PERANCANGAN SISTEM DETEKSI MALWARE MENGGUNAKAN METODE DEEP NEURAL NETWORK

TUGAS AKHIR

Diajukan Untuk Melengkapi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer

OLEH :

ROMI BAGASKARA
09121001015

JURUSAN SISTEM KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
2019

HALAMAN PENGESAHAN

PERANCANGAN SISTEM DETEKSI MALWARE MENGGUNAKAN METODE DEEP NEURAL NETWORK

TUGAS AKHIR

Diajukan Untuk Melengkapi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer

Oleh :

ROMI BAGASKARA
09121001015

Inderalaya, Juli 2019

Mengetahui,

Ketua Jurusan Sistem Komputer	Pembimbing Tugas Akhir
Rossi Passarella, M. Eng.	**Prof. Dr. Ir. Siti Nurmaini, M.T**
NIP. 197806112010121004	**NIP. 196908021994012001**

HALAMAN PERSETUJUAN

Telah di uji dan lulus pada :

Hari : Sabtu
Tanggal : 15 Juni 2019

Tim Penguji :

1. **Ketua** : **Ir. Bambang Tutuko, M.T.** (.....) 22/7/2019
2. **Anggota I** : **Rossi Passarella, M. Eng.** (.....)
3. **Anggota II** : **Huda Ubaya, M.T.** (.....)

Mengetahui,
Ketua Jurusan Sistem Komputer

Rossi Passarella, M. Eng.
NIP. 19780611 201012 1 004

HALAMAN PERNYATAAN

Yang bertanda tangan dibawah ini :

Nama **: Romi Bagaskara**

NIM **: 09121001015**

Judul **: Perancangan Sistem Deteksi Malware Menggunakan Metode Deep Neural Network**

Hasil Pengecekan *Software iThenticate/Turnitin* : 7%

Menyatakan bahwa laporan tugas akhir saya merupakan hasil karya sendiri dan bukan hasil penjiplakan / *plagiat*. Apabila ditemukan unsure penjiplakan / *plagiat* dalam laporan tugas akhir ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya sesuai dengan ketentuan yang berlaku.

Demikian, pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.

Palembang, Juli 2019

Romi Bagaskara

NIM. 09121001015

KATA PENGANTAR

Bismillahirrahmanirrahim,

Alhamdulillah puji syukur penulis haturkan kehadirat ALLAH Subhanahu Wa Ta'alaatas rahmat dan karunia NYA yang telah memberikan kesehatan dan kelancaran dalam menyelesaikan penulisan tugas akhir ini yang berjudul "**Perancangan Sistem Deteksi *Malware* Menggunakan Metode *Deep Neural Network***"

Harapan dari penulisan laporan tugas akhir ini dapat memberikan referensi bagi pembaca dan menjadi pengetahuan yang berguna pada bidang keamanan informasi khususnya forensic digital.

Dalam penyusunan laporan ini, penulis tidak lepas dari bantuan berbagai pihak yang membantu berupa bimbingan, saran, dukungan dan doa dari awal dan akhir pembuatan tugas akhir, hanya ALLAH SWT yang mampu membalas semua kebaikan. Karena itu penulis banyak mengucapkan terima kasih kepada :

1. ALLAH SWT
2. Rasulullah Muhammad SAW
3. Papa, Mama dan Mami yang selalu mendoakan dan menyayangi diriku sejak lahir
4. Saudara saya, Rangga dan Robi yang saya sayangi
5. Keluarga besar yang selalu mendukung dan mendoakan
6. Bapak Jaidan Jauhari, M.T. selaku Dekan Fakultas Ilmu Komputer
7. Bapak Rossi Pasarella, M.Eng. selaku Ketua Jurusan Sistem Komputer
8. Ibu Ir. Siti Nurmaini, M.T. selaku Pembimbing Tugas Akhir
9. Bapak Firdaus, S.T., M.Kom. selaku Pembimbing Akademik
10. Bapak Serta Ibu Dosen Fakultas Ilmu Komputer Universitas Sriwijaya
11. Teman seperjuangan Sistem Komputer 2012
12. Sahabat GGSKom, Abda, Maman, Adhi, Adi, Agung, Ayep, Bang Jek, Sam, Anggoro, Toci, Acong, Avid, Bayu, Beni, Candra, Deni, Denny, Dwiky, Edi, Edoy, Ekik, Eko, Erick, Ojik, Fahmi, Faiz, Pajik, Adik Faris, Halim, Anip, Mamat, Siu, Jaar, Maulanos, Bio, Imam, Sulkhan, Nando,

Randra, Rashad, Bram, Ojan, Sahat, Sam, Syukron, Tahta, Tomo, Visca, Agus, Yogi dan Pia.

13. Civitas Akademika Fakultas Ilmu Komputer Universitas Sriwijaya
14. Teman seperjuangan di FASILKOM UNSRI, UNSRI, HIMASISKO dan mahasiswa diluar sana yang mengerjakan skripsi, SEMANGAT!

Palembang, Juli 2019

Penulis

DAFTAR ISI

Halaman

HALAMAN JUDUL i
HALAMAN PENGESAHAN ii
HALAMAN PERSETUJUAN iii
HALAMAN PERNYATAAN iv
HALAMAN PERSEMBAHAN v
KATA PENGANTAR vi
ABSTRAK viii
ABSTRACT ix
DAFTAR ISI x
DAFTAR GAMBAR xii
DAFTAR TABEL xvi

BAB I PENDAHULUAN
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Tujuan 3
1.4 Sistematika Penulisan 3

BAB II TINJAUAN PUSTAKA
2.1 Malware 5
2.1.1 Kriteria Virus 7
2.1.1.1 Mendapatkan Informasi 8
2.1.1.2 Memeriksa Suatu Program 8
2.1.1.3 Menggandakan Diri 8
2.1.1.4 Manipulasi 9
2.1.1.5 Menyembunyikan Diri 10
2.2 Neural Network 10
2.2.1 Perancangan 12
2.2.2 Pencegahan Overfitting 13

2.2.3 Mempercepat Belajar 13
2.2.4 Pelatihan Neural Network 14
2.2.5 SMOTE (Synthetic Minority Oversampling Technique).....16
2.3 Feature Engineering 17
2.3.1 Agresi Jenis Fitur 17
2.3.2 Pelabelan 18
2.4 Deep Neural Network 18

BAB III METODOLOGI
3.1 Pendahuluan 21
3.2 Kerangka Kerja 21
3.3 Penerapan Dataset Malware 23
3.4 Ekstraksi Fitur 24
3.5 Pra-Processing.....25
3.6 Struktur Deep Neural Network 26
3.7 Validasi 36
3.8 Analisa 38

BAB IV IMPLEMENTASI DAN PENGUJIAN
4.1 Implementasi 41
4.2 Simulasi.....73
4.2.1 Hasil Nilai Akurasi Kerja.....73
4.3 Hasil Simulasi 77
4.3.1 Hasil Output dari Fungsi Aktivasi.....77
4.3.2 Hasil Model Accuracy dan Model Loss.....86

BAB V KESIMPULAN DAN SARAN
5.1 Kesimpulan 97
5.2 Saran.....97

DAFTAR PUSTAKA.....98

DAFTAR GAMBAR

Halaman

Gambar 2.1 Struktur Umum Neural Network 11
Gambar 2.2 Klasifikasi Malware.....17
Gambar 3.1 Kerangka Kerja Penelitian.....22
Gambar 3.2 Mengubah Benign dan Malware menjadi Value 24
Gambar 3.3 Algoritma Resampling.....25
Gambar 3.4 Algoritma Training 26
Gambar 3.5 Flowchart Deep Neural Network.....28
Gambar 3.6 Struktur Deep Neural Network.....29
Gambar 3.7 Flowchat Simulasi Deep Neural Network 30
Gambar 3.8 Algoritma Model Deep Neural Network 36
Gambar 3.9 Algoritma Fit Model Deep Neural Network.....37
Gambar 3.10 Algoritma Evaluasi Deep Neural Network.....38
Gambar 3.11 Algoritma Model Accuracy 40
Gambar 3.12 Algoritma Model Loss.....40
Gambar 4.1 Mengimplementasikan Library Python 42
Gambar 4.2 Memasukkan Dataset ke sistem.....42
Gambar 4.3 Jumlah Row dan Column pada setiap Variable.....43
Gambar 4.4 Proses Resampling.....44
Gambar 4.5 Proses Training Data 44
Gambar 4.6 Model Relu – Sigmoid 7 Hidden Layer.....45
Gambar 4.7 Model Relu – Sigmoid 6 Hidden Layer.....46
Gambar 4.8 Model Relu – Sigmoid 5 Hidden Layer.....46
Gambar 4.9 Model Relu – Sigmoid 4 Hidden Layer.....46
Gambar 4.10 Model Relu – Sigmoid 3 Hidden Layer.....47
Gambar 4.11 Model Relu – Sigmoid 2 Hidden Layer.....47
Gambar 4.12 Model Relu – Sigmoid 1 Hidden Layer.....47
Gambar 4.13 Model Relu – Softmax 7 Hidden Layer.....48
Gambar 4.14 Model Relu – Softmax 6 Hidden Layer.....48

Gambar 4.15 Model Relu – Softmax 5 Hidden Layer.....49
Gambar 4.16 Model Relu – Softmax 4 Hidden Layer.....49
Gambar 4.17 Model Relu – Softmax 3 Hidden Layer.....49
Gambar 4.18 Model Relu – Softmax 2 Hidden Layer.....50
Gambar 4.19 Model Relu – Softmax 1 Hidden Layer.....50
Gambar 4.20 Model Sigmoid – Softmax 7 Hidden Layer.....50
Gambar 4.21 Model Sigmoid – Softmax 6 Hidden Layer.....51
Gambar 4.22 Model Sigmoid – Softmax 5 Hidden Layer.....51
Gambar 4.23 Model Sigmoid – Softmax 4 Hidden Layer.....51
Gambar 4.24 Model Sigmoid – Softmax 3 Hidden Layer.....52
Gambar 4.25 Model Sigmoid – Softmax 2 Hidden Layer.....52
Gambar 4.26 Model Sigmoid – Softmax 1 Hidden Layer.....52
Gambar 4.27 Model Relu – Tanh 7 Hidden Layer.....53
Gambar 4.28 Model Relu – Tanh 6 Hidden Layer.....53
Gambar 4.29 Model Relu – Tanh 5 Hidden Layer.....54
Gambar 4.30 Model Relu – Tanh 4 Hidden Layer.....54
Gambar 4.31 Model Relu – Tanh 3 Hidden Layer.....54
Gambar 4.32 Model Relu – Tanh 2 Hidden Layer.....55
Gambar 4.33 Model Relu – Tanh 1 Hidden Layer.....55
Gambar 4.34 Model Sigmoid – Relu 7 Hidden Layer.....55
Gambar 4.35 Model Sigmoid – Relu 6 Hidden Layer.....56
Gambar 4.36 Model Sigmoid – Relu 5 Hidden Layer.....56
Gambar 4.37 Model Sigmoid – Relu 4 Hidden Layer.....56
Gambar 4.38 Model Sigmoid – Relu 3 Hidden Layer.....57
Gambar 4.39 Model Sigmoid – Relu 2 Hidden Layer.....57
Gambar 4.40 Model Sigmoid – Relu 1 Hidden Layer.....57
Gambar 4.41 Model Softmax – Relu 7 Hidden Layer.....58
Gambar 4.42 Model Softmax – Relu 6 Hidden Layer.....58
Gambar 4.43 Model Softmax – Relu 5 Hidden Layer.....59
Gambar 4.44 Model Softmax – Relu 4 Hidden Layer.....59
Gambar 4.45 Model Softmax – Relu 3 Hidden Layer.....59
Gambar 4.46 Model Softmax – Relu 2 Hidden Layer.....60

Gambar 4.47 Model Softmax – Relu 1 Hidden Layer.....60
Gambar 4.48 Model Softmax – Sigmoid 7 Hidden Layer.....60
Gambar 4.49 Model Softmax – Sigmoid 6 Hidden Layer.....61
Gambar 4.50 Model Softmax – Sigmoid 5 Hidden Layer.....61
Gambar 4.51 Model Softmax – Sigmoid 4 Hidden Layer.....61
Gambar 4.52 Model Softmax – Sigmoid 3 Hidden Layer.....62
Gambar 4.53 Model Softmax – Sigmoid 2 Hidden Layer.....62
Gambar 4.54 Model Softmax – Sigmoid 1 Hidden Layer.....62
Gambar 4.55 Model Tanh – Relu 7 Hidden Layer.....63
Gambar 4.56 Model Tanh – Relu 6 Hidden Layer.....63
Gambar 4.57 Model Tanh – Relu 5 Hidden Layer.....64
Gambar 4.58 Model Tanh – Relu 4 Hidden Layer.....64
Gambar 4.59 Model Tanh – Relu 3 Hidden Layer.....64
Gambar 4.60 Model Tanh – Relu 2 Hidden Layer.....65
Gambar 4.61 Model Tanh – Relu 1 Hidden Layer.....65
Gambar 4.62 Model Sigmoid – Tanh 7 Hidden Layer.....65
Gambar 4.63 Model Sigmoid – Tanh 6 Hidden Layer.....66
Gambar 4.64 Model Sigmoid – Tanh 5 Hidden Layer.....66
Gambar 4.65 Model Sigmoid – Tanh 4 Hidden Layer.....66
Gambar 4.66 Model Sigmoid – Tanh 3 Hidden Layer.....67
Gambar 4.67 Model Sigmoid – Tanh 2 Hidden Layer.....67
Gambar 4.68 Model Sigmoid – Tanh 1 Hidden Layer.....67
Gambar 4.69 Model Softmax – Tanh 7 Hidden Layer.....68
Gambar 4.70 Model Softmax – Tanh 6 Hidden Layer.....68
Gambar 4.71 Model Softmax – Tanh 5 Hidden Layer.....69
Gambar 4.72 Model Softmax – Tanh 4 Hidden Layer.....69
Gambar 4.73 Model Softmax – Tanh 3 Hidden Layer.....69
Gambar 4.74 Model Softmax – Tanh 2 Hidden Layer.....70
Gambar 4.75 Model Softmax – Tanh 1 Hidden Layer.....70
Gambar 4.76 Model Tanh – Softmax 7 Hidden Layer.....70
Gambar 4.77 Model Tanh – Softmax 6 Hidden Layer.....71
Gambar 4.78 Model Tanh – Softmax 5 Hidden Layer.....71

Gambar 4.79 Model Tanh – Softmax 4 Hidden Layer 71
Gambar 4.80 Model Tanh – Softmax 3 Hidden Layer 72
Gambar 4.81 Model Tanh – Softmax 2 Hidden Layer 72
Gambar 4.82 Model Tanh – Softmax 1 Hidden Layer 72
Gambar 4.83 Sistem Deep Neural Network dijalankan 73
Gambar 4.84 Hasil Akurasi Sistem 74
Gambar 4.85 Input grafik Model Accuracy.....74
Gambar 4.86 Input grafik Model Loss 75
Gambar 4.87 Hasil Cross – Entropy Loss 75
Gambar 4.88 Variable TP, FP, FN dan TN 76
Gambar 4.89 Hasil Sesitivity dan Specifity 76
Gambar 4.90 Accuracy dan Loss Model Relu Sigmoid 7 Hidden Layer 87
Gambar 4.91 Accuracy dan Loss Model Relu Sigmoid 6 Hidden Layer 87
Gambar 4.92 Accuracy dan Loss Model Relu Sigmoid 5 Hidden Layer 87
Gambar 4.93 Accuracy dan Loss Model Relu Sigmoid 4 Hidden Layer 88
Gambar 4.94 Accuracy dan Loss Model Relu Sigmoid 3 Hidden Layer 88
Gambar 4.95 Accuracy dan Loss Model Relu Sigmoid 2 Hidden Layer 88
Gambar 4.96 Accuracy dan Loss Model Relu Sigmoid 1 Hidden Layer 89
Gambar 4.97 Perbandingan Accuracy dan Loss Relu - Sigmoid 89
Gambar 4.98 Accuracy dan Loss Model Softmax Relu 1 Hidden Layer 90
Gambar 4.99 Accuracy dan Loss Model Softmax Relu 3 Hidden Layer 90
Gambar 4.100 Perbandingan Accuracy dan Loss Softmax - Relu 91
Gambar 4.101 Accuracy dan Loss Model Softmax Sigmoid 1 Hidden Layer ..91
Gambar 4.102 Accuracy dan Loss Model Softmax Sigmoid 2 Hidden Layer ..92
Gambar 4.103 Accuracy dan Loss Model Softmax Sigmoid 3 Hidden Layer ..92
Gambar 4.104 Perbandingan Accuracy dan Loss Softmax - Sigmoid 93
Gambar 4.105 Accuracy dan Loss Model Sigmoid Tanh 7 Hidden Layer 93
Gambar 4.106 Perbandingan Accuracy dan Loss Sigmoid - Tanh 94
Gambar 4.107 Accuracy dan Loss Model Softmax Tanh 3 Hidden Layer 95
Gambar 4.108 Perbandingan Accuracy dan Loss Sigmoid - Tanh 95

DAFTAR TABEL

Halaman

Tabel 3.1 Dataset Row 0 sampai 9 23
Tabel 3.2 Keterangan Struktur Deep Neural Network Relu Sigmoid.....31
Tabel 3.3 Keterangan Struktur Deep Neural Network Relu Softmax.....31
Tabel 3.4 Keterangan Struktur Deep Neural Network Sigmoid Softmax.....32
Tabel 3.5 Keterangan Struktur Deep Neural Network Relu Tanh 32
Tabel 3.6 Keterangan Struktur Deep Neural Network Sigmoid Relu.....32
Tabel 3.7 Keterangan Struktur Deep Neural Network Softmax Relu.....33
Tabel 3.8 Keterangan Struktur Deep Neural Network Softmax Sigmoid.....33
Tabel 3.9 Keterangan Struktur Deep Neural Network Tanh Relu 34
Tabel 3.10 Keterangan Struktur Deep Neural Network Sigmoid Tanh 34
Tabel 3.11 Keterangan Struktur Deep Neural Network Softmax Tanh 35
Tabel 3.12 Keterangan Struktur Deep Neural Network Tanh Softmax 35
Tabel 4.1 Hasil Akhir Model Deep Neural Network Relu - Sigmoid.....77
Tabel 4.2 Confusion Matrix Model Relu - Sigmoid.....78
Tabel 4.3 Hasil Akhir Model Deep Neural Network Relu – Softmax 78
Tabel 4.4 Confusion Matrix Model Relu - Softmax.....78
Tabel 4.5 Hasil Akhir Model Deep Neural Network Sigmoid – Softmax 78
Tabel 4.6 Confusion Matrix Model Sigmoid – Softmax.....79
Tabel 4.7 Hasil Akhir Model Deep Neural Network Relu – Tanh.....79
Tabel 4.8 Confusion Matrix Model Relu - Tanh.....79
Tabel 4.9 Hasil Akhir Model Deep Neural Network Sigmoid – Relu 80
Tabel 4.10 Confusion Matrix Model Sigmoid – Relu.....80
Tabel 4.11 Hasil Akhir Model Deep Neural Network Softmax – Relu 80
Tabel 4.12 Confusion Matrix Model Softmax – Relu.....81
Tabel 4.13 Hasil Akhir Model Deep Neural Network Softmax - Sigmoid.....81
Tabel 4.14 Confusion Matrix Model Softmax - Sigmoid 81
Tabel 4.15 Hasil Akhir Model Deep Neural Network Tanh – Relu.....82
Tabel 4.16 Confusion Matrix Model Tanh – Relu 82

Tabel 4.17 Hasil Akhir Model Deep Neural Network Sigmoid - Tanh 82
Tabel 4.18 Confusion Matrix Model Sigmoid - Tanh.....83
Tabel 4.19 Hasil Akhir Model Deep Neural Network Softmax - Tanh 83
Tabel 4.20 Confusion Matrix Model Softmax - Tanh.....83
Tabel 4.21 Hasil Akhir Model Deep Neural Network Tanh - Softmax 84
Tabel 4.22 Confusion Matrix Model Tanh - Softmax.....84
Tabel 4.23 Analisa Hasil Akhir Model Deep Neural Networ…….....85

Designing a Malware Detection System Using the Deep Neural Network Method

Romi Bagaskara (09121001015)
Computer Engineering Department, Faculty of Computer Science
Universitas Sriwijaya
Email: bagasthewolf13@gmail.com

ABSTRACT

This research will focus on how accurately the Deep Neural Network (DNN) can detect Malware files from the dataset provided. Malware is malicious software that refers to programs that intentionally exploit vulnerabilities in computing systems for dangerous purposes. Today, with the increasing number of malware generated every day, the need for more automated and smart methods to learn, adapt and capture malware is very important, a number of advanced solutions are offered by security companies to prevent malicious malware attacks. One of them is using Machine Learning. The results of this Malware detection test showed the results of accuracy amounted to 0.9995 using the Relu and Sigmoid activation functions. Testing using another activation function also shows an accuracy value of 0.9991 by using the Softmax and Relu activation functions.

Keywords *:Deep Neural Network (DNN), Malware, Machine Learning, Dataset, Activation Function , Confusion Matrix.*

Mengetahui,

Ketua Jurusan Sistem Komputer	**Pembimbing Tugas Akhir**
Rossi Passarella, M.Eng	**Prof. Dr. Ir. Siti Nurmaini, M.T**
NIP. 197604252010121100 1	**NIP. 196908021994012001**

Perancangan Sistem Deteksi *Malware* Menggunakan Metode *Deep Neural Network*

Romi Bagaskara (09121001015)
Computer Engineering Department, Faculty of Computer Science
Universitas Sriwijaya
Email: bagasthewolf13@gmail.com

ABSTRAK

Penelitian ini akan berfokus pada seberapa akurat Deep Neural Network (DNN) dapat mendeteksi file - file Malware dari Dataset yang sudah disediakan. Malware adalah perangkat lunak berbahaya yang mengacu pada program yang secara sengaja mengeksploitasi kerentanan dalam sistem komputasi untuk tujuan yang berbahaya. Saat ini, dengan peningkatan jumlah malware yang dihasilkan setiap hari, kebutuhan untuk metode yang lebih otomatis dan cerdas untuk belajar, beradaptasi, dan menangkap malware sangat penting, sejumlah solusi mutakhir ditawarkan para perusahaan keamanan untuk mencegah serangan malware berbahaya. Salah satunya adalah menggunakan Machine Learning. Hasil dari pengujian deteksi Malware ini menunjukkan hasil akurasi berjumlah 0.9995 dengan menggunakan fungsi aktivasi Relu dan Sigmoid. Pengujian menggunakan fungsi aktivasi lain juga menunjukkan nilai akurasi berjumlah 0.9991 dengan menggunakan fungsi aktivasi Softmax dan Relu.

Kata Kunci : *Deep Neural Network (DNN), Malware, Machine Learning, Dataset, Activation Function , Confusion Matrix.*

Mengetahui,

Ketua Jurusan Sistem Komputer	**Pembimbing Tugas Akhir**
Rossi Passarella, M.Eng	**Prof. Dr. Ir. Siti Nurmaini, M.T**
NIP. 197604252010121100 1	**NIP. 196908021994012001**

BAB I

PENDAHULUAN

1.1 Latar Belakang

Malware adalah perangkat lunak berbahaya yang mengacu pada program yang secara sengaja mengeksploitasi kerentanan dalam sistem komputasi untuk tujuan yang berbahaya [1]. Dengan komputer dan Internet yang sudah menjadi bagian penting dalam kehidupan sehari-hari, malware merupakan ancaman serius bagi keamanan setiap pengguna komputer. Malware terus memfasilitasi serangan cyber, dimana sebagai penyerang, malware tetap menjadi salah satu alat utama kampanye mereka. Oleh karena itu, Untuk melawan penjahat cyber, penting bagi para pembela untuk memahami perilaku malware, seperti pola penyebaran atau rekrutmen keanggotaan, ukuran botnet, dan distribusi bot [2].

Berbagai kategori pendekatan deteksi telah diusulkan, termasuk *Signature Based*, yang membutuhkan aturan buatan tangan sendiri untuk bendapatkan data yang relevan agar bisa melakukan deteksi, dan *Machine Learning*, yang secara otomatis memberikan alasan tentang data *Malware* dan *Benignware* agar sesuai dengan parameter model deteksi [3]. Sampai saat ini, komputer industri keamanan lebih memilih menggunakan metode *Signature Based* karena dilihat dari tingkat *Low False Positive Rates* yang dapat dicapai oleh metode tersebut. Namun, dalam beberapa tahun terakhir, *Machine Learning* berhasil mencapai deteksi tingkat tinggi pada tingkat *Low False Positive Rates* tanpa beban generasi tangan manusia yang diperlukan dengan metode manual.

Saat ini, dengan peningkatan jumlah malware yang dihasilkan setiap hari, kebutuhan untuk metode yang lebih otomatis dan cerdas untuk belajar, beradaptasi, dan menangkap malware sangat penting, sejumlah solusi mutakhir ditawarkan para perusahaan keamanan untuk mencegah serangan malware berbahaya. Yang terbaru adalah kapabilitas *Machine Learning* untuk menangkal *Malware* secara real-time. *Machine Learning* sendiri adalah sebuah cabang aplikasi dari kecerdasan buatan yang fokus pada pengembangan sebuah sistem yang mampu belajar "sendiri" tanpa harus berulang kali di program oleh manusia. Salah satu arsitektur kerja dari *Machine Learning* adalah *Artificial Neural Network*, yang merupakan sistem komputasi yang diilhami oleh jaringan saraf biologis yang memiliki struktur seperti otak hewan, sedangkan *Deep Neural Network* memiliki struktur yang lebih rumit.

Deep Neural Network adalah sebuah *Artificial Neural Network* dengan beberapa lapisan antara lapisan input dan output, *Deep Neural Network* sudah menjadi alternatif pembelajaran *Machine Learning* karena kemajuan yang signifikan dalam algoritma pelatihan [3]. *Deep Neural Network* dapat menemukan manipulasi matematis yang benar untuk mengubah input menjadi output, apakah itu hubungan linear atau hubungan non-linear. Jaringan bergerak melalui lapisan-lapisan yang menghitung probabilitas setiap keluaran.

Pada tugas akhir ini akan dilakukan perancangan sebuah algoritma *Deep Neural Network* menggunakan tool *Python* yang dilakukan pada sistem operasi windows dengan menggunakan dataset yang didapatkan dari Keggle, kemudian akan dibandingkan dengan dataset lain untuk mengklarifikasi akurasi dari

rancangan *Deep Neural Network* untuk membandingan sebuah file terinfeksi Malware atau tidak.

1.2. Rumusan Masalah

Berdasarkan uraian diatas maka peneliti akan mengidentifikasikan rumusan masalah yang akan dibahas pada tugas akhir ini adalah bagaimana merancang sistem pendeteksi Malware menggunakan *Deep Neural Network*, dimana dari rancangan *Deep Neural Network* tersebut akan menganalisis sebuah Dataset yang berisi kumpulan bahwa sebuah file terinfeksi oleh *Malware* atau tidak yang dapat menghasilkan hasil *Test Accuracy* yang tinggi.

1.3. Tujuan

Berdasarkan uraian diatas, maka peneliti akan memiliki tujuan yang dapat dicapai dari tugas akhir ini adalah sebagai berikut :

1. Aplikasi yang akan dibuat dapat memudahkan dalam melakukan analisa deteksi *Malware* menggunakan metode *Deep Neural Network*
2. Aplikasi yang akan dibuat dapat menghasilkan *Test Accuracy* yang tinggi

1.4. Sistematika Penulisan

Pada penelitian tugas akhir ini peneliti akan menggunakan sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Bab ini akan menguraikan tentang latar belakang permasalahan, mencoba merumuskan inti permasalahan yang dihadapi, menentukan tujuan dan kegunaan penelitian, yang kemudian diikuti dengan pembatasan masalah, asumsi,metodelogi penelitian serta sistematika penulisan.

BAB II. LANDASAN TEORI

Bab ini akan membahas berbagai konsep dasar dan teori-teori yang berkaitan dengan topik penelitian yang dilakukan dan hal-hal yang berguna dalam proses analisis permasalahan serta tinjauan terhadap penelitian-penelitian serupa yang telah pernah dilakukan sebelumnya termasuk sintesisnya.

BAB III. METODOLOGI

Bab ini akan menjelaskan tentang langkah-langkah (metodologi) perancangan software yang akan dibahas serta merancang aplikasi yang akan dibangun.

BAB IV. HASIL DAN ANALISA

Bab ini akan membahas mengenai hasil dari pengujian yang telah dilakukan

BAB V. KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran yang sudah diperoleh dari hasil penulisan Tugas Akhir.

DAFTAR PUSTAKA

[1] S. J. Mihai Christodorescu, D. Maughan, D. Song, and C. Wang, 2007. "Malware Detection".

[2] Kavitha.U and S.Shanmugapriya, 2017. "Malware Propagation in Large Scale Networks".

[3] S. Wo´zniak, A.-D. Alm´asi, V. Cristea, Y. Leblebici, and T. Engbersen., 2015. "Review of advances in neural networks: Neural design technology stack. In Proceedings of ELM- 2014 Volume 1", 367–376.

[4] I. You and K. Yim, 2010. "Malware obfuscation techniques: A brief survey", 297-300.

[5] X. Hu, T. Chiueh and K. G. Shin, 2009. "Large-scale malware indexing using function-call graphs", 611-620.

[6] Bishop, C. M., 2006. "Pattern Recognition and Machine Learning, Springer", ISBN 978-0-387-31073-2.

[7] Hopfield, J. J., 1982. "Neural networks and physical systems with emergent collective computational abilities". Proc. Natl. Acad. Sci. U.S.A. 79 (8): 2554–2558.

[8] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov., 2014. "Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research", 15(1):1929–1958.

[9] K. He, X. Zhang, S. Ren, and J. Sun., 2015. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". arXiv preprint arXiv:1502.01852.

[10] J. Friedman, T. Hastie, and R. Tibshirani., 2001. "The elements of statistical learning, volume 1". Springer series in statistics Springer, Berlin.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton., 2012. "Imagenet

classification with deep convolutional neural networks". In Advances in neural information processing systems, 1097–1105.

[12] Bengio and Yoshua., 2009. "Learning Deep Architectures for AI" (PDF). Foundations and Trends in Machine Learning. 2 (1): 1–127.

[13] Szegedy, Christian, Toshev, Alexander, Erhan and Dumitru., 2013. "Deep neural networks for object detection".

[14] Hof, Robert D., 2018. "Is Artificial Intelligence Finally Coming into Its Own?". MIT Technology Review.

[15] V. N. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer., 2002. "SMOTE: Synthetic Minority Over-sampling Technique", 16:321-357.