

PENENTUAN DIALEK JAWA MENGGUNAKAN METODE RECURRENT NEURAL NETWORK

DETERMINATION OF JAVA DIALEK USING RECURRENT NEURAL NETWORK METHOD

¹Hanif Nilam Pratama Adiarso, ²Andrew Briand Osmond, ³ Anggunmeka Luhur Prasasti.

¹²³Program Studi S1 Sistem Komputer, Fakultas Teknik Elektro, Universitas Telkom

¹hanif.nilam@gmail.com, ²abosmond@telkomuniversity.ac.id, ³Anggunmeka@gmail.com

Abstrak

Indonesia merupakan negara dengan beragam suku dan bahasa, dalam satu suku pun banyak menggunakan dialek atau bahasa yang berbeda-beda. Dalam hal ini penulis membuat suatu sistem aplikasi Speech Processing berteknologi Speech Recognition dengan metode atau algoritma yang digunakan yaitu Recurrent Neural Network (RNN) untuk penentuan dialek Jawa Malang, Jember, Solo, dan Banyumas. Pada saat ini banyak sekali penelitian dan percobaan yang telah dilakukan oleh ahli teknologi modern menggunakan Speech Recognition contohnya seperti google voice milik Google, dan Siri milik Iphone dimana mengolah data input berupa suara menjadi output yang sesuai dengan apa yang diinginkan oleh pengguna aplikasi tersebut. Sama halnya dengan sistem aplikasi penentuan dialek Jawa Malang, Jember, Solo, dan Banyumas menggunakan metode Recurrent Neural Network ini dapat menentukan dan mengenali tidak hanya satu dialek bahasa jawa melainkan empat dialek bahasa jawa, yaitu dialek Jawa Malang, Jawa Jember, Jawa Solo dan Jawa Banyumas.

Recurrent Neural Network sendiri adalah metode yang digunakan dalam pengolahan data pada sistem ini, dimana dapat mencapai akurasi tertinggi 64% dalam pencocokan antara input dan output yang dihasilkan. Sistem yang dibuat memiliki dasar yang sama seperti google voice, atau google translate akan tetapi dalam fitur google voice hanya bisa dalam bahasa-bahasa tertentu seperti Indonesia, English, dan Jawa. Dalam sistem aplikasi ini input yang berupa suara bahasa berdialek Jawa akan diolah hingga menjadi output dapat menentukan dialek Jawa Malang, Jember, Solo, dan Banyumas. Tidak hanya mengenali satu dialek Jawa saja akan tetapi juga dapat menentukan dialek Jawa dari daerah mana yang dihasilkan.

Kata Kunci : Speech Processing, Speech Recognition, Recurrent Neural Network, dialek

Abstract

Indonesia is a country with a variety of tribes and languages, in one tribe, many use different dialects or languages. In this case the author makes a Speech Processing application system with Speech Recognition technology using the method or algorithm used, namely the Recurrent Neural Network (RNN) for the determination of the Malang, Jember, Solo, and Banyumas dialects. At this time there are a lot of researches and experiments that have been done by modern technology experts using Speech Recognition, for example, Google's Google voice and Iphone's Siri, where processing input data in the form of sound becomes output that is in accordance with what the application user wants. Similar to the application system for determining the dialects of Javanese Malang, Jember, Solo, and Banyumas using the Recurrent Neural Network method can determine and recognize not only one Javanese dialect but four Javanese dialects, namely Javanese dialects in Malang, Java Jember, Java Solo and Java Banyumas.

Recurrent Neural Network itself is a method used in processing data on this system, which can achieve the highest accuracy of 64% in matching between input and output produced. The system created has the same basis as google voice, or google translate, but in google voice features can only be in certain languages such as Indonesia, English, and Java. In this application system the input in the form of a Javanese dialect language sound will be processed until it can be output can determine the Java dialects of Malang, Jember, Solo, and Banyumas. Not only recognize one Javanese dialect but also can determine the Javanese dialect of which area is produced.

Keywords: Speech Processing, Speech Recognition, Recurrent Neural Network, dialect

Keywords: Speech Processing, Speech Recognition, Recurrent Neural Network, dialect

1. Pendahuluan

1.1 Latar Belakang

Indonesia merupakan negara yang memiliki beragam suku, setiap suku juga memiliki bahasa yang berbeda-beda. Menurut data dari BPS Terdapat 1211 bahasa (1158 bahasa daerah) tahun 2010[1]. Bahasa Jawa merupakan salah satu bahasa daerah yang ada di Indonesia yang terletak di pulau Jawa yang berada di provinsi Jawa Timur dan Jawa Tengah. Banyak daerah di provinsi Jawa Timur dan Jawa Tengah yang menggunakan bahasa dan dialek Jawa seperti Solo, Banyumas, Malang dan Jember. Meskipun dari semua daerah di Jawa Timur dan Jawa Tengah menggunakan bahasa Jawa akan tetapi setiap daerah memiliki dialek yang berbeda-beda.

Masih banyak daerah-daerah dengan berbagai macam suku bahasa di Indonesia yang masih mengalami kesulitan dalam memahami atau mengenali bahasa antar suku atau daerah yang ada. Sehingga masih sering terjadi kesalahpahaman dalam memahami maksud dan tujuan bahasa daerah tersebut.

Penelitian tentang pengenalan antar bahasa daerah yang ada di Indonesia masih sedikit saat ini. Oleh karena itu dibuatlah penelitian yang mendeteksi dialek atau bahasa antar suku daerah yang ada di Indonesia.

Dalam penelitian ini akan dibuat sebuah aplikasi yang akan menentukan bahasa atau dialek Jawa sebagai masukannya dan keluarannya berupa teks dan dialek dari bahasa Jawa.

Metode yang akan digunakan adalah Recurrent Neural Network. Pada penelitian sebelumnya dengan metode yang sama menggunakan bahasa Manado proses kinerja sistem didapatkan hasil akurasi rata-rata sebesar 87%. Semakin banyak epoch dan data latih yang banyak akan meningkatkan tingkat akurasi. Pada penelitian kali ini metode yang digunakan diharapkan mampu memiliki akurasi hingga 80% dan dapat menentukan dialek Jawa.

1.2 Tujuan

Adapun tujuan yang akan dicapai pada Tugas Akhir ini :

Tujuan dari tugas akhir ini adalah :

1. Membuat aplikasi Speech Recognition menggunakan bahasa Jawa.
2. Mengimplementasikan metode Recurrent Neural Network kedalam aplikasi tersebut.
3. Menampilkan hasil output program berupa teks bahasa Jawa dan dapat menentukan dialek bahasa Jawa.

1.3 Identifikasi Masalah

Adapun identifikasi masalah dalam tugas akhir sebagai berikut:

1. Input hanya berupa bahasa dialek Jawa yang terdiri dari Malang, Jember, Solo, Banyumas.
2. Data latih dibuat sendiri.
3. Aplikasi yang dibuat menggunakan Matlab.
4. Metode yang digunakan hanya Recurrent Neural Network dan Long Short-Term Memory.

2. Dasar Teori

Bagian ini berisi tentang dasar teori yang digunakan untuk merancang speech recognition dengan mengimplementasikan metode recurrent neural network untuk menentukan Dialek Jawa Malang, Jawa Jember, Jawa Solo, Jawa Banyumas. Adapun teori-teori yang digunakan adalah sebagai berikut.

2.1 Speech Processing

Speech yaitu sinyal ucapan yang merupakan bentuk khusus dari sinyal suara (voice)[2,3]. Adapun pemrosesan khususnya yaitu Speech processing. Speech Processing adalah analisis ucapan manusia (menggunakan teknik pemrosesan sinyal digital). Ada beberapa aspek dari pemrosesan ucapan, sesuai dengan fokus analisis: sintesis ucapan, pengenalan suara atau ucapan, pengenalan suara, analisis suara, pengkodean dan kompresi ucapan, peningkatan bicara, diarsiasi speaker[19].

Speech merupakan bentuk komunikasi yang paling natural sesama manusia. Sinyal biasanya diproses dalam representasi digital, sehingga pemrosesan ucapan dapat dianggap sebagai kasus khusus pemrosesan sinyal digital, yang diterapkan pada sinyal ucapan. Aspek pemrosesan ucapan meliputi akuisisi, manipulasi, penyimpanan, pemindahan dan keluaran sinyal ucapan. Masukannya disebut Speech recognition dan outputnya disebut Speech synthesis.

penelitian di bidang Speech sangat luas dan baru berkembang di Indonesia, dan relatif sedikit diminati.

Teknologi yang terkenal terkait Speech processing adalah :

- Speech Recognition, Speech to Text
- Text to Speech

2.2 Speech Recognition

Speech Recognition adalah suatu pengembangan teknik dan sistem yang menggunakan komputer untuk menerima masukan berupa kata yang diucapkan. Teknologi ini memungkinkan suatu perangkat untuk mengenali dan memahami kata-kata yang diucapkan dengan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola tertentu yang tersimpan dalam suatu perangkat. Kata-kata yang diucapkan diubah bentuknya menjadi sinyal digital dengan cara mengubah gelombang suara menjadi sekumpulan angka yang kemudian disesuaikan dengan kode-kode tertentu untuk mengidentifikasi kata-kata tersebut. Hasil dari identifikasi kata yang diucapkan dapat ditampilkan dalam bentuk tulisan atau dapat dibaca oleh perangkat teknologi sebagai sebuah komando untuk melakukan suatu pekerjaan, misalnya penekanan tombol pada telepon genggam yang dilakukan secara otomatis dengan komando suara [5,6].

Parameter yang dibandingkan adalah tingkat penekanan suara yang kemudian akan dicocokkan dengan template database yang tersedia. Pada jaman sekarang, Speech Recognition sudah banyak digunakan, namun sayangnya aplikasi yang menggunakan teknologi Speech Recognition, lebih banyak dirancang untuk ter built-in pada mobile phone tertentu, contoh : (Siri pada IOS), dan pada umumnya, teknologi Speech Recognition digunakan untuk salah satu alat bantu menjalankan perintah – perintah singkat, seperti mengetik teks, dan pencarian kontak.

2.2 Mel-Frequency Cepstral Coefficient (MFCC)

MFCC merupakan salah satu metode ekstraksi ciri untuk sinyal akustik terbaik [4]. Analisis suara pada mel-frequency didasarkan pada persepsi pendengaran manusia, karena telinga manusia telah diamati dapat berfungsi sebagai filter pada frekuensi tertentu. *Mel Frequency Cepstrum Coefficients* (MFCC) merupakan satu metode yang banyak dipakai dalam bidang speech recognition. Metode ini digunakan untuk melakukan *feature extraction*, sebuah proses yang mengkonversikan sinyal suara menjadi beberapa parameter. Filter ini digunakan untuk menangkap karakteristik fonetis penting dari sebuah ucapan. MFCC digambarkan dalam skala mel-frekuensi yang merupakan frekuensi linier dibawah 1000Hz dan logaritmik di atas 1000Hz.

2.3 Recurrent Neural Network

RNN adalah salah satu bagian dari keluarga Neural Network untuk memproses data yang bersambung (sequential data). Cara yang dilakukan RNN untuk dapat menyimpan informasi dari masa lalu adalah dengan melakukan looping di dalam arsitekturnya, yang secara otomatis membuat informasi dari masa lalu tetap tersimpan [10].

Model Recurrent Neural Network (RNN) memiliki loop di dalam lapisan tersembunyi. Recurrent Neural Network biasanya digunakan untuk pemodelan skema data. Inti dari RNN berasal dari representasi penyimpanan semua lapisan masukan sebelumnya ke lapisan tersembunyi dan lapisan tersembunyi berikutnya. RNN memiliki bentuk feedforward dan berlaku untuk semua neuron (koneksi penuh). Karena itu, koneksi jaringan yang memungkinkan bentuknya dinamis. RNN memiliki tiga jenis lapisan: x lapisan input, lapisan tersembunyi A, dan lapisan keluaran.

Jika kita membuka model RNN standar maka dapat dianggap sebagai penyalinan struktur yang sama beberapa kali, maka huruf A dari setiap salinan akan digunakan sebagai input berikutnya[11,12,13].

2.4 Long Short-Term Memory

Long short-term memory - biasanya hanya disebut "LSTM" - adalah jenis RNN khusus, yang mampu mempelajari ketergantungan jangka panjang. LSTM diperkenalkan oleh Hochreiter & Schmidhuber (1997), kemudian disempurnakan dan dipopulerkan kembali oleh banyak orang.

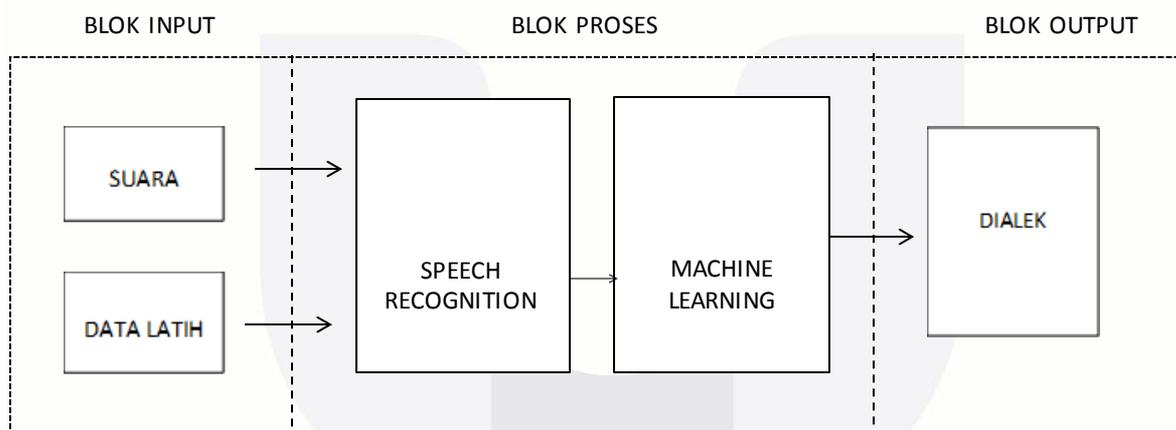
LSTM secara eksplisit dirancang untuk menghindari masalah ketergantungan jangka panjang. Mengingat informasi untuk jangka waktu yang lama, LSTM juga terkenal praktis[15,16].

LSTM berisi blok khusus yang disebut blok memori pada recurrent hidden layer. Blok memori yang berisi memory cell yang menyimpan dalam keadaan sementara. Setiap memori blok pada arsitektur sebenarnya memiliki 2 gate yaitu input dan output gate. Input gate mengontrol aktifitas aliran yang masuk kedalam cell memory. Dan output gate mengontrol aliran aktifitas setelahnya, dan kemudian forget gate ditambahkan pada memory block[10]. Baru-baru ini, konsep autoencoder telah menjadi lebih banyak digunakan untuk belajar model data generative [8].

3. Perancangan

3.1 Perancangan Sistem Secara Umum

Agar aplikasi *desktop* yang dibuat dapat berfungsi dan mengeluarkan *output* sesuai apa yang di harapkan, maka dibuat sebuah perancangan. Berikut adalah gambaran sistem yang akan di buat.



Gambar 3. 1 Diagram Blok Aplikasi

Metode yang digunakan dalam penelitian ini adalah Recurrent Neural Network yang berguna untuk membentuk pola yang akan mempresentasikan pengklasifikasian. Langkah utama yang dilakukan adalah pencarian sampel suara yang akan digunakan pada proses Input sampel berupa audio rekaman dari narasumber yang berbicara bahasa manado untuk dijadikan data latih.

Setelah data terkumpul maka dilakukan proses preprocessing dimana data masukan yang diolah agar dapat menghasilkan bentuk dan nilai setelah itu dilakukan ekstraksi ciri dengan menggunakan Mel-Frequency Cepstral Coefficient (MFCC) agar mendapatkan nilai ciri dari masing-masing data masukan. Setelah itu masuk kedalam Recurrent Neural Network

3.2 Noise Reduction

Sampel yang didapatkan dari narasumber tidak sepenuhnya jernih, dikarenakan proses pengambilan sampel dilakukan di tempat umum jadi dilakukan proses penghilangan noise agar suara dapat terdengar lebih jelas, pada proses Noise reduction penulis menggunakan aplikasi Audacity.

3.3 Flow Diagram Sistem

Cara kerja aplikasi yaitu dengan masukan berupa suara berbahasa Jawa kemudian kalimat yang di ucapkan merupakan susunan dari beberapa kata berbahasa Jawa. Sebelumnya kita membuat sebuah data latih yang dibuat secara mandiri. Data latih yang telah di buat akan dimasukan kedalam metode Recurrent Neural Network menggunakan algoritma LSTM. Setelah itu hasil yang dikeluarkan berupa teks berbahasa Jawa dan mengetahui dialek Jawa itu sendiri.

saat kita membuka aplikasi ada beberapa fitur untuk pengucapan suara berbahasa Jawa. Kemudian akan muncul sebuah keterangan dari aplikasi tersebut bahwa kalimat atau kata yang ducapkan berdialek Jawa

3.4 Blok Diagram Fase Pelatihan dan Pengujian

Untuk membuat sistem speech recognition, diperlukan dua buah subsistem (subsistem pelatihan dan subsistem pengujian).

3.5 Preprocessing

Preprocessing atau pemrosesan awal bertujuan untuk mengolah suara agar dapat diambil karakteristiknya. Sehingga masukan tahap ekstraksi ciri lebih mudah diproses. Pada processing terdapat dua proses, yaitu:

1. Input Audio

Pada tahap ini suara rekaman dari narasumber yang akan dijadikan sebagai data uji dan data latih yang akan dimasukan kedalam program.

2. Normalisasi

Pada proses ini, sinyal di-resampling dengan frekuensi yang nantinya akan ditentukan. Setelah itu dilakukan centering amplitudo, yaitu penggeseran distribusi amplitudo sehingga pusatnya berada pada sumbu=0 dan rata-rata amplitudo sinyal sama dengan nol. Kemudian, sinyal masukan yang bertipe stereo diubah menjadi mono dengan mencari rata-ratanya. Setelah itu, dilakukan proses normalisasi, tujuannya untuk menyamakan amplitudo maksimum sinyal suara sehingga tidak ada pengaruh perubahan amplitudo pada pemrosesan berikutnya.

3. Segmentasi

Tujuan proses ini adalah menyederhanakan sinyal masukan yang berukuran besar. Pada proses pengujian, sinyal masukan berupa rekaman suara dalam bentuk kalimat sehingga diperlukan proses segmentasi sinyal untuk menjadi kata. Sedangkan pada proses pelatihan, sinyal masukan berupa rekaman suara dalam bentuk kata sehingga tidak diperlukan proses segmentasi. Dengan semakin sederhananya sinyal masukan maka proses komputasi sistem akan lebih mudah dan cepat.

3.6 Feature Extraction

Feature extraction atau ekstraksi ciri dilakukan pada dua proses, yaitu ekstraksi ciri untuk pembuatan database sebagai template dan ekstraksi ciri masukan data uji. Secara umum, speech recognizer memproses sinyal suara yang masuk dan menyimpannya dalam bentuk digital. Hasil proses digitalisasi tersebut kemudian dikonversi dalam bentuk spectrum suara yang akan dianalisa dengan membandingkannya dengan template suara pada database sistem.

3.7 Make Class

Pada tahap ini dilakukan pembuatan Class untuk setiap nilai klasifikasi. Terdapat 4 Class yaitu Malang, Jember, Solo, Banyumas.

3.8 Recurrent Neural Network Training

Pada tahap ini, keluaran dari modul pencocokan ciri akan dibandingkan dengan set data latih. Perbandingan dilakukan per kata. Metode yang digunakan pada tahap pencocokan ciri yaitu RNN.

3.9 Recurrent Neural Network Test

Pada tahap ini, adalah serangkaian uji coba yang akan dilakukan untuk menguji akurasi aplikasi terhadap masukan berupa suara. Parameter yang akan diuji adalah sebagai berikut:

1. Akurasi Aplikasi

Pada pengujian ini akan diuji apakah aplikasi dapat menangkap masukan suara dengan jelas. Akan diukur tingkat ketepatan dalam proses speech recognition, apakah data latih yang di proses kemudian di cocokan dengan data uji memiliki kesamaan yang tinggi.

2. Performansi Sistem

Performansi sistem dinilai berdasarkan : Jumlah akurasi sistem / jumlah pengujian x 100%.

3. Hasil

Bentuk keluaran yang di inginkan adalah dapat menentukan berapa akurasi dari dialek yang di uji dari masukan yang berupa kata atau kalimat dari dialek Jawa Malang, Jawa Jember, Jawa Solo dan Jawa Banyumas.

4. Pengujian

Setelah dilakukannya perancangan sistem, untuk mengetahui sistem yang telah dirancang dapat bekerja dengan baik, maka diperlukan pengujian pada sistem untuk mengetahui tingkat keberhasilan dengan menganalisis beberapa parameter yang menjadi tolak ukur keberhasilan sistem tersebut. Pada penelitian ini dilakukan beberapa pengujian untuk menguji keakurasian pada metode yang telah digunakan pada system dengan cara merubah parameter yang terdapat dalam sistem agar mendapatkan akurasi terbaik.

4.1 Pengujian Validasi Sistem.

4.1 Tujuan Pengujian

Tujuan dalam pengujian terhadap sistem ini adalah sebagai berikut :

1. Mengetahui apakah parameter yang ada dalam sistem mempengaruhi hasil tingkat akurasi.
2. Mengetahui apakah metode dari Recurrent Neural Network cocok pada sistem pengenalan ucapan.
3. Mengetahui apakah jumlah data yang di proses dapat mempengaruhi sistem dan parameter yang ada.

4.2 Pengujian Validasi

4.2.1 Pengujian Hidden Layer Dialek Jawa Malang

Dalam pengujian hidden units Dialek Jawa Malang dengan data latih 400 dan data uji 5 untuk mendapatkan hasil akurasi yang tinggi akan ditentukan dengan mencari nilai hidden units dari 10 – 150 dengan kelipatan 10. Berikut tabel hidden units dialek Jawa Malang :

Tabel 4. 1 Pengujian Hidden Units Dialek Jawa Malang

| Class / Data Latih | | Malang / 400 | | | |
|--------------------|------------|--------------|--------------|-------------|---------------------|
| Data Uji | | Malang (5) | | | |
| No | Epoch | Mini Batch | Hidden Units | Acuracy | Time |
| 1 | 150 | 10 | 10 | 80% | 5 min 5 sec |
| 2 | 150 | 10 | 20 | 80% | 5 min 30 sec |
| 3 | 150 | 10 | 30 | 80% | 5 min 24 sec |
| 4 | 150 | 10 | 40 | 80% | 5 min 38 sec |
| 5 | 150 | 10 | 50 | 80% | 6 min 19 sec |
| 6 | 150 | 10 | 60 | 100% | 6 min 29 sec |
| 7 | 150 | 10 | 70 | 80% | 5 min 46 sec |
| 8 | 150 | 10 | 80 | 100% | 2 min 43 sec |

| | | | | | |
|----|-----|----|-----|------|--------------|
| 9 | 150 | 10 | 90 | 100% | 3 min 0 sec |
| 10 | 150 | 10 | 100 | 100% | 4 min 12 sec |
| 11 | 150 | 10 | 110 | 80% | 4 min 24 sec |
| 12 | 150 | 10 | 120 | 20% | 6 min 22 sec |
| 13 | 150 | 10 | 130 | 40% | 6 min 32 sec |
| 14 | 150 | 10 | 140 | 80% | 8 min 25 sec |
| 15 | 150 | 10 | 150 | 80% | 9 min 42 sec |

Dalam tabel telah ditentukan hidden units 80 dengan akurasi 100% untuk dialek Jawa Malang.

4.2.2 Pengujian Hidden Units Dialek Jawa Jember

Dalam pengujian hidden units Dialek Jawa Jember dengan data latih 400 dan data uji 5 untuk mendapatkan hasil akurasi yang tinggi akan ditentukan dengan mencari nilai hidden units dari 10 – 150 dengan kelipatan 10. Berikut tabel hidden units dialek Jawa Jember :

Tabel 4. 2 Pengujian Hidden Units Dialek Jawa Jember

| Class / Data Latih | | Jember / 400 | | | |
|--------------------|-----------|--------------|--------------|-------------|---------------------|
| Data Uji | | Jember (5) | | | |
| No | Epoch | Mini Batch | Hidden Units | Acuracy | Time |
| 1 | 90 | 10 | 10 | 80% | 4 min 1 sec |
| 2 | 90 | 10 | 20 | 60% | 2 min 32 sec |
| 3 | 90 | 10 | 30 | 80% | 2 min 41 sec |
| 4 | 90 | 10 | 40 | 40% | 4 min 8 sec |
| 5 | 90 | 10 | 50 | 40% | 3 min 57 sec |
| 6 | 90 | 10 | 60 | 100% | 4 min 27 sec |
| 7 | 90 | 10 | 70 | 80% | 4 min 3 sec |
| 8 | 90 | 10 | 80 | 80% | 4 min 9 sec |
| 9 | 90 | 10 | 90 | 40% | 4 min 49 sec |
| 10 | 90 | 10 | 100 | 60% | 5 min 13 sec |
| 11 | 90 | 10 | 110 | 40% | 5 min 18 sec |
| 12 | 90 | 10 | 120 | 40% | 5 min 0 sec |
| 13 | 90 | 10 | 130 | 20% | 5 min 21 sec |
| 14 | 90 | 10 | 140 | 40% | 5 min 13 sec |
| 15 | 90 | 10 | 150 | 100% | 5 min 27 sec |

Dalam tabel telah ditentukan hidden units 60 dengan akurasi 100% untuk dialek Jawa Jember

4.2.3 Pengujian Hidden Units Dialek Jawa Solo

Dalam pengujian hidden units Dialek Jawa Solo dengan data latih 400 dan data uji 5 untuk mendapatkan hasil akurasi yang tinggi akan ditentukan dengan mencari nilai hidden units dari 10 – 150 dengan kelipatan 10. Berikut tabel hidden units dialek Jawa Solo :

Tabel 4. 3 Pengujian Hidden Units Dialek Jawa Solo

| Class / Data Latih | | Solo / 400 | | | |
|--------------------|-----------|------------|--------------|------------|---------------|
| Data Uji | | Solo (5) | | | |
| No | Epoch | Mini Batch | Hidden Units | Acuracy | Time |
| 1 | 10 | 20 | 10 | 80% | 14 sec |
| 2 | 10 | 20 | 20 | 80% | 14 sec |
| 3 | 10 | 20 | 30 | 80% | 21 sec |
| 4 | 10 | 20 | 40 | 80% | 19 sec |
| 5 | 10 | 20 | 50 | 80% | 17 sec |
| 6 | 10 | 20 | 60 | 80% | 17 sec |
| 7 | 10 | 20 | 70 | 80% | 18 sec |
| 8 | 10 | 20 | 80 | 80% | 16 sec |
| 9 | 10 | 20 | 90 | 80% | 17 sec |
| 10 | 10 | 20 | 100 | 80% | 19 sec |

| | | | | | |
|----|----|----|-----|-----|--------|
| 11 | 10 | 20 | 110 | 80% | 17 sec |
| 12 | 10 | 20 | 120 | 80% | 16 sec |
| 13 | 10 | 20 | 130 | 80% | 17 sec |
| 14 | 10 | 20 | 140 | 80% | 15 sec |
| 15 | 10 | 20 | 150 | 80% | 20 sec |

Dalam tabel telah ditentukan hidden units 10 dengan akurasi 80% untuk dialek Jawa Solo

4.2.4 Pengujian Hidden Layer Dialek Jawa Banyumas

Dalam pengujian hidden units Dialek Jawa Banyumas dengan data latih 400 dan data uji 5 untuk mendapatkan hasil akurasi yang tinggi akan ditentukan dengan mencari nilai hidden units dari 10 – 150 dengan kelipatan 10. Berikut tabel hidden units dialek Jawa Banyumas :

Tabel 4. 4 Pengujian Hidden Units Dialek Jawa Banyumas

| Class / Data Latih | | Banyumas / 400 | | | |
|--------------------|-----------|----------------|--------------|------------|---------------------|
| Data Uji | | Banyumas (5) | | | |
| No | Epoch | Mini Batch | Hidden Units | Acuracy | Time |
| 1 | 70 | 10 | 10 | 60% | 1 min 57 sec |
| 2 | 70 | 10 | 20 | 60% | 2 min 5 sec |
| 3 | 70 | 10 | 30 | 60% | 2 min 3 sec |
| 4 | 70 | 10 | 40 | 60% | 2 min 11 sec |
| 5 | 70 | 10 | 50 | 60% | 2 min 29 sec |
| 6 | 70 | 10 | 60 | 40% | 2 min 25 sec |
| 7 | 70 | 10 | 70 | 60% | 2 min 34 sec |
| 8 | 70 | 10 | 80 | 60% | 2 min 27 sec |
| 9 | 70 | 10 | 90 | 60% | 2 min 43 sec |
| 10 | 70 | 10 | 100 | 60% | 2 min 34 sec |
| 11 | 70 | 10 | 110 | 80% | 1 min 49 sec |
| 12 | 70 | 10 | 120 | 60% | 3 min 8 sec |
| 13 | 70 | 10 | 130 | 80% | 3 min 0 sec |
| 14 | 70 | 10 | 140 | 40% | 2 min 39 sec |
| 15 | 70 | 10 | 150 | 80% | 2 min 40 sec |

Dalam tabel telah ditentukan hidden layer 110 dengan akurasi 80% untuk dialek Jawa Banyumas .

4.2.3 Pengujian Pengguna Dialek Jawa Malang

Pengujian kalimat pengguna dialek Jawa Malang yang menggunakan 5 kalimat berdialek Jawa Malang dan 5 orang yang berbeda dengan epoch 150, mini batch 10, hidden units 80, dan data latih 400. Berikut tabel pengujian kalimat pengguna dialek Jawa Malang :

Tabel 4. 5 Pengujian Pengguna Dialek Jawa Malang

| Class / Data Latih | | Malang / 400 | | | |
|-------------------------------|---------------------------|----------------|------------|--------------|-------------------|
| Data Uji / Pengguna / kalimat | | Malang / 5 / 5 | | | |
| No | Kalimat | Epoch | Mini Batch | Hidden Units | Rata-rata Kalimat |
| 1 | Yok opo kabare, sehat a? | 150 | 10 | 80 | 64% |
| 2 | Awakmu wes mangan a? | 150 | 10 | 80 | 60% |
| 3 | Kapan awakmu budal? | 150 | 10 | 80 | 52% |
| 4 | Kerjoanku sek gurung mari | 150 | 10 | 80 | 60% |
| 5 | Mene onok tugas opo? | 150 | 10 | 80 | 60% |
| Rata- Rata Total | | | | | 59% |

Dari tabel 4. 5 setiap kalimatnya telah diuji sebanyak 5 pengujian dengan orang yang berbeda untuk taber setiap kalimat terdapat pada lampiran. Rata-rata total yang dihasilkan untuk dialek Jawa Malang adalah 59% dengan 400 data latih.

4.2.4 Pengujian Pengguna Dialek Jawa Jember

Pengujian kalimat pengguna dialek Jawa Jember yang menggunakan 5 kalimat berdialek Jawa Jember dan 5 orang yang berbeda dengan epoch 90, mini batch 10, hidden units 60, dan data latih 400. Berikut tabel pengujian kalimat pengguna dialek Jawa Jember :

Tabel 4. 6 Pengujian Pengguna Dialek Jawa Jember

| Class / Data Latih | | Jember / 400 | | | |
|-------------------------------|----------------------------|----------------|------------|--------------|-------------------|
| Data Uji / Pengguna / kalimat | | Jember / 5 / 5 | | | |
| Pengujian | Kalimat | Epoch | Mini Batch | Hidden Units | Rata-rata Kalimat |
| 1 | Yok opo kabarmu sak iki? | 90 | 10 | 60 | 60% |
| 2 | Jeneng lengkapmu sopo? | 90 | 10 | 60 | 64% |
| 3 | Loro opo awakmu? | 90 | 10 | 60 | 64% |
| 4 | Ojok lali sesok onok tugas | 90 | 10 | 60 | 68% |
| 5 | Wes bengi ayo mole | 90 | 10 | 60 | 64% |
| Rata- Rata Total | | | | | 64% |

Dari tabel 4. 6 setiap kalimatnya telah diuji sebanyak 5 pengujian dengan orang yang berbeda untuk taber setiap kalimat terdapat pada lampiran. Rata-rata total yang dihasilkan untuk dialek Jawa Jember adalah 64% dengan 400 data latih.

4.2.5 Pengujian Pengguna Dialek Jawa Solo

Pengujian kalimat pengguna dialek Jawa Solo yang menggunakan 5 kalimat berdialek Jawa Solo dan 5 orang yang berbeda dengan epoch 10, mini batch 20, hidden units 10, dan data latih 400. Berikut tabel pengujian kalimat pengguna dialek Jawa Solo :

Tabel 4. 7 Pengujian Pengguna Dialek Jawa Solo

| Class / Data Latih | | Solo / 400 | | | |
|-------------------------------|-------------------------------|--------------|------------|--------------|-------------------|
| Data Uji / Pengguna / Kalimat | | Solo / 5 / 5 | | | |
| Pengujian | Kalimat | Epoch | Mini Batch | Hidden Units | Rata-rata Kalimat |
| 1 | Bapak sampun sepuh yuswonipun | 10 | 20 | 10 | 80% |
| 2 | Kulo remen kaleh panjenengan | 10 | 20 | 10 | 60% |
| 3 | Asmane panjenengan sinten? | 10 | 20 | 10 | 76% |
| 4 | Dinten minggu wayahe leren | 10 | 20 | 10 | 60% |
| 5 | Bengi niki hawanipun asrep | 10 | 20 | 10 | 60% |
| Rata- Rata Total | | | | | 67% |

Dari tabel 4. 7 setiap kalimatnya telah diuji sebanyak 5 pengujian dengan orang yang berbeda untuk taber setiap kalimat terdapat pada lampiran. Rata-rata total yang dihasilkan untuk dialek Jawa Solo adalah 67% dengan 400 data latih.

4.2.6 Pengujian Pengguna Dialek Jawa Banyumas

Pengujian kalimat pengguna dialek Jawa Banyumas yang menggunakan 5 kalimat berdialek Jawa Banyumas dan 5 orang yang berbeda dengan epoch 70, mini batch 10, hidden units 110, dan data latih 400. Berikut tabel pengujian kalimat pengguna dialek Jawa Banyumas :

Tabel 4. 8 Pengujian Pengguna Dialek Jawa Banyumas

| Class / Data Latih | | Banyumas / 400 | | | |
|-------------------------------|---|------------------|------------|--------------|-------------------|
| Data Uji / Pengguna / Kalimat | | Banyumas / 5 / 5 | | | |
| Pengujian | Kalimat | Epoch | Mini Batch | Hidden Units | Rata-rata Kalimat |
| 1 | Asikk, gae aku mas? iye, seriusan? serius | 70 | 10 | 110 | 60% |
| 2 | Keprigen carane motong kayu kue, tapi ora konangan sopo bae | 70 | 10 | 110 | 60% |
| 3 | Fad misale koe dadi wong sugih, koe pengen apa? | 70 | 10 | 110 | 68% |
| 4 | Pada ngapa kie? hey mas nangndi,ki badhe nang alas | 70 | 10 | 110 | 72% |
| 5 | Kon mak e inyong, nyong kon tuku iwak seger | 70 | 10 | 110 | 64% |
| Rata- Rata Total | | | | | 65% |

Dari tabel 4. 8 setiap kalimatnya telah diuji sebanyak 5 pengujian dengan orang yang berbeda untuk taber setiap kalimat terdapat pada lampiran. Rata-rata total yang dihasilkan untuk dialek Jawa Banyumas adalah 65% dengan 400 data latih.

4.3 Rata-rata Pengujian Peformansi Pengguna

Setelah melakukan pengujian kalimat pengguna dari setiap dialek yang menjadi inputan pada sistem maka didapatkan rata-rata pengujian peformansi pengguna, apakah sistem yang dibuat dapat menentukan dialek daerah Jawa. Berikut tabel rata-rata pengujian peformansi pengguna dengan 400 dan 800 data latih :

Tabel 4. 9 Rata-Rata Pengujian Peformansi Pengguna 400 Data Latih

| No | Dialek | Rata-rata Dialek |
|---------------------------------|---------------|------------------|
| 1 | Jawa Malang | 59% |
| 2 | Jawa Jember | 64% |
| 3 | Jawa Solo | 67% |
| 4 | Jawa Banyumas | 65% |
| Rata-Rata Peforma sistem | | 64% |

Pada tabel 4. 9 di dapatkan rata-rata pengujian peformansi pengguna yaitu 64% untuk 400 data latih, dimana akurasi pada pengujian pengguna dapat menentukan dialek Jawa Malang, Jawa Jember, Jawa Solo, dan Jawa Banyumas.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan Berdasarkan hasil pengujian dan analisa yang telah dilakukan dari Tugas Akhir ini, maka dapat ditarik kesimpulan sebagai berikut :

1. Recurrent Neural Network dapat digunakan untuk menentukan dialek Jawa Malang, Jawa Jember, Jawa Solo, dan Jawa Banyumas.
2. Pada proses pengujian peformansi pengguna didapatkan hasil akurasi rata-rata sebesar 64% untuk 400 data latih dan 62% untuk 800 data latih.
3. Selisih rata-rata total dari pengujian peformansi pengguna dengan 400 dan 800 data latih adalah 2%.
4. Setiap parameter memiliki pengaruh terhadap akurasi dan peformansi pada sistem.
5. Maksimal Epoch dan minimal MiniBatchSize merupakan penilaian penting untuk menghasilkan tingkat akurasi yang tinggi pada sistem.

5.2 Saran

Berdasarkan dari Penelitian Tugas akhir ini maka, penulis menyarankan untuk penelitian selanjutnya adalah sebagai berikut :

1. Diperlukan penelitian lanjutan untuk melatih sistem Recurrent neural network khususnya untuk penelitian tentang speech recognition.
2. Untuk pengambilan data latih dan data uji dapat berupa suara rekaman dan dilakukan di tempat yang tidak memiliki tingkat kebisingan rendah.
3. Penelitian lebih lanjut dapat menambahkan parameter untuk menghitung tingkat akurasi.
4. Penelitian lebih lanjut dapat menambahkan *Class* agar klasifikasi lebih kompleks.

Daftar Pustaka:

- [1] <http://www.nafiun.com/2013/02/bahasa-dialek-perbedaan-mitos-legenda-dongeng-lisan-pengertian.html>, diakses pada 19 september 2017
- [2] <http://www.metode-algoritma.com/2016/01/voiceSpeech-recognition-pengenalan.html>, diakses pada 19 september 2017
- [3] <http://www.ece.ucsb.edu>
- [4] Benesty, Jacob, M. Mohan Sondhi, dan Yiteng Huang (2008). Handbook of Speech Processing. Springer-Verlag. p. 6. ISBN 159904840X.
- [5] Hu, Yu Hen dan Jenq-Neng Hwang. (2002). Handbook of Neural Network Signal Processing. Florida: CRC Press LLC.
- [6] Jelinek, Frederick (1997). Statistical Methods For Speech Recognition. Massachusetts Institute of Technology. p. 17. ISBN 0262100665.
- [7] R. S. Chavan dan G. S. Sable, "An Overview of Speech Recognition Using HMM," International Journal of Computer Science and Mobile Computing, vol. 2, no. 6, 2013.
- [8] Gurney, K. (1997) An Introduction to Neural Networks London: Routledge. ISBN 1-85728-673-1 (hardback) or ISBN 1-85728-503-4 (paperback)
- [9] Fausett, L. (1994). Fundamentals of Neural Network (Archetectors, Algorithms, and Applications). Upper Saddle River, New-Jersey: Prentice-Hall.
- [10] <http://kkurniawan.com/blog/mn-bahasa-indonesia>, diakses pada 21 september 2017
- [11] Naiyang Guan, Yuan Li, Xiang Zhang, Zhigang Luo Weijiang Feng, "Audio Visual Speech Recognition with Multimodal," in Neural Networks (IJCNN), 2017 International Joint Conference , Anchorage, AK, USA, 2017, pp. 681-688.
- [12] Abhinav Sethy, Bhuvana Ramabhadran, Stanley Chen Ebru Arisoy, "Bidirectional recurrent neural network language models for automatic speech recognition," in Acoustics, Speech and Signal Processing (ICASSP), Brisbane, QLD, Australia, 2015, p. 5421.
- [13] Saliza Ismail, Den Fairol SamaonL Abdul Manan Ahmad, "Recurrent neural network with backpropagation through time for speech recognition," in Communications and Information Technology, 2004. ISCIT 2004, Sapporo, Japan, Japan, 2004, pp. 98-102.
- [14] <https://karpathy.github.io/2015/05/21/mn-effectiveness/>, diakses pada 21 september 2017
- [15] LiMin Fu, "Neural Networks in Computer Intelligence. 1994.
- [16] Klaus Greff; Rupesh Kumar Srivastava; Jan Koutník; Bas R. Steunebrink; Jürgen Schmidhuber (2015). "LSTM: A Search Space Odyssey"
- [17] <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>, diakses pada 21 september 2017
- [18] <http://id.affdu.com/top-5-free-voice-recognition-apps-for-your-android-phone.html>, diakses pada 05 oktober 2017
- [19] <http://www.vicomtech.org/en/t4/e12/speech-processing>