses informasi. *Neuropsychiatrist* melaporkan bahwa manusia menggunakan sekitar 6.000 – 20.000 kata setiap hari [1]. Itulah mengapa wicara sangatlah penting bagi manusia. Dengan wicara, informasi dapat disampaikan dengan lebih detail. Dengan banyaknya variasi tata bahasa menyebabkan penerjemahan makna dari sinyal wicara menjadi semakin kompleks. Penelitian ini bertujuan untuk mengembangkan teknologi kecerdasan buatan yang dapat memahami perintah Bahasa Indonesia yang diaplikasikan pada prototipe robot pelayan dan keamanan.

Sinyal suara adalah suatu sinyal analog yang membutuhkan pemrosesan lebih lanjut sebelum dapat dipahami oleh mesin. Pada mulanya, sinyal suara dipisahkan dari sinyal jeda (diam) menggunakan *voice activity detection*. Kemudian ciri sinyal diekstrak menggunakan *bark-frequency cepstral coefficients*. Sistem ini mengemulasikan sistem pendengaran manusia dengan menganalisis spektrum frekuensi ke dalam beberapa *filter* spektral tertentu. *Neural network* akan digunakan dalam proses identifikasi *cepstral* yang dihasilkan. *Neural network* (NN) adalah sekelompok jaringan unit pemroses yang dapat memodelkan sesuatu berdasarkan sistem jaringan syaraf manusia. NN mampu untuk memproses data statistik non-linier. NN akan memberikan penilaian statistik tingkat kecocokan sinyal suara dengan kata-kata yang dilatihkan. NN juga digunakan dalam pemrosesan bahasa natural untuk menentukan penilaian koneksi antar kata yang berurutan.

*Natural language processing* (NLP) adalah salah satu cabang teknologi kecerdasan buatan yang berfokus pada penyelesaian masalah yang muncul dalam interaksi antara mesin dengan bahasa natural. Bahasa natural adalah bahasa yang secara umum digunakan oleh manusia untuk berkomunikasi dengan yang lain. Dalam NLP, kosakata akan dipetakan ke dalam *vector space model* [2]. *Vector space model* (VSM) adalah sebuah pemodelan bahasa yang meliputi karakteristik semantik dan sintaksis dalam pemrosesan bahasa natural. Dalam VSM, kosakata dan frase dipetakan dalam sebuah vektor bilangan nyata dalam dimensi tertentu. Setiap kombinasi dari kata atau frase dapat menghasilkan perbedaan makna. *Recursive neural network* (RNN) dapat digunakan untuk memprediksi hierarki struktur pohon kalimat yang dapat digunakan untuk mencari makna yang terkandung di dalamnya.

## **1.2. Perumusan Masalah**

Pada penelitian ini terdapat beberapa permasalahan yang harus diselesaikan, yaitu :

1. Bagaimana memisahkan sinyal suara yang berisi informasi dengan sinyal suara yang tidak berisi informasi.
2. Bagaimana mengekstraksi ciri sinyal suara menjadi data untuk proses pengolahan selanjutnya.
3. Bagaimana mengidentifikasi data berdasarkan pola yang dapat dipelajari menggunakan *neural network*.
4. Bagaimana pengaruh variasi kata dan *noise* dalam proses pengenalan pola data sinyal wicara.
5. Bagaimana pengaruh variasi pewicara dalam proses pengenalan pola data sinyal wicara.
6. Bagaimana menerjemahkan kosakata ke dalam model representasi distribusi dalam proses pengolahan bahasa natural.
7. Bagaimana memodelkan perintah berdasarkan struktur pohon sintaksis kalimat yang dimaksud kemudian menerjemahkannya menjadi urutan perintah untuk robot.

### **1.3. Tujuan Dan Manfaat Penelitian**

Tujuan dari penelitian ini adalah :

1. Memisahkan sinyal suara yang berisi informasi dengan sinyal suara yang tidak berisi informasi.
2. Mengekstraksi ciri sinyal suara menjadi data yang dapat digunakan untuk proses pengolahan selanjutnya.
3. Mengidentifikasi pola data menggunakan *neural network*.
4. Mengetahui pengaruh variasi kata dan *noise* dalam proses pengenalan pola data sinyal wicara.
5. Mengetahui pengaruh perbedaan pewicara dalam proses pengenalan pola data sinyal wicara.
6. Menerjemahkan kosakata ke dalam model representasi distribusi.
7. Mengurai perintah menjadi struktur dasarnya berdasarkan pohon sintaksis kalimat yang dimaksud.

Penelitian ini akan berkontribusi terhadap pengembangan sistem interaksi manusia-mesin. Sistem interaksi manusia-mesin diharapkan berkembang menuju sistem interaksi yang lebih alami dan interaktif. Artinya robot tidak hanya menerima perintah saja, tetapi memberikan respon balik terhadap perintah wicara yang diberikan. Diharapkan sistem interaksi yang terbentuk akan mendekati sistem interaksi manusia.

*(Halaman ini sengaja dikosongkan)*

## **BAB 2**

### **KAJIAN PUSTAKA DAN DASAR TEORI**

#### **2.1. Kajian Pustaka**

Robot adalah sebuah mesin yang dapat mengerjakan tugas berdasarkan perintah yang diberikan. Robot harus memiliki antarmuka penggunaan yang memudahkan pengguna untuk berinteraksi dengannya. Antarmuka adalah sebuah sarana dimana interaksi antara pengguna dan mesin terjadi. Secara alami, manusia berinteraksi dengan beberapa cara seperti bahasa isyarat, kontak mata dan wicara.

Bahasa isyarat adalah gerakan satu atau beberapa bagian tubuh yang berguna untuk menyampaikan informasi tertentu. Untuk menghadirkan interface yang atraktif, sebuah robot dilengkapi dengan kamera untuk mendeteksi pergerakan tangan dan isyarat dari pengguna [3]. Secara visual, penerjemahan perintah membutuhkan penanganan khusus karena teknologi ini sangat tergantung dengan pencahayaan, kedalaman gambar dan deteksi obyek [4]. Dalam kasus tertentu seperti penerjemahan perintah kompleks, sistem komunikasi dengan bahasa isyarat akan menemui keterbatasannya. Penginderaan visual hanya mengizinkan robot untuk memahami sebuah perintah sederhana. Bahasa tubuh memiliki keterbatasan dalam menerjemahkan perintah kompleks.

Kontak mata adalah salah satu bentuk dari komunikasi non-verbal yang disebut *oculesic*. Penelitian yang telah dilakukan seperti [5], menggunakan kedipan mata untuk mengontrol pergerakan kursi roda. Antarmuka jenis ini sangat cocok digunakan untuk menerjemahkan perintah sederhana yang membutuhkan respon cepat. Penggunaan kontak mata sebagai antarmuka memiliki kelemahan terutama pada akurasi yang sangat terpengaruh oleh kondisi lingkungan [6].

Antarmuka berbasis wicara mengizinkan robot untuk memahami perintah yang diberikan oleh pengguna menggunakan perintah suara. Antarmuka ini dapat mengakomodasi perintah sederhana maupun kompleks. Sistem ini menggunakan sistem penerjemah wicara. Sistem penerjemah wicara akan memproses sinyal suara menjadi data dan menerjemahkannya menjadi makna kata-kata yang sesuai. Pengenalan wicara memiliki banyak variasi penggunaan sebagai contoh pengendalian robot mobil [7] maupun penugasan robot industri [8].

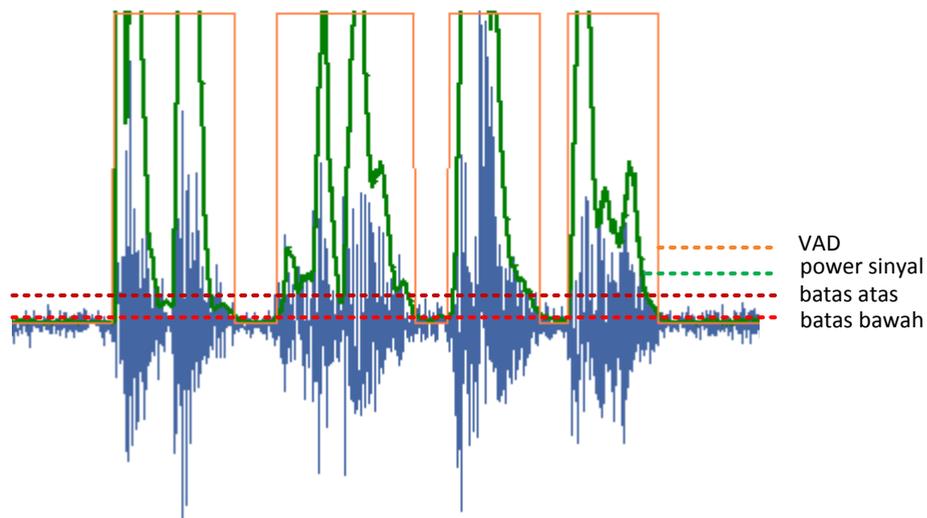
Secara umum, sistem penerjemah wicara dibagi kedalam 2 proses, meliputi ekstraksi ciri dan pengenalan pola. Salah satu sarana identifikasi untuk merepresentasikan ciri sinyal wicara adalah dengan menggunakan cepstral. Reprerentasi cepstral dari spektrum wicara memberikan reprerentasi dari sifat-sifat spektral lokal sinyal untuk analisis frame yang diketahui. *Mel-frequency cepstral coefficients* (MFCC) dan *bark-frequency cepstral coefficients* (BFCC) menjadi kandidat untuk analisis spektral. BFCC memberikan hasil relatif lebih baik dibandingkan MFCC dalam menangani *noise channel* dan *spectral distortion* [9]. Sedangkan pada proses pengenalan pola *cepstral*, akan digunakan neural network (NN). NN menghasilkan akurasi pengenalan lebih baik dibandingkan metode yang telah ada sebelumnya [10].

Sinyal wicara yang telah dikenali akan diolah lebih lanjut menggunakan *natural language processing* (NLP) untuk menghasilkan sistem penerjemah wicara yang lebih natural. NLP diaplikasikan menggunakan *recursive neural network* (RNN). RNN memiliki kemampuan untuk mempelajari aturan sintaksis and sematik kalimat [11].

## **2.2. Dasar Teori**

### **2.2.1. Voice Activity Detection**

Fungsi utama dari *voice activity detection* (VAD) adalah mendeteksi keberadaan dari wicara dengan maksud untuk membantu pemrosesan wicara untuk memberikan batasan awal dan akhir dari segmen sinyal wicara. Fungsi dasar dari sebuah algoritma VAD adalah untuk mengekstrak beberapa fitur atau kuantitas dari sebuah *input* sinyal dan untuk membandingkan besaran tersebut dengan nilai ambang batas, biasanya ciri tersebut diekstrak dari karakteristik *noise* dan sinyal wicara. Pengambilan keputusan bahwa sinyal dalam kondisi aktif dimulai jika besarnya nilai hasil pengujian mendekati nilai batas atas dan berakhir jika nilai mendekati nilai batas bawah. Pemilihan ambang batas yang tepat akan menentukan kesuksesan VAD dalam memberikan keputusan mengenai kondisi sinyal, apakah sinyal dalam kondisi aktif ataupun tidak aktif.



Gambar 2.1. Voice activity detection

Metode yang biasa digunakan adalah menghitung *power* sinyal dalam satu satuan waktu tertentu.

$$p = \frac{1}{n} \sum_{j=1}^N (x_j^2) \quad (2.1)$$

dimana  $p$  adalah *power* sinyal,  
 $x_j$  adalah sinyal wicara pada detik ke- $j$  dan  
 $N$  adalah panjang *moving average filter*

### 2.2.2. Bark-Frequency Cepstral Coefficients

Sinyal suara adalah sinyal yang berubah terhadap waktu secara perlahan (*quasi-stationary*). Untuk rentang waktu yang pendek antara 5 sampai 100 milidetik, karakteristik sinyal ini bisa dianggap stasioner. Namun demikian untuk periode yang cukup panjang (1/5 seconds atau lebih) karakteristik sinyal berubah yang mencerminkan perbedaan bunyi yang diucapkan. Oleh karena itu, *short-time spectral analysis* biasa digunakan untuk mengkarakterisasi sinyal suara. Pada penelitian ini, BFCC digunakan sebagai metode ekstraksi ciri sinyal wicara.

Sebuah sinyal suara yang keluar dari sistem diakibatkan oleh eksitasi masukan dan juga respon dari sistem. Dari sudut pandang pemrosesan sinyal, output dari sistem dapat diperlakukan sebagai konvolusi dari eksitasi input dengan respon sistem. Kadang-kadang dalam proses pembelajaran dan pengolahan sinyal, masing-masing komponen harus dipisah. Proses memisahkan dua komponen disebut sebagai dekonvolusi.

Wicara terdiri dari sumber eksitasi dan komponen sistem saluran vokal. Untuk menganalisis sinyal wicara, model eksitasi dan sistem komponen wicara harus dipisahkan. Tujuan dari analisis *cepstral* adalah untuk memisahkan wicara menjadi sumber dan sistem komponennya tanpa pengetahuan khusus mengenai sumber dan/atau sistem tersebut. Menurut teori sumber produksi wicara, sinyal wicara yang dihasilkan dapat dianggap sebagai konvolusi model urutan eksitasi dan vokal karakteristik saluran filter. Jika  $e(n)$  adalah urutan eksitasi dan  $h(n)$  adalah urutan saluran penyaring vokal, maka wicara  $s(n)$  dapat dinyatakan sebagai berikut.

$$s(n) = e(n) \times h(n) \quad (2.2)$$

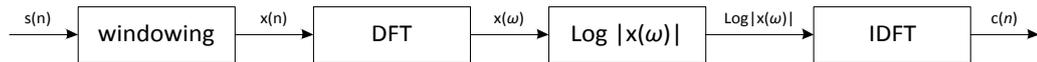
Dari Persamaan (2.2), besarnya spektrum wicara dapat direpresentasikan sebagai berikut.

$$S(\omega) = E(\omega) \times H(\omega) \quad (2.3)$$

Untuk menggabungkan secara linier  $E(\omega)$  dan  $H(\omega)$  dalam domain frekuensi, representasi logaritmik digunakan. Jadi representasi logaritmik dari Persamaan (2.3) akan dituliskan sebagai berikut.

$$\log |S(\omega)| = \log |E(\omega)| + \log |H(\omega)| \quad (2.4)$$

Seperti ditunjukkan dalam Persamaan (4), operasi log mengubah *magnitude* spektrum wicara di mana komponen eksitasi dan komponen saluran vokal dari semula menggunakan operator perkalian menjadi operator penjumlahan. Dengan operator penjumlahan, otomatis kedua komponen tersebut dapat dipisahkan.



Gambar 2.2. Blok diagram perhitungan cepstrum

Pemisahan dapat dilakukan dengan melakukan *inverse discrete fourier transform* (IDFT) dari gabungan log spektrum eksitasi dan sistem saluran vokal komponen. Perlu dicatat bahwa IDFT akan mengubah spektrum linier kembali ke domain waktu, sedangkan IDFT spektrum log mengubah ke domain *quefrensy* atau domain *cepstral* yang mirip dengan domain waktu. Hal ini secara matematis dijelaskan dalam Persamaan (2.4).

$$c(n) = IDFT \{ \log |S(\varphi)| \} = IDFT \{ \log |E(\omega)| \} + IDFT \{ \log |H(\omega)| \} \quad (2.5)$$

*Discrete cosinus transform* (DCT) biasa digunakan terutama dalam BFCC untuk menggantikan IDFT. DCT digunakan untuk menemukan proyeksi ortogonal dari data berdimensi banyak. DCT mirip dengan DFT dengan tanpa mengkalkulasi bagian imajiner dan menghasilkan pengompresan energi yang lebih baik dibandingkan DFT.

Secara garis besar, proses untuk mendapatkan ekstraksi ciri dari sinyal suara menggunakan *BFCC* dapat dilakukan melalui beberapa tahapan. Tahapan tersebut meliputi *pre-emphasis filtering*, *frame blocking*, *windowing*, *fast fourier transform*, *bark-frequency wrapping* dan *discrete cosinus transform*.

### 2.2.2.1. Pre-emphasis Filtering

Pre-emphasis filetering merupakan salah satu jenis *filter* yang sering digunakan sebelum sebuah sinyal diproses lebih lanjut. *Filter* ini mempertahankan frekuensi-frekuensi tinggi pada sebuah spektrum, yang umumnya tereliminasi pada saat proses produksi suara.

$$y_n = x_n - \alpha x_{n-1} \quad (2.6)$$

dimana  $y_n$  adalah sinyal hasil pre-emphasis *filtering*,  
 $x_n$  adalah sinyal hasil proses sebelumnya dan  
 $\alpha$  adalah konstanta pre-emphasis antara 0.9 - 1.

#### 2.2.2.2. Frame Blocking

Karena sinyal suara terus mengalami perubahan akibat adanya pergeseran artikulasi dari organ produksi vokal, sinyal harus diproses dalam satuan waktu tertentu. Panjang *frame* yang digunakan adalah antara 20 - 30 milidetik. Panjang *frame* yang digunakan sangat mempengaruhi keberhasilan dalam analisa spektral. Di satu sisi, ukuran dari *frame* harus sepanjang mungkin untuk dapat menunjukkan resolusi frekuensi yang baik. Tetapi di lain sisi, ukuran *frame* juga harus cukup pendek untuk dapat menunjukkan resolusi waktu yang baik.

Proses *frame blocking* ini dilakukan terus sampai seluruh sinyal dapat diproses. Selain itu, proses ini dilakukan secara *overlapping* untuk setiap *frame*-nya. Panjang daerah *overlap* yang digunakan adalah 25% - 50% dari panjang *frame*. *Overlapping* dilakukan untuk menghindari hilangnya ciri atau karakteristik suara pada perbatasan perpotongan setiap *frame*.

#### 2.2.2.3. Windowing

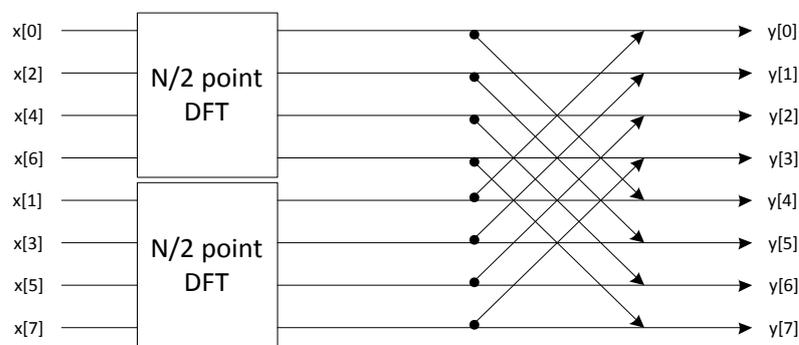
Proses *framing* dapat menyebabkan terjadinya kebocoran spektral atau *aliasing*. *Aliasing* adalah sinyal baru dimana memiliki frekuensi yang berbeda dengan sinyal aslinya. Efek ini dapat terjadi karena rendahnya jumlah sampling rate, ataupun karena proses *frame blocking* sehingga menyebabkan sinyal menjadi *discontinue*. Untuk mengurangi kemungkinan terjadinya kebocoran spektral, maka hasil dari proses *frame blocking* harus melewati proses *windowing*. Sebuah fungsi *window* yang baik harus menyempit pada bagian *main lobe* dan melebar pada bagian *side lobe*-nya.

$$y_n = x_n \left( 0.54 - 0.46 \cos \frac{2\pi n}{N-1} \right) \quad (2.7)$$

dimana  $y_n$  adalah sinyal hasil *windowing* dan  $x_n$  adalah sinyal hasil proses sebelumnya.

#### 2.2.2.4. Fast Fourier Transform

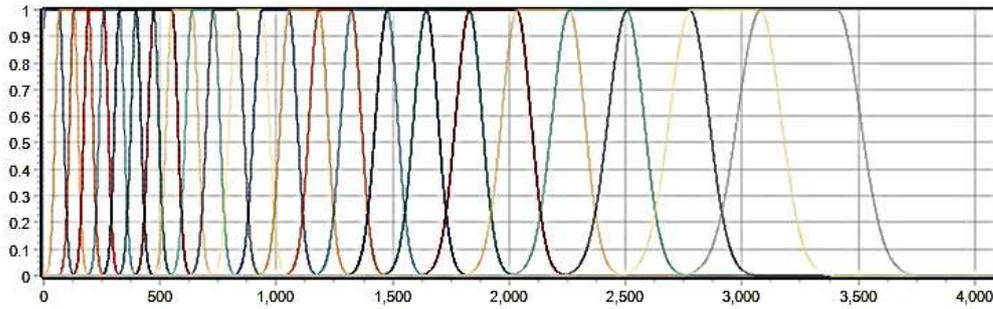
*Fast fourier transform* (FFT) menjadi solusi yang dapat digunakan dalam analisa frekuensi karena kecepatan dan efektifitasnya dalam pemrosesan data. Transformasi ini dapat menghitung deret *fourier* dengan lebih cepat dibandingkan dengan transformasi *fourier* diskrit. Untuk data dengan jumlah ribuan atau bahkan jutaan, penggunaan transformasi *fourier* cepat dapat mengurangi waktu komputasi hingga beberapa kali lipat. FFT memanfaatkan sifat periodik pada DFT seperti pada Gambar 2.3. FFT dalam rentang waktu pendek biasa disebut *short-time fourier transform* (STFT). Adapun ide dibalik metode ini adalah membuat *non-stationary signal* menjadi suatu representasi *stationary signal* dengan memasukkan suatu fungsi *window*. Dalam hal ini, sinyal yang ada dibagi menjadi beberapa *frame* dimana *segmen* yang didapatkan, diasumsikan sebagai *stationary signal*.



Gambar 2.3. Diagram kupu-kupu FFT radix-2 DIT

#### 2.2.2.5. Frequency Wrapping

Tahap ini merupakan proses konvolusi dengan  $n$ -*filter* untuk tiap frame spektrum sinyal. Persepsi ini dinyatakan dalam skala *bark* yang mempunyai hubungan tidak linier dengan frekuensi suara. *Frequency wrapping* umumnya dilakukan dengan menggunakan *filter bank*.



Gambar 2.4. Bark-Frequency *Filter Bank*

*Filter bank* adalah salah satu bentuk dari *filter* yang dilakukan dengan tujuan untuk mengetahui ukuran energi dari frekuensi pada *band* tertentu dalam sinyal suara. *Filter bank* diterapkan pada domain domain frekuensi. Dalam kedua kasus ini, *filter* berjumlah 24 channel yang dilakukan secara linier terhadap frekuensi 0-4 kHz. *Bark filter* dibentuk menggunakan Persamaan 2.8 dan 2.9.

$$f_c = \frac{1960(B_c+0.53)}{26.28-B_c} \quad (2.8)$$

$$B_w = \frac{52548}{B_c^2 - 52.56B_c + 690.39} \quad (2.9)$$

dimana  $f_c$  adalah frekuensi tengah dalam Hz,  
 $B_c$  adalah frekuensi tengah dalam bark dan  
 $B_w$  adalah bandwidth.

Gambar 2.4 menunjukkan *magnitude* dari *bark filter bank*. Bentuk filter akan saling overlapping. *Bark-frequency wrapping* menggunakan representasi konvolusi dalam melakukan *filter* terhadap sinyal, yaitu dengan mengalikan antara spektrum sinyal dengan koefisien *filter bank*. Semakin banyak *channel* yang digunakan, maka detail ciri sinyal semakin tinggi dan jumlah data menjadi lebih besar. Berikut ini adalah rumus yang digunakan dalam perhitungan *bark-frequency wrapping*.

$$y_k = \sum_{n=1}^{N_{Filter}} x_n h_n \quad (2.10)$$

Dimana  $y_k$  adalah hasil konvolusi dengan *magnitude filter bank*,  
 $x_n$  adalah frekuensi sinyal hasil analisa fourier dan  
 $h_n$  adalah koefisien *filter bank*.

### 2.2.2.6. Discrete Cosinus Transform

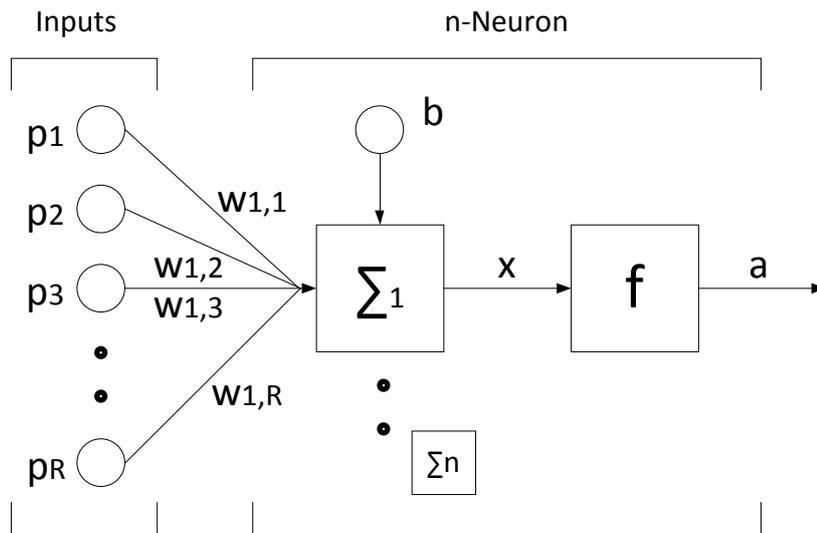
*Cepstrum* didapat dari *discrete cosine transform* (DCT) untuk mendapatkan kembali sinyal dalam domain waktu. Hasilnya disebut sebagai *bark-frequency cepstral coefficient* (BFCC). BFCC bisa didapat dari pendekatan Persamaan (2.11).

$$C_j = \sum_{i=1}^{NFilter} X_i \cos \left( \frac{j\pi (i-0.5)}{NFilter} \right) \quad (2.11)$$

Dimana  $j = 1, 2, 3, \dots, NFilter$ ;  
 $X_i$  adalah koefisien yang didapatkan dari persamaan 2.  
 $C_j$  adalah bark-frequency *cepstral coefficients* ke-j.

### 2.2.3. Neural Network

Secara umum *neural network* (NN) adalah jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan jaringan syaraf manusia. NN ini merupakan sistem adaptif yang dapat merubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan tersebut. Secara sederhana NN adalah sebuah alat pemodelan data statistik non-linier. NN dapat digunakan untuk memodelkan hubungan yang kompleks antara *input* dan *output* untuk menemukan pola-pola pada data. Secara mendasar, sistem pembelajaran merupakan proses penambahan pengetahuan pada NN yang sifatnya kontinuitas sehingga pada saat digunakan pengetahuan tersebut akan dieksploitasikan secara maksimal dalam mengenali suatu objek. *Neuron* adalah bagian dasar dari pemrosesan suatu *neural network*. Di bawah ini merupakan bentuk dasar dari suatu *neuron*.



Gambar 2.5. Struktur dasar *neuron*

Suatu *neuron* dengan  $R$  *input* diperlihatkan dalam Gambar 2.5. Masukan tunggal  $p_1, p_2, \dots, p_R$  masing-masing diberi pembobot oleh unsur-unsur yang bersesuaian dengan  $w[1,1]; w[1,2]; \dots; w[1,R]$  dari matriks bobot  $W$ . Nilai keluaran *neuron* dihitung dengan Persamaan (2.12). Kemudian nilai keluaran akan diskalakan menggunakan fungsi aktivasi  $f(x)$ . Fungsi ini akan menskalakan rentang nilai keluaran menjadi antara -1 hingga 1.

$$x = b + \sum_{i=1}^R (w_{i,j} + p_i) \quad (2.12)$$

$$a = f(x) \quad (2.13)$$

$$f(x) = \frac{2}{1+e^{-ax}} - 1 \quad (2.14)$$

Pada proses propagasi balik, akan dihitung nilai delta berdasarkan selisih target yang ingin dicapai dengan nilai keluaran saat ini dikalikan dengan turunan dari fungsi aktivasi yang digunakan. Proses tersebut dituliskan dalam persamaan berikut ini.

$$\Delta = (a_{target} - a)f'(x) \quad (2.15)$$

$$f'(x) = \frac{(1-x)(1+x)\alpha}{2} \quad (2.16)$$

Nilai pembaruan pembobot dan *bias* dapat dihitung dengan persamaan berikut ini.

$$w_{1,i} = w_{1,i} + \mu\Delta p_i \quad (2.17)$$

$$b = b + \mu\Delta \quad (2.18)$$

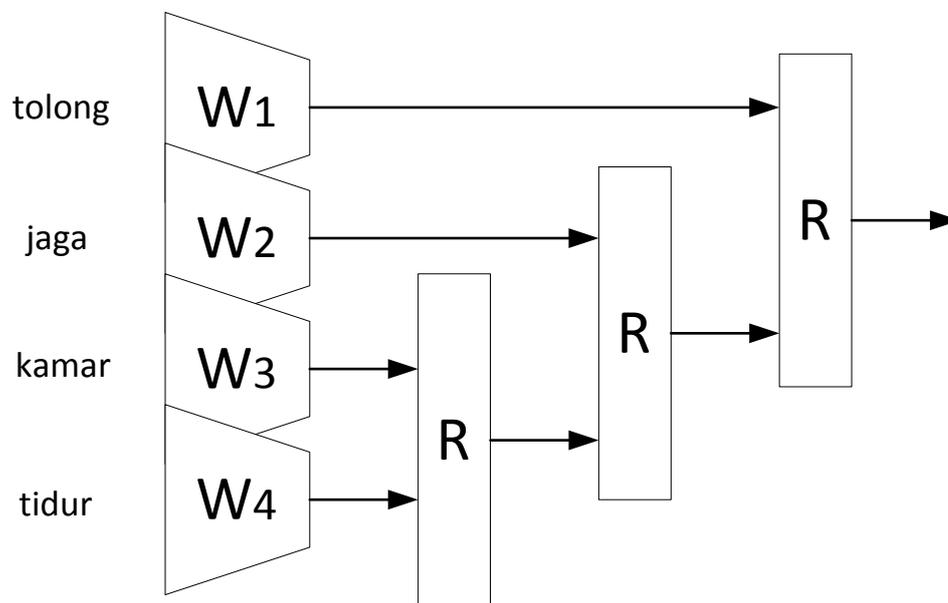
Pada akhirnya, nilai pembaharuan *weight* dan *bias* ini yang dijadikan acuan pada proses pengujian. Nilai ini ibarat memori yang dapat digunakan dalam proses pengenalan pola. Proses pengujian itu sendiri merupakan proses penggolongan pola baru berdasarkan sampel pola yang telah dipelajari sebelumnya.

#### 2.2.4. Recursive Neural Network

Salah satu struktur rekursif yang banyak ditemukan adalah kalimat percakapan sehari-hari. Pada prinsipnya, struktur ini merupakan gabungan dari beberapa struktur kecil yang saling memiliki keterkaitan. Misalkan pada kalimat 'tolong-jaga-kamar-tidur'. Kalimat tersebut tersusun dari gabungan antar kata dimana gabungan tersebut akan membentuk struktur yang lebih besar. Struktur tersebut akan bergabung kembali dengan kata di sebelahnya hingga struktur tersebut lengkap menjadi sebuah kalimat utuh. Struktur pohon sintaksis kalimat tersebut seperti tampak pada Gambar 2.6.

Setiap kata dalam *vector space* ( $W$ ) memiliki keterikatan dengan kata di sebelahnya. Kuat lemahnya ikatan ini mempengaruhi kecondongan arah penggabungan kata. Modul (R) pada Gambar 2.6 merupakan *recursive neural network*. *Recursive neural network* dibentuk dengan mengumpankan balik keluaran dari *neural network* sebagai masukannya. Struktur seperti ini mudah diaplikasikan dalam bahasa pemrograman karena hanya memanfaatkan fungsi pengulangan. RNN dapat diaplikasikan untuk mengurai pohon sintaksis kalimat karena

kemampuannya yang dapat menghitung nilai potensial koneksi antar dua masukannya.



Gambar 2.6. Struktur pohon sintaksis kalimat 'tolong-jaga-kamar-tidur'

## **BAB 4**

### **PENGUJIAN DAN ANALISA SISTEM**

Pengujian dan analisa dilakukan untuk mengetahui fungsional tiap-tiap sub sistem yang telah dibuat. Pengujian yang dilakukan terdiri dari:

1. Pengujian *voice activity detection*.
2. Pengujian ekstraksi ciri sinyal suara.
3. Pengujian sistem identifikasi pola dengan *neural network*.
4. Pengujian variasi kata dan *noise* dalam proses pengenalan pola data sinyal wicara.
5. Pengujian variasi pewicara dalam proses pengenalan pola data sinyal wicara.
6. Pengujian pemetaan kosakata ke dalam model representasi distribusi.
7. Pengujian pemodelan kalimat berdasarkan struktur pohon sintaksisnya.
8. Pengujian variasi model struktur pohon sintaksis dalam proses penerjemahan kalimat.

#### **4.1. Pengujian Voice Activity Detection.**

Tujuan dari pengujian *voice activity detection* adalah menguji pengaruh beberapa variasi variable yang terkait dalam proses pemisahan sinyal suara yang berisi informasi dengan sinyal suara yang tidak berisi informasi.

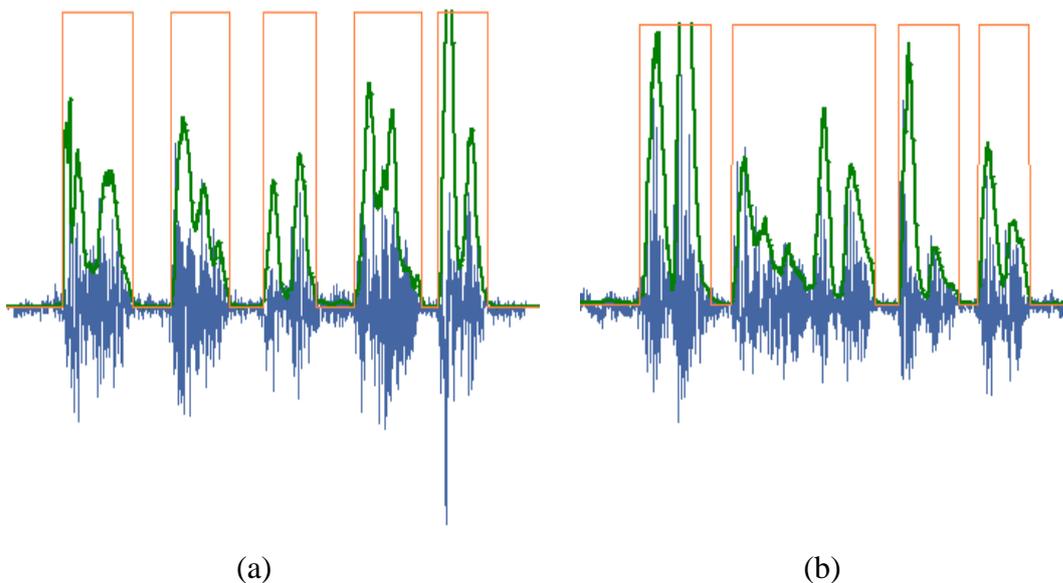
##### **4.1.1. Prosedur Pengujian**

1. Pengujian dilakukan dengan beberapa variasi kecepatan pengucapan kata, kemudian dilakukan pengamatan keputusan VAD yang dihasilkan.
2. Selanjutnya dilakukan pengujian pengaruh variasi lebar *moving average filter* untuk pengambilan keputusan pada VAD.
3. Menguji pengaruh variasi nilai *threshold* untuk pengambilan keputusan pada VAD.

4. Dengan lebar *moving average filter* dan nilai ambang batas yang optimal, dilakukan pengujian akurasi VAD dengan menaikkan rasio signal to *noise* secara bertahap.
5. Akurasi VAD didapat dengan membandingkan tingkat keberhasilan VAD dalam memotong sinyal wicara yang berisi informasi dalam 20 kali percobaan.

#### 4.1.2. Hasil Pengujian dan Analisa

Kecepatan pengucapan kata sangat berpengaruh terhadap kemampuan VAD dalam mendeteksi sinyal yang berisi informasi. Sinyal wicara pada kalimat 'jarvis-tolong-maju-lima-langkah', dapat diproses dengan baik seperti tampak pada Gambar 4.1 (a). Keseluruhan pengucapan kalimat tersebut selesai dalam kurun waktu 2,5 detik. Berarti setiap kata rata-rata diucapkan dalam 0,5 detik. VAD yang telah dirancang, masih dapat bekerja dengan baik pada rentang waktu pengucapan yang lebih lama (dengan jeda antar kata lebih panjang). Akan tetapi untuk pengucapan dengan rentang waktu lebih pendek, ada kalanya VAD masih kesulitan dalam membedakan dua sinyal yang berdampingan (Gambar 4.1 (b)).



Gambar 4.1. VAD pada sinyal wicara 'jarvis-tolong-maju-lima-langkah' pada (a) kondisi ideal (b) kondisi pengucapan terlalu cepat

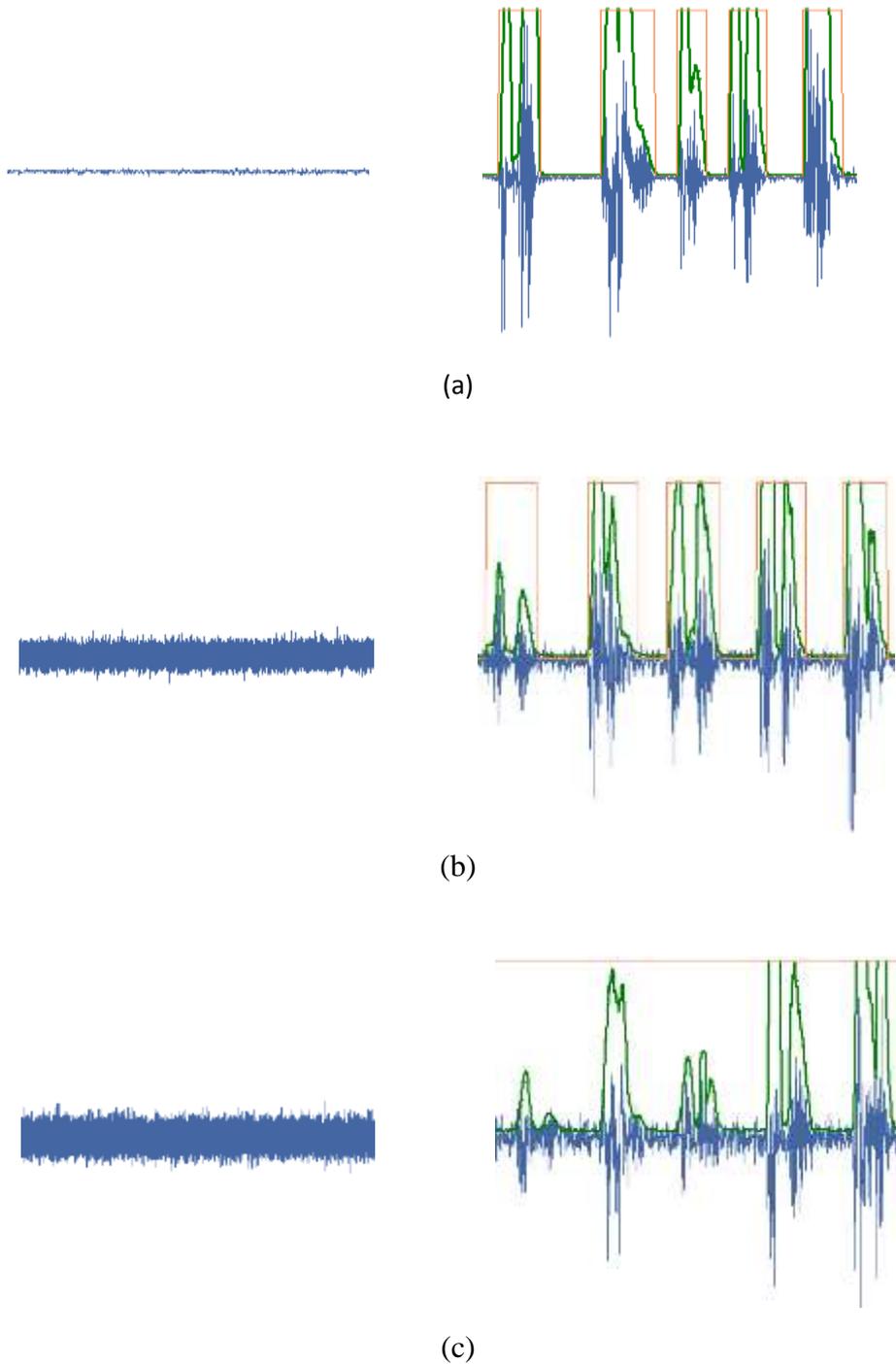
Tabel 4.1. Akurasi *voice activity detection* dengan beberapa variasi SNR

Lebar Filter Moving Average	Threshold	SNR (dB)	Akurasi (%)
165	$1,29 \times 10^{-5}$	66	65
220	$1,29 \times 10^{-5}$	66	85
275	$1,29 \times 10^{-5}$	66	95
330	$1,29 \times 10^{-5}$	66	95
440	$1,29 \times 10^{-5}$	66	80(*)
550	$1,29 \times 10^{-5}$	66	75(*)
275	$9,01 \times 10^{-6}$	66	80
275	$1,03 \times 10^{-5}$	66	85
275	$1,16 \times 10^{-5}$	66	85
275	$1,29 \times 10^{-5}$	66	95
275	$1,42 \times 10^{-5}$	66	90(**)
275	$1,55 \times 10^{-5}$	66	90(**)
275	$1,29 \times 10^{-5}$	10	35
275	$1,29 \times 10^{-5}$	15	35
275	$1,29 \times 10^{-5}$	20	45
275	$1,29 \times 10^{-5}$	25	90
275	$1,29 \times 10^{-5}$	30	95

Pengujian dilanjutkan dengan menguji efektivitas VAD terhadap variasi lebar *moving average filter*, nilai *threshold* dan SNR. Pengujian dilakukan dengan *noise* paling minimal, kemudian dilakukan perhitungan akurasi dari beberapa sampel kata yang diuji. Kemudian *noise* ditingkatkan secara bertahap dan diuji kembali akurasi yang dihasilkan.

Pengujian dengan lebar filter sebesar 275-330 menghasilkan akurasi yang terbaik. Untuk lebar filter  $> 330$ , terdapat catatan khusus (\*). Akurasi yang dihasilkan tidak konstan untuk beberapa kali percobaan. Hal ini dapat diakibatkan jumlah filter yang terlalu panjang menyebabkan VAD sulit untuk menentukan nilai

*threshold* yang optimal. *Filter moving average* yang terlalu panjang menyebabkan nilai rata-rata keluaran menjadi terlalu landai. Hal ini mengakibatkan sulit untuk menentukan kondisi aktivitas sinyal sehingga akurasi VAD menjadi berkurang.



Gambar 4.2. Sinyal noise dan pengambilan keputusan VAD pada sinyal wicara 'jarvis-tolong-maju-lima-langkah' dengan (a) SNR 56, (b) SNR 25 dan (c) 21 dB

Kemudian untuk pengujian variasi nilai *threshold* yang telah dinormalisasi, didapat nilai optimal  $1,29 \times 10^{-5}$  (Gambar 4.2). Nilai *threshold* lebih dari  $1,29 \times 10^{-5}$  mendapat catatan khusus (\*\*). Hasil yang didapat cukup baik, hanya saja akurasi tidak konsisten untuk beberapa kali percobaan. Hal ini disebabkan VAD terlalu sensitif dalam menentukan aktivitas sinyal sehingga ada kalanya suatu kata yang terdiri lebih dari satu suku kata, akan terdeteksi sebagai dua bagian terpisah. VAD yang terlalu sensitif juga mengakibatkan penentuan sinyal kondisi aktif menjadi kurang sempurna karena bagian awal dan akhir sinyal akan terpotong.

Untuk pengujian variasi SNR, didapat data bahwa kondisi optimal VAD dapat menghasilkan akurasi yang baik dengan  $SNR \geq 25$  dB. Pengujian dengan  $SNR < 25$  dB menghasilkan tingkat akurasi yang sangat rendah. VAD kesulitan membedakan sinyal yang berisi informasi maupun tidak karena dalam kondisi jeda, *power* sinyal sudah sangat tinggi sehingga dikategorikan sebagai sinyal aktif. Perlu dilakukan penyesuaian kembali nilai ambang batas yang diberikan agar VAD dapat memberikan akurasi yang lebih tinggi. Hal ini tampak pada Gambar 4.3, dimana VAD tampak aktif di sepanjang sinyal jeda maupun sinyal wicara.

## **4.2. Pengujian Ekstraksi Ciri Sinyal Suara.**

Tujuan dari pengujian ekstraksi ciri sinyal suara adalah menguji setiap tahapan ekstraksi ciri menggunakan BFCC dan menganalisa pengaruh variabel yang terkait dengan keluaran yang dihasilkan.

### **4.2.1. Prosedur Pengujian**

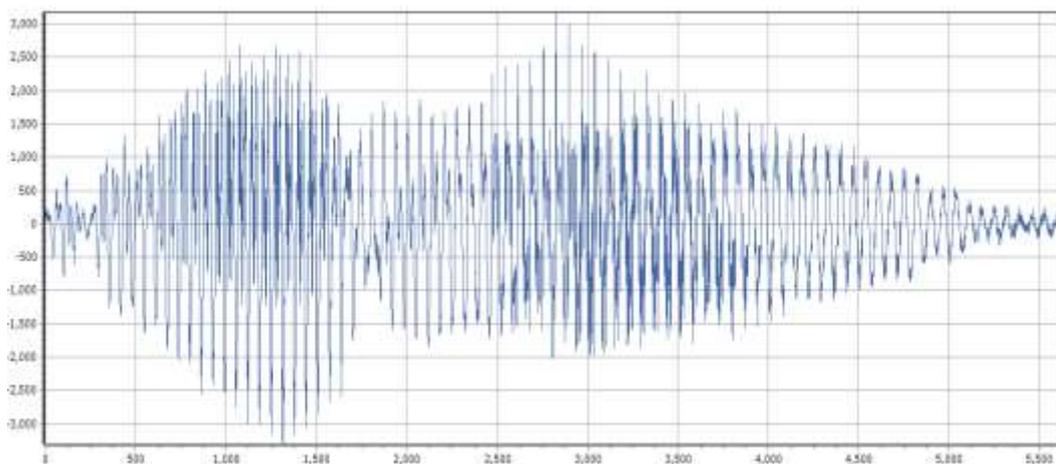
1. Pengujian dilakukan dengan melihat keluaran yang dihasilkan untuk setiap tahapan dalam BFCC.
2. Pengucapan sebuah kata diuji menggunakan *neural network*, kemudian dilakukan pengujian kembali untuk kata yang sama.
3. Selanjutnya dilakukan pengamatan apakah ada kemiripan dari keluaran yang dihasilkan.

4. Untuk kata yang berbeda, BFCC diujikan kembali hingga menghasilkan keluaran tertentu, kemudian diuji untuk kedua kalinya kata yang sama dengan pengucapan yang sedikit berbeda dan dilakukan pengamatan apakah ada kesamaan keluaran dengan keluaran sebelumnya.

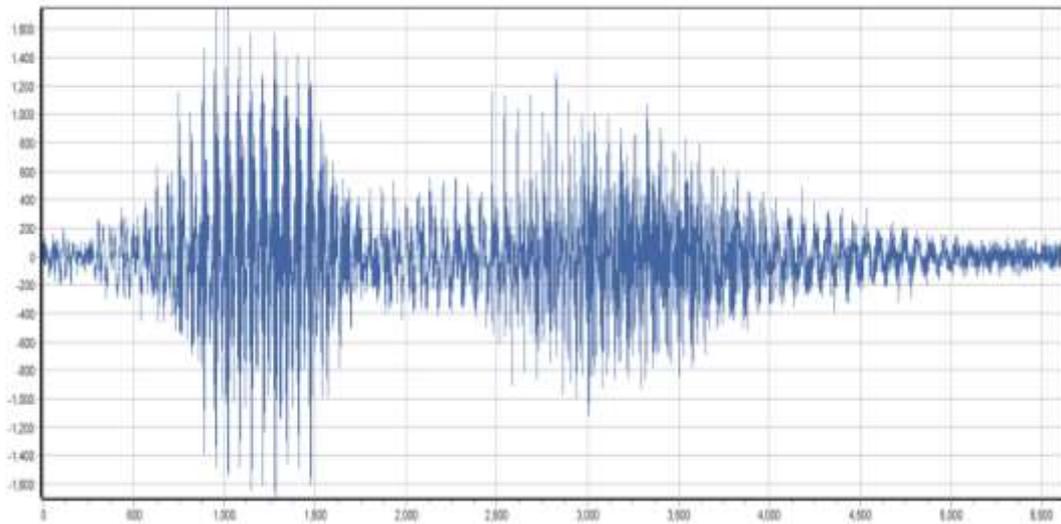
#### 4.2.2. Hasil Pengujian dan Analisa

Sinyal keluaran dari *pre-emphasis filtering* tampak mengalami penguatan pada frekuensi tinggi. Pada keluaran *pre-emphasis*, sinyal tampak lebih padat tetapi dengan bentuk yang mirip dengan sinyal asli. Apabila sinyal tidak melalui *pre-emphasis filtering*, detail pada frekuensi tinggi tidak terbaca. Padahal, beberapa sinyal wicara memiliki karakteristik yang dominan pada frekuensi tinggi. Penggunaan *pre-emphasis* sangat mempengaruhi keberhasilan dari sistem penerjemah wicara. Keluaran sinyal hasil *pre-emphasis filtering* terlihat pada Gambar 4.4.

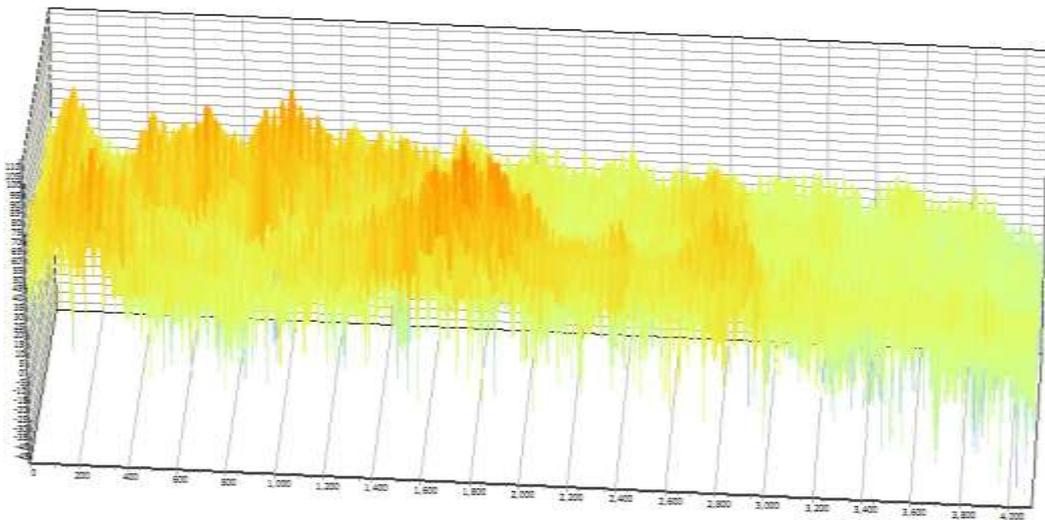
Selanjutnya sinyal akan dipotong-potong dalam kondisi overlap 30 %. Hasilnya seperti tampak pada Gambar 4.5. Sinyal dalam domain waktu diubah ke dalam domain frekuensi dengan rentang 0 – 4 kHz. Analisa *fourier* dilakukan per-*frame* sehingga setiap *frame*-nya akan menghasilkan 4000 data *magnitude* frekuensi sinyal.



Gambar 4.3. Sinyal mentah hasil pemotongan VAD

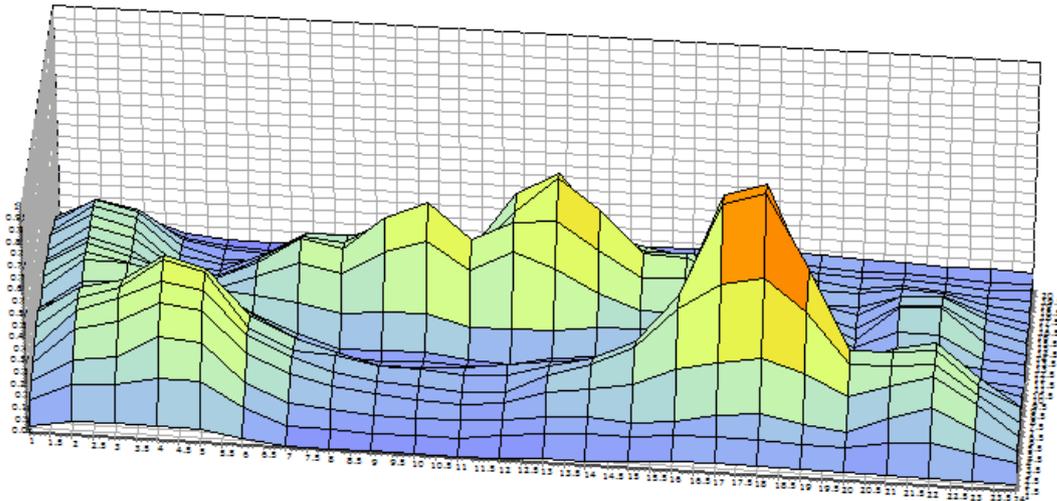


Gambar 4.4. Keluaran sinyal setelah melalui pre-emphasis *filtering*

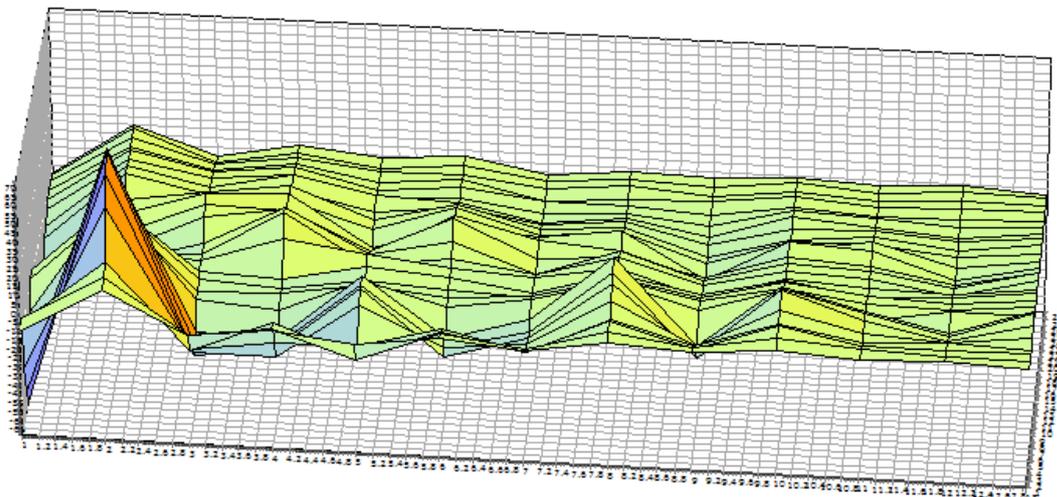


Gambar 4.5. Keluaran sinyal pada proses transformasi fourier cepat

Keluaran dari proses *frequency wrapping* tampak seperti pada Gambar 4.6. Dari 4000 data untuk setiap *frame* akan dimampatkan menjadi 24 data untuk setiap *frame*. *Frequency wrapping* dilakukan dengan mengalikan *magnitude* pada setiap frekuensi sinyal dengan 24 *filter bank* yang telah dibentuk sebelumnya. Pada pengujian kata 'jarvis', keluaran *bark frequency wrapping* berubah setiap waktu mengikuti pengucapan setiap huruf yang merangkai kata yang diucapkan. Kondisi perubahan sinyal inilah yang dapat dijadikan acuan untuk mempelajari karakteristik setiap sinyal wicara.

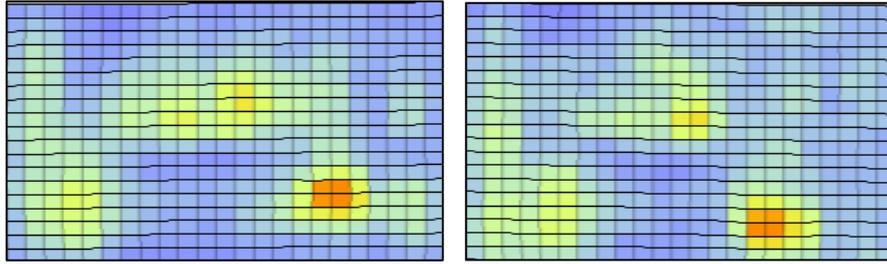


Gambar 4.6. Ciri sinyal wicara setelah melalui *bark-frequency wrapping*



Gambar 4.7. Ciri sinyal setelah melalui proses DCT

*Bark cepstral* didapat dengan menghitung DCT pada data *frequency wrapping* seperti tampak pada Gambar 4.7. Data dalam domain frekuensi akan diubah kembali ke dalam domain waktu. Proses ini mirip dengan perhitungan *invers fourier transform* dimana sebagian kecil fitur pada frekuensi tinggi akan diabaikan. BFCC pada dasarnya mengambil informasi frekuensi dari sinyal wicara kemudian diproses *filter-filter* dengan rentang berbeda sehingga dihasilkan *cepstral*. Pada penelitian ini, fitur yang diproses adalah *bark frequency wrapping*. Untuk pengucapan kata yang sama, keluaran yang dihasilkan akan terlihat hampir sama seperti pada Gambar 4.8.



Gambar 4.8. Ciri sinyal wicara untuk kata 'jarvis' dalam dua kali percobaan.

Kemiripan keluaran akan semakin berbeda jika pengucapan yang dilakukan dengan frekuensi dan gaya yang berbeda pula. Akan tetapi masih ada detail informasi yang menunjukkan kesamaan sinyal untuk kata yang sama. Salah satu cara untuk mengatasi keragaman intonasi maupun warna suara adalah dengan melakukan pengenalan pada *neural network* untuk setiap variasi yang ada.

### 4.3. Pengujian Sistem Identifikasi Pola Dengan Neural Network.

Tujuan dari pengujian sistem identifikasi pola adalah menguji pengaruh variasi variable *neural network* seperti *beta*, *alpha* dan *learning rate*.

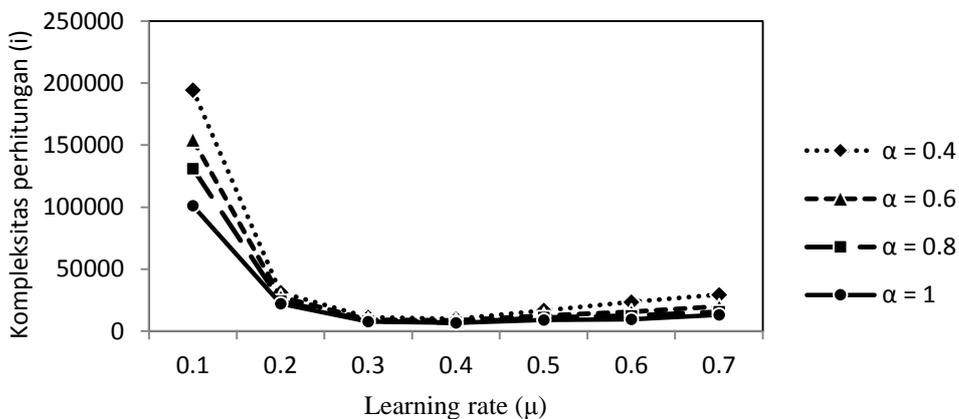
#### 4.3.1. Prosedur Pengujian

1. Pengujian dilakukan dengan mengubah-ubah nilai *beta* ( $\beta$ ), *alpha* ( $\alpha$ ) dan *learning rate* ( $\mu$ ).
2. Kemudian dilakukan pengamatan terhadap kompleksitas perhitungan dimana banyaknya dapat dilihat dari jumlah iterasi yang dibutuhkan dalam proses *learning* untuk mencapai konvergen.
3. Selanjutnya dilakukan pengujian pengaruh variasi *beta* ( $\beta$ ) terhadap akurasi yang dihasilkan.
4. Akurasi dihitung dengan membandingkan keluaran pada proses pengujian dengan keluaran yang seharusnya sesuai target. Percobaan ini dilakukan beberapa kali, kemudian dilakukan perhitungan rata-rata dari akurasi yang didapat. Akurasi didapat dengan membandingkan keluaran yang diharapkan dengan keluaran *neuron* yang sebenarnya.

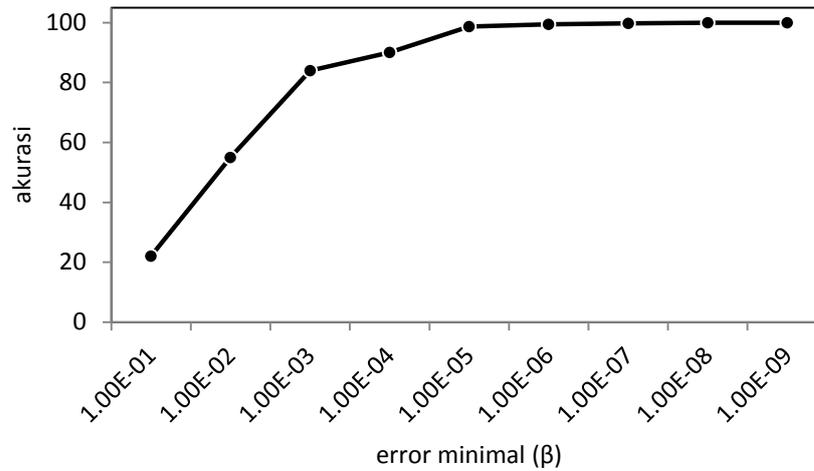
### 4.3.2. Hasil Pengujian dan Analisa

Setiap variabel *neural network* sangat berpengaruh terhadap kompleksitas perhitungan. Kompleksitas perhitungan semakin bertambah dengan semakin kecilnya konstanta pembelajaran ( $\mu$ ). Nilai  $\mu$  melambangkan kecepatan pembelajaran *neuron* dalam memperbarui nilai *weight* dan *bias* agar sesuai dengan target yang ingin dicapai. Melalui percobaan yang telah dilakukan, nilai  $\mu$  yang optimal adalah sekitar 0,3. Semakin besar nilai  $\mu$  tidak berpengaruh banyak terhadap kompleksitas perhitungan. Akan tetapi nilai  $\mu$  yang lebih kecil dapat menambah kompleksitas perhitungan secara dramatis. Konstanta kemiringan kurva ( $\alpha$ ) juga mempengaruhi kompleksitas perhitungan. Semakin rendah nilai  $\alpha$ , maka kompleksitas perhitungan semakin bertambah. Kompleksitas perhitungan akan mempengaruhi lama proses pembelajaran. Dengan perhitungan yang lebih efisien, *resource* komputer dapat lebih dihemat dan pekerjaan dapat diselesaikan lebih cepat.

Pada pengujian akurasi proses klasifikasi, didapat data bahwa semakin kecil nilai beta yang diberikan, semakin tinggi akurasi. akurasi mulai membaik dengan nilai  $\beta \leq 10^{-5}$ . Perlu diingat bahwa nilai  $\beta$  yang semakin kecil akan meningkatkan kompleksitas perhitungan sehingga waktu pembelajaran akan semakin panjang. Nilai  $\beta$  yang terlalu kecil malah akan memperlambat proses pembelajaran untuk mencapai konvergen.



Gambar 4.9. Pengaruh variasi variabel *neural network* terhadap kompleksitas perhitungan



Gambar 4.10. Pengaruh variasi minimum error terhadap akurasi pengenalan pola

#### 4.4. Pengujian Variasi Kata Dan *Noise* Dalam Proses Pengenalan Pola Data Sinyal Wicara.

Tujuan dari pengujian variasi kata dan *noise* dalam proses pengenalan pola sinyal wicara adalah untuk menguji pengaruh banyaknya variasi kata dan variasi *noise* dengan akurasi yang dihasilkan.

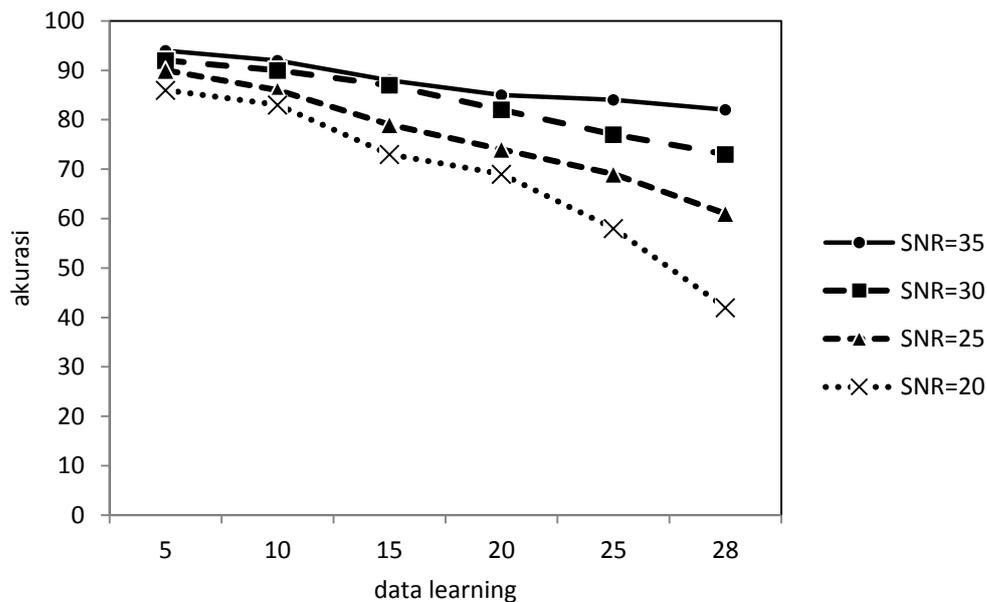
##### 4.4.1. Prosedur Pengujian

1. Pengujian dilakukan dengan menyusun 30 kosakata yang akan digunakan untuk proses identifikasi.
2. Mengambil sampel data sebanyak 10 buah variasi sinyal wicara untuk setiap kosakata.
3. Menjalankan proses pelatihan *neural network* hingga nilai *mean square error* kurang dari  $\beta$ .
4. Menghitung akurasi pengenalan wicara dengan mencoba variasi sinyal wicara sesuai dengan kosakata yang dilatihkan. Pengujian ini dilakukan dengan mencatat SNR.
5. Selanjutnya langkah ke-4 diulang kembali dengan variasi nilai SNR yaitu 35 dB, 30 dB, 25 dB dan 20 dB.

#### 4.4.2. Hasil Pengujian dan Analisa

Pengujian dilakukan dengan 2 *hidden layer* yang masing-masing *layer* memiliki jumlah *neuron* sesuai dengan *neuron output*. Sesuai dengan Gambar 4.11, semakin bertambahnya variasi data pembelajaran, maka akurasi *neural network* semakin menurun. Dengan semakin banyaknya *sample* pembelajaran, proses untuk mencapai konvergen juga semakin lama dan ada kemungkinan proses tersebut gagal. Kegagalan biasa diakibatkan kombinasi *weight* dan *bias* NN yang tidak dapat mengakomodasi variasi pola yang diberikan. Memperbesar jumlah *neuron* dapat dijadikan alternatif agar proses pembelajaran dapat mencapai konvergen dengan konsekuensi kompleksitas perhitungan semakin meningkat.

Kegagalan penerjemahan wicara juga bisa disebabkan pengucapan kata yang mirip, misalkan pada kata ‘maju’ dan ‘laju’. Karena tidak termasuk ke dalam pengucapan 28 kosakata yang telah ditentukan sebelumnya, maka kata ‘laju’ maupun kata lain yang mirip dalam hal pengucapan harus dilatihkan dengan target keluaran bukan merupakan salah satu dari kosakata yang dimaksud. Perlu dilakukan proses *training* yang lebih banyak untuk kata-kata dengan pengucapan yang mirip sehingga nantinya NN mampu menghasilkan detail diferensiasi yang akurat untuk kata-kata tersebut.



Gambar 4.11. Variasi pengaruh SNR terhadap akurasi pengenalan kata

Pada pengujian selanjutnya, dilihat pengaruh variasi SNR terhadap akurasi identifikasi pola sinyal wicara. SNR didapat dengan mengurangkan *power* rata-rata sinyal wicara dengan *power* rata-rata sinyal *noise* dalam dB. Masih berdasarkan Gambar 4.11, semakin tinggi SNR menghasilkan akurasi pengenalan pola yang semakin tinggi. SNR yang semakin besar megindikasikan bahwa jarak perbedaan sinyal wicara dan sinyal *noise* semakin lebar. Hal ini mempengaruhi akurasi VAD dalam proses pemotongan sinyal wicara. Nilai SNR yang besar akan mempermudah batasan yang diberikan untuk membedakan pola sinyal wicara dengan *silent*. Sebaliknya SNR yang rendah akan mengakibatkan data yang diproses VAD menjadi tidak akurat. Akibatnya NN banyak mengalami kesalahan dalam proses identifikasi pola. Untuk meningkatkan akurasi, sampel pembelajaran NN dapat ditingkatkan dengan penambahan data pola sinyal wicara pada kondisi SNR rendah. Akurasi tertinggi didapat dengan SNR 35 dB yaitu mencapai 82 %.

#### **4.5. Pengujian Variasi Pewicara Dalam Proses Pengenalan Pola Data Sinyal Wicara.**

Tujuan dari pengujian variasi pewicara adalah menguji seberapa akurat sistem penerjemah kata dapat membedakan warna suara beberapa pewicara yang berbeda.

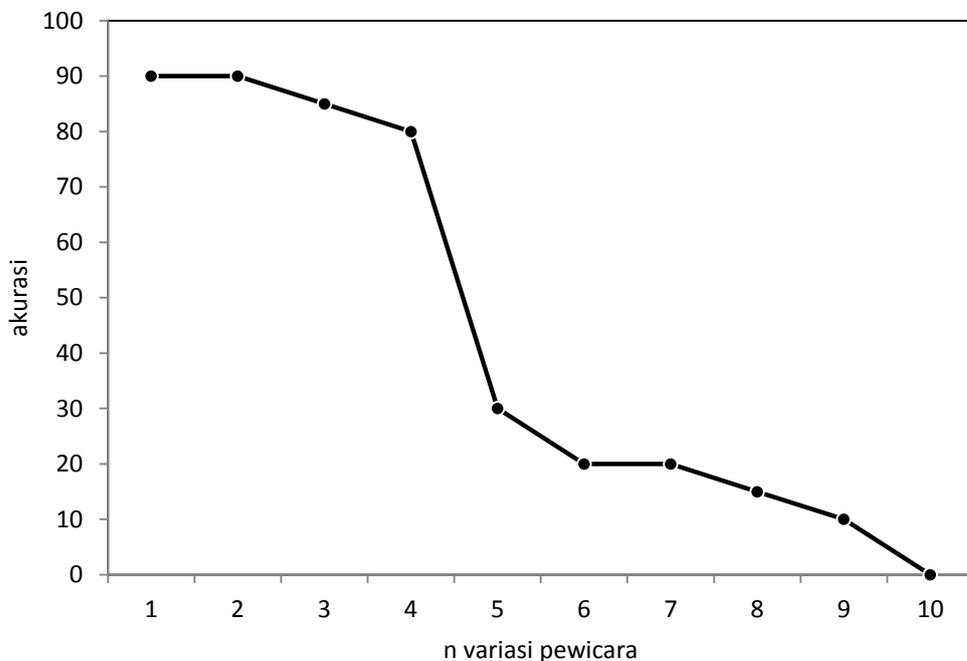
##### **4.5.1. Prosedur Pengujian**

1. Pengujian dilakukan dengan kosakata yang diucapkan oleh 10 orang pewicara.
2. Masing-masing pewicara mengucapkan kata yang telah ditentukan sebelumnya.
3. Selanjutnya dilakukan pengujian dimana pewicara akan mengucapkan kata yang telah diujikan sebelumnya sebanyak 20 kali secara acak, kemudian dicatat akurasi yang dihasilkan sistem pembelajaran pola dalam menentukan pewicara yang dimaksud.

#### 4.5.2. Hasil Pengujian dan Analisa

Akurasi sangat ditentukan oleh banyaknya variasi keluaran yang diharapkan. Semakin tinggi variasi obyek klasifikasi, maka semakin rendah akurasi yang dihasilkan sesuai dengan Gambar 4.12. Setiap kata yang diucapkan oleh pewicara yang berbeda memiliki ciri khusus yang berbeda satu sama lain. Hal ini sangat bergantung pada warna suara masing-masing individu. Akan tetapi pada dasarnya, BFCC dan NN dapat digunakan untuk mempelajari pola data untuk pewicara beserta sinyal wicaranya. Hanya saja, akurasi NN akan menurun seiring dengan semakin banyaknya pola data yang mampu untuk diidentifikasi.

Akurasi sangat menurun pada pengujian dengan variasi lima pewicara atau lebih. Hal ini dapat disebabkan karena sinyal wicara untuk satu kata bisa dikatakan memiliki ciri yang hampir mirip. Untuk mempelajari pola warna suara masing-masing individu, perlu dilakukan pengambilan sampel sinyal yang lebih banyak dan kompleks. Hal ini yang menyebabkan sinyal wicara satu kata yang singkat tidak dapat dijadikan acuan untuk mengidentifikasi identitas pewicara. Pengenalan pewicara dapat dilakukan dengan sampel data yang lebih kompleks dan detail.



Gambar 4.12. Pengaruh jumlah pewicara terhadap akurasi pengenalan pewicara

#### **4.6. Pengujian Pemetaan Kosakata Ke Dalam Model Representasi Distribusi.**

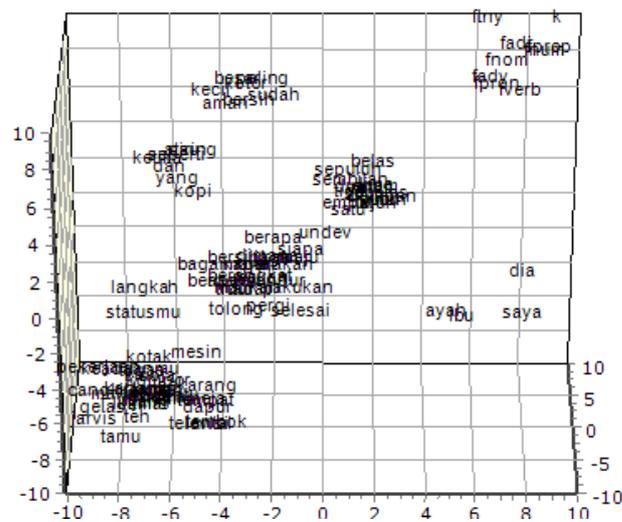
Tujuan dari pengujian pemetaan kosa kata ke dalam model representasi distribusi adalah menguji seberapa efektif *neural network* dapat mengklasifikasikan obyek yang baru ke dalam vektor yang dekat dengan obyek dengan kelas yang sama.

##### **4.6.1. Prosedur Pengujian**

1. Pengujian dilakukan dengan menyusun 100 kosakata beserta label kelas kata yang berkorelasi dengan perintah bagi robot.
2. Kemudian NN dikondisikan untuk mempelajari pembagian kelas kata dari 100 kosakata yang telah disusun.
3. Masukan NN berupa 100 kosakata direpresentasikan dalam representasi one-hot.
4. Keluaran NN diatur sesuai dimensi VSM yang diinginkan. Pada penelitian ini, 100 kosakata akan direpresentasikan dalam VSM 3 dimensi.
5. Sampel pembelajaran diambil sejumlah kelas kata akan dibedakan, sedangkan kosakata yang lainnya akan menjadi bahan pengujian untuk melihat hubungan jarak dalam vctor space sesuai dengan kelas kata yang dimaksud.

##### **4.6.2. Hasil Pengujian dan Analisa**

Pemetaan SVM dari 100 kosakata tampak pada Gambar 4.13. Setiap kata akan diberi label kelasnya masing-masing. Keluaran dari *neural network* yang diujikan adalah vektor 3 dimensi yang sesuai dengan kategori kosakata yang dimaksud. Dari 8 kelas kata yang diberikan, kosakata dapat diklasifikasikan dengan baik. Hal ini terlihat dari penggerombolan kata yang memiliki label kelas yang sama.



Gambar 4.13. Pemetaan 100 kosakata menggunakan VSM

Beberapa kata yang bergerombol dalam VSM berarti memiliki jarak *vector space* akan semakin dekat seiring dengan kesamaan kelas kata sedangkan untuk kelas kata yang berbeda, jarak yang dihasilkan juga akan bertambah. Pemetaan seperti ini akan sangat menentukan keberhasilan dari pemrosesan bahasa natural dimana proses tersebut membutuhkan korelasi jarak vektor antar kata untuk menentukan hubungan sintaksis yang sesuai dengan sistem tata bahasa.

#### 4.7. Pengujian Penguraian dan Penerjemahan Perintah Berdasarkan Struktur Pohon Sintaksisnya.

Tujuan dari pengujian penguraian struktur kalimat adalah menguji kemampuan RNN dalam mengurai struktur sintaksis pada kalimat yang baru diujikan. Kemudian diuji pula seberapa efektif RNN dapat mempelajari beberapa variasi pohon sintaksis dan seberapa akurat proses penguraian dapat dilakukan.

##### 4.7.1. Prosedur Pengujian

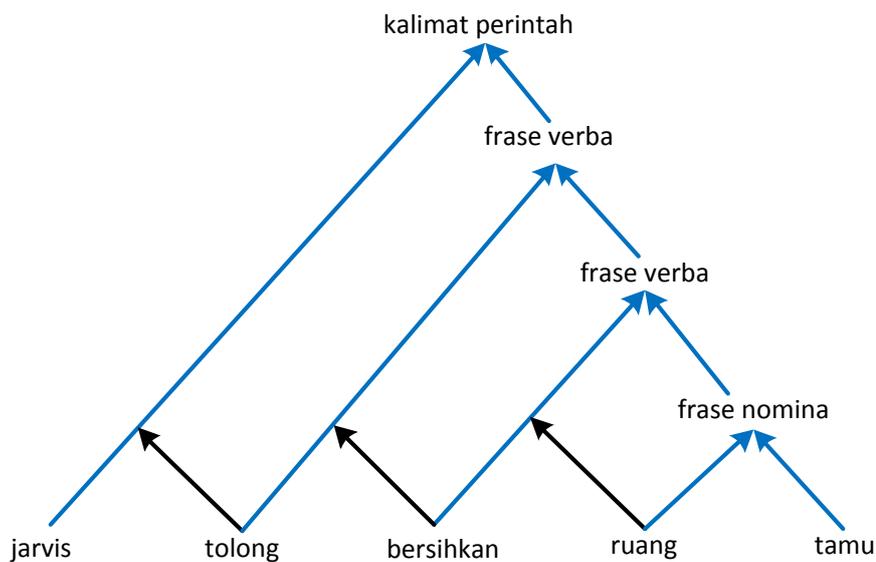
1. Pengujian dilakukan dengan menyusun struktur pohon pada kalimat 'jarvis-tolong-bersihkan-ruang-tamu'.
2. Kemudian RNN dikondisikan untuk mempelajari hubungan setiap kata yang saling berdampingan pada kalimat tersebut.

3. *Input neuron* diatur sebagai *vector space* 3 dimensi dari 100 kosakata dan frase yang telah dibuat sebelumnya.
4. Keluaran *neuron* diatur sebagai klasifikasi frase dari dua kata masukan sesuai susunan sintaksis Bahasa Indonesia.
5. Keluaran berupa klasifikasi frase dengan koefisien terbesar diubah menjadi *vector space* berdasarkan *database* yang telah disusun sebelumnya.
6. Keluaran berupa *vector space* frase akan diumpankan kembali sebagai *input* yang akan bersaing kembali untuk memperoleh nilai koefisien terbesar dari keluaran RNN.
7. Proses ini akan berulang hingga struktur kata dalam kalimat utuh selesai dipelajari.
8. Pengujian dilakukan dengan kalimat yang mirip dalam proses pembelajaran, seperti contoh berikut:  
'jarvis-tolong-bersihkan-ruang-tidur',  
'jarvis-tolong-jaga-kamar-tidur',  
'jarvis-tolong-jaga-ruang-tamu' dsb.
9. Menganalisa struktur hasil parsing yang dihasilkan dengan RNN.
10. Selanjutnya dilakukan pengujian akurasi penguraian kalimat dengan menyusun segala kemungkinan koneksi yang dapat dibentuk dari beberapa variasi kosakata.
11. Melakukan proses pembelajaran RNN dari hubungan koneksi yang telah disusun sebelumnya.
12. Menghitung akurasi dari proses pengujian untuk 100 kosakata dalam *vector space model*.

#### **4.7.2. Hasil Pengujian dan Analisa**

Pembelajaran struktur pada RNN dapat dilakukan dengan memenggal setiap koneksi kata dalam kalimat, menyusunnya satu persatu kemudian memberikan target keluaran sesuai dengan struktur yang akan dibentuk. Kalimat yang diujikan adalah 'tolong-bersihkan-ruang-tamu'. Kemudian nama 'jarvis' ditambahkan pada awal kalimat sebagai penanda bagi robot bahwa dialah yang sedang diperintah.

Kalimat 'jarvis-tolong-bersihkan-ruang-tamu', memiliki beberapa koneksi antar kata yaitu 'jarvis-tolong', 'tolong-bersihkan', 'bersihkan-ruang', dan 'ruang-tamu' seperti tampak pada Gambar 4.14. Penentuan klasifikasi kelas frase harus ditentukan dari hubungan koneksi antar kata. Koneksi 'ruang-tamu -> frasa nomina' digunakan sebagai sampel pembelajaran yang memiliki nilai potensial tinggi, sedangkan koneksi yang lain memiliki nilai potensial rendah.



Gambar 4.14. Proses pengenalan struktur sintaksis pada kalimat 'jarvis-tolong-bersihkan-ruang-tamu'

```

K.{
  KBnd.jarvis FVerb.{
    KKrj.tolong FVerb.{
      KKrj.bersihkan FPron.{
        FNom.{
          KBnd.ruang KPron.tamu}
        KPron.saya
      }
    }
  }
}

```

Gambar 4.15. Keluaran struktur sintaksis pada aplikasi yang dibuat

Keluaran dari proses parsing kalimat seperti tampak pada Gambar 4.15. Terlihat bahwa kalimat yang sebelumnya dijadikan sampel pembelajaran dapat dikenali dengan baik pada proses pengujian. Kemudian dilakukan pengujian untuk kalimat yang memiliki kemiripan struktur yaitu 'jarvis-tolong-bersihkan-ruang-tidur', 'jarvis-tolong-jaga-kamar-tidur' dan 'jarvis-tolong-jaga-ruang-tamu'. Ternyata struktur ketiga kalimat tersebut juga dapat dikenali dengan baik.

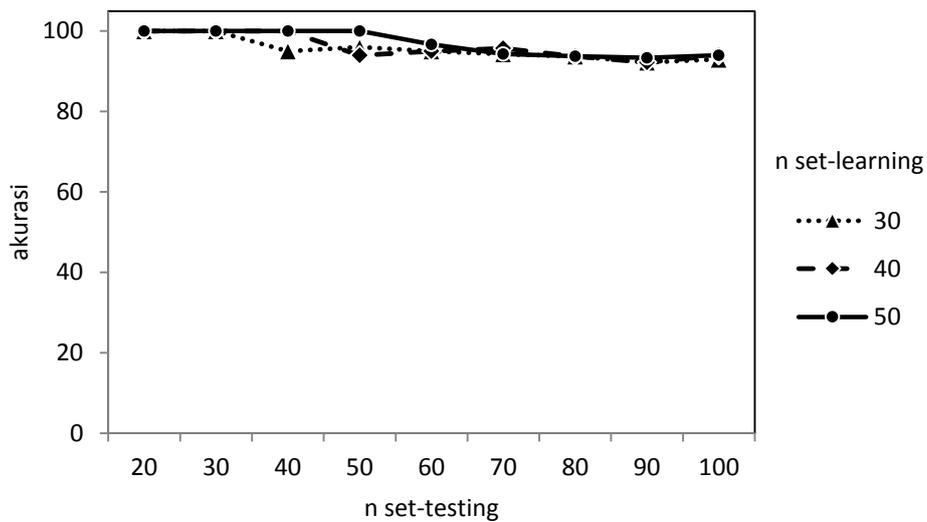
```
check(current_location)
while(current_location<>coordinate(25,42)){
    find(x,y)
}
activate(cleaning_mode)
```

Gambar 4.16. Keluaran terjemahan dalam bahasa mesin sesuai dengan perintah yang diberikan

Keluaran dari kalimat perintah 'jarvis-tolong-bersihkan-ruang-tamu' seperti tampak pada Gambar 4.16. Untuk setiap koneksi kata yang berhasil terjalin, proses *decoding* dilakukan sebagai respon untuk frase baru yang terbentuk. Misalnya untuk frase 'ruang-tamu' akan diterjemahkan sebagai koordinat lokasi ruang tamu dalam persepsi robot, sehingga robot dapat memahami lokasi yang akan dia tuju. Begitu pula untuk frase kerja 'bersihkan-(ruang-tamu)', akan diterjemahkan oleh robot untuk membersihkan lokasi yang dituju sebelumnya. Proses ini berjalan sekuensial sehingga perintah yang dimaksud dapat dikerjakan oleh robot secara berurutan.

Pada Gambar 4.17, akurasi pengenalan kata semakin menurun sesuai dengan semakin banyaknya hubungan koneksi kosakata yang diujikan. Dengan menambah sampel pembelajaran, akurasi dapat meningkat walaupun tidak signifikan. Akurasi didapat sekitar 93 % untuk 100 kosakata VSM dengan 50 data pembelajaran. Dengan jumlah data pembelajaran sebanyak ini, RNN memiliki kemungkinan untuk sulit mencapai konvergen dikarenakan menipisnya jumlah kombinasi *weight* yang dapat digunakan dalam mempelajari pola. Untuk mengatasi hal ini, maka jumlah *neuron* dapat ditambah dengan konsekuensi kompleksitas perhitungan semakin meningkat.

Keberhasilan RNN dalam mempelajari pola struktur kalimat sangat dipengaruhi oleh pemetaan kata dengan VSM. Pemetaan yang baik harus mampu mengakomodasi struktur sintaksis dan semantik setiap kata. Kesalahan dalam memetakan VSM dan banyaknya variasi kata dapat mengakibatkan pembelajaran RNN tidak mencapai konvergen.



Gambar 4.17. Pengaruh variasi data learning dan testing terhadap akurasi pengenalan struktur kalimat

## **BAB 5**

### **PENUTUP**

Dari hasil perancangan, pembuatan, pengujian dan analisa sistem pada penelitian dengan judul “SISTEM IDENTIFIKASI PERINTAH BAHASA INDONESIA PADA PROTOTYPE ROBOT PELAYAN DAN KEAMANAN” ini, penulis mendapatkan kesimpulan serta memberikan saran yang akan berguna bagi pengembangan penelitian ini.

#### **5.1. Kesimpulan**

Pada penelitian ini, telah dirancang VAD dengan akurasi sekitar  $\geq 85\%$  untuk  $SNR \geq 25$ . Keberhasilan VAD dalam menyeleksi sinyal wicara akan menentukan keberhasilan proses pengenalan wicara. Pada proses pengenalan pola ciri sinyal wicara, didapat nilai koefisien  $\alpha = 1$ ,  $\mu = 0.4$  and  $\beta = 10^{-5}$  sebagai nilai parameter *neural network*. Nilai ini dapat menghasilkan kombinasi kompleksitas pemrosesan *neural network* yang paling rendah dan akurasi yang tinggi. Sistem ekstraksi ciri berbasis BFCC dikombinasikan dengan *neural network* menghasilkan akurasi 82 % untuk 28 kosakata Bahasa Indonesia yang diujikan. Akurasi akan semakin menurun dengan semakin tingginya SNR maupun semakin banyaknya pola kosakata yang dipelajari.

Representasi distribusi memungkinkan pemetaan kata dalam dimensi yang lebih rendah. *Neural network* dapat digunakan untuk mempelajari pola distribusi dari kosakata dalam *vector space*. *Vector space* inilah yang digunakan selanjutnya dalam pemrosesan bahasa natural. Sistem pemrosesan bahasa natural menggunakan RNN menghasilkan akurasi sebesar 93 % dengan 50 sampel pola sintaksis dalam Bahasa Indonesia. RNN dapat mengurai perintah wicara yang telah dikenali menjadi struktur dasarnya kemudian menerjemahkannya menjadi bahasa yang lebih dekat dengan bahasa mesin. Selanjutnya robot akan lebih mudah untuk mengeksekusi urutan perintah yang dimaksud.

## 5.2. Saran

Proses ekstraksi ciri pada kata terisolasi menghasilkan jumlah data yang besar. Apalagi jumlah kata terisolasi dalam Bahasa Indonesia memiliki jumlah yang besar sehingga mempengaruhi kompleksitas identifikasi data. Proses ekstraksi ciri untuk fonem lebih menjanjikan dikarenakan jumlah fonem yang berkisar puluhan dan data yang dihasilkan juga lebih kecil. Sistem ekstraksi ciri berbasis fonem perlu dikembangkan dengan penggunaan *time-delay neural network* sebagai unit klasifikasi pola data fonem tersebut. Sistem ini dapat bekerja lebih fleksibel tanpa terpengaruh akurasi VAD dalam memisahkan sinyal wicara.

*Neural network* memiliki kelemahan pada akurasi ketika dihadapkan pada proses identifikasi pola data dengan banyak variasi keluaran. Menambah banyak *neuron* dapat menjadi solusi sementara untuk mengakomodasi banyaknya data sampel pembelajaran. Akan tetapi pada akhirnya keterbatasan distribusi *weight* menjadi penghalang NN dalam mengidentifikasi banyak variasi pola. Perlu dirancang sebuah NN modular dimana jaringan syaraf ini dibentuk dari setiap jaringan pemroses kecil yang digunakan untuk mempelajari pola data yang lebih sederhana.

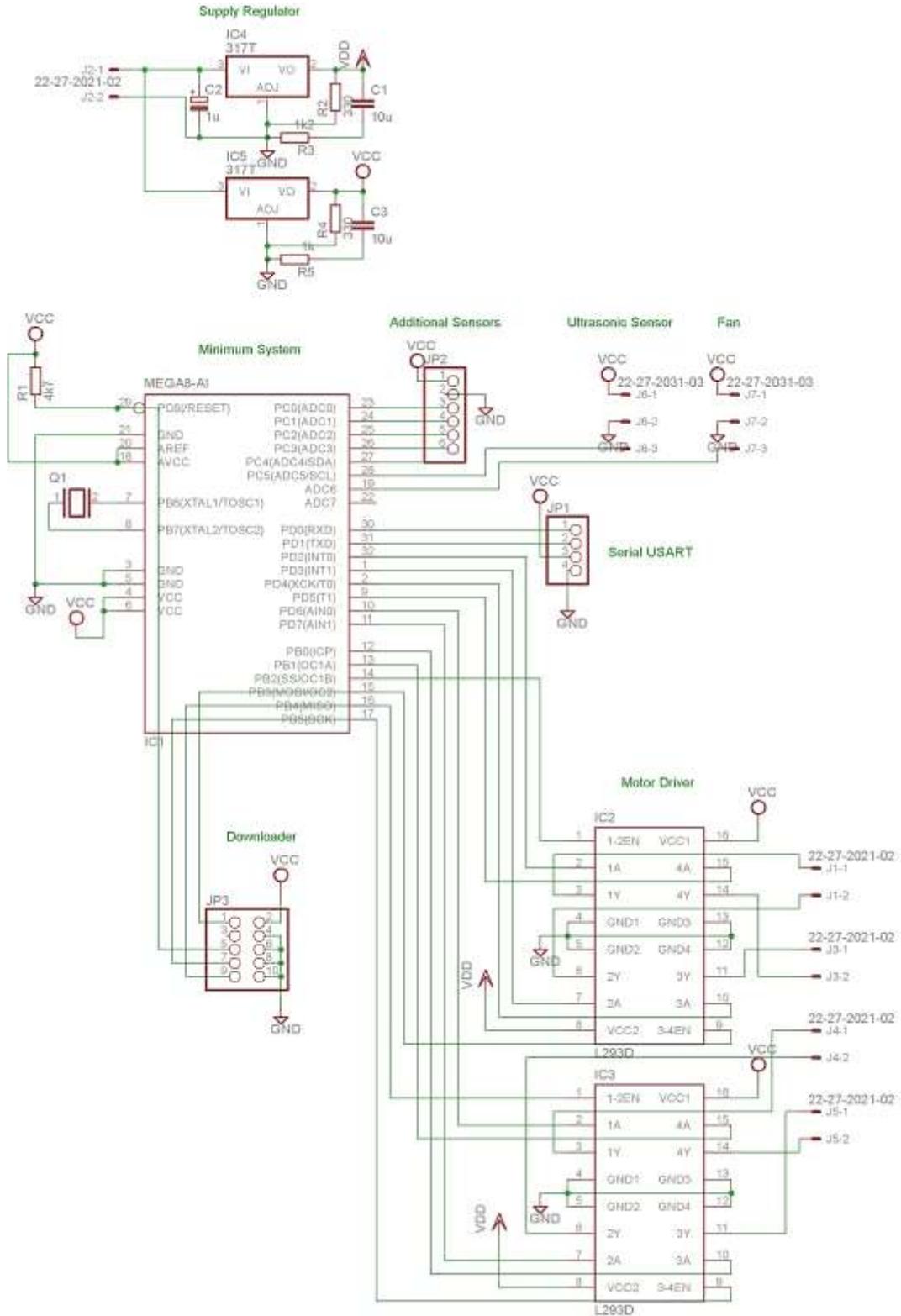
## DAFTAR PUSTAKA

- [1] Mehl, M. R., Vazire, S., Ramírez-Esparza, N., Slatcher, R. B., Pennebaker, J. W. (2007), “*Are women really more talkative than men?*”, *Science* 317 (5834), 82-82.
- [2] Socher, R., Lin, C. C., Yan-Tak, Ng. A., Manning, C. D. (2011), “*Parsing Natural Scenes and Natural Language with Recursive Neural networks*”, Proceedings of the 26th International Conference on Machine Learning (ICML).
- [3] Fardana, A.R., Jain, S., Jovancevic, I., Suri, Y., Morand, C. and Robertson, N.M. (2013), “*Controlling a Mobile Robot with Natural Commands based on Voice and Gesture*”, Proceedings of the IEEE International Conference on Robotics and Automation (ICRA).
- [4] Olson, C. F., Matthies, L. H., Schoppers, M., Maimone, M. W. (2010), “*Rover navigation using stereo ego-motion*”, *Robotics and Autonomous Systems* 43 (4): page 215–229.
- [5] Purwanto, D., Mardiyanto, R., Arai, K. (2009), “*Electric wheelchair control with gaze direction and eye blinking*” Proceedings of the 14th International Symposium on Artificial Life and Robotics, Oita, Japan.
- [6] Damaryam, G., Dunbar, G. (2005), “*A Mobile Robot Vision System for Self navigation using the Hough Transform and neural networks*”, Proceedings of the EOS Conference on Industrial Imaging and Machine Vision, Munich, pp. 72.
- [7] Jangmyung, L., MinCheol, L. (2013), “*A Robust Control of Intelligent Mobile Robot Based on Voice Command*”. Proceedings of the 6th International Conference, ICIRA.
- [8] Teller, S., Walter, M. R., Antone, M., Correa, A., Davis, R., Fletcher, L., Frazzoli, E., Glass, J., How, J. P., Huang, A. S., Jeon, J. H., Karaman, S., Luders, B., Roy, N., Sainath, T. (2010), “*A Voice-Commandable Robotic Forklift Working Alongside Humans in Minimally-Prepared Outdoor Environments*”, Proceedings of the Robotics and Automation (ICRA). 2010.

- [9] Kumar, P., Biswas, A., Mishra, A .N., and Chandra, M. (2010), "*Spoken Language Identification Using Hybrid Feature Extraction Methods*", Journal of Telecommunications. Volume 1. Issue 2.
- [10] Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., Lang, K. (1988), "*Phoneme Recognition: Neural Networks vs Hidden Markov Models*", Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP).
- [11] Chen, D., and Manning, C. D. (2014), "*A Fast and Accurate Dependency Parser using Neural networks*", Proceedings of the 2014 conference on Empirical Methods in *Natural language processing* (EMNLP).
- [12] Socher, R., Bauer, J., Manning, C., D., Yan-Tak Ng., A. (2013), "*Parsing with compositional vector grammars*", Proceedings of the ACL conference.

# LAMPIRAN

## SKEMATIK PERANGKAT KERAS VERSI AT MEGA 8A



*(Halaman ini sengaja dikosongkan)*

## **RIWAYAT HIDUP PENULIS**



**Karisma Trinanda Putra** dilahirkan di Bondowoso, 19 Juni 1990. Merupakan anak ke tiga dari tiga bersaudara pasangan Bapak Moh. Yasin dan Ibu Indah Juniarti. Penulis memulai pendidikan di SD Negeri Dabasah 07 Bondowoso, lalu melanjutkan di SMP Negeri 01 Bondowoso. Penulis menempuh jenjang pendidikan di SMA Negeri 02 Bondowoso. Pada tahun 2008 penulis diterima di Program D4 Jurusan Teknik Elektronika-Politeknik Elektronika Negeri Surabaya dan menyelesaikan studinya pada tahun 2012. Setelah menyelesaikan studi di tahap sarjana terapan, penulis meneruskan pendidikan di Program Magister Jurusan Teknik Elektro Institut Teknologi Sepuluh Nopember dengan memilih bidang keahlian Elektronika Industri.

e-mail: [hello.karisma@gmail.com](mailto:hello.karisma@gmail.com)

*(Halaman ini sengaja dikosongkan)*