# Penerapan *Speech Recognition* Menggunakan Metode *Long Short-Term Memory* (LSTM) untuk Presentasi Dinamis

**Satriya Adhitama[1], Donny Avianto[2]**
Program Studi Informatika, Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta
DIY Yogyakarta, Indonesia
Email: [1]satadhitama@gmail.com, [2]donnyavianto@gmail.com

***Abstract. Speech Recognition Implementaion Using Long Short-Term Memory (LSTM) for Dynamic Presentation.*** *Presentasi merupakan salah satu metode untuk dapat mengkomunikasikan suatu ide maupun gagasan yang disajikan sedemikian rupa sehingga audiens dapat dengan mudah memahami apa yang disampaikan oleh pembicara. Pengemasan dan penyampaian materi presentasi menjadi hal yang sangat penting untuk dapat meningkatkan terjadinya komunikasi efektif antara pembicara dan audiens. Komunikasi efektif dapat ditingkatkan dengan menggunakan media presentasi interaktif seperti powerpoint. Pembicara sering kali menggunakan metode dynamic presentation yang umum disediakan di berbagai platform presentasi. Namun, pengoperasian dynamic presentation ini terkadang menjadi kendala bagi pembicara ketika ingin menuju ke suatu section yang diinginkan karena membutuhkan operator atau perangkat pendukung lainnya. Hal ini menjadi sebuah distraksi bagi presenter yang harus terus fokus kepada audiens dan materi serta penguasaan panggung daripada memikirkan teknis, sehingga dapat lebih menguasai jalannya presentasi. Salah satu teknologi yang dapat digunakan untuk membantu presentasi adalah dengan menggunakan speech recognition. Speech recognition dapat membantu presenter dalam memberikan perintah untuk mengoperasikan perangkat lunak penampil presentasi yang telah disusun secara dinamis. Secara otomatis, sistem akan mendeteksi suara untuk dapat menentukan konten mana yang dituju, sehingga presentation software akan memindahkan slide yang sesuai. Sistem pengenalan suara untuk presentasi dinamis ini akan menggunakan metode machine learning, yaitu Long Short-Term Memory (LSTM) yang termasuk dalam Artificial Neural Network (ANN) untuk dapat mengenali suara secara baik. Model LSTM yang dibuat mmapu menghasilkan akurasi yang bagus, yaitu sebesar 94.18% untuk training, 94.34% untuk validation, dan 0.9398% untuk testing.*

***Keywords:*** Speech Recognition, Long Short-Term Memory, Speech Command, Presentation Software, Presentation

***Abstrak.*** *Presentasi merupakan salah satu metode untuk dapat mengkomunikasikan suatu ide maupun gagasan yang disajikan sedemikian rupa sehingga audiens dapat dengan mudah memahami apa yang disampaikan oleh pembicara. Pengemasan dan penyampaian materi presentasi menjadi hal yang sangat penting untuk dapat meningkatkan terjadinya komunikasi efektif antara pembicara dan audiens. Komunikasi efektif dapat ditingkatkan dengan menggunakan media presentasi interaktif seperti powerpoint. Pembicara sering kali menggunakan metode dynamic presentation yang umum disediakan di berbagai platform presentasi. Namun, pengoperasian dynamic presentation ini terkadang menjadi kendala bagi pembicara ketika ingin menuju ke suatu section yang diinginkan karena membutuhkan operator atau perangkat pendukung lainnya. Hal ini menjadi sebuah distraksi bagi presenter yang harus terus fokus kepada audiens dan materi serta penguasaan panggung daripada memikirkan teknis, sehingga dapat lebih menguasai jalannya presentasi. Salah satu teknologi yang dapat digunakan untuk membantu presentasi adalah dengan menggunakan speech recognition. Speech recognition dapat membantu presenter dalam memberikan perintah untuk mengoperasikan perangkat lunak penampil presentasi yang telah disusun secara dinamis. Secara otomatis, sistem akan mendeteksi suara untuk dapat menentukan konten mana yang dituju, sehingga presentation software akan memindahkan slide yang sesuai. Sistem pengenalan suara untuk presentasi dinamis ini akan menggunakan metode machine learning, yaitu Long Short-Term Memory*

*(LSTM) yang termasuk dalam Artificial Neural Network (ANN) untuk dapat mengenali suara secara baik. Model LSTM yang dibuat mmapu menghasilkan akurasi yang bagus, yaitu sebesar 94.18% untuk training, 94.34% untuk validation, dan 0.9398% untuk testing.*

***Kata Kunci:*** Pengenalan Ucapan, Long Short-Term Memory, Perintah Ucapan, Perangkat Presentasi, Presentasi

## 1. Pendahuluan

Presentasi merupakan adalah aktivitas mengungkapkan pikiran, gagasan, ide, pendapat, argumen, dan yang lainnya menggunakan bahasa lisan [1]. Presentasi dibuat sedemikian rupa sehingga penyampaian dan penampilan materi menjadi lebih menarik sehingga muncul rasa keingintahuan audiens terhadap informasi yang akan disampaikan. Tak jarang juga presentasi dibuat dalam bentuk yang interaktif sehingga dapat tercapai komunikasi yang efektif antara pembicara dengan audiens. Software presentasi seperti Power Point, Google Slide, Canva menjadi alat pendukung untuk menyusun materi tampilan materi presentasi yang dapat dioperasikan melalui komputer. Selain itu, software presentasi juga digunakan dalam penyampaian materi melalui mode slide show dengan proyektor sebagai alat pendukungnya.

Namun, terkadang materi yang disusun secara dinamis pada software presentasi akan mengakibatkan pembicara kesusahan untuk berinteraksi. Pembicara harus selalu mengarahkan pointer ke bagian yang dinginkan atau membutuhkan asisten untuk dapat membantu melakukan interaksi dengan software presentasi. Pembicara memerlukan alat tambahan untuk dapat mengoperasikan software presentasi seperti keyboard, mouse, atau remote control. Hal ini menimbulkan distraksi bagi pembicara karena harus menangani hal teknis sekaligus menyampaikan materi secara bersamaan.

Distraksi penggunaan software presentasi ini dapat diatasi dengan adanya teknologi pengenalan suara otomatis (automatic speech recognition/ASR). ASR merupakan proses untuk mengubah sinyal suara menjadi rangkaian kata atau entitas linguistik lainnya dengan algoritma tertentu menggunakan komputer. Saat ini pengguna lebih memilih untuk mengunakan perangkat seperti komputer, ponsel pintar, atau perangkat terkoneksi lainnya melalui ucapan [2].

Implementasi pengenalan ucapan (speech recognition) dilakuakan menggunakan model machine learning sequential Long Short-Term Memory (LSTM). LSTM yang merupakan turunan dari RNN telah mencapai peningkatan performa untuk Automatic Speech Recognition [3]. LSTM dapat menangkap informasi time series yang cocok digunakan untuk pengenalan ucapan (speech recognition) [4]. LSTM akan dapat mengenali kata tertentu yang muncul pada sinyal suara (keyword spotting).

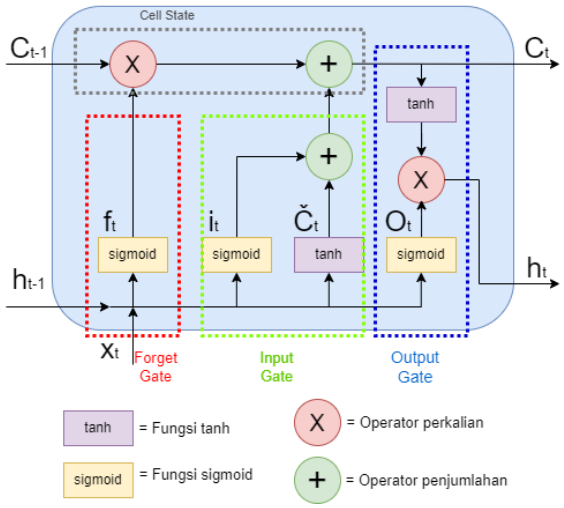
## 2. Tinjauan Pustaka

### 2.1 *Automatic Speech Recognition* (ASR)

ASR adalah metode untuk dapat mengubah ucapan kata menjadi teks dengan menggunakan komputer yang berbasis perangkat lunak. Sistem dirancang dengan teknik tertentu yang bertujuan untuk mengenali dan memproses suara manusia [5]. Prinsip dasar dari ASR yaitu seseorang berbicara mengeluarkan variasi tekanan suara pada *larynx* (pangkal tenggorokan), kemudian suara yang dihasilkan akan digitalisasi menggunakan *microphone* dan dikirimkan melalui sebuah perantara atau jaringan [2].

### 2.1. Long Short-Term Memory (LSTM)

Long short-term memory (LSTM) merupakan pengembangan metode dari recurrent neural network (RNN) yang dirancang dengan memory cell yang mampu merepresentasikan dependensi jangka panjang (long-term dependency) terhadap urutan waktu yang terjadi pada data [6]. Penanganan vanishing gradient problem pada RNN dapat diatasi menggunakan LSTM. LSTM secara khusus dirancang untuk menghindari masalah dependensi jangka panjang.




***Gambar 1 Arsitektur LSTM***

LSTM memiliki tiga gates atau gerbang yang masing-masing memiliki peran untuk melindungi dan mengontrol cell state. Cell state merupakan garis horizontal yang melewati bagian atas diagram sel LSTM yang memiliki kemampuan untuk menghapus atau menambahkan informasi baru yang masuk dalam waktu t dengan memanfaatkan struktur cermat yang disebut gerbang. Gates atau gerbang sendiri adalah sebuah cara yang digunakan oleh LSTM untuk melakukan seleksi terhadap informasi yang masuk ke dalam sel.

**2.2 Sparse Categorical Cross Entropy**

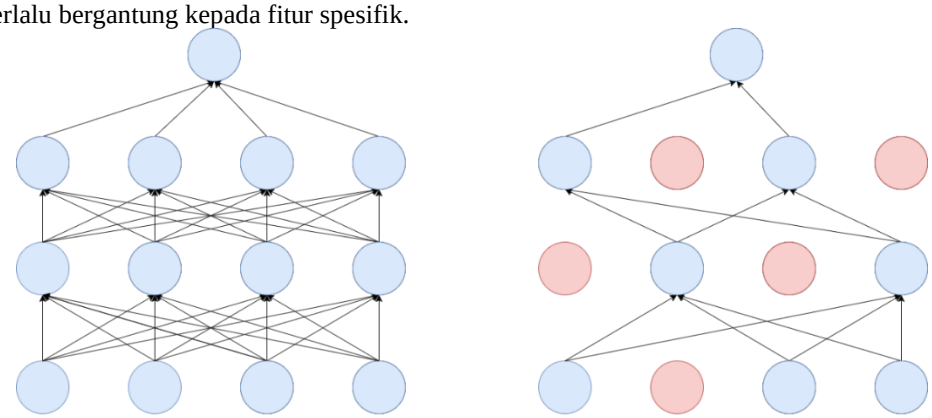
*Cost function* menjadi kunci utama untuk menyesuaikan bobot *neural network* sehingga dapat menghasilkan model *machine learning* yang baik. Secara spesifik, pada saat propagasi maju, *neural network* dijalankan untuk melatih data dan dihasilkan *output* klasifikasi yang mengindikasikan probabilitas kebenaran label [7]. *Sparse categorical crossentropy* adalah fungsi *loss* yang umum digunakan untuk kasus klasifikasi multi kelas ketika label target merupakan bilangan bulat (*integer*). Fungsi *loss* ini berasal dari *categorical crossentropy* namun label direpresentasikan sebagai matriks dengan format *sparse*. Matriks *sparse* merepresentasikan label sebagai sebuah nilai index tunggal daripada vektor *one-hot encoding*.

$$\text{SCCE} = -\log(softmax(y)[t] = -\log(\frac{exp(y)}{\sum_{j=1}^{n} exp(y)}[t] \quad (1)$$

**2.3 Dropout Regularization**

Regularisasi adalah metode sebuah proses untuk mengatasi permasalahan overfitting pada model yang dapat memengaruhi performa model. Konsep dari fully connected layer menimbulkan masalah eksponensial memori yang disebut overfitting sebagai akibat dari koneksi neuron yang terlalu banyak dan berlebih sehingga pembelajaran yang dilakukan terlalu berlebihan [8].

Regularisasi dropout merupakan teknik untuk secara acak menjadikan bagian input unit menjadi 0 pada setiap langkah training. Beberapa neuron akan dinonaktifkan untuk mengurangi interdependensi neuron dan terlalu bergantung kepada fitur spesifik.



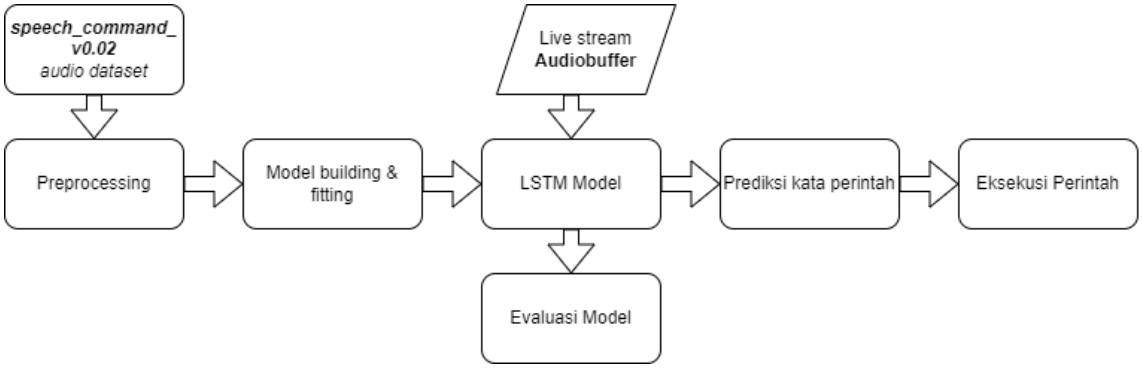
(a) Sebelum regulasisasi *dropout* (b) Sesudah regulasisasi *dropout*

***Gambar 2 Dropout Regularization***

## 3. Metodologi Penelitian

### 3.1 Tahapan Penelitian

Tahapan yang dilaksanakan pada penelitian ini dapat dilihat pada Gambar 3 berikut:

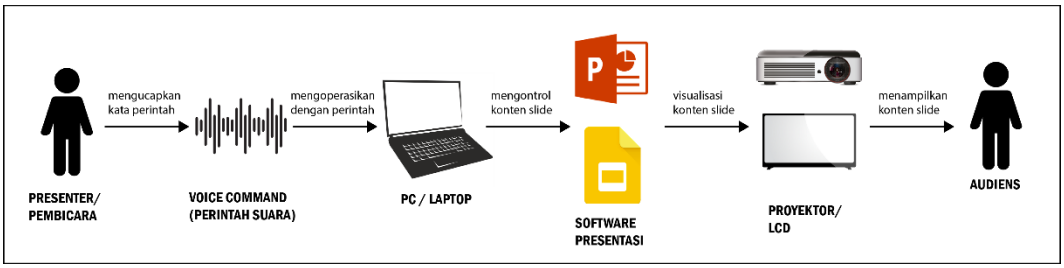



***Gambar 3 Tahapan Penelitian***

Gambar 3 menunjukkan tahapan-tahapan yang dilakukan dalam membuat sistem pengenalan suara untuk mengoperasikan kata perintah. Data *speech_command_v0.02* digunakan untuk melatih dan mengevaluasi model *machine learning* sehingga dapat mengenali input sinyal suara yang termuat kata perintah di dalamnya. Data ini dilakukan preprocessing untuk mengubah input file audio menjadi bentuk *spectrogram*. Setelah data siap, model dapat dibangun dengan menggunakan layer LSTM. Hasil dari *fitting* model kemudian dievaluasi untuk mengetahui performayang dihasilkan. Model yang telah disimpan akan digunakan pada prediksi input audiobuffer untuk menemukan kata perintah yang dapat dieksekusi untuk mengoperasikan *software* presentasi.

### 3.2 Rancangan Sistem

Rancangan arsitektur sistem yang dibuat pada penelitian ini terdapat pada Gambar 4 berikut:




***Gambar 4 Rancangan sistem pengenalan suara***

Gambar 4 menujukkan alur kerja sistem untuk pengoperasian *software* presentasi menggunakan perintah suara. Pengguna (*presenter/*pembicara) memberikan input berupa sinyal suara pada *microphone* yang terhubung dengan *device* presentasi seperti PC atau laptop. Sinyal suara ini kemudian akan dilakukan pemrosesan dan prediksi untuk mengetahui kata perintah yang diucapkan oleh pengguna. Audiobuffer yang dihasilkan dari input *michrophone* akan diolah per 16000 sample rate, yaitu durasi 1 detik untuk setiap kali prediksinya. Apabila kata perintah ditemukan, maka program akan melakukan kontrol pada *device* untuk mengubah konten tampilan pada *software* presentasi yang ditampilkan kepada audiens melalui proyektor.

### 3.3 Pengumpulan dan Pembagian Data

*speech_command_v0.02* merupakan dataset suara yang dikumpulkan oleh Pete Warden [9]. Data ucapan terdiri atas kata berbahasa inggris berdurasi 1 detik yang diambil melalui *microphone* telepon atau laptop pengguna. Para subjek diminta untuk merekam suara pada ruangan tertutup sendirian dengan pintu tertutup untuk menghindari percakapan lain yang bersifat privasi. Subjek ini terdiri dari sejumlah orang yang berbeda-beda sehingga model dapat melakukan pelatihan dengan baik.

Dataset *speech_command_v0.02* memuat jumlah sebanyak 34 ragam kata. Sistem yang dirancang hanya akan menggunakan 8 kata sebagai perintah untuk mengoperasikan *software* presentasi. Kata-kata yang dipilih akan digunakan dalam proses pembuatan model *machine learning*. Tabel 1 berikut merupakan rincian jumlah dan kegunaan kata perintah:

***Tabel 1 Kata kunci suara dan perintah***

| *No* | *Keywords* | *Audio files* | *Commands* |
|---|---|---|---|
| 1 | Down | 3,917 | Menuju ke slide berikutnya ketika tidak sedang slide show |
| 2 | Go | 3,880 | Memainkan media player |
| 3 | Left | 3,801 | Menuju ke slide sebelumnya ketika sedang slide show |
| 4 | Off | 3,754 | Mengubah state app menajadi OFF (tidak menerima perintah) hingga dihidupkan kembali |
| 5 | On | 3,845 | Mengubah state app menjadi ON (mampu menerima perintah) |
| 6 | Right | 3,778 | Menuju ke slide berikutnya ketika sedanga slide show |
| 7 | Stop | 3,872 | Mematikan streaming audio dan terminate program |
| 8 | Up | 3,723 | Menuju ke slide sebelumnya ketika tidak sedang slide show |

Data *speech_command_v0.02* dimuat dengan menggunakan *library* tensorflow pada python untuk mengubah kumpulan direktori file audio menjadi *tensorflow dataset*. Dataset ini kemudian akan membagi secara rata terhadap 8 kelas yang tersedia menjadi 3 bagian sesuai dengan kegunananya, yaitu *training*, *validation*, dan *testing*. Tabel 2 berikut menunjukkan jumlah dataset untuk setiap bagian:

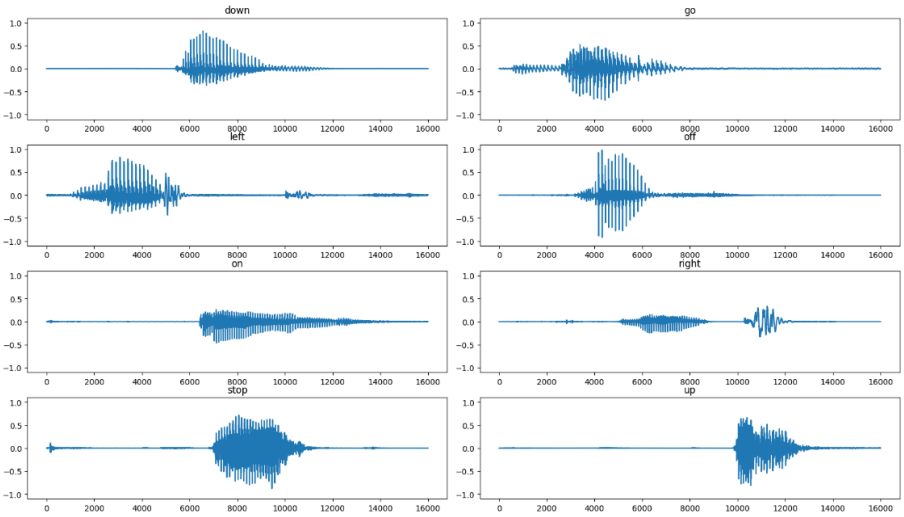
***Tabel 2 Pembagian dataset***

| *Partition* | *Audio Files* |
|---|---|
| *Training* | 24449 |
| *Validation* | 3056 |
| *Testing* | 3056 |

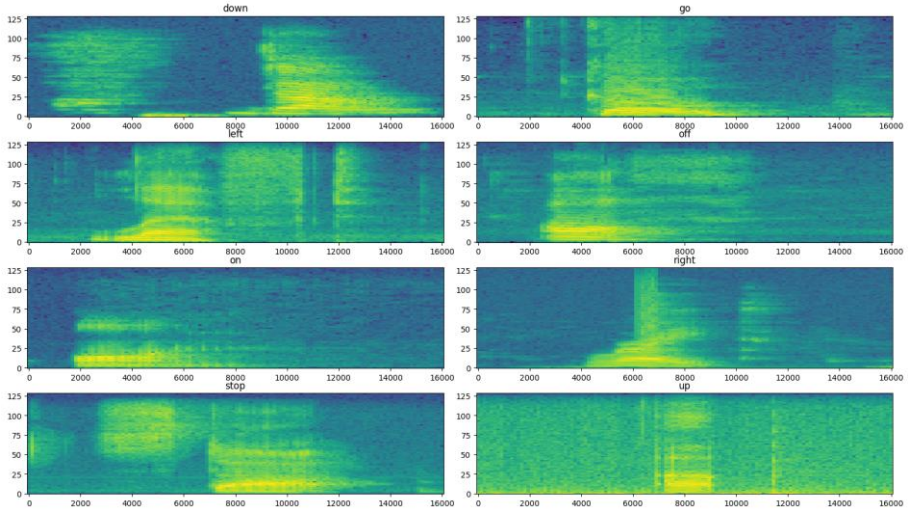
## 4. Hasil dan Diskusi

### 4.1 Hasil *Waveform* dan *Spectrogram*

Sinyal suara dari dapat direpresentasikan dalam bentuk *waveform* dan *spectrogram*. *Waveform* merepresentasikan sinyal suara berdasarkan domain waktu dengan menunjukkan perubahan amplitudo yang terjadi. S*pectrogram* menjadi representasi sinyal suara berdasarkan domain frekuensi. Kedua bentuk representasi sinyal suara tersebut dapat divisualisasikan untuk menujukkan perbedaan pola suara yang dihasilkan oleh masing-masing kata ucapan. Berikut pada Gambar 5 dan Gambar 6 di bawah menunjukkan hasil visualisasi *waveform* dan *spectrogram* untuk 8 kata perintah:




***Gambar 5 Visualisasi Waveform***




***Gambar 6 Visualisasi Spectrogram***

### 4.2 Pengujian Model LSTM

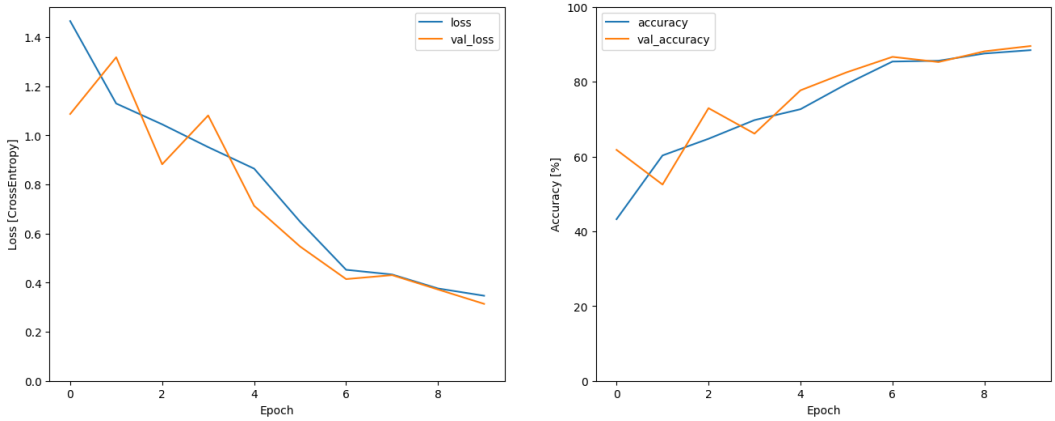
Sistem pengenalan suara (*speech recognition*) memerlukan model *machine learning* yang telah melalui proses *fitting* sehingga bobot pada setiap unit model dapat menghasilkan prediksi tepat mengenai kata perintah yang muncul. Penelitian ini menggunakan LSTM sebagai layer penyusun model *deep learning*. LSTM dinilai mampu untuk menghasilkan performa yang baik pada data *sequential* (menggunakan urutan waktu). Tabel 3 berikut menunjukkan susunan *layer* model *machine learning* yang digunakan untuk *fitting* dataset:

***Tabel 3 Model summary***

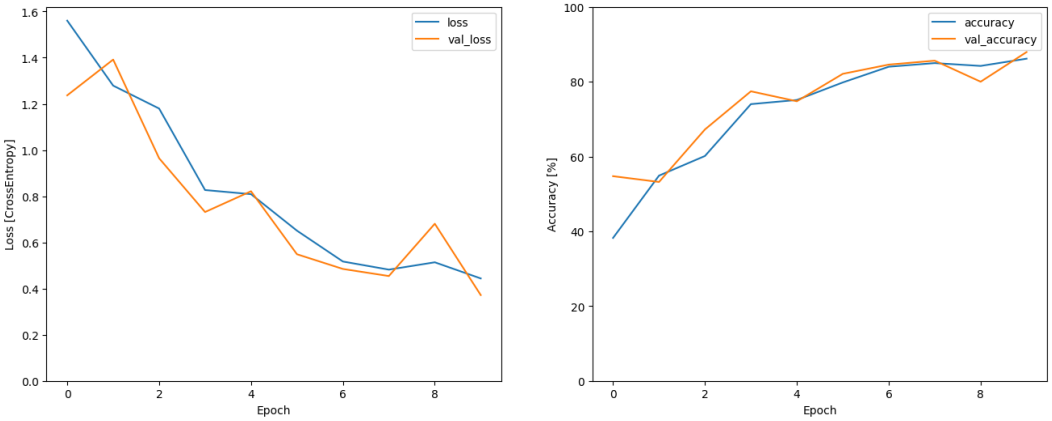
| Nama model | *Layers* | *Units* | *Output Shape* | *Param* | *Total Params* |
|---|---|---|---|---|---|
| LSTM_1 | LSTM | 128 | (,124,128) | 132096 | 182024 |
| | LSTM | 64 | (,64) | 49408 | |
| | Dense | 8 | (,8) | 520 | |
| LSTM_2 | LSTM | 128 | (,124,128) | 132096 | 182024 |
| | Dropout(0.5) | - | (,124,128) | 0 | |
| | LSTM | 64 | (,64) | 49408 | |
| | Dense | 8 | (,8) | 520 | |
| LSTM_3 | LSTM | 256 | (,124,256) | 395264 | 593416 |
| | LSTM | 128 | (,128) | 197120 | |
| | Dense | 8 | (,8) | 1032 | |
| LSTM_4 | LSTM | 256 | (,124,256) | 395264 | 593416 |
| | Dropout(0.5) | - | (,124,256) | 0 | |
| | LSTM | 128 | (,128) | 197120 | |
| | Dense | 8 | (,8) | 1032 | |

Tabel 3 di atas menunjukkan 4 arsitektur model *machine learning* dengan layer LSTM sebagai eksperimen guna menentukan hasil model yang terbaik. Jumlah unit ditentukan untuk variabel pengujian dengan dua layer pada masing-masing model memiliki jumlah neuron yang berbeda. LSTM_1 dan LSTM 2 menggunakan layer LSTM 128 dan 64, sedangkan LSTM_3 dan LSTM_4 menggunakan layer LSTM 256 dan 128. Selain itu, *dropout regularization* juga diterapkan pada model LSTM_3 dan LSTM_4.

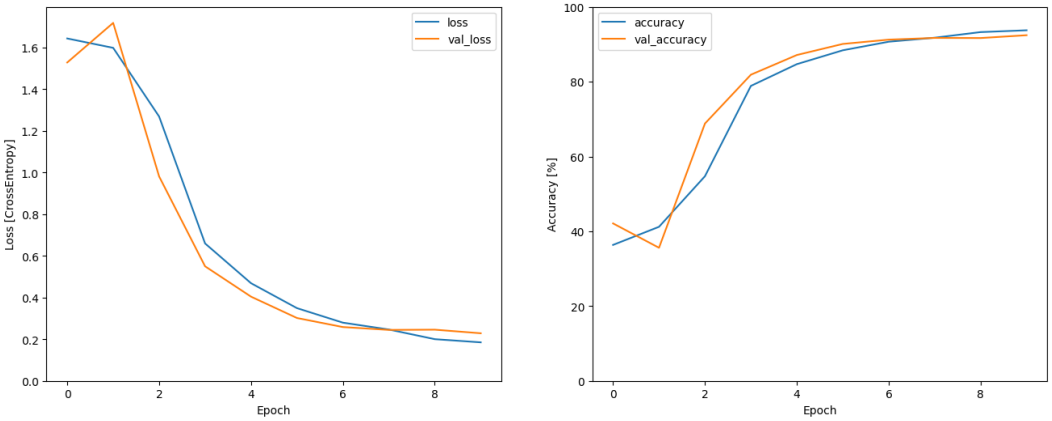
Keempat model di atas kemudian dilakukan proses *fitting* untuk mencari nilai bobot unit terbaik. *Fitting* dilakukan dalam 10 kali epoch dengan *optimizer* Adam yang memiliki *learning rate* sebesar 0,001. Metrik yang digunakan untuk mengukur evaluasi model adalah *sparse categorical crossentropy* (fungsi *loss*) dan akurasi. Gambar 7, Gambar 8, Gambar 9, dan Gambar 10 berikut menunjukkan riwayat *loss* dan *accuracy* pada setiap epoch selama proses *fitting* berlangsung:



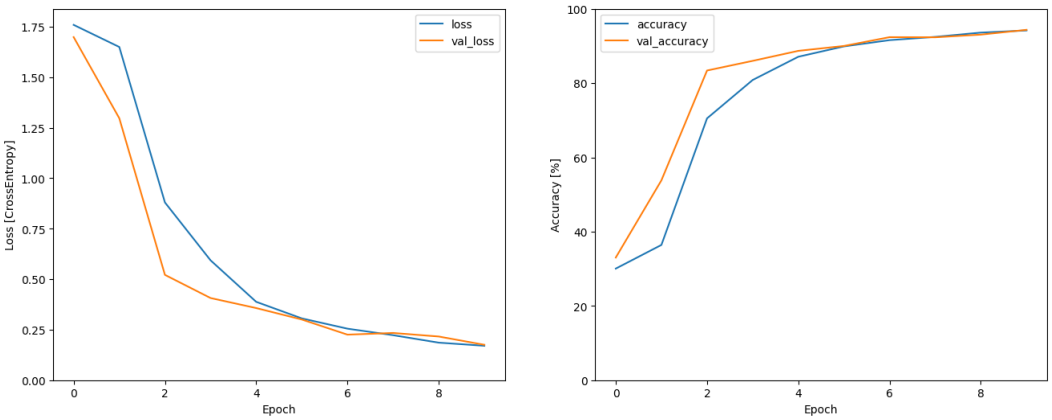

***Gambar 7 Riwayat Fitting LSTM_1***




***Gambar 8 Riwayat Fitting LSTM_2***




***Gambar 9 Riwayat Fitting LSTM_3***




***Gambar 10 Riwayat Fitting LSTM_3***

Berikut pada Tabel 4 menunjukkan hasil evaluasi setiap model yang telah melalui proses *fitting*:

***Tabel 4 Evaluasi Model LSTM***

| Nama Model | *Training* | | *Validaiton* | | *Testing* | | *Fitting time* |
|---|---|---|---|---|---|---|---|
| | *Loss* | *Accuracy* | *Loss* | *Accuracy* | *Loss* | *Accuracy* | |
| LSTM_1 | 0.3466 | 0.8844 | 0.3136 | 0.8954 | 0.3188 | 0.8896 | 80m 47.3s |
| LSTM_2 | 0.4438 | 0.8616 | 0.3717 | 0.8793 | 0.3889 | 0.8691 | 106m 37.7s |
| LSTM_3 | 0.1850 | 0.9375 | 0.2286 | 0.9243 | 0.2528 | 0.9167 | 172m 3.9s |
| LSTM_4 | 0.1704 | 0.9418 | 0.1752 | 0.9434 | 0.1883 | 0.9398 | 143m 42.2s |

Pada model LSTM_1 dan LSTM_2 dibuat model menggunakan 2 layer LSTM. Layer LSTM pertama berisi 128 unit dan layer LSTM kedua berisi 64 unit. LSTM_2 memiliki perbedaan yaitu terdapat tambahan layer *Dropout* sebagai regularisasi yang diletakkan di antara kedua layer LSTM. Hasil menunjukkan LSTM_1 lebih baik dari LSTM_2 dengan mengungguli total waktu *fitting* selama 80 menit 47,3 detik dibandingkan 106 menit 37,7 detik. Nilai loss yang dihasilkan LSTM_1 lebih kecil, yaitu 0,3466 untuk training, 0,3136 untuk validation, dan 0,3188 untuk testing. Akurasi yang dihasilkan LSTM menugguli sekitar 2% untuk setiap datanya, dengan rincian 88,44% untuk training, 89,54% untuk validation, dan 88,96% untuk testing. Penggunaan dropout layer pada LSTM_2 memengaruhi hasil modelnya, sehingga regularisasi tidak diperlukan untuk struktur model tersebut.

LSTM_3 dan LSTM_4 memiliki aturan pengujian yang sama seperti 2 model pendahulunya. Jumlah unit menjadi pembeda dengan layer pertama berjumlah 256 unit dan kedua 128 unit. Hasil eksperimen menunjukkan LSTM_4 dengan *dropout layer* menungguli LSTM_3. LSTM_4 memakan waktu *fitting* selama 143 menit 45,2 detik, unggul sekitar 30 menit dari LSTM_3. Performa model LSTM_4 sedikit mengungguli dengan nilai loss 0,1704 untuk training, 0,1752 untuk validation, dan 0,1883 untuk testing. Akurasi yang dihasilkan LSTM_4 menungguli model-model terdahulunya, yaitu memiliki akurasi 94,18% untuk training, 94,34% untuk validation, dan 93,98% untuk testing.

Hasil eksperimen 4 model dengan struktur yang berbeda menunjukkan bahwa LSTM_4 jauh mengungguli performa LSTM_1 dan LSTM_2, serta sedikit lebih baik daripada LSTM_3. Jumlah unit dan penggunaan layer dropout terbukti berpengaruh terhadap performa dan lama *fitting* model.

Regularisasi dropout hanya dapat berpengaruh baik untuk model dengan jumlah unit yang banyak untuk mencegah terjadinya overfitting sebagai akibat terlalu banyaknya unit yang terhubung. *Fully connected layer* menyebabkan pelatihan yang berlangsung menjadi lebih lama dan cenderung menyebabkan *overfit*. Namun, hasil eksperimen dengan dropout tidak selalu dapat ditambahkan untuk segala jenis struktur model. LSTM_4 dapat dengan baik menggunakan regularisasi dropout sehingga performa modelnya dapat jauh mengungguli model lain.

## 5. Kesimpulan dan Saran

Sistem pengenalan ucapan (*speech recognition*) menggunakan LSTM dapat dengan baik diterapkan untuk mengetahui kata perintah yang muncul pada input suara. Terdapat 4 model sebagai perbandingan dengan jumlah unit dan penggunaan *dropout regularization* sebagai pembedanya. Dari keempat model tersebut, model LSTM_4 menghasilkan performa terbaik, dengan rincian nilai loss 0.1704 dan *accuracy* 94.18% untuk training, nilai loss 0.1752 dan *accuracy* 94.34% untuk *validation*, serta nilai

loss 0.1883 dan *accuracy* 93.98% untuk *testing*. Penggunaan regularisasi *dropout* dapat berguna pada LSTM_4 untuk meningkatkan performa dan menghindari *overfitting*.

## 6. Ucapan Terima Kasih

Penulis mengucapkan kepada Universitas Tekonolgi Yogyakarta yang telah mendukung tersusunnya penelitian ini serta Pak Donny Avianto, S.T, M.T selaku dosen pembimbing yang telah memberikan arahan dengan baik dan detail.

## Referensi

[1] I. Lisnawati and Y. Ertinawati, "Literat Presentasi," *Jurnal Metaedukasi*, vol. 1, no. 1, pp. 1–12, 2019.

[2] J. L. K. E. Fendji, D. C. M. Tala, B. O. Yenke, and M. Atemkeng, "Automatic Speech Recognition Using Limited Vocabulary: A Survey," *Applied Artificial Intelligence*, vol. 36, no. 1, Dec. 2022, doi: 10.1080/08839514.2022.2095039.

[3] J. Oruh, S. Viriri, and A. Adegun, "Long Short-Term Memory Recurrent Neural Network for Automatic Speech Recognition," *IEEE Access*, vol. 10, pp. 30069–30079, 2022, doi: 10.1109/ACCESS.2022.3159339.

[4] L. Xiang, S. Lu, X. Wang, H. Liu, W. Pang, and H. Yu, "Implementation of LSTM Accelerator for Speech Keywords Recognition," in *2019 IEEE 4th International Conference on Integrated Circuits and Microsystems (ICICM)*, IEEE, Oct. 2019, pp. 195–198. doi: 10.1109/ICICM48536.2019.8977176.

[5] S. Sen, A. Dutta, and N. Dey, *Audio Processing and Speech Recognition*. Singapore: Springer Singapore, 2019. doi: 10.1007/978-981-13-6098-5.

[6] A. Sagheer and M. Kotb, "Time series forecasting of petroleum production using deep LSTM recurrent networks," *Neurocomputing*, vol. 323, pp. 203–213, Jan. 2019, doi: 10.1016/j.neucom.2018.09.082.

[7] Y. Ho and S. Wookey, "The Real-World-Weight Cross-Entropy Loss Function: Modeling the Costs of Mislabeling," *IEEE Access*, vol. 8, pp. 4806–4813, 2020, doi: 10.1109/ACCESS.2019.2962617.

[8] B. Jabir and N. Falih, "Dropout, a basic and effective regularization method for a deep learning model: a case study," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 2, p. 1009, Nov. 2021, doi: 10.11591/ijeecs.v24.i2.pp1009-1016.

[9] P. Warden, "Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition," Mountain View, California, 2018.