e-ISBN 978-623-93343-3-8
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 20 April 2021

Model Klasifikasi Emosi Berdasarkan Suara Manusia Dengan Metoode Multilater Perceptron

Deni Ardiyansyah [1], Jayanta [2]
Informatika
Universitas Pembangunan Nasional Veteran Jakarta
Jalan Rumah Sakit Fatmawati, Pondok Labu, Jakarta Selatan 12450
denia@upnvj.ac.id

Abstrak. Teknologi interaksi manusia dengan komputer sudah semakin berkembang, misalnya pengenalan suara atau *speech recognition.* Salah satu kegunaan dari pengenalan suara adalah untuk mengenali emosi manusia. Komputer dapat mengenali dan mengklasifikasi emosi manusia berdasarkan suara. Sudah banyak penelitian terkait dengan berbagam metode ekstraksi ciri dan klasifikasi namun hasilnya masih belum mendekati sempurna. Adapun metode ekstraksi ciri menggunakan Mel Frequency Ceptral Coefficient (MFCC). Data yang digunakan adalah data sekunder yang bersumber dari *Ryerson Audio-Visual Database of Emotional Speech and Song* (RAVDESS). Model sistem akan dapat mengenali 8 jenis emosi yaitu netral, tenang, senang, sedih, marah, takut, jijik dan terkejut. Hasil dari model didapatkan akurasi untuk emosi netral sebesar 98%, emosi tenang sebesar 97%, emosi senang sebesar 94%, emosi sedih sebesar 97%, emosi marah sebesar 97%, emosi takut sebesar 94%, emosi jijik sebesar 97% dan emosi terkejut sebesar 96%. Sehingga hasil akurasi rata-rata dari model yang telah dibuat sebesar 96%

Kata Kunci: *Speech Recognition*, Emosi, Suara, Klasifikasi

1 Pendahuluan

Teknologi saat ini sudah semakin maju seiring perkembangan zaman. Khususnya di bidang interaksi manusia dengan komputer. Manusia terus berusaha melakukan penelitian supaya komputer dapat merepresentasikan apa yang manusia inginkan. Misalkan dalam pengenalan emosi. Komputer dapat melakukan pengenalan emosi salah satunya berdasarkan suara. Suara manusia memiliki banyak informasi penting yang dapat diidentifikasi dan dikenali berupa ciri. Setiap orang memiliki ciri suara yang berbeda maka dari itu perlu dilakukan ekstraksi ciri untuk mendapat hasil ciri suara tersebut yang nantinya akan dilakukan klasifikasi untuk menentukan emosi apa yang sedang dirasakan oleh orang tersebut.

Sudah banyak penelitian yang berfokus pada pengenalan emosi berdasarkan suara dengan beragam metode ekstraksi ciri dan metode klasifikasi. Penelitian-penelitian tersebut menggunakan dataset ucapan yang tersedia sebagai standar seberapa baik sebuah sistem melakukan pengenalan emosi. Banyak *database* emosi yang disediakan oleh beberapa lembaga diluar negeri seperti SUSAS (*Speech Under Simulated and Actual Stress*), *Berlin database of emotional speech* (Emo-DB), *Surrey Audio-Visual Expressed Emotion* (SAVEE) *Database*, *The Ryerson Audio-Visual Database of Emotional Speech and Song* (RAVDESS) dan masih banyak lagi. Adapun beberapa penelitian yang dilakukkan untuk mengenali emosi diantaranya oleh A. Iqbal & K. Barua pada tahun 2019 dengan judul *A Real-time Emotion Recognition from Speech using Gradient Boosting* dengan menggunakan dataset RAVDESS dan metode klasifikasi SVM, KNN dan *Gradient Boosting*. Hasil akurasi untuk data suara pria dengan metode SVM sebersar 65%, metode KNN sebesar 56% dan metode *Gradient Boosting* sebesar 70%. Lalu hasil akurasi untuk data suara wanita dengan metode SVM sebersar 45%, metode KNN sebesar 55% dan metode *Gradient Boosting* sebesar 62%. Kemudian penelitian serupa lainnya dilakukan oleh D. Issa et.al pada tahun 2020 dengan judul *Speech Emotion Recognition With Deep Convolutional Neural Networks* dengan menggunakan dataset yang sama yaitu RAVDESS dan mencapai akurasi sebesar 71,61%.

Adapun metode ekstraksi ciri yang paling sering digunakan dalam pengolahan suara adalah *Mel Frequency Cepstral Coefficient* (MFCC) seperti penelitian yang telah dilakukan oleh R. Umar et.al pada tahun 2019 dengan judul Analisis Bentuk Pola Suara Menggunakan Ekstraksi Ciri *Mel-Frequencey Cepstral Coefficients* (MFCC) dengan menggunakan data suara dari beberapa pengguna dengan pengucapan yang sama yaitu “login”. Hasilnya

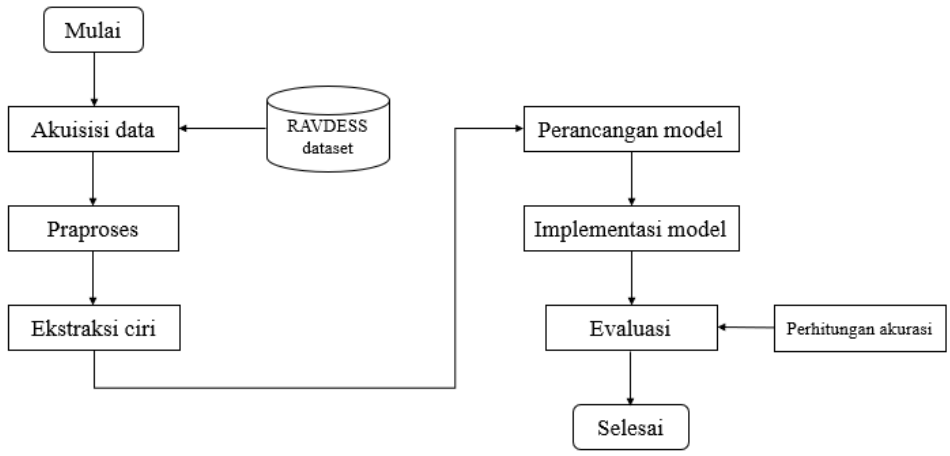
metode MFCC dapat melakukan ekstraksi ciri dengan baik sehingga memberikan hasil bentuk pola suara yang berbeda dari setiap penggunanya.

Lalu metode klasfikasi yang memiliki tingkat akurasi tinggi adalah Multi Layer Perceptron (MLP). Metode ini merupakan turunan dari Jaringan Syaraf Tiruan (JST). Akurasi tinggi dari MLP dalam proses klasifikasi dibuktikan dari penelitian yang telah dilakukan oleh N. Husain & N. Aji pada tahun 2019 dengan judul Klasifikasi Sinyal EEG Dengan *Power Spectra Density* Berbasis Metode *Welch* Dan MLP *Backpropagation* dengan menggunakan 5 kelas dataset (set A, set B, set C, set D dan set E) dari database yang dibuat oleh Dr. Ralph Andrzejak dari Pusat Epilepsi di Universitas Bonn, Jerman. Hasilnya MLP dapat mengklasifikasi sinyal EEG dengan nilai akurasi yang tinggi yaitu 99,68 %.

Dengan latarbelakangi beberapa referensi tersebut, penulis ingin berkontribusi dengan melakukan penelitian tentang pengenalan emosi dengan judul Model Klasifikasi Emosi Berdasarkan Suara Dengan Metode Multilayer Perceptron. Dataset yang digunakan merupakan data sekunder karena penelitian ini dilakukan pada masa pandemi Covid-19. Adapun dataset bersumber dari database *The Ryerson Audio-Visual Database of Emotional Speech and Song* (RAVDESS). Lalu metode ekstraksi ciri yang digunakan adalah *Mel Frequency Cepstral Coefficient* (MFCC) karena metode ini dapat merepresentasikan sinyal dengan baik sehingga dapat memberikan hasil bentuk pola suara yang berbeda dari setiap pengguna. Lalu Multi Layer Perceptron (MLP) dipilih untuk metode klasifikasi dalam penelitian ini karena mampu melakukan klasifikasi pada dataset multiclass dengan baik. Adapun algoritma pembelajaran yang digunakan adalah *Backpropagation* untuk mengupdate nilai bobot.

2 Metodologi Penelitian

Pada tahap ini akan menjelaskan alur penelitian yang akan dilakukan. Tahapan alurnya dapat dilihat pada Gambar.1.



Gambar 1 Kerangka Pikir

2.1 Pengumpulan Data

Tahap pengumpulan data yang dilakukan dari *Ryerson Audio-Visual Database of Emotional Speech and Song* (RAVDESS) dengan sampling 48kHz dan 16 bit. https://www.kaggle.com/uwrfkaggler/ravdess-emotional-speech-audio RAVDESS adalah *database* multimodal ucapan dan lagu emosional yang tervalidasi. Emosi dan

lagu ini direkam menggunakan microphone Sennheiser pada frekuensi 48 kHz, yang disuarakan 24 aktor profesional, menyuarakan pernyataan yang cocok secara leksikal dalam aksen Amerika Utara yang netral. Jumlah data yang digunakan adalah 1430 data yang terdiri dari 95 emosi netral, 183 emosi tenang, 192 emosi senang, 192 emosi sedih, 192 emosi marah, 192 emosi takut, 192 emosi jijik dan 192 emosi terkejut.

2.2 Praproses

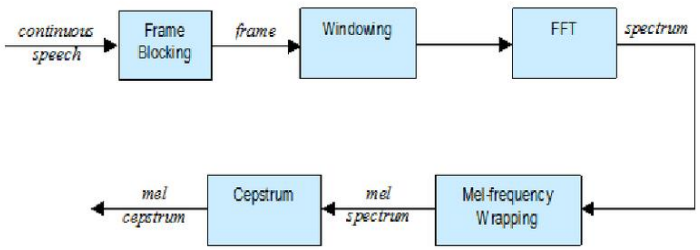
Praproses pada penelitian ini dengan melakukan normalisasi menggunakan filter *Pre-emphasis*. Filter *pre-emphasis* merupakan filter *Finite Impulse Response* (FIR) orde satu (*single tap*) pelolos frekuensi tinggi (*High Pass Filter*, HPF). Dengan menggunakan filter ini, fluktuasi spektrum frekuensi rendah dan tinggi menjadi lebih halus dibandingkan sebelumnya. [1] Proses pre emphasis ditampilkan pada persamaan:

$y[n] = s[n] - a.s\ [n-1],\ 0.9 \leq a \leq 1.0$ (1)

Dimana:
y[n] = signal hasil pre - emphasis
s[n] = signal sebelum pre - emphasis
a = nilai alpha

2.3 Ekstraksi ciri

Merupakan tahapan ekstraksi ciri dengan menggunakan metode *Mel Frequency Cepstral Coefficient* (MFCC). Ekstraksi ciri dalam proses ini ditandai dengan pengubahan data suara menjadi data citra berupa spektrum gelombang. Lalu hasil dari ekstraksi berupa nilai *vector* yang nantinya akan dilakukan proses klasifikasi. Berikut tahapan pada metode MFCC:



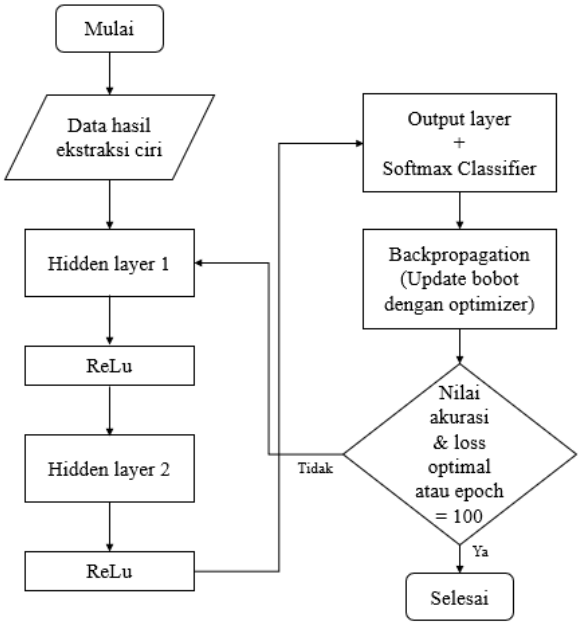
Gambar 2 Tahapan MFCC

2.4 Perancangan Model

Setelah dilakukan ekstraksi ciri, data akan dilakukan pembagian data menjadi 80% data latih dan 20% data uji. Selanjutnya dilakukan penentuan parameter arsitektur MLP yaitu terdiri dari penentuan jumlah hidden layer dan ukuran hidden layer, fungsi aktivasi, *optimizer*, ukuran *batch* dan jumlah *epoch*.
Pada tahap selanjutnya data latih akan menjadi masukan proses *training* menggunakan arsitektur *Multilayer Perceptron Backpropagation*. Adapun algoritma pembelajarannya adalah *Backpropagation* untuk mengupdate nilai bobot supaya hasilnya dapat nilai error yang rendah. Berikut persamaan untuk algoritma *Backpropagation* [2].

Wbaru = Wlama + ΔW (2)
Dimana
W = Weight atau bobot

Berikut rancangan arsitektur model MLP dan parameter yang digunakan



Gambar 3 Rancangan Model MLP Backpropagation

a. Pemilihan jumlah *hidden layer* dan ukuran *hidden layer*

Ditentukan *hidden layer* sejumlah 2 dan ukuran masing-masing *hidden layer* sebesar 256 dan 64.

b. Pemilihan Fungsi Aktivasi

Pada penelitian ini, fungsi aktivasi yang digunakan adalah *ReLu* dan *softmax* pada output layer. Untuk menghitung input, *weight* dan bias.

c. Pemilihan Fungsi Optimasi

Fungsi optimasi atau *Optimizer* yang digunakan adalah Adam untuk mendapatkan bobot yang optimal.

d. Pemilihan Fungsi *Loss*

Fungsi *Loss* adalah fungsi yang menghitung seberapa besar nilai error dari hasil klasifikasi/prediksi. Semakin rendah nilai *error* maka semakin baik hasil klasifikasi/prediksi tersebut.

e. Pemilihan ukuran *batch*

Batch size adalah jumlah sampel data yang diambil dari seluruh dataset. *Batch size* yang digunakan pada penelitian ini sebesar adalah 64. Artinya sample data pertama dan seterusnya yang diambil berjumlah 64 sampai seluruh data pada dataset selesai diambil dan kemudian dilakukan proses training.

f. Pemilihan jumlah *epoch*

Epoch ialah jumlah iterasi untuk mengulangi proses pembelajaran. Epoch ditentukan supaya proses pelatihan dapat berhenti baik hasilnya sudah optimal ataupun belum optimal. Jumlah *epoch* yang dipilih yaitu 100. Setelah seluruh data pada dataset selesai diambil dengan batch size yang telah ditentukan, maka dilakukan proses pelatihan dimulai dari proses pelatihan maju dan pelatihan mundur. Satu proses ini dapat dikatakan

satu kali penggulangan atau 1 epoch dan akan berulang sampai epoch yang ditentukan yaitu 100 epoch supaya model dapat lebih optimal dalam mengenali data.

2.5 Implementasi Model

Arsitektur MLP yang telah dirancang dengan parameter di dalamnya kemudian akan digunakan untuk proses pelatihan dengan menggunakan data latih sebesar 80% dari total dataset yang digunakan. Kemudian hasil pelatihan berupa model dilakukan pengujian dengan menggunakan data uji sebesar 20% dari total dataset. Dari proses pelatihan didapat juga waktu pelatihan untuk melihat seberapa lama waktu komputasi dalam model melakukan proses pelatihan.

2.6 Evaluasi

Pada tahap ini dilakukan evaluasi dari hasil pengujian data dengan menggunakan *confusion matrix*. Nilai yang dihasilkan melalui metode *confusion Matrix* dalam bentuk akurasi. Akurasi adalah presentase jumlah data yang diklasifikasikan (prediksi) secara benar. Berikut tabel dari *confusion matrix*.[2]

Tabel 1 Confusion Matrix

Confussion Matrix		Kelas Prediksi	
		Positif	Negatif
Kelas Sebenernya	Positif	TP	FN
	Negatif	FP	TN

Berdasarkan tabel Confusion Matrix diatas:

a. *True Positive* (TP) = data positif yang terprediksi dengan benar
b. *False Negative* (FP) = data negatif yang terprediksi dengan salah
c. *False Negative* (FN) = data positif yang terprediksi dengan salah
d. *True Negative* (TN) = data negatif yang terprediksi dengan benar

Sehingga rumus yang digunakan untuk menghitung nilai akurasi adalah sebagai berikut:

$$Akurasi = \frac{TN + TP}{TN + TP + FP + FN}$$

3. Hasil Dan Pembahasan

Pada hasil dan pembahasan akan membahas proses persiapan data, praproses, ekstraksi ciri, perancangan model, implementasi model dan evaluasi.

3.1 Pengumpulan Data

Data suara didapat dari database RAVDESS dengan jumlah data 1430 yang terdiri dari 95 emosi netral, 183 emosi tenang, 192 emosi senang, 192 emosi sedih, 192 emosi marah, 192 emosi takut, 192 emosi jijik dan 192 emosi terkejut. Kemudian diseleksi dan dikelompokkan secara manual dari berdasarkan aktor menjadi berdasarkan jenis emosi ke dalam folder seperti pada gambar berikut.

Gambar 4 Folder Dataset

3.2 Praproses data

Pada tahap ini data dipersiapkan untuk dilakukan ekstraksi ciri. Praproses pada penelitian ini yaitu pre emphasis dan silent removal. Tujuan pre emphasis untuk mengurangi noise pada sinyal dengan frekuensi tinggi dan sinyal dengan frekuensi rendah sehingga suara menjadi lebih halus. Lalu tujuan silent removal adalah untuk memotong bagian diam di awal dan akhir suara secara manual dengan menggunakan audio cutter.

3.1.1 Preemphasis

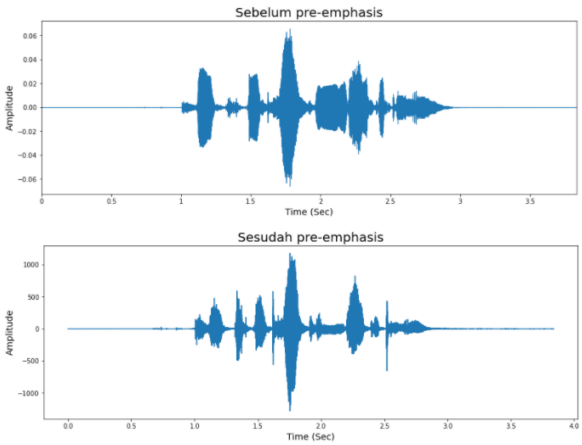
Kemudian dilakukan proses *pre emphasis* untuk mengurangi *noise* pada sinyal yang berfrekuensi tinggi dan rendah supaya suara menjadi lebih halus dengan dilakukan perkalian dengan nilai koefisien dari *alpha* yaitu 0.97. Pada data suara di penelitian ini hanya akan mengambil sinyal yang mengandung wicara saja. Berikut formula dan perhitungan *pre emphasis* pada data suara dengan emosi tenang. [1]

$$y[n] = s[n] - a.s\,[n-1] \quad (3)$$

$$= -5.7104176e\text{-}06 - 0.97x\,[-5.7104176e\text{-}06 - 1]$$

$$= 5.48200099e15$$

Sehingga hasil dari pre emphasis sebagai berikut.

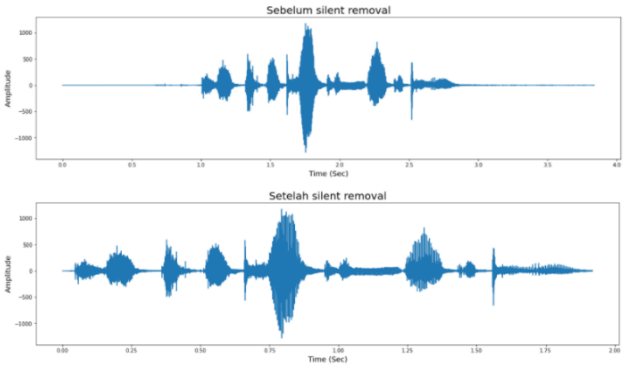


Gambar 5 Preemphasis

Dapat dilihat dari hasil *pre emphasis*, sinyal suara menjadi lebih halus karena telah dilakukan pengurangan *noise* pada sinyal yang berfrekuensi tinggi dan rendah.

3.1.2 Silent Removal

Kemudian dilakukan *silent removal* untuk memotong bagian diam di awal dan akhir karena pada bagian diam tersebut terdapat nilai yang tidak dibutuhkan dalam proses ekstraksi ciri. Seperti penelitian yang telah dilakukan oleh Prasetyo dan Wirawan pada tahun 2017. Hasil keberhasilan dalam pengenalan suara 3,5 kali lebih tinggi dibanding tanpa menggunakan silent removal. Berikut hasil s*ilent removal* pada data suara emosi tenang.



Gambar 6 Silent Removal

3.1 Ekstraksi Ciri

Data suara yang telah dilakukan praproses kemudian masuk ke tahap ekstraksi ciri mengunakan metode MFCC untuk mendapatkan nilai ciri dari setiap data. Metode MFCC terdiri dari tahap *frame blocking*, *windowing*, FFT, *Mel Frequency Wrapping* dan *Cepstrum* dalam bentuk nilai vektor. Adapun parameter yang digunakan yaitu Sample Rate (Fs) 44100 Hz, Overlap 50%, Lebar *frame* 8 ms dan besar jumlah koefisien MFCC 40 Mel. [3]

3.1.1 Frame Blocking

Pada proses ini sinyal audio dibagi ke dalam sebuah *frame* dengan nilai sampling rate yaitu 44100 Hz dengan panjang *frame* (N) 352 sampel yang didapat dari perhitungan dibawah ini

$$N = \frac{Fr}{1000}\ x\ Fs \tag{4}$$

$$N = \frac{8}{1000}\ x\ 44100 = 352\ sampel$$

dan dengan panjang overlapping (M) 100 sampel yang didapat dari perhitungan dibawah ini

$$Fo = Fr\ x\ persentase\ overlap$$

$$Fo = 8\ x\ \frac{50}{100} = 4\ ms$$

$$M = \frac{Fo}{1000}\ x\ Fs \tag{5}$$

$$M = \frac{4}{1000}\ x\ 44100 = 176\ sampel$$

Sehingga didapat jumlah frame sebagai berikut

$$K = \frac{(Fs-N)}{M} + 1 \tag{6}$$

$$K = \frac{(44100-352)}{176} + 1$$

$$= \frac{43748}{176} + 1 = 440\ frame/detik$$

3.1.2 Windowing

Setelah dilakukan *frame blocking* kemudian masuk ke tahap *windowing*. Tujuan dari *windowing* adalah menjaga nilai agar meminimalisir diskontinus atau kehilangan informasi dari setiap *frame*. Fungsi *windowing* yang digunakan pada penelitian ini adalah *Haming window* atau jika dinotasikan adalah *w (n)*. Berikut rumus perhitungan dari Haming window.

$$w(n) = 0.52 - 0.46\ cos\ cos\ \left(\frac{2\pi n}{N-1}\right), 0 \le i \le N-1 \tag{7}$$

$$w(1) = 0.52 - 0.46\ cos\ cos\ \left(\frac{2\pi 1}{352-1}\right) = 0{,}059$$

3.1.3 Fast Fourier Transform

Proses selanjutnya adalah *fast fourier transform* yang bertujuan untuk mengubah domain waktu ke dalam domain frekuensi dengan mengimplementasikan *discrete fourier transform*. Berikut persamaan *fast fourier transform* Dimana X(n) adalah frekuensi, k bernilai 0,1,2 sampai (N-1) N adalah jumlah sampel pada tiap-tiap *frame* j merupakan bilangan imajiner

$$Xn = \sum_{k=0}^{N-1} \quad Xk\, e^{-2\pi_j kn/N} \tag{8}$$

Sehingga perhitungan FFT menggunakan rumus sebagai berikut

$$F(k) = \sum_{n=1}^{N} \quad f(n)\, cos\, cos\, \left(\frac{2\pi nk}{N}\right) - j \sum_{n=1}^{N} \quad f(n) sin \left(\frac{2\pi nk}{N}\right) \tag{9}$$

$$Fo = \left[0{,}059 \left(cos\left(\frac{2\pi * 0 * 0}{1}\right)\right)\right] - j\, sin\, sin\, \left(cos\left(\frac{2\pi * 0 * 0}{1}\right)\right)$$
$$= 0.420997 - 0{,}84146j = -0.28705$$

3.1.4 Mel Frequency Wrapping

Langkah selanjutnya yaitu *Mel frequency wrapping* yang berfungsi untuk menentukan ukuran energi dari sebuah frekuensi agar sesuai dengan frekuensi pendengaran manusia. Berikut perhitungan dari *mel frequency wrapping*.

$$mel(f) = \frac{2595 * log\,(1+\frac{f}{700})}{\frac{Si}{2}} \tag{10}$$

dimana f adalah frekuensi dan Si adalah hasil F FT. Lalu penulis menggunakan hasil FFT 1 ke dalam persamaan tersebut sehingga F*mel* 1 adalah sebagai berikut

$$Fmel = \frac{2595 * log\,(1+\frac{1000}{700})}{\frac{-0.287057}{2}} = -1091684.5$$
$$Fmel\,1 = -0.287057 * 1091684.5 = -313375.768$$

3.1.5 DCT

Langkah terakhir dari MFCC adalah mengubah *mel spectrum* menjadi domain waktu menggunakan *discrete cosine transform* (DCT). Output dari proses ini adalah hasil akhir dari metode MFCC berupa nilai koefisien vektor. Berikut hasil data yang telah dilakukan ekstraksi ciri.

Tabel 2 Hasil Ekstraksi Ciri

Data	Mfcc 1	Mfcc 2	…	Mfcc 12	Mfcc 13
0	-799.6416625976562	-3.8857173919677734	…	0.6061770915985107	1.9533488750457764
2	-801.0642700195312	-3.1395621299743652	…	0.2409388870008392	1.5719976425170898
…	…	…	…	…	…
1429	-733.8917236328125	-0.04636732116341591	…	-0.042407654225826263	0.09197572618722916
1430	-733.6569213867188	-0.15460969507694244	…	0.4010993242263794	-0.10754294693470001

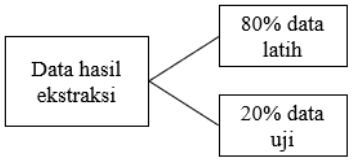
3.2 Perancangan Model

Selanjutnya data hasil ekstraksi ciri kemudian dilakukan klasifikasi dengan menggunakan algoritma MLP *Backpropagation* yang sebelumnya dibuat rancangan arsitektur MLP dengan parameter yang telah ditentukan. Berikut alur cara kerja dari arsitektur MLP yang telah dibuat. [4]

Data hasil ekstraksi ciri menjadi input ke dalam algoritma MLP *Backpropagation* diawali dengan masuk ke dua *hidden layer* yang menggunakan fungsi aktivasi *ReLu* pada setiap layernya. Kemudian masuk ke output layer yang menggunakan fungsi aktifasi *Softmax* dan kemudian masuk ke tahap *Backpropagation* untuk mengupdate bobot kemudian jika nilai akurasi dan loss/error belum optimal atau *epoch* belum sampai 100 kali maka akan mengulangi dengan masuk ke *hidden layer* dan seterusnya hingga mendapat nilai akurasi dan *loss* yang optimal atau akan berhenti saat mencapai *epoch* 100 walaupun nilai akurasi dan *loss/error* belum sampai ke hasil yang diharapkan.

3.3 Pembagian Data

Setelah perancangan arsitektur MLP kemudian data terlebih dahulu dibagi menjadi data latih dan data uji. Pada penelitian ini dilakukan pembagian data dengan perbandingan 80% data latih dan 20% data uji sehingga dari total data sebanyak 1.430 dibagi menjadi 1144 data latih dan 286 data uji.



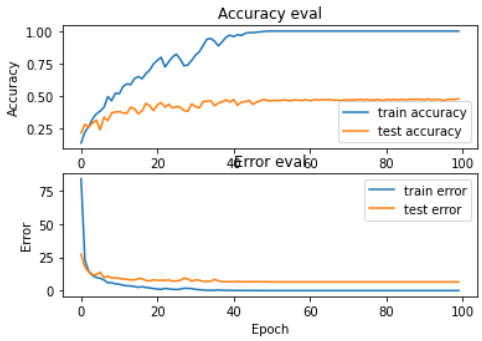
Gambar 7 Pembagian Data

3.4 Implementasi Model

Implementasi model berisikan hasil pelatihan model dari data latih sebesar 80% dengan menggunakan algoritma MLP *Backpropagation* dan pengujian menggunakan data uji sebesar 20% lalu hasil pengujian model dengan menggunakan *confusion matrix* untuk mengetahui tingkat akurasi.

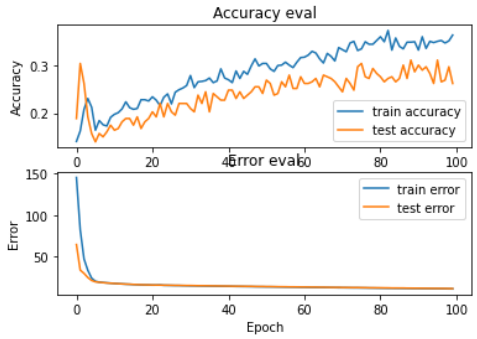
3.5 Hasil Pelatihan model

Pada data uji 80%, dari 1430 data hasil ekstraksi ciri didapat 1144 data latih yang akan dilakukan proses pelatihan dengan menggunakan algoritma MLP *Backpropagation*. Nilai *rate* adalah 0.0001, Optimasi yang digunakan adalah Adam dan fungsi *loss* adalah *sparse categorical crossentropy* yang merupakan teknik untuk mengevaluasi nilai loss terhadap hasil yang bersifat kategori. Adapun proses pelatihan dengan *epoch* sebanyak 100 kali dan ukuran *batch* adalah 64. Berikut hasil parameter dengan menggunakan algoritma MLP Backpropagation.



Gambar 8 Hasil Pelatihan

Terlihat dari gambar diatas bahwa hasil pelatihan pada skema kedua masih *overfit*. Maka penulis mencoba melakukan *Dropout* dengan menghapus jumlah *neuron* secara acak sebesar 30% dan regularisasi L1 dengan memberi nilai *norm*/norma supaya dalam mengupdate bobot menghasilkan bobot yang lebih kecil. Nilai *norm* yang digunakan sebesar 0,1%.



Gambar 9 Hasil Pelatihan dengan Dropout dan Regularisasi

Tabel 3 Hasil Pelatihan Model

Training		*Validation*	
loss	*Accuracy*	*val_loss*	*val_accuracy*
0.0010	10.000	6.4997	0.4755

Tabel 4 Hasil Pelatihan dengan Dropout dan Regularisasi

Training		*Validation*	
loss	*Accuracy*	*val_loss*	*val_accuracy*
10.7217	0.3636	11.0585	0.2622

Kesimpulannya untuk hasil pelatihan model masih *overfit* karena model terlalu mengenali data latih dibanding data uji. Lalu setelah model menggunakan *dropout* dan regularisasi, *overfit* cukup teratasi namun hasil akurasi menurun. Sehingga model yang digunakan untuk proses pengujian adalah model yang telah dilakukan pelatihan tanpa menggunakan *dropout* dan regularisasi.

3.6 Waktu Pelatihan

Tabel 5 Waktu Pelatihan

Waktu Pelatihan (detik)	
Model Skema Pertama	Model Skema Pertama dengan *Dropout* dan Regularisasi
30.79 detik	54.19 detik
Model Skema Kedua	Model Skema Kedua dengan *Dropout* dan Regularisasi
50.96 detik	60.08 detik

Pada tabel diatas terlihat bahwa pada model skema pertama dan kedua dengan *droput* dan regularisasi memiliki waktu pelatihan lebih lama karena regularisasi bekerja dengan menambahkan nilai *norm* pinalti pada *objective function* dan berpanguruh terhadap nilai bobot. Semakin besar nilai bobot maka semakin besar pula nilai *norm* yang ditambahkan yang menjadikan waktu pelatihan lebih lama. Sedangkan dropout tidak membuat waktu pelatihan menjadi lebih lama karena hanya memotong jumlah neuron.

3.7 Hasil Pengujian

Pada tahap ini dilakukan pengujian model skema kedua hasil pelatihan data. Model diuji dengan menggunakan data baru dengan menggunakan parameter ekstraksi ciri yang sama. Pengujian dilakukan untuk mengukur performa model dalam mengenali data baru dan mendapat nilai akurasi yang relatif sama dengan hasil pelatihan. Adapun data uji yang digunakan sebanyak 286 data yang didapat dari 20% data uji dan hasil pengujian menggunakan metode *confusion matrix* untuk mengetahui tingkat akurasi. Dari setiap jenis emosi akan dilakukan *confusion matrix*. Berikut hasil dari *confusion matrix*.

Tabel 6 *Confusion matriks* Semua Kelas Emosi

Confusion matrix		Prediksi Kelas							
		0	1	2	3	4	5	6	7
Kelas Sebenarnya	Netral (0)	13	1	0	0	1	1	0	1
	Tenang (1)	2	23	0	3	0	0	0	0
	Senang (2)	0	0	26	0	0	2	0	1
	Sedih (3)	1	0	1	31	0	0	0	1
	Marah (4)	1	0	0	0	44	0	0	0

	Takut (5)	1	1	1	0	1	33	2	1
	Jijik (6)	0	0	0	1	1	1	37	2
	Terkejut (7)	1	0	4	0	0	0	1	35

3.8 Evaluasi

Evaluasi dilakukan untuk mempresentasikan hasil pengujian dengan menghitung akurasi dengan menggunakan *confusion matrix* yang telah didapatkan dari data uji. Berikut perhitungan akurasi dari setiap kelas emosi.

$$Akurasi = \frac{TN + TP}{TN + FP + FN + TP} \; x \; 100\%$$

$$Akurasi\ Emosi\ Netral = \frac{263 + 13}{263 + 4 + 6 + 13} \; x \; 100\% = \frac{276}{285} \; x \; 100 = 96\%$$

$$Akurasi\ Emosi\ Tenang = \frac{246 + 33}{246 + 5 + 2 + 33} \; x \; 100\% = \frac{279}{286} \; x \; 100 = 97\%$$

$$Akurasi\ Emosi\ Senang = \frac{241 + 26}{241 + 3 + 6 + 26} \; x \; 100\% = \frac{267}{276} \; x \; 100 = 96\%$$

$$Akurasi\ Emosi\ Sedih = \frac{238 + 31}{238 + 3 + 4 + 31} \; x \; 100\% = \frac{269}{276} \; x \; 100 = 97\%$$

$$Akurasi\ Emosi\ Marah = \frac{227 + 44}{227 + 1 + 3 + 44} \; x \; 100\% = \frac{271}{275} \; x \; 100 = 98\%$$

$$Akurasi\ Emosi\ Takut = \frac{232 + 33}{232 + 7 + 4 + 33} \; x \; 100\% = \frac{265}{276} \; x \; 100 = 96\%$$

$$Akurasi\ Emosi\ Jijik = \frac{231 + 37}{231 + 5 + 3 + 37} \; x \; 100\% = \frac{268}{276} \; x \; 100 = 97\%$$

$$Akurasi\ Emosi\ Terkejut = \frac{229 + 35}{229 + 6 + 6 + 35} \; x \; 100\% = \frac{264}{276} \; x \; 100 = 95\%$$

Tabel 7 Parameter Evaluasi

Parameter	Jumlah
Data uji	20%
Learning rate	0,0001
Batch size	64
Epoch	100

Dapat disimpulkan dari pengujian dengan menggunakan data uji 20%, *learning rate* = 0,0001, *batch size* = 64 dan *epoch* = 100 menghasilkan nilai akurasi dari kelas emosi netral sebesar 96%, kelas emosi tenang sebesar 97%, kelas emosi senang sebesar 96%, kelas emosi sedih sebesar 97%, kelas emosi marah sebesar 98%, kelas emosi takut sebesar 96%, kelas emosi jijik sebesar 97% dan kelas emosi terkejut sebesar 95%. Sehingga akurasi rata-ratanya sebesar 96%.

4. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan penulis terkait pembuatan model klasifikasi emosi berdasarkan suara manusia, penulis memberi kesimpulan yakni (1) Metode MFCC dapat melakukan ekstraksi ciri dengan baik pada dataset yang digunakan di penelitian ini dengan menghasilkan ciri dari setiap suara dan jenis emosi yang

berbeda berupa nilai vektor yang memudahkan dalam proses klasifikasi. (2) Klasifikasi emosi berdasarkan suara manusia dengan menggunakan algoritma MLP *Backpropagation* memiliki hasil akurasi rata-rata sebesar 96%. (3) Parameter pelatihan model yang digunakan adalah fungsi aktifasi pada *hidden layer* = *ReLu*, fungsi aktivasi pada *output layer* = *Softmax*, *optimizer* = Adam, *Learning rate* = 0,0001, *batch size* = 64 dan *epoch* = 100.

5. Referensi

[1] S. Helmiyah, A. Fadlil, and A. Yudhana, "Pengenalan Pola Emosi Manusia Berdasarkan Ucapan Menggunakan Ekstraksi Fitur Mel-Frequency Cepstral Coefficients (MFCC)," *CogITo Smart J.*, vol. 4, no. 2, p. 372, 2019, doi: 10.31154/cogito.v4i2.129.372-381.

[2] A. Yani, "Analisa Kelayakan Kredit Menggunakan Artifcial Neural Network dan Backpropogation (Studi Kasus German Credit Data)," *J. Ilm. Komputasi*, vol. 18, no. 4, pp. 385–390, 2019, doi: 10.32409/jikstik.18.4.2672.

[3] R. Umar, I. Riadi, and A. Hanif, "Analisis Bentuk Pola Suara Menggunakan Ekstraksi Ciri Mel-Frequencey Cepstral Coefficients (MFCC)," *CogITo Smart J.*, vol. 4, no. 2, p. 294, 2019, doi: 10.31154/cogito.v4i2.130.294-304.

[4] N. Purwaningsih, "Penerapan multilayer perceptron untuk klasifikasi jenis kulit sapi tersamak," *J. TEKNOIF*, vol. 4, no. 1, pp. 1–7, 2016.