

Klasifikasi Suara Paru Normal Dan Abnormal Menggunakan *Deep Neural Network* dan *Support Vector Machine*

Adnan Hassal Falah¹, Jondri, M.Si²

^{1,2}Fakultas Informatika, Universitas Telkom, Bandung

¹adnanhassalfalah@students.telkomuniversity.ac.id, ²jondri@telkomuniversity.ac.id

Abstrak

Penyakit pernapasan masih menjadi pembunuh tertinggi setelah stroke dan penyakit jantung, hal ini disebabkan teknik diagnosis yang masih terbatas pada auskultasi. Melalui auskultasi ditemukan bahwa paru-paru memiliki suara yang berbeda-beda, sesuai dengan kondisi kesehatan seseorang. Oleh karena itu, dimulailah penelitian untuk mengklasifikasikan jenis suara paru. Berbagai metode telah digunakan untuk penelitian di bidang tersebut, tidak terkecuali *deep learning*. Diantara sekian banyak metode yang berkembang di bawah label *deep learning*, ternyata *Autoencoder* hanya digunakan sekali dalam sejarah penelitian klasifikasi data suara paru. *Autoencoder* (AE) merupakan salah satu arsitektur *Deep Neural Network* yang mampu merekonstruksi suatu data. Kemampuan ini dapat dimanfaatkan sebagai metode ekstraksi ciri sehingga *classifier* dapat mengklasifikasikan suatu data dengan lebih baik. Oleh karena itu, *autoencoder* diajukan sebagai metode ekstraksi ciri pada tugas akhir ini. Kemampuan *Autoencoder* sebagai metode ekstraksi ciri akan diuji oleh *Support Vector Machine* (SVM). Vektor ciri dipersiapkan dengan *continuous wavelet transform* (CWT) dan tiga pemrosesan lebih lanjut, lalu diinputkan ke dalam *Autoencoder*. Dari dua macam pengujian, sistem klasifikasi AE-SVM berhasil mencapai akurasi sebesar 82,38%.

Kata kunci : *suara paru, continuous wavelet transform, wavelet energy, deep neural network, autoencoder, support vector machine*

Abstract

Respiratory disease is still the highest killer after strokes and heart disease, this is due to diagnostic techniques that are still limited to auscultation. Through auscultation it is found that the lungs have different sounds, according to one's health condition. Therefore, research began to classify the types of lung sounds. Various methods have been used for research in this field, including deep learning. Among the many methods that develop under the label of deep learning, it turns out that *Autoencoder* is only used once in the history of research in classification of pulmonary sound data. *Autoencoder* (AE) is one of the *Deep Neural Network* architectures that is capable of reconstructing data. This capability can be used as a feature extraction method so that *classifier* can classify data better. Therefore, *autoencoder* is proposed as a feature extraction method in this final project. The ability of *Autoencoder* as a feature extraction method will be tested by *Support Vector Machine* (SVM). The feature vector is prepared with *continuous wavelet transform* (CWT) and three further processing, then it inputted into *Autoencoder*. From two experiments on *Autoencoder's* parameter, the AE-SVM classification system achieved an accuracy of 82.38%.

Keywords: *lungs sound, continuous wavelet transform, wavelet energy, deep neural network, autoencoder, support vector machine*

1. Pendahuluan

Latar Belakang

Penyakit pernapasan merupakan salah satu penyakit berbahaya selain penyakit jantung dan stroke[8]. Penyakit ini tidak membunuh penderitanya dengan cepat seperti serangan jantung. Diawali dengan gejala-gejala yang sering dianggap remeh seperti batuk atau napas yang tidak teratur, penderita pada umumnya tidak akan memeriksa kondisi paru-paru. Pemeriksaan lengkap baru dilakukan ketika gejala sudah berkembang dari batuk ringan menjadi batuk yang berkepanjangan atau napas tidak teratur yang menjadi sesak napas. Sedikitnya teknologi *monitoring* kesehatan pernapasan juga menjadi faktor terhambatnya penanganan dini terhadap kemunculan gejala penyakit. Hingga saat ini, teknik diagnosis yang dilakukan dokter untuk pemeriksaan paru-paru umumnya menggunakan auskultasi. Auskultasi merupakan teknik diagnosa yang dilakukan dengan mendengarkan suara organ tubuh pasien dengan menggunakan stetoskop. Namun ketepatan hasil diagnosis dari auskultasi bergantung dari kemampuan pendengaran setiap

dokter[20]. Mangione et al [11] dalam laporannya tentang pelatihan teknik auskultasi kepada masyarakat umum, menuliskan bahwa para dokter yang memiliki spesialisasi di bidang pernapasan pun hanya mampu menebak satu dari sepuluh jenis suara secara tepat, yaitu *wheeze*, jenis suara paru abnormal yang sering didengar dari penderita asma.

Dengan perkembangan teknologi saat ini, sudah terdapat stetoskop yang bisa merekam suara organ tubuh pada saat melakukan auskultasi. Setiap rekaman suara paru-paru ini dapat diarsipkan untuk menjadi bahan belajar dokter atau sebagai bahan penelitian di bidang teknologi digital. Dengan adanya data rekaman suara paru-paru, dapat dirancang sebuah sistem yang dapat mendeteksi gejala penyakit pernapasan berdasarkan jenis suara paru. Sistem tersebut nantinya dapat dikembangkan menjadi alat bantu dokter dalam melakukan diagnosis, atau aplikasi personal untuk *monitoring* kondisi kesehatan paru-paru.

Rancangan sistem tersebut dapat dimulai dengan klasifikasi data suara paru, sistem diajarkan untuk dapat membedakan berbagai jenis suara paru. Klasifikasi data suara paru secara digital telah dimulai sejak tahun 1988[5]. Data pada suara paru memiliki dimensi yang sangat tinggi. *Features* unik pada setiap jenis suara paru bisa saja tersebar secara acak, sehingga dibutuhkan metode untuk mengekstrak *features* tersebut. Telah banyak metode yang digunakan untuk ekstraksi *features* pada data suara paru, seperti *Mel-Frequency Cepstral Coefficient*[9, 12], *Auto Regressive*[2, 7], dan *wavelet coefficients*[1]. Dari berbagai macam metode tersebut, diajukan sistem klasifikasi yang menggunakan *Autoencoder* sebagai metode ekstraksi ciri. Pada kegiatan ini, kemampuan *Autoencoder* dalam mereduksi dimensi data akan diuji oleh *Support Vector Machine*. Secara singkat, *Autoencoder* merupakan salah satu arsitektur *Deep Neural Network* yang didesain untuk merekonstruksi ulang suatu data, memiliki struktur *hidden layer* yang terdiri dari *encoder*, *code* dan *decoder*.

Perumusan Masalah

Berikut rumusan masalah dalam pengerjaan tugas akhir:

1. Seberapa akurat *Autoencoder* dan *Support Vector Machine* sebagai sistem klasifikasi suara paru?
2. Berdasarkan nilai akurasi data testing, apakah *Autoencoder* dan *Support Vector Machine* merupakan sistem yang lebih baik daripada sistem klasifikasi suara paru dengan *Support Vector Machine* saja?

Berikut hal-hal yang dibatasi pada tugas akhir ini:

1. Pengamatan terhadap performa *Autoencoder* menjadi fokus utama dalam kegiatan ini, sehingga setting parameter dan skenario pengujian hanya didesain untuk *Autoencoder*.
2. Penulisan *source code* untuk membuat *Autoencoder* dan SVM tidak sepenuhnya dibuat manual oleh penulis, namun dibantu oleh penggunaan dua *library*, yaitu Keras dan Scikit-Learn[4, 15].
3. Tugas akhir ini menggunakan data dari R.A.L.E Lung Sounds 3.2, dengan format file **wav*.

Tujuan

Berikut tujuan yang ingin dicapai dalam pengerjaan tugas akhir:

1. Akurasi *Autoencoder* dan *Support Vector Machine* sebagai sistem klasifikasi suara paru.
2. Menentukan *Autoencoder* dan *Support Vector Machine* merupakan sistem yang lebih baik daripada sistem klasifikasi suara paru dengan *Support Vector Machine* atau tidak, berdasarkan nilai akurasi pada data *testing*

Organisasi Tulisan

Jurnal tugas akhir ini disusun sebagai berikut, pada Bagian dua, akan dijelaskan lebih lanjut tentang *autoencoder*, jenis suara paru, dan *support vector machine*, dan metrik evaluasi yang akan digunakan. Bagian tiga membahas tentang rancangan sistem dan skenario percobaan yang akan diujikan. Hasil dan analisis dibahas pada bagian empat. Terakhir, kesimpulan ditulis pada bagian lima.

2. Studi Terkait

Jenis Suara Paru [3]

Secara garis besar suara paru-paru dibagi menjadi dua jenis, yaitu normal dan tidak normal. Suara paru normal terdengar seperti angin yang melewati pipa. Adapun jenis suara paru-paru tidak normal yang diantara lain *wheeze* dan *crackle*. *Wheeze* merupakan jenis suara paru yang memiliki nada tinggi dengan frekuensi 400 Hz ke atas, suara ini ditimbulkan oleh penyempitan saluran pernapasan, sehingga suara ini terdengar seperti siulan. *Crackle* merupakan jenis suara paru yang berada pada rentang 350 Hz hingga 650 Hz, suara ini kadang terdengar seperti suara gelembung air yang pecah atau suara kembang api *sparkles*, yang disebabkan oleh pembukaan tiba-tiba atau bola gas pada saluran pernapasan.

Ekstraksi Ciri dengan *Continuous Wavelet Transform*

Karakteristik sinyal suara paru dapat diekstrak melalui berbagai metode pemrosesan sinyal, salah satunya dengan menggunakan transformasi Wavelet. Jenis transformasi Wavelet yang paling sering digunakan dalam penelitian suara paru adalah transformasi Wavelet diskrit atau *Wavelet Packet*. Namun pada pengerjaan tugas akhir ini, digunakan jenis transformasi Wavelet kontinu (CWT), yang ditulis sebagai berikut

$$X(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} s(t) \psi^* \left(\frac{t-b}{a} \right) dt \quad (2.1)$$

dengan $s(t)$ sebagai fungsi sinyal di ranah waktu, ψ^* adalah fungsi analisis gelombang (*Mother Wavelet*), b sebagai dilatasi waktu, dan a sebagai *scale*.

Koefisien Wavelet yang terdapat di tiap *scale* kemudian diproses dengan operasi $|\cdot|$, *smoothing*, dan perhitungan total energi [17, 14]. Pada *smoothing*, digunakan *Gaussian window* berikut

$$w(n) = e^{-n^2/2\sigma^2} \quad (2.2)$$

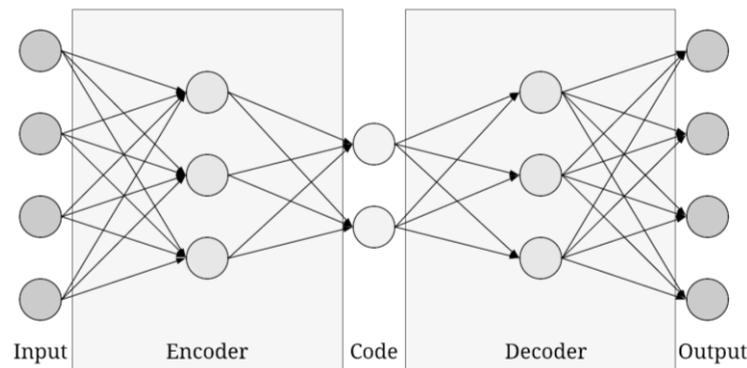
sebagai filter sinyal suara paru. Nilai n sebagai panjang filter akan disesuaikan dengan nilai *scale*. Sementara itu, nilai σ adalah $(n-1)/2$ [21]. Setelah proses *smoothing*, dilakukan perhitungan total energi pada tiap *scale* sebagai berikut

$$E_i = \sqrt{\frac{1}{N_i} \sum_{j=1}^{N_i} (c_j)^2} \quad (2.3)$$

Total energi inilah yang digunakan sebagai *features*. koefisien Wavelet (c) yang terdapat pada tiap *scale* dijumlahkan lalu dibagi dengan panjang *scale* (N_i). Adapun $\frac{1}{N_i}$ dan akar pada persamaan 2.3 disisipkan untuk menghasilkan nilai E_i yang tidak terlalu besar. Jenis *mother wavelet* yang akan digunakan pada tugas akhir ini adalah Morlet [17].

Autoencoder

Autoencoder merupakan salah satu arsitektur *Deep Neural Network*. Autoencoder termasuk ke dalam *unsupervised learning* karena *output layer* tidak berisi informasi kelas, melainkan data itu sendiri. Autoencoder dapat melakukan *encoding* dan *decoding* terhadap suatu data [10], sehingga dapat digunakan untuk mereduksi data dengan jumlah *features* yang besar. Struktur Autoencoder terdiri dari *encoder*, *code*, dan *decoder*. Masing-masing bagian *autoencoder* dapat dilihat pada gambar 2.1. Code layer, yang berada di tengah, dapat dianggap sebagai representasi *features*, bagian inilah yang menjadi hasil reduksi dimensi dari suatu data [13]. Namun, *autoencoder* juga sebuah *neural network*, sehingga jenis representasi data yang tersimpan didalam *code layer* tidak diketahui. Sebuah model *autoencoder* juga tidak bisa mengambil representasi *features* dari sembarang data, kecuali data tersebut pernah dipelajari sebelumnya [10].



Gambar 2.1 Ilustrasi training Autoencoder dalam mereduksi dimensi data dari empat dimensi ke dua dimensi. Data direduksi menjadi tiga dimensi dan direduksi lagi menjadi dua dimensi oleh *encoder*, hasil *encoding* terakhir kemudian disimpan pada *code*. Representasi data di dalam *code* digunakan untuk rekonstruksi data oleh *decoder*. Proses tersebut terus diulang, hingga didapatkan nilai error antara data rekonstruksi dan data asli yang paling rendah atau mencapai jumlah epoch yang ditentukan.

Support Vector Machine (SVM)

SVM merupakan sebuah model klasifikasi *supervised* biner dan non-probabilistik[1]. SVM menentukan wilayah setiap kelas data dengan menggunakan *hyperplane*

$$w^T x_i + b = 0 \quad (2.4)$$

yang terbentuk dari nilai bobot (w) dan bias (b)[18]. *Hyperplane* bisa berupa garis linear atau non-linear, tergantung dari kondisi data (x). *Hyperplane* yang berupa garis non-linear bisa didapatkan melalui penggunaan metode kernel. Beberapa jenis kernel yang sering digunakan diantaranya *radial basis function* dan polinomial. *Hyperplane* didapat dari margin, jarak terdekat titik-titik data di setiap kelas dari *hyperplane*. Setiap titik data yang menjadi margin disebut *support vector*. *Support vector* inilah yang mempengaruhi bentuk *hyperplane*, sehingga SVM dapat melakukan klasifikasi data dengan persamaan berikut

$$y = \text{sign}(w^T x_i + b) \quad (2.5)$$

SVM bersifat biner sehingga hanya bisa digunakan untuk kasus *binary classification*, dalam kasus *multiclass classification*, masalah ini dapat diatasi dengan pembuatan model SVM sebanyak jumlah kelas. Cara ini disebut *One-Against-All* (OAA). Dengan OAA, satu record data dapat diklasifikasi jika hanya dikenali oleh satu model SVM saja.

Metrik Evaluasi [6]

Performa *Autoencoder* dan SVM dalam mengklasifikasi data suara paru dievaluasi dengan persentase akurasi data *testing*. Akurasi adalah metrik yang mendeskripsikan seberapa banyak data yang diprediksi dengan tepat oleh sistem klasifikasi. Berikut perhitungan untuk mendapatkan nilai akurasi

$$\text{Accuracy} = \frac{T}{N} \times 100\% \quad (2.6)$$

dengan T sebagai jumlah data testing yang diprediksi dengan tepat, dari N sebagai total data testing.

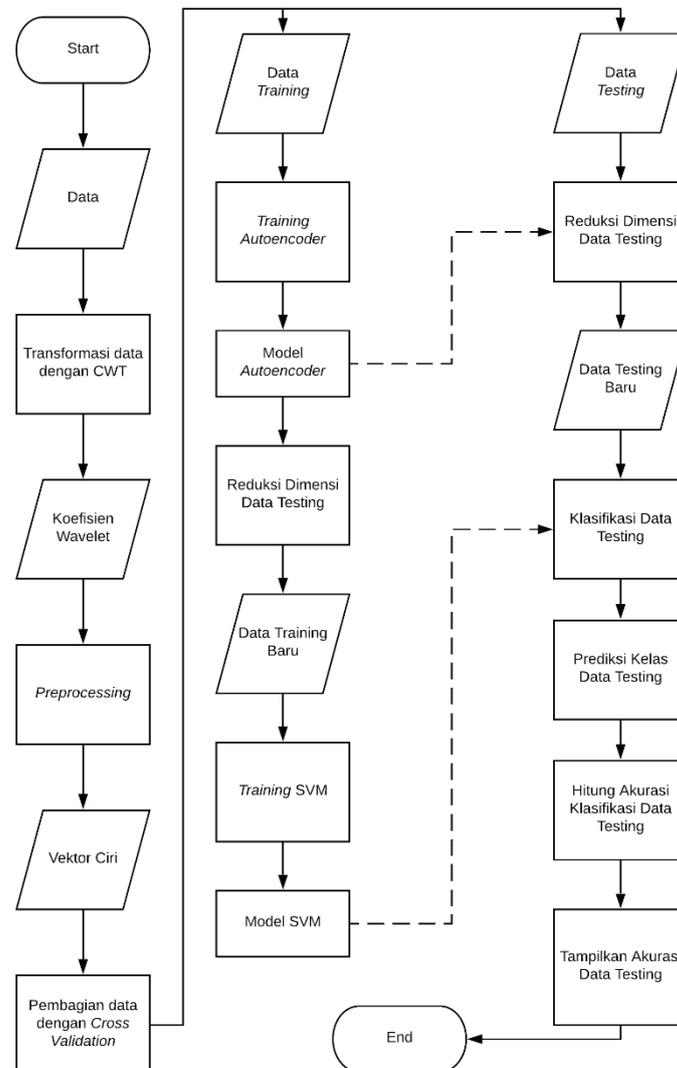
3. Desain Sistem

Deskripsi Sistem

Alur sistem klasifikasi data pengerjaan tugas akhir secara umum terdiri dari *input* data, *preprocessing*, reduksi dimensi dengan *Autoencoder*, membagi data menjadi dua bagian (*training* dan *testing*) disertai dengan *cross validation*, kemudian *learning* dan *testing* model SVM. Alur rancangan sistem secara umum dapat dilihat pada gambar 3.1, pemaparan lebih lanjut dilanjutkan pada sub-bagian selanjutnya.

Dataset dan Pembagian Data

Data suara paru-paru diambil dari R.A.L.E Lung Sounds versi 3.2[16] dengan *frequency sampling* 48kHz dan rata-rata durasi 1,8 detik. Total data yang digunakan sebanyak 68 suara, terdiri dari 21 suara normal, 12 *wheeze*, dan 35 *crackle*. Setiap data berisi satu siklus inspirasi (mengambil udara) dan ekspirasi (mengeluarkan udara). Data kemudian dibagi menjadi dua bagian yaitu training 70% dan testing 30%, pembagian data juga disertai dengan cross validation sebanyak 10 buah. Cross validation yang digunakan adalah *Shuffle Split*, gabungan dari *K-Fold* dan *Random Permutation*, yang tersedia di salah satu library Python, yaitu *Scikit-Learn*[Pedregosa]. Contoh data dari tiap kelas dapat dilihat pada gambar 3.2.



Gambar 3.1 Flowchart Deskripsi Sistem

Preprocessing

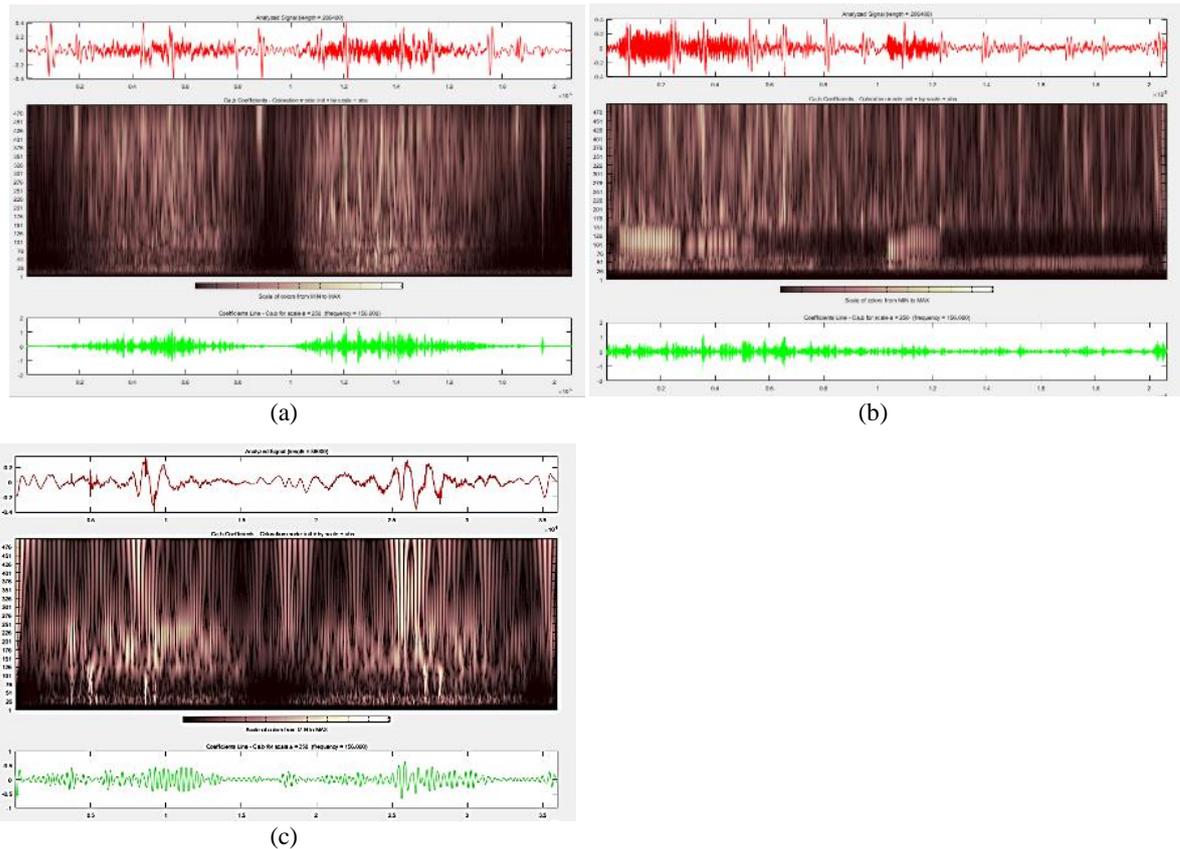
Setelah data masuk ke dalam sistem, data pada setiap kelas diamati terlebih dahulu dengan CWT. CWT ini diatur dengan *range scale* 1 - 500 dan *mother wavelet* Morlet. Berdasarkan pengamatan dari gambar 3.2, dipilihlah *range scale* 20 - 200. Setelah pengamatan, dilakukan tahap *preprocessing* seperti yang diilustrasikan gambar 3.3. *Preprocessing* diakhiri dengan pembagian data yang disertai *cross validation*, sebagaimana telah dijelaskan pada sub-bagian kedua.

Training dan Testing

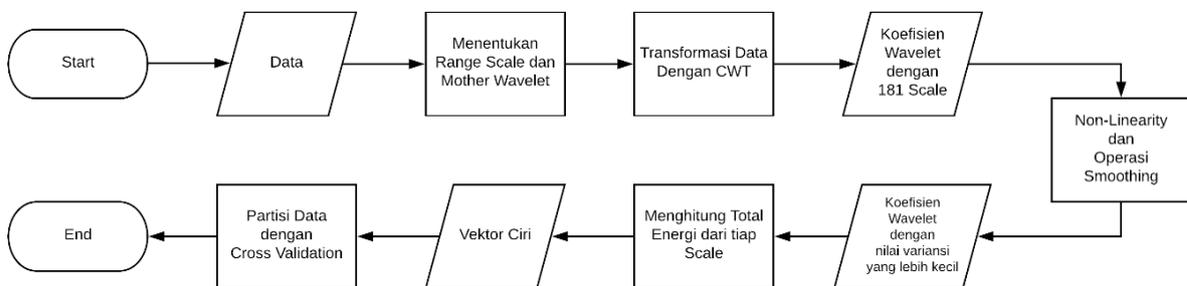
Sesuai gambar 3.1, Autoencoder dan SVM dilatih secara berurutan. Pembuatan *autoencoder* terdiri dari input layer, tiga *encoder*, satu *code*, tiga *decoder*, dan output layer. Jumlah neuron pada masing-masing layer, diilustrasikan pada gambar 3.6. Adapun parameter autoencoder yang akan diuji coba pada

tugas akhir ini adalah jumlah *epoch* dan jumlah neuron pada *code layer*. Kandidat jumlah epoch yang diuji coba adalah 50, 100, 300, 500, 700, 1000, 1300, 1500, 1700, dan 2000. Sedangkan kandidat jumlah neuron yang akan diujikan pada code layer adalah 5, 10, 15, dan 20. Langkah-langkah pada tahap *training* telah diilustrasikan pada gambar 3.4.

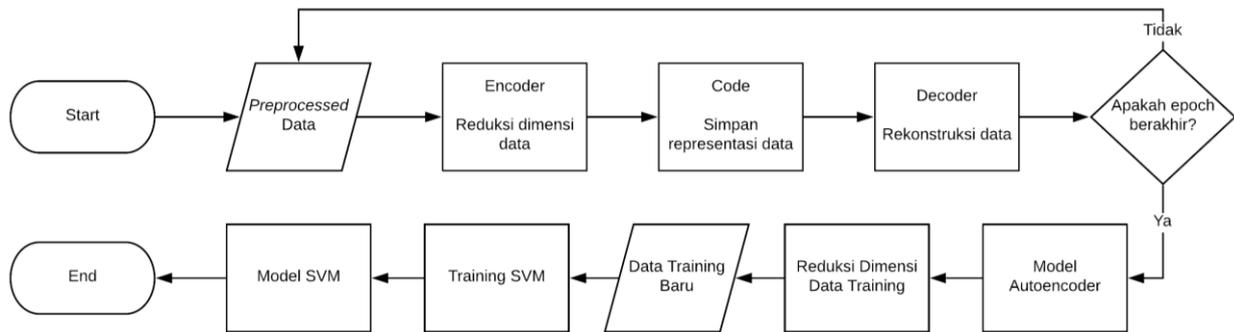
Setelah *training* selesai, setengah bagian autoencoder (input, *encoder*, dan *code*) disimpan untuk diujikan pada data *testing*. Model *autoencoder* yang telah disimpan kemudian digunakan untuk mereduksi data *training*. Dengan data *training* yang telah diproses oleh *autoencoder*, dilakukan proses *training* SVM. SVM yang dibangun menggunakan kernel *radial basis function*. Sementara itu, langkah *testing* meliputi input data, pengujian model *autoencoder* dan model SVM. Hasil prediksi kemudian dievaluasi dengan menggunakan nilai akurasi, *F1 score*, dan *confusion matrix*. Untuk diagram alur pada tahap *testing*, bisa dilihat pada gambar 3.5.



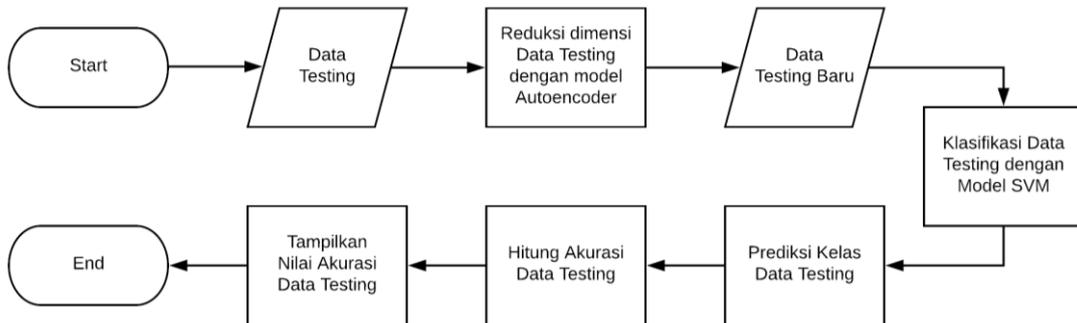
Gambar 3.2 Contoh sinyal suara dari masing-masing kelas. (a) suara paru normal. (b) suara paru wheeze. (c) suara paru crackle. Tiga gambar ini diambil saat dilakukan pengamatan data untuk penentuan *range scale* yang akan digunakan pada tahap *preprocessing*.



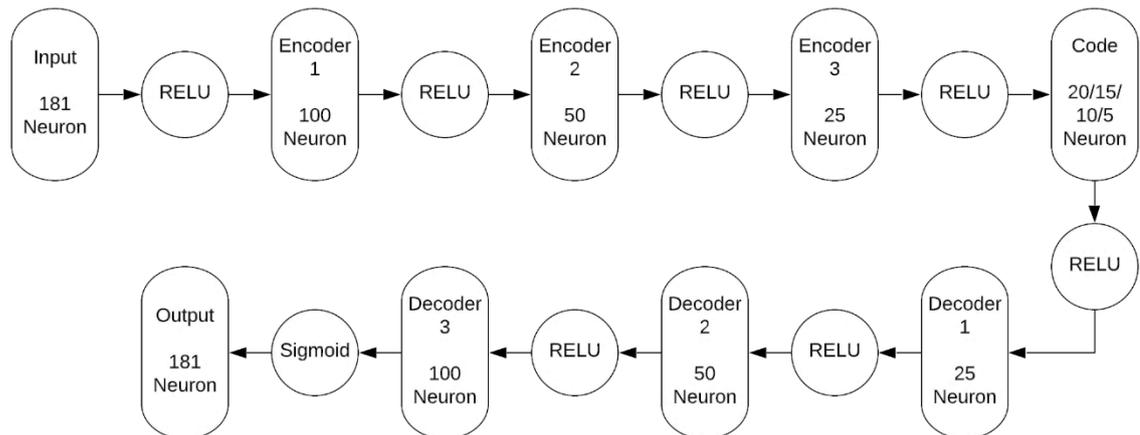
Gambar 3.3 Flowchart tahap *preprocessing*



Gambar 3.4 Flowchart training autoencoder dan SVM.



Gambar 3.5 Flowchart testing model autoencoder dan SVM.



Gambar 3.6 Desain autoencoder beserta keterangan jumlah neuron dan fungsi aktivasi pada tiap layer.

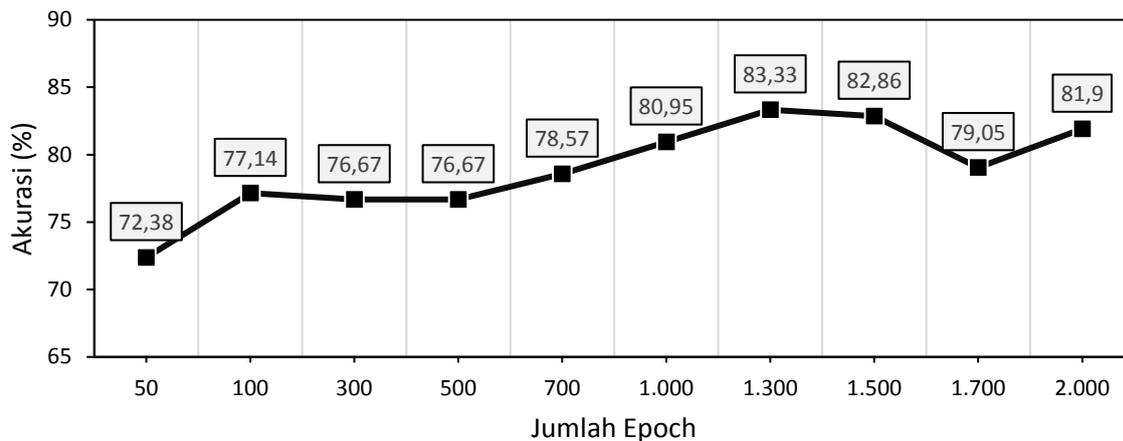
4. Hasil Pengujian dan Analisis

Pertama-tama, dilakukan percobaan jumlah epoch. Percobaan ini dilakukan dengan autoencoder yang mereduksi dimensi data hingga 5 *features*. Berdasarkan tampilan grafik pada gambar 4.1, jumlah epoch 1300 memiliki nilai *F1 score* terbesar. Dengan menggunakan 1300 *epoch*, dilakukan percobaan kedua, yaitu jumlah neuron pada *code layer*. Sebagai pembandingan, dibangun juga sebuah sistem klasifikasi yang sama, hanya saja tidak menggunakan *autoencoder*. Hasil percobaan kedua ditampilkan pada tabel 4.1, sistem klasifikasi yang tidak menggunakan *autoencoder* memiliki akurasi sebesar 75,71%, sedangkan sistem klasifikasi yang menggunakan *autoencoder* memiliki akurasi tertinggi 82,38%.

5. Kesimpulan

Seluruh hasil pengujian menunjukkan bahwa *autoencoder* berhasil mempelajari representasi *features* data *testing* dengan baik dan memiliki akurasi tertinggi sebesar 82,38%. Adapun saran yang dapat diberikan pada tugas akhir ini, sebaiknya sistem diujikan kembali pada dataset yang berbeda dan dibandingkan dengan paduan antara SVM dan metode reduksi dimensi yang sudah ada. Diharapkan pada kegiatan selanjutnya, skema pengujian lain yang melibatkan regularisasi L2, jumlah layer yang menjadi *encoder*

atau *decoder*, jumlah neuron pada tiap *layer*, serta penentuan fungsi aktivasi pada tiap *layer*, dapat dilaksanakan.



Gambar 4.1 Grafik akurasi AE-SVM pada percobaan jumlah epoch.

Tabel 4.1 Perbandingan akurasi antara SVM dan AE-SVM.

Metode	Jumlah Feature	Akurasi
SVM	181	75,71%
AE-SVM	5	81,90%
	10	80%
	15	81,90%
	20	82,38%

Daftar Pustaka

- [1] Abbasi, S., Derakhshanfar, R., Abbasi, A., & Sarbaz, Y. (2013). *Classification of normal and abnormal lung sounds using neural network and support vector machines*. In Electrical Engineering (ICEE), 2013 21st Iranian Conference on (pp. 1-4). IEEE.
- [2] Alsmadi, S., & Kahya, Y. P. (2008). *Design of a DSP-based instrument for real-time classification of pulmonary sounds*. Computers in biology and medicine, 38(1), 53-61.
- [3] Bohadana, A., Izbicki, G., & Kraman, S. S. (2014). *Fundamentals of lung auscultation*. New England Journal of Medicine, 370(8), 744-751.
- [4] Chollet, F. (2018). *Keras: The python deep learning library*. Astrophysics Source Code Library.
- [5] El-Dhaher, A. H. G., Mustafa, K. Y., & Abu-Irhayem, F. D. (1988). *Microcomputer-based system to measure, record and process flow-volume curves, respiratory questionnaire data and environmental exposure*. Computer methods and programs in biomedicine, 26(1), 63-70.
- [6] Géron, A. (2017). *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. " O'Reilly Media, Inc."
- [7] Kahya, Y. P., Yeginer, M., & Bilgic, B. (2006, August). *Classifying respiratory sounds with different feature sets*. In Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE (pp. 2856-2859). IEEE.
- [8] Kochetov, K., Putin, E., Azizov, S., Skorobogatov, I., & Filchenkov, A. (2017, September). *Wheeze detection using convolutional neural networks*. In Portuguese Conference on Artificial Intelligence (pp. 162-173). Springer, Cham.
- [9] Li, L., Xu, W., Hong, Q., Tong, F., & Wu, J. (2016, October). *Classification between normal and adventitious lung sounds using deep neural network*. In Chinese Spoken Language Processing (ISCSLP), 2016 10th International Symposium on (pp. 1-5). IEEE.
- [10] LISA Lab. (2014). *Deep Learning Tutorial Release 0.1*, University of Montreal.

- [11] Mangione, S., & Nieman, L. Z. (1999). *Pulmonary auscultatory skills during training in internal medicine and family practice*. American journal of respiratory and critical care medicine, 159(4), 1119-1124.
- [12] Mayorga, P., Druzgalski, C., Morelos, R. L., Gonzalez, O. H., & Vidales, J. (2010, August). *Acoustics based assessment of respiratory diseases using GMM classification*. In Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE (pp. 6312-6316). IEEE.
- [13] Mishra, M., Menon, H., & Mukherjee, A. (2018). *Characterization of S_1 and S_2 Heart Sounds Using Stacked Autoencoder and Convolutional Neural Network*. IEEE Transactions on Instrumentation and Measurement.
- [14] Murguía, J. S., & Rosu, H. C. (2011). *Discrete wavelet analyses for time series*. In *Discrete Wavelet Transforms-Theory and Applications*. InTech.
- [15] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J. (2011). *Scikit-learn: Machine learning in Python*. Journal of machine learning research, 12(Oct), 2825-2830.
- [16] Pixsoft Inc. 2008. R.A.L.E.® Lung Sounds 3.2. [Online] Available at: <http://www.rale.ca/lungsounds.htm> [Accessed 3 April 2018].
- [17] Ray, S., & Chan, A. (2001). *Automatic feature extraction from wavelet coefficients using genetic algorithms*. In *Neural Networks for Signal Processing XI*. Proceedings of the 2001 IEEE Signal Processing Society Workshop (pp. 233-241). IEEE.
- [18] Santosa, B. (2007). *Data mining teknik pemanfaatan data untuk keperluan bisnis*. Yogyakarta: Graha Ilmu, 978(979), 756.
- [19] Shmilovici, A. (2009). *Support vector machines*. In *Data mining and knowledge discovery handbook* (pp. 231-247). Springer, Boston, MA.
- [20] Taplidou, S. A., & Hadjileontiadis, L. J. (2007). *Wheeze detection based on time-frequency analysis of breath sounds*. Computers in biology and medicine, 37(8), 1073-1083.
- [21] The MathWorks, Inc. 2018. MATLAB Gausswin. [Online] Available at: <https://www.mathworks.com/help/signal/ref/gausswin.html>