Pembuatan Text-To-Speech Synthesis System Untuk Penutur Berbahasa Indonesia

Handi Dwi Rachma, Zonda Rugmiaga, Miftahul Huda Politeknik Elektronika Negeri Surabaya, Institut Teknologi Sepuluh November Surabaya Kampus ITS Sukolilo, Surabaya 60111, Indonesia
e-mail: handee_eepis@yahoo.com

ABSTRAK

Perkembangan teknologi telekomunikasi yang sangat pesat dihasilkan berbagai produk teknologi telekomunikasi yang sangat beragam. Produk telekomunikasi yang dihasilkan dari waktu ke waktu merupakan produk yang lebih efektif dan efisien dalam hal penggunaan dan pemeliharaan, baik secara teknis maupun biaya.

Pada paper ini diusulkan suatu metode diphone concatenation untuk mengkonversikan penulisan teks pada PC atau laptop menjadi output suara sesuai dengan teks yang dituliskan. Pembuatan ini dilakukan dengan menggunakan metode synthesis system yang terdiri dari tiga proses yaitu text pre processing, pembangkitan prosody dan proses concatenation.

Pembuatan perangkat lunak ini didahului dengan perancangan sistem aplikasi, kemudian rancangan tersebut diimplementasikan dengan text to speech synthesis system yang terdiri dari proses text pre processing, pembangkitan prosody, dan proses concatenation. Setelah diimplementasikan, perangkat lunak ini diuji coba sesuai dengan spesifikasi kebutuhan dan kemampuan yang dimiliki yaitu melakukan pengkonversian dari input kata atau kalimat ke bentuk representasi diphone yang kemudian diphone-diphone tersebut akan disambungkan (concatenate) untuk menjadi suara seperti teks yang diinputkan. Dengan demikian aplikasi perangkat lunak ini dapat digunakan untuk membantu para tuna netra agar dapat membaca berita dari internet ataupun membaca email..

Kata kunci: *diphone, text pre processing, prosody, concatenation.*

1. Pendahuluan

Perkembangan teknologi komputer yang sangat pesat, memicu perkembangan di berbagai bidang. Komputer diharapkan mampu berinteraksi secara lisan dengan pemakainya menggunakan bahasa sehari-hari, bukan bahasa mesin yang terkesan rumit. Tentunya, komputer harus dilengkapi dengan perangkat lunak untuk mengendalikan semua sistem serta menjalankan fungsi-fungsinya. Perangkat keras serta sebagaian perangkat lunak akan bersifat generik, tetapi sebagian komponen perangkat lunaknya akan bersifat language dependent, yaitu perangkat lunak yang melakukan pemrosesan bahasa alami secara lisan.

Teknologi bahasa adalah teknologi yang berhubungan dengan penggunaan bahasa, baik bahasa lisan maupun bahasa tulisan. Bahasa merupakan alat komunikasi paling relevan dan tepat sasaran untuk menyampaikan keinginan dan maksud manusia. Bentuk representasinya adalah berupa suara atau ucapan (spoken language), tetapi sering pula dinyatakan dalam bentuk tulisan. Sistem pemrosesan bahasa alami secara lisan dapat dibentuk dari sistem text to speech.

Pada paper ini, akan dilakukan perancangan sistem yang mengkonversikan sebuah teks bahasa Indonesia ke dalam bentuk ucapan. Text to speech synthesis system meliputi : proses text pre-processing, prosody dan proses concatenation yang menggabungkan diphone dari database suara.

2. Teori Penunjang

2.1 Teknologi Pemrosesan Bahasa

Bahasa dapat dibedakan menjadi 2 , yaitu Bahasa Alami dan Bahasa Buatan. Bahasa alami adalah bahasa yang biasa digunakan untuk berkomunikasi antar manusia, misalnya bahasa Indonesia, Sunda, Jawa, Inggris, Jepang, dan sebagainya. Bahasa Buatan adalah bahasa yang dibuat secara khusus untuk memenuhi kebutuhan tertentu, misalnya bahasa pemodelan atau bahasa pemrograman komputer.

Suatu sistem pemrosesan bahasa alami secara lisan dapat dibentuk dari tiga sub-sistem, yaitu sebagai berikut :

a. Sub-Sistem Natural Language Processing (NLP), berfungsi untuk melakukan pemrosesan secara simbolik terhadap bahasa tulisan. Beberapa bentuk aplikasi sub-sistem ini adalah translator bahasa alami (misalnya dari bahasa Inggris ke Bahasa Indonesia), sistem pemeriksaan sintaks bahasa, sistem yang dapat menyimpulkan suatu narasi, dan sebagainya.
b. Sub-Sistem Text-to-Speech (TTS), berfungsi untuk mengubah text (bahasa tulisan) menjadi ucapan (bahasa lisan).

Sub-Sistem Speech Recognation (SR), merupakan kebalikan teknologi Text to Speech, yaitu sistem yang berfungsi untuk mengubah atau mengenali suatu ucapan (bahasa lisan) menjadi teks (bahasa tulisan).

2.2 Kaidah Bahasa Indonesia

Bahasa Indonesia mengenal bahasa tulisan maupun bahasa lisan. Kadangkala terdapat beberapa perbedaan dalam kedua jenis bahasa ini. Dalam bahasa lisan, dikenal istilah fonem, yang merupakan kesatuan bahasa terkecil yang dapat membedakan arti. Dalam bahasa tulisan, fonem dilambangkan dengan huruf. Dengan kata lain, huruf adalah tulisan dari fonem. Seringkali istilah fonem disamakan dengan huruf, padahal tidak selamanya berlaku demikian. Berikut adalah konsep bahasa Indonesia berdasarkan pedoman umum ejaan bahasa Indonesia yang disempurnakan.

2.2.1 Abjad

Abjad yang digunakan dalam bahasa Indonesia terdiri atas 52 huruf, yaitu 26 huruf besar (A-Z) dan 26 huruf kecil (a-z).

2.2.2 Fonem

Fonem adalah istilah linguistik dan merupakan satuan terkecil dalam sebuah bahasa yang masih bisa menunjukkan perbedaan makna. Untuk bahasa Indonesia memiliki 35 fonem.

2.2.3 Diphone

Diphone adalah gabungan dari dua buah fonem bahasa Indonesia. Jumlah diphone dalam bahasa Indonesia kurang lebih sebanyak 1024 diphone.

2.3 Text To Speech

Pada dasarnya *Text-to-Speech* adalah suatu sistem yang dapat mengubah text menjadi ucapan. Suatu pensintesa ucapan atau *Text to Speech* pada prinsipnya terdiri dari dua sub sistem, yaitu :

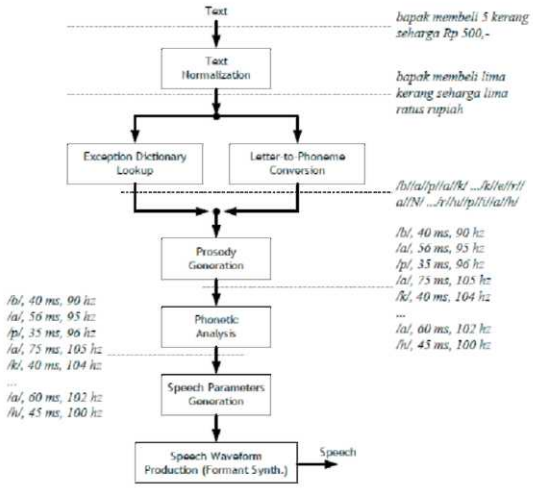
a.Bagian konverter teks ke fonem (Text to Phoneme)

b.Bagian konverter fonem ke ucapan (Phoneme to Speech)

Bagian konverter teks ke fonem berfungsi untuk mengubah kalimat masukan dalam suatu bahasa tertentu yang berbentuk teks menjadi rangkaian kode-kode bunyi yang biasanya direpresentasikan dengan kode fonem, durasi serta pitch-nya. Bagian ini bersifat sangat language dependent. Untuk suatu bahasa baru, bagian ini harus dikembangkan secara lengkap khusus untuk bahasa tersebut.

Bagian konverter fonem ke ucapan akan menerima masukan berupa kode-kode fonem serta pitch dan durasi yang dihasilkan oleh bagian sebelumnya. Berdasarkan kode-kode tersebut, bagian konverter fonem ke ucapan akan menghasilkan bunyi atau sinyal ucapan yang sesuai dengan kalimat yang ingin diucapkan.

Konversi dari teks ke fonem sangat dipengaruhi oleh aturan-aturan yang berlaku dalam suatu bahasa. Pada prinsipnya proses ini melakukan konversi dari simbol-simbol tekstual menjadi simbol-simbol fonetik yang merepresentasikan unit bunyi terkecil dalam suatu bahasa. Setiap bahasa memiliki aturan cara pembacaan dan cara pengucapan teks yang sangat spesifik. Hal ini menyebabkan implementasi unit konverter teks ke fonem menjadi sangat spesifik terhadap suatu bahasa.

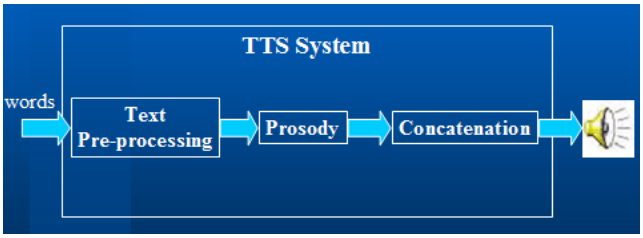


Gambar 1. Urutan Konversi Dari Teks Menjadi Ucapan

3. Perancangan Sistem

3.1 Text To Speech Synthesis System

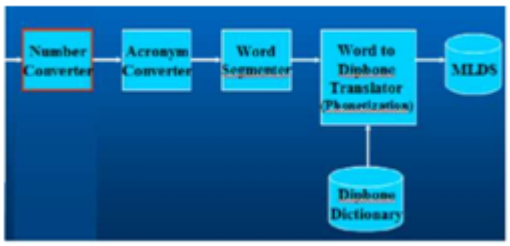
Text to Speech synthesis system terdiri dari 3 bagian, yaitu *text pre-processing*, pembangkitan *prosody* dan *concatenation.* Di bawah ini adalah diagram blok *text to speech synthesis system* :



Gambar 2. Blok Diagram *Text to speech synthesis system*

3.1.1 Text pre-processing

Yaitu pengkonversian dari input yang berupa teks menjadi *diphone* (gabungan dua buah fonem). Ketika input yang berupa teks, akronim (singkatan) ataupun angka maka bagian ini akan mengkonversikan menjadi *diphone* yang telah tersedia di database *diphone*. Diagram blok untuk proses *text pre-processing* adalah :



Gambar 3. Blok diagram *text pre processing*

Dari blok diagram sistem dapat dijelaskan cara kerja sistem yaitu :

- **Number Converter**

Jika input pada sistem berupa angka, maka sistem mengkonversikan angka ke dalam representasi diphone (gabungan dua buah fonem).

- **Acronym Converter**

Jika input pada sistem berupa kata singkatan dalam bahasa Indonesia, maka sistem mengkonversika singkatan ke dalam representasi diphone (gabungan dua buah fonem).

- **Word Segmenter**

Jika input pada sistem berupa kata atau kalimat maka sistem mengkonversikan kata atau kalimat ke dalam representasi diphone (gabungan dua buah fonem).

- **Diphone Dictionary**

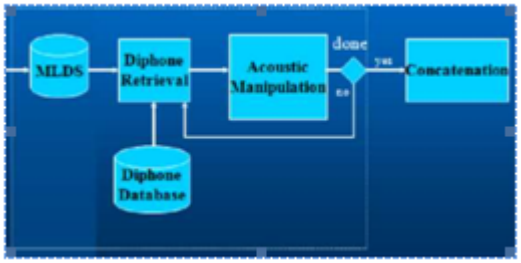
Merupakan database yang berupa kumpulan dari diphone – diphone. Pembuatan diphone dilakukan dengan melakukan pelabelan pada sinyal wicara. Jumlah diphone yang telah terkumpul sebanyak 1024 diphone.

- **MLDS (Multi Level Data Structure)**

Terdiri dari semua data yang diperlukan untuk sub system berikutnya, dalam hal ini adalah proses prosody. MLDS terdiri dari representasi diphonediphone hasil pengkonversian inputan

3.1.2 Prosody

Yaitu untuk mendapatkan ucapan yang lebih alami, ucapan yang dihasilkan harus memiliki intonasi *(prosody).* Secara kuantisasi, prosodi adalah perubahan nilai pitch (frekuensi dasar) selama pengucapan kalimat dilakukan atau pitch sebagai fungsi waktu. Prosodi bersifat sangat spesifik untuk setiap bahasa, sehingga model yang diperlukan untuk membangkitkan data-data prosodi menjadi sangat spesifik juga untuk suatu bahasa. Diagram blok untuk prosodi adalah :



Gambar 4. Blok diagram *Prosody*

- MLDS (Multi Level Data Structure), tediri dari semua data yang diperlukan untuk sub sistem berikutnya. MLDS terdiri atas kata, reperesentasi *diphone*, Prosodic parameter untuk tiap diphone (ini perpaduan antara level kata dan level *prosody* kalimat). MLDS mengizinkan untuk modulasi.
- Diphone Retrieval didalamnya terdapat tiga tahapan yang terjadi, yaitu database hasil perekaman *diphone*, menyimpan bentuk gelombang *diphone* dan *Prosodic* parameter dalam variabel.
- Accoustic Manipulation di dalamnya terdapat proses pengenalan file-file gelombang .WAV(load, play, write), vast array dari peralatan signal processing, built-in function, ease debugging , GUI-capable

3.1.3 Concatenation

Yaitu penggabung-gabungan segmen-segmen bunyi yang telah direkam sebelumnya. Setiap segmen berupa *diphone* (gabungan dua buah fonem). Pada perekaman suara dilakukan beberapa kali agar mendapatkan hasil yang akurat.

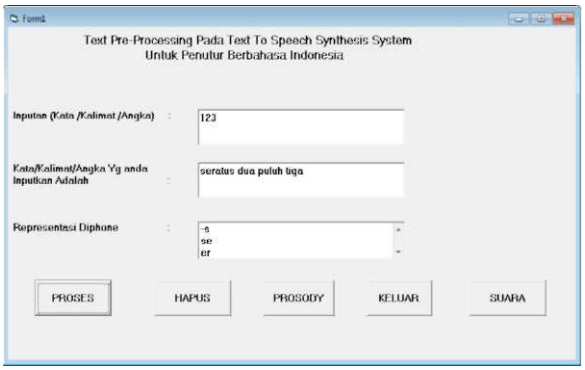
4. Hasil Pengujian dan Analisa

Dari hasil pembuatan perangkat lunak pada blok perancangan sistem, kemudian pengujian dilakukan dengan masukan kata, kalimat, atau angka ke dalam blok *text pre-processing.* Kata atau kalimat yang dimasukan akan dikonversikan kedalam bentuk representasi *diphone* (gabungan dari 2 fonem). Jika masukan sistem berupa angka, maka sistem akan mengkonversikan dari angka (numerik) ke string. Dari bentuk string inilah kemudian dikonversikan ke dalam bentuk representasi *diphone* seperti contoh di bawah ini.

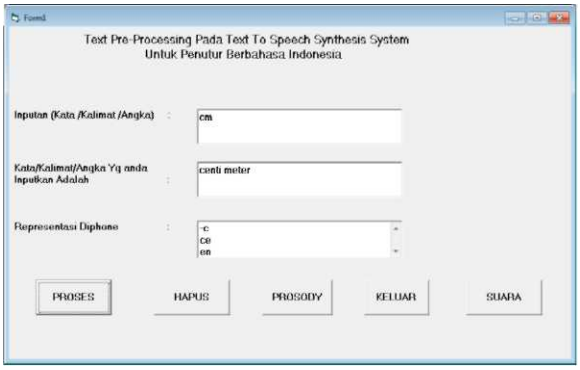
Dari tampilan program diatas, masukan sistem berupa angka yaitu "123". Selanjutnya masukan angka tersebut dikonversikan terlebih dahulu ke dalam bentuk string menjadi "seratus dua puluh tiga". Dari bentuk string inilah dilakukan pengonversian kedalam bentuk representasi diphone menjadi "-s, se, er, ra, at, tu, us, s- ,-d, du, ua, a-, -p, pu, ul, lu, uh, h-, -t, ti, ig, ga, a- ".

Jika masukan sistem berupa kata atau kalimat, maka program langsung mengkonversikan masukan ke dalam bentuk representasi diphone. Jika masukan sistem

berupa akronim atau singkatan, pada program harus dilakukan inisialisasi terlebih dahulu untuk beberapa singkatan yang populer digunakan. Misalnya singkatan untuk "cm" maka kepanjangannya adalah "centi meter". Sebagaimana yang terlihat pada tampilan program dibawah ini.



Gambar 5. Proses *Text Pre-Processing* dengan masukan angka



Gambar 6. Proses *Text Pre-Processing* dengan masukan akronim

Dari tampilan program diatas, masukan sistem berupa akronim atau singkatan yaitu "cm". Sebelumnya pada program, dilakukan inisialisasi untuk beberapa singkatan beserta kepanjangannya yang populer di sekitar kita. Misalnya "cm", "km", "kg", "jl". Untuk masukan "cm", maka sistem terlebih dahulu akan mengkonversikannya kepanjangan dari "cm" yaitu "centi meter". Dari kepanjangannya inilah dilakukan pengonversian ke bentuk representasi diphone menjadi " -c, ce, en, nt, ti, i-, -m, me, et, te, er, r- ".

Pada proses prosodi, dilakukan pemodelan terhadap representasi diphone hasil dari text pre-processing. Masing-masing diphone ditambahkan satu pitch periode data didepan dan dibelakang. Tujuan pemberian satu pitch periode data ini untuk mengaplikasikan algoritma PSOLA *(Pitch Synchronous Overlap Add Method)* untuk proses concatenation.

Panjang data dari *pitch* awal dan akhir sangat penting karena sangat menentukan berapa besar data yang di *overlap*-kan dan di tambahkan dengan *diphone* asli, hal itu yang akan menjadi dasar dalam penyambungan *diphone* dengan menggunakan metode PSOLA.

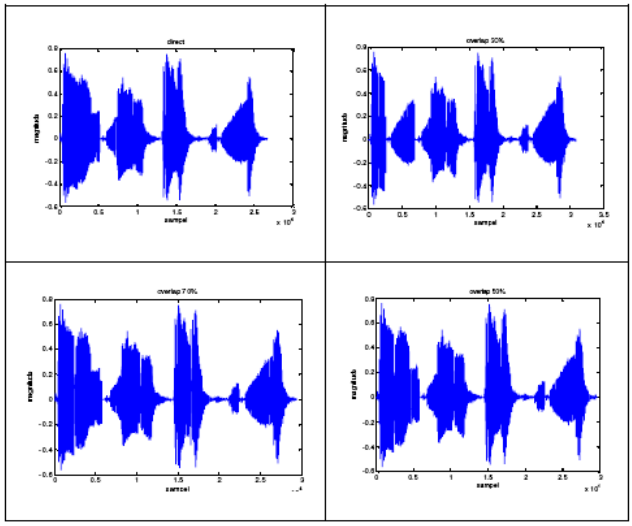
Pada metode PSOLA ada beberapa bagian yang menjadi prinsip dasar:

- Jika sinyal wav tersebut dalam proses *concatenation* terletak di paling depan, maka hanya satu *pitch period* yang paling terakhir itulah yang akan di proses.
- Jika sinyal vaw tersebut dalam proses *concatenation* terletak di tengah, maka ujung yang paling depan dan yang paling akhirlah yang akan diproses.
- Jika sinyal wav tersebut dalam proses *concatenation* paling belakang, maka hanya satu *pitch period* yang paling awal itulah yang akan di proses.

Pada proses concatenation terjadi penggabungan kembali diphone-diphone pada proses prosody dengan menggunakan algoritma PSOLA *(Pitch Synchronous Overlap Add Method).* Pada PSOLA, proses *overlap* di lakukan setelah *diphone* asli di tambahkan satu *pitch* yang telah mengalami *windowing* di ujung depan dan belakang pada *diphone.* Selanjutnya panjang data dari masingmasing *pitch* kedua *diphone* yang akan disambungkan tersebut di kalikan dengan besar *overlap* yang akan di gunakan. Jika *overlap* 30%, maka panjang data dari masing-masing *pitch* dari kedua *diphone* yang akan disambungkan tersebut adalah 0.3 dari panjang data *pitch* tersebut . jika *overlap* 50%, dari masing-masing *pitch* dari kedua *diphone* yang akan disambungkan tersebut adalah 0.5 dari panjang data *pitch* tersebut, dan jika menggunakan *overlap* 70%, maka masing-masing *pitch* dari kedua *diphone* yang akan disambungkan tersebut adalah 0.7 dari panjang data *pitch* tersebut.

Panjang data yang akan di *overlap*-kan dari masing-masing *pitch* dari kedua *diphone* yang akan disambungkan tersebut akan di bandingkan dan dicari mana yang paling pendek. Panjang data yang paling pendek dari keduanya akan di jadikan patokan, karena jika memakai panjang data *pitch* yang paling besar, akan menyebabkan *diphone* asli pada *pitch* yang panjang datanya lebih pendek akan tertumpuk, akibatnya akan terjadi pencemaran pada sinyal diphone-nya, dan suara yang dihasilkan semakin kacau, dampak tersebut akan semakin besar jika persentase yang di *overlap-kan* juga semakin besar.

Pembangkitan *pitch* dengan metode PSOLA membuat ujung-ujung *pitch* dari suatu *diphone* berada pada *magnitude* 0 dan terlihat proses penyambungannya semakin halus antar sinyal diphone-nyabila di bandingkan dengan penyambungan secara langsung.



Gambar 7. Hasil penyambungan *diphone-diphone* yang membentuk kata" politeknik" secara *direct,* PSOLA *overlap* 30%, 50% dan 70%

Penyambungan dengan PSOLA membuat panjang data hasil penyambungan *diphone* tersebut lebih panjang bila di bandingkan dengan panjang data hasil penyambungan dengan metode *direct* (langsung). Pada penyambungan kata "politeknik", penyambungan secara langsung memiliki panjang data keseluruhan 26661 sampel, sedangkan pada penyambungan dengan menggunakan PSOLA, terdapat tiga pengkondisian yaitu menggunakan *overlap* 30%, 50% dan 70%.

Pada penyambungan dengan menggunakan metode PSOLA *overlap* 30% panjang data penyambungan sinyal *diphone-diphone* yang membentuk kata "politeknik" menjadi sepanjang 30832 sampel, jika menggunakan PSOLA *overlap* 50%, panjang datanya menjadi 29598 sampel. Dan jika menggunakan PSOLA dengan *overlap* 70%, panjang datanya sepanjang 29454 sampel. Dari ketiga metode PSOLA dengan besar *overlap* yang berbeda terlihat bahwa semakin panjangnya data yang akan di *overlap-kan* maka menyebabkan panjang data dari keseluruhan hasil penyambungan akan semakin pendek, hal ini di karenakan dua data yang di *overlap*-kan itu menjadi satu kesatuan atau di lebur menjadi satu, sehingga dalam hasil penyambungan memiliki panjang data sebesar panjang kedua *diphone* ditambah sisa kedua panjang data *pitch* yang tidak ikut di *overlap* kan dan panjang data yang telah di *overlap-kan.* Hal ini juga membuat *magnitude* dari data yang di *overlap*-kan juga semakin besar.

Dari hasil survey yang merupakan penilaian subyektif dari 20 responden untuk semua penyambungan dengan menggunakan masukan yang berbeda di dapatkan hasil:

- Penyambungan dengan metode *direct* kualitasnya cukup bagus atau suara yang cukup jernih dengan nilai MOS rata-rata 3.31.
- Penyambungan menggunakan metode PSOLA dengan *overlap* 30% dapat dikatakan bahwa hasil penyambungannya, memiliki kualitas cukup bagus atau suara yang cukup jernih dengan nilai MOS rata-rata 2.85.
- Penyambungan menggunakan metode PSOLA dengan *overlap* 50% dapat dikatakan bahwa hasil penyambungannya memiliki kualitas cukup bagus atau suara yang cukup jernih dengan nilai MOS rata-rata 2.81.
- Penyambungan menggunakan metode PSOLA dengan *overlap* 70% dapat dikatakan bahwa hasil penyambungannya memiliki kualitas cukup bagus atau suara yang cukup jernih dengan nilai MOS rata-rata 2.79.

5. Kesimpulan

Berdasarkan hasil pengujian dan analisa dapat disimpulkan bahwa :

- Pada proses *text pre-processing,* dapat dilakukan dengan benar pengonversian masukan berupa angka, akronim, kata dan kalimat ke bentuk representasi *diphone.*
- Penyambungan menggunakan metode PSOLA dengan *overlap* 30% memberikan hasil yang lebih baik dari responden dengan nilai MOS rata-rata adalah 2.85 dibandingkan dengan overlap 50% dengan nilai MOS rata-rata 2.81 dan overlap 70% dengan nilai MOS rata-rata 2.79.

Daftar Pustaka

[1]. Ian McLoughlin, *"'Applied Speech And Audio Processing"*, Cambridge University Press, Singapore, 2009.

[2]. Michael Beddaoui, Abdel Aziz El-Solh, "A *Text To Speech Synthesis System"*, 2002.

[3]. Ary Akhmad Arman, *"Konversi Dari Teks Ke Ucapan"*, Institut Teknologi Bandung.

[4]. Pusat Bahasa Departeman Pendidikan Nasional, *"Kamus Besar Bahasa Indonesia"*, Jakarta, 2008.