Mimic Human Speech in Bahasa Indonesia Using Speech Recognition and Speech Synthesis

Valens Prabagita Ivan Susilo
Department of Computer Science
President University
Cikarang, Bekasi, 17550, Indonesia
prabagita12@gmail.com

Abstrak

Orang - orang menggunakan speech recognition dan speech synthesis untuk membantu meningkatkan dan mendukung kegiatan mereka sehari - hari. Dengan salah satu teknologi tersebut saja, pengembang dapat menghasilkan berbagai macam software. Menyatukan kedua teknologi tersebut, pengembang dapat menghasilkan lebih banyak berbagai software. Salah satu kombinasi itu adalah mimic speech, atau menirukan suara. Penelitian ini akan membahas tentang Speech Recognition menggunakan Convolutional Neural Network sebagai model machine learning dan Speech Synthesis yang menggunakan Concatenative Synthesis dengan suku kata sebagai satuannya. Berbeda dengan software – software terkait baru - baru ini, penelitian ini memiliki pendekatan yang lebih sederhana untuk menirukan suara dalam Bahasa Indonesia. Tujuan dari penelitian ini adalah untuk membuat aplikasi untuk mengumpulkan, melatih, dan menirukan suara dalam Bahasa Indonesia. Pengguna dapat berpartisipasi dengan merekam suara mereka. Suara tersebut dikumpulkan untuk dilatih agar dapat digunakan di aplikasi nantinya untuk mengenali suara. Dengan model yang telah dilatih, kini pengguna dapat membuat komputer meniru suara mereka. Pertama, pengguna harus mengidentifikasi suara mereka apakah dikenali oleh aplikasi. Langkah ini diperlukan untuk membuat suara digital pengguna. Setelah dibuat, berdasarkan suku kata terdaftar, yang mana suaranya telah dikenali oleh aplikasi, pengguna dapat menghasilkan suara dengan membuat kalimat dari suku kata tersebut. Aplikasi untuk mengumpulkan dan menirukan suara akan dikembangkan dalam website dan aplikasi untuk melatih dikembangkan dalam command promt.

1 Pendahuluan

Speech recognition adalah proses untuk mendapatkan data dari menganalisa suara. Kebalikan dari speech recognition adalah speech synthesis, yaitu proses untuk membuat suara buatan. Maka dari itu speech recognition dikenal dengan istilah speech-to-text dan speech synthesis dengan isitilah text-to-speech. Dengan salah satu teknologi tersebut saja, pengembang dapat menghasilkan berbagai macam software. Menyatukan kedua teknologi tersebut, pengembang dapat menghasilkan lebih banyak berbagai software. Salah satu kombinasi itu adalah mimic speech, atau menirukan suara. Penggunaan mimic speech yang dikenal adalah menciptakan suara digital yang akan digunakan sebagai suara vokal asisten buatan. Dengan itu akan membuat asisten buatan lebih personal bagi pengguna.

Penelitian ini bertujuan untuk membuat aplikasi untuk mengumpulkan suara - suara, melatih machine learning dengan suara yang terkumpul, dan menirukan suara dalam Bahasa Indonesia.

2 Batasan

Batasan aplikasi ini adalah sebagai berikut:

- Ada 9 suku kata yang dipilih untuk digunakan dalam aplikasi, a, i, na, ma, mu, di, ri, ku, dan kan. Suku kata akan digunakan sebagai satuan unit.
- Suara direkam dalam 1 detik dengan sample rate 16000 dan mono.
- Speech recognition digunakan untuk mengenali suara ketika user sedang membuat suara digital pada proses identifikasi suara.
- Speech synthesis digunakan untuk menghasilkan suara berdasarkan suara digital yang dipilih dan teks yang dimasukan pada proses pembuatan suara.

3 Metodologi

Pendekatan yang digunakan untuk mencapai tujuan penelitian ini menggunakan sebagai teknik berikut:

3.1 Concatenative Synthesis

Concatenative synthesis menghubungkan suara atau ucapan yang telah direkam. Concatenative synthesis merupakan cara termudah untuk menghasilkan suara buatan yang terdengar jelas dan alami. Salah satu aspek terpenting dalam concatenative synthesis adalah menentukan satuan suara yang benar. Dalam sistem sekarang ini, satuan yang digunakan biasanya berupa kata, suku kata, *demisyllables*, fonem, *diphones*, dan kadang pula triphones [Hande, 2014].

3.2 MFCC

MFCC adalah salah satu metode ekstraksi fitur yang paling umum digunakan dalam speech recognition yang diperkenalkan oleh Davis dan Mermelstein pada 1980-an [practicalcrypthography.com, 2012].

3.3 Convolutional Neural Network

Convolutional neural network (CNN) adalah salah satu varian model neural network yang sangat dikenal. Model ini dirancang untuk mengenali objek tidak peduli dimana permukaan objek itu. Model ini tidak harus mempelajari kembali perkiraan setiap permukaan yang mungkin muncul [Geitgey, 2016].

4 Hasil Percobaan

Untuk mengevaluasi efektivitas metode yang diusulkan di bagian sebelumnya, tes dilakukan untuk memastikan aplikasi berjalan dengan baik.

Dataset yang digunakan selama evaluasi sebanyak 10.000 suara laki - laki dan perempuan, yang masing - masing 500 suara pada setiap suku kata pada latar yang tidak berisik dan masing - masing 500 suara pada suara tidak dikenal atau suara acak. Hasil yang dianggap baik ditunjukan dari akurasi lebih dari 75% dengan suku kata yang bersangkutan. Evaluasi juga dilakukan di suku kata diluar batasan dan suara acak seperti kondisi diam, suku kata o dan mi.

Ada 4 penguji dalam skenario ini. Penguji 1 yaitu pengguna laki - laki yang suaranya telah digunakan dalam pembelajaran machine learning. Penguji 2 yaitu pengguna perempuan yang suaranya telah digunakan dalam pembelajaran machine learning. Penguji 3 yaitu pengguna laki - laki yang suaranya belum pernah digunakan dalam pembelajaran machine learning. Penguji 4 yaitu pengguna perempuan yang suaranya belum pernah digunakan dalam pembelajaran machine learning. Setiap penguji akan melakukan evaluasi dalam keadaan seperti berikut:

- Latar berisik (Musik keras atau orang sekitar sedang mengobrol).
- Latar lumayan berisik (Suara hujan atau suara dari ruangan sebelah).
- Latar tidak berisik.

Tabel 1: Hasil Penguji 1 pada latar berisik.

No	Spoken Syllable	Noisy Background			Correct Result
		1	2	3	
1	a	52 *(a)*	100 *(a)*	100 *(unknown)*	1/3
2	i	62 *(na)*	85 *(ri)*	96 *(ma)*	0/3
3	na	100 *(unknown)*	86 *(kan)*	100 *(unknown)*	0/3
4	ma	100 *(a)*	84 *(unknown)*	100 *(a)*	0/3
5	mu	99 *(kan)*	71 *(ku)*	88 *(unknown)*	0/3
6	di	59 *(mu)*	100 *(ri)*	97 *(ri)*	0/3
7	ri	50 *(na)*	100 *(ri)*	92 *(a)*	1/3
8	ku	100 *(ku)*	85 *(a)*	100 *(kan)*	1/3
9	kan	72 *(unknown)*	61 *(kan)*	79 *(kan)*	3/3
10	Unknown 1 *(silent)*	99 *(unknown)*	100 *(a)*	99 *(unknown)*	2/3
11	Unknown 2 *(o)*	100 *(na)*	100 *(a)*	100 *(ku)*	0/3
12	Unknown 3 *(mi)*	100 *(a)*	51 *(kan)*	96 *(ma)*	1/3

Tabel 2: Hasil Penguji 1 pada latar lumayan berisik.

No	Spoken Syllable	Semi Noisy Background			Correct Result
		1	2	3	
1	a	100 *(a)*	100 *(a)*	100 *(a)*	3/3
2	i	100 *(i)*	36 *(ri)*	100 *(i)*	2/3
3	na	97 *(mu)*	48 *(ma)*	99 *(na)*	1/3
4	ma	49 *(i)*	98 *(ri)*	79 *(mu)*	0/3
5	mu	55 *(ri)*	99 *(ku)*	50 *(ri)*	0/3
6	di	57 *(i)*	100 *(di)*	100 *(i)*	1/3
7	ri	100 *(ri)*	75 *(ri)*	100 *(ri)*	3/3
8	ku	92 *(ku)*	68 *(ku)*	91 *(ku)*	2/3
9	kan	100 *(kan)*	94 *(na)*	100 *(kan)*	2/3
10	Unknown 1 *(silent)*	100 *(a)*	99 *(unknown)*	100 *(ma)*	2/3
11	Unknown 2 *(o)*	100 *(ku)*	89 *(ku)*	72 *(ku)*	1/3
12	Unknown 3 *(mi)*	100 *(ri)*	94 *(ri)*	100 *(ri)*	0/3

Tabel 3: Hasil Penguji 1 pada latar tidak berisik.

No	Spoken Syllable	Not Noisy Background			Correct Result
		1	2	3	
1	a	100 (a)	100 (a)	100 (a)	3/3
2	i	94 (i)	48 (kan)	94 (i)	2/3
3	na	80 (na)	99 (na)	100 (na)	3/3
4	ma	96 (di)	90 (i)	57 (di)	0/3
5	mu	100 (i)	86 (ku)	95 (i)	0/3
6	di	99 *(ri)*	100 *(i)*	80 *(ri)*	0/3
7	ri	100 *(ri)*	96 *(ri)*	91 *(ri)*	3/3
8	ku	85 *(ku)*	96 *(ku)*	100 *(ku)*	3/3
9	kan	92 *(kan)*	100 *(kan)*	100 *(kan)*	3/3
10	Unknown 1 *(silent)*	98 *(unknown)*	98 *(unknown)*	98 *(unknown)*	3/3
11	Unknown 2 *(o)*	100 *(ku)*	98 *(ku)*	65 *(ku)*	1/3
12	Unknown 3 *(mi)*	54 *(mu)*	100 *(ri)*	67 *(mu)*	2/3

Tabel 4: Hasil Penguji 2 pada latar berisik.

No	Spoken Syllable	Noisy Background			Correct Result
		1	2	3	
1	a	99 *(unknown)*	99 *(a)*	99 *(a)*	2/3
2	i	82 *(unknown)*	98 *(ma)*	99 *(ri)*	0/3
3	na	52 *(ri)*	99 *(ma)*	61 *(ma)*	0/3
4	ma	50 *(unknown)*	37 *(unknown)*	95 *(ri)*	0/3
5	mu	99 *(unknown)*	52 *(di)*	99 *(unknown)*	0/3
6	di	99 *(i)*	99 *(unknown)*	99 *(unknown)*	0/3
7	ri	86 *(i)*	20 *(kan)*	57 *(di)*	0/3
8	ku	99 *(ku)*	68 *(kan)*	90 *(unknown)*	1/3
9	kan	84 *(kan)*	99 *(kan)*	78 *(a)*	2/3
10	Unknown 1 *(silent)*	100 *(unknown)*	100 *(unknown)*	100 *(unknown)*	3/3
11	Unknown 2 *(o)*	99 *(ma)*	99 *(ku)*	99 *(ku)*	0/3
12	Unknown 3 *(mi)*	36 *(di)*	96 *(mu)*	99 *(di)*	1/3

Tabel 5: Hasil Penguji 2 pada latar lumayan berisik.

No	Spoken Syllable	Semi Noisy Background			Correct Result
		1	2	3	
1	a	100 *(a)*	100 *(unknown)*	99 *(a)*	2/3
2	i	92 *(ri)*	97 *(unknown)*	96 *(ri)*	0/3
3	na	83 *(ma)*	62 *(unknown)*	75 *(kan)*	0/3
4	ma	87 *(unknown)*	64 *(ma)*	90 *(na)*	2/3
5	mu	76 *(ma)*	78 *(ku)*	66 *(di)*	0/3
6	di	58 *(mu)*	48 *(mu)*	30 *(kan)*	0/3
7	ri	99 *(kan)*	89 *(kan)*	99 *(ri)*	1/3
8	ku	99 *(unknown)*	99 *(ku)*	88 *(unknown)*	1/3
9	kan	99 *(kan)*	65 *(kan)*	85 *(kan)*	2/3
10	Unknown 1 *(silent)*	100 *(unknown)*	79 *(a)*	99 *(ri)*	1/3
11	Unknown 2 *(o)*	99 *(a)*	82 *(ku)*	99 *(na)*	0/3
12	Unknown 3 *(mi)*	89 *(kan)*	35 *(ku)*	96 *(ri)*	1/3

Tabel 6: Hasil Penguji 2 pada latar tidak berisik.

No	Spoken Syllable	Not Noisy Background			Correct Result
		1	2	3	
1	a	88 *(a)*	99 *(a)*	99 *(a)*	3/3
2	i	99 *(i)*	99 *(i)*	52 *(i)*	2/3
3	na	97 *(a)*	54 *(kan)*	99 *(a)*	0/3
4	ma	99 *(na)*	99 *(na)*	98 *(na)*	0/3
5	mu	81 *(i)*	95 *(i)*	44 *(i)*	0/3
6	di	99 *(i)*	36 *(i)*	96 *(i)*	0/3
7	ri	47 *(ri)*	32 *(unknown)*	48 *(i)*	0/3
8	ku	98 *(ku)*	67 *(kan)*	99 *(ku)*	2/3
9	kan	84 *(na)*	85 *(na)*	87 *(na)*	0/3
10	Unknown 1 *(silent)*	99 *(unknown)*	99 *(unknown)*	100 *(unknown)*	3/3
11	Unknown 2 *(o)*	100 *(ku)*	99 *(ku)*	95 *(ku)*	0/3
12	Unknown 3 *(mi)*	67 *(ri)*	78 *(di)*	77 *(i)*	1/3

Tabel 7: Hasil Penguji 3 pada latar berisik.

No	Spoken Syllable	Noisy Background			Correct Result
		1	2	3	
1	a	100 *(a)*	99 *(a)*	100 *(a)*	3/3
2	i	71 *(ma)*	99 *(ri)*	99 *(ri)*	0/3
3	na	70 *(ku)*	99 *(a)*	57 *(unknown)*	0/3
4	ma	99 *(a)*	100 *(a)*	100 *(a)*	0/3
5	mu	74 *(mu)*	96 *(kan)*	74 *(mu)*	0/3
6	di	99 *(ri)*	96 *(ri)*	89 *(ri)*	0/3
7	ri	100 *(unknown)*	90 *(na)*	86 *(a)*	0/3
8	ku	98 *(kan)*	64 *(mu)*	99 *(a)*	0/3
9	kan	99 *(na)*	100 *(kan)*	98 *(kan)*	2/3
10	Unknown 1 *(silent)*	99 *(a)*	99 *(a)*	100 *(a)*	0/3
11	Unknown 2 *(o)*	92 *(na)*	84 *(na)*	61 *(mu)*	1/3
12	Unknown 3 *(mi)*	64 *(kan)*	53 *(na)*	99 *(ri)*	2/3

Tabel 8: Hasil Penguji 3 pada latar lumayan berisik.

No	Spoken Syllable	Semi Noisy Background			Correct Result
		1	2	3	
1	a	100 *(a)*	99 *(a)*	99 *(a)*	3/3
2	i	99 *(ri)*	90 *(di)*	92 *(di)*	0/3
3	na	100 *(kan)*	99 *(kna)*	92 *(ma)*	0/3
4	ma	99 *(a)*	100 *(kan)*	100 *(kan)*	0/3
5	mu	99 *(mu)*	99 *(mu)*	65 *(ku)*	2/3
6	di	99 *(ri)*	100 *(ri)*	99 *(ri)*	0/3
7	ri	95 *(kan)*	76 *(i)*	78 *(di)*	0/3
8	ku	99 *(a)*	96 *(a)*	100 *(a)*	0/3
9	kan	99 *(a)*	96 *(a)*	99 *(a)*	0/3
10	Unknown 1 *(silent)*	99 *(ri)*	73 *(di)*	60 *(unknown)*	2/3
11	Unknown 2 *(o)*	100 *(a)*	99 *(ku)*	100 *(a)*	0/3
12	Unknown 3 *(mi)*	99 *(na)*	99 *(ri)*	99 *(i)*	0/3

Tabel 9: Hasil Penguji 3 pada latar tidak berisik.

No	Spoken Syllable	Not Noisy Background			Correct Result
		1	2	3	
1	a	100 *(a)*	100 *(a)*	100 *(a)*	3/3
2	i	92 *(ri)*	95 *(ri)*	41 *(ku)*	0/3
3	na	99 *(na)*	74 *(ku)*	80 *(a)*	1/3
4	ma	59 *(a)*	84 *(ku)*	48 *(kan)*	0/3
5	mu	99 *(mu)*	99 *(ri)*	99 *(ri)*	1/3
6	di	100 *(ri)*	99 *(i)*	99 *(i)*	0/3
7	ri	99 *(ri)*	96 *(ri)*	94 *(i)*	2/3
8	ku	73 *(di)*	100 *(mu)*	99 *(mu)*	0/3
9	kan	100 *(kan)*	99 *(kan)*	96 *(kan)*	3/3
10	Unknown 1 *(silent)*	98 *(unknown)*	98 *(unknown)*	98 *(unknown)*	3/3
11	Unknown 2 *(o)*	99 *(ku)*	68 *(mu)*	100 *(ku)*	1/3
12	Unknown 3 *(mi)*	99 *(ri)*	93 *(i)*	100 *(ri)*	0/3

Tabel 10: Hasil Penguji 4 pada latar berisik.

No	Spoken Syllable	Noisy Background			Correct Result
		1	2	3	
1	a	100 *(unknown)*	99 *(unknown)*	94 *(ma)*	0/3
2	i	97 *(unknown)*	99 *(ri)*	99 *(ri)*	0/3
3	na	100 *(na)*	99 *(ri)*	99 *(ma)*	1/3
4	ma	93 *(na)*	99 *(ma)*	100 *(na)*	1/3
5	mu	55 *(ri)*	99 *(unknown)*	43 *(unknown)*	0/3
6	di	99 *(ri)*	54 *(i)*	99 *(ma)*	0/3
7	ri	97 *(ri)*	74 *(ri)*	100 *(ri)*	2/3
8	ku	68 *(ma)*	92 *(ku)*	52 *(52)*	1/3
9	kan	48 *(ku)*	22 *(ku)*	99 *(a)*	0/3
10	Unknown 1 *(silent)*	100 *(unknown)*	100 *(unknown)*	100 *(unknown)*	3/3
11	Unknown 2 *(o)*	100 *(a)*	100 *(a)*	100 *(a)*	0/3
12	Unknown 3 *(mi)*	55 *(mu)*	99 *(ri)*	94 *(i)*	1/3

Tabel 11: Hasil Penguji 4 pada latar lumayan berisik.

No	Spoken Syllable	Semi Noisy Background			Correct Result
		1	2	3	
1	a	73 *(a)*	91 *(unknown)*	99 *(unknown)*	0/3
2	i	81 *(ri)*	99 *(i)*	100 *(ri)*	1/3
3	na	100 *(ma)*	99 *(unknown)*	99 *(unknown)*	0/3
4	ma	99 *(ma)*	99 *(na)*	78 *(ma)*	2/3
5	mu	81 *(unknown)*	99 *(unknown)*	99 *(unknown)*	0/3
6	di	100 *(ri)*	99 *(i)*	99 *(i)*	0/3
7	ri	85 *(ri)*	99 *(ri)*	98 *(ri)*	3/3
8	ku	96 *(ku)*	89 *(ku)*	99 *(a)*	2/3
9	kan	100 *(a)*	99 *(a)*	64 *(ma)*	0/3
10	Unknown 1 *(silent)*	99 *(unknown)*	99 *(ri)*	100 *(unknown)*	2/3
11	Unknown 2 *(o)*	99 *(a)*	100 *(a)*	99 *(a)*	0/3
12	Unknown 3 *(mi)*	43 *(a)*	59 *(ri)*	99 *(unknown)*	3/3

Tabel 12: Hasil Penguji 4 pada latar tidak berisik.

No	Spoken Syllable	Not Noisy Background			Correct Result
		1	2	3	
1	a	99 *(unknown)*	75 *(a)*	100 *(a)*	2/3
2	i	88 *(i)*	99 *(i)*	99 *(ri)*	2/3
3	na	99 *(na)*	99 *(ma)*	99 *(ma)*	1/3
4	ma	55 *(ma)*	72 *(kan)*	99 *(na)*	0/3
5	mu	99 *(unknown)*	94 *(unknown)*	28 *(i)*	0/3
6	di	89 *(i)*	99 *(ri)*	99 *(ri)*	0/3
7	ri	99 *(ri)*	100 *(ri)*	99 *(ri)*	3/3
8	ku	39 *(kan)*	99 *(ku)*	58 *(mu)*	1/3
9	kan	100 *(a)*	99 *(a)*	100 *(a)*	3/3
10	Unknown 1 *(silent)*	40 *(unknown)*	24 *(unknown)*	60 *(unknown)*	3/3
11	Unknown 2 *(o)*	100 *(a)*	99 *(a)*	100 *(a)*	0/3
12	Unknown 3 *(mi)*	60 *(ri)*	100 *(i)*	99 *(ri)*	1/3

Meskipun latar berisik dan lumayan berisik, setiap penguji tidak menunjukuan hasil yang cukup baik, 2 atau 3 suku kata masih dapat diprediksi dengan benar dengan akurasi yang tinggi. Latar tidak berisik juga tidak menjamin hasil yang sempurna bahkan pada penguji 1 dan 2. Tetapi, hasilnya lebih baik daripada latar berisik dan lumayan berisik. Selain itu, pengurangan latar suara tidak diterapkan ketika proses ekstrasi yang dilakukan sebelum pembelajaran.

Penguji 1 dan 2 sering mendapat hasil yang lebih baik dari pada penguji 3 dan 4. Pada saat tertentu penguji 3 dan 4 mendapat hasil yang lebih baik. Ini terjadi karena latar suaranya atau bisa juga kemampuan mikrofon untuk merekam suara.

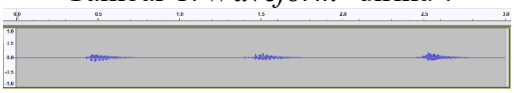
Suku kata ma, mu, dan di, serta suku kata acak o dan mi memiliki hasil yang buruk hampir setiap waktu. Pada suku kata acak o dan mi hal ini disebabkan pembelajaran suara acak berisi lebih banyak latar suara belakang dari pada suku kata o dan mi. Pada suku kata ma, mu, dan di seringkali hasil prediksi benar tapi dengan akurasi yang rendah atau memprediksi suku kata relatif, contoh hasil ri atau i pada suku kata di. Hasil ini dapat ditingkatkan dengan menambahkan data lagi untuk dipelajari karena data yang digunakan terhitung masih relatif sedikit. Model machine learning yang optimum juga dapat meningkatkan hasil prediksi dan akurasi.

Teks acak untuk tes suara yang dihasilkan adalah ‘diriku’, ‘aku makan ikan’, ‘di mana mamamu’, dan ‘halo namaku ivan’

‘diriku’, ‘aku makan ikan’, dan ‘di mana mamamu’ dapat dihasilkan. Namun, ‘halo namaku ivan’ tidak bisa. ‘h’ tidak ditemukan di dalam database saat teks sedang dianalisa dari depan. Aplikasi akan menampilkan peringatan di browser bahwa ‘h’ tidak ditemukan dan akan menampilkan daftar suku kata yang sudah teridentifikasi. Sebenarnya ‘h’ tidak termasuk dalam suku kata pilihan. Teks yang berisikan suku kata yang tidak terpilih maupun terpilih namun belum dianalisa dan didaftarkan maka aplikasi akan menampilkan peringatan dan menghentikan proses pembuatan suara.

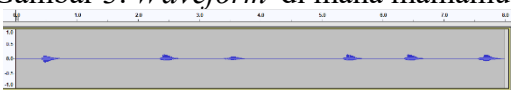
Berikut suara yang dihasilkan dari ‘diriku’, ‘aku makan ikan’, dan ‘di mana mamamu’ secara berurutan:

Gambar 1: *Waveform* ‘diriku’.



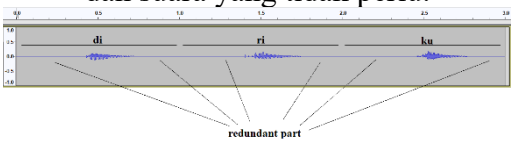
Gambar 2: *Waveform* 'aku makan ikan'.

Gambar 3: *Waveform* 'di mana mamamu'.



Berdasarkan gambar diatas, setiap teks dapat dibedah dan dibedakan secara mudah. Setiap suara direkam dalam 1 detik, maka durasi suara yang dihasilkan dari jumlah suku kata pada teks. Meskipun suara yang dihasilkan baik, dapat didengar dan dimengerti, namun masih ada hening atau suara yang tidak perlu kecuali spasi. Hal itu membuat suara yang dihasilkan kurang terdengar lancer. Hal ini dapat terjadi dikarenakan tidak ada proses lebih lanjut untuk menganalisa dan menghapus suara yang tidak perlu pada proses penggabungan ataupun setelah proses identifikasi dalam aplikasi ini.

Gambar 4: *Waveform* 'diriku' menunjukan suku kata dan suara yang tidak perlu.



5 Diskusi

Pada bagian ini, ada beberapa diskusi mengapa metode yang diusulkan digunakan untuk penelitian ini. Berikut adalah poin - poin penting:

- Concatenative Synthesis untuk Speech Synthesis
- Suku kata untuk satuan suara
- MFCC untuk Speech Recognition
- CNN untuk Machine Learning model

Dalam mimic speech, suara diambil dari suara yang telah dianalisa. Concatenative synthesis dapat menjadi pendekatan terbaik untuk masalah ini. Selain caranya paling mudah untuk digunakan, cara ini juga cepat untuk dikembangkan kedalam aplikasi. Articulatory dan Formant synthesis terlalu kompleks karena parameter yang dibutuhkan banyak untuk membuat system vokal suara ataupun peraturan yang dapat menyesuaikan semua pengguna.

Sangat sulih untuk menemukan penelitian mengenai jumlah pasti *demisyllables*, fonem atau satuan suara yang lebih rendah dalam Bahasa Indonesia. Maka dari itu suku kata adalah pilihan yang paling tepat karena kata akan terlalu banyak memakan memori dan kurang fleksibel dalam membuah suara dalam bentuk kalimat.

Daripada Linear Prediction Coefficients (LPC). MFCC mampu menirukan sistem pendengaran manusia dengan baik. Meskipun Perceptually Based Linear Predictive Analysis (PLP) dapat melakukan hal yang sama [Dave & Pipalia, 2014], MFCC tetap akan dipilih karena MFCC adalah proses ekstrasi yang paling sering digunakan. Teknik tersebut sudah banyak tersebar menjadikan mudah untuk dikembangkan dan di-*debug* dalam aplikasi.

Dua alasan utama menggunakan CNN. Pertama, concatenative synthesis memakai satuan suara. Tiap satuan suara adalah bagian kecil dari suatu kalimat. CNN dianggap sebagai pendekatan optimal untuk menganalisa *small-footprint keyword*, yakni satuan suara itu sendiri, dari pada pendekatan machine learning yang lain [Sainath & Parada, 2015]. Kedua, proses ekstrasi MFCC dapat di plot menjadi spektogram. Dan CNN adalah pendekatan paling umum untuk menganalisa mengenai visual dan gambar, termasuk spektogram.

6 Conclusion

Berikut kesimpulan yang diperoleh dari penilitian ini:

- Aplikasi ini dapat mengumpulkan data suara melalui website.
- Aplikasi ini dapat melatih model machine learning dengan data yang sudah dikumpulkan melalui command prompt.
- Aplikasi ini dapat menirukan suara Bahasa Indonesia melalui website.
- Aplikasi ini dapat mengenali suara dari rekaman suara walaupun prediksi dan akurasi tidak sempurna. Namun, hampir setiap waktu machine learning dapat memprediksi dengan baik.
- Aplikasi ini dapat menghasilkan suara dari text yang dimasukan walaupun masih ada hening atau suara yang tidak perlu. Namun, hasil suara tersebut masih dapat didengar dan dimengerti.

Di masa yang akan datang, penilitian lebih lanjut. Pada speech recognition, model machine learning dapat ditingkatkan. Tidak ada salah atau benar dalam memodelkan machine learning, tetapi selalu ada model yang optimum yang dapat memprediksi dengan akurasi yang paling baik. Pada speech synthesis, menambakan proses untuk menghilangkan hening atau suara tidak perlu dari rekaman suara dan juga menghilangkan atau mengurangi latar suara akan menghasilkan pembuatan suara yang lebih lancar dan baik didengar. Pada Bahasa Indonesia, penelitian tentang fonem Bahasa Indonesia dapat meningkatkan speech recognition dan speech synthesis secara signifikan karena aplikasi ini menggunakan suku kata sebagai satuan suara.

7 Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Bapak Tjong Wan Sen sebagai pembimbing skripsi atas saran dan dukungan selama proses skripsi. Juga kepada dosen yang lain atas dukungan, ilmu, dan pengalaman selama kuliah.

Referensi

[Hande, 2014] Hande, S. S. (2014). A Review on Speech Synthesis an Artificial Voice Production. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 8.

[practicalcrypthography.com, 2012] practicalcryptography.com. (2012). *Mel Frequency Cepstral Coefficient (MFCC) tutorial*. Retrieved

from Practical Cryptography: http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/

[Geitgey, 2016] Geitgey, A. (2016, Juny 14). *Machine Learning is Fun! Part 3: Deep Learning and Convolutional Neural Networks*. Retrieved from Medium:https://medium.com/@ageitgey/machine-learning-is-fun-part-3-deep-learning-and-convolutional-neural-networks-f40359318721

[Sainath & Parada, 2015] Sainath, T. N., & Parada, C. (2015). *Convolutional Neural Networks for Small-footprint Keyword Spotting.* New York, NY, U.S.A: Google, Inc.