

1. Pendahuluan

Latar Belakang

Di era Revolusi Industri 4.0, teknologi digital dan internet seperti kecerdasan buatan semakin berkembang, terutama pada sistem keamanan perangkat [1]. Android telah menjadi sistem operasi perangkat seluler paling populer dalam beberapa tahun terakhir, dengan pangsa pasar global sekitar 71,96% (November 2022) [2]. Dengan demikian, jumlah pengguna *smartphone* terus meningkat dan pertumbuhan pengguna aplikasi Android juga meningkat pesat. Seiring bertambahnya jumlah pengguna aplikasi Android, jumlah dan kompleksitas data malware Android juga terus meningkat [3], menimbulkan ancaman yang signifikan terhadap keamanan perangkat seluler dan layanan yang mereka sediakan. Sehingga semakin banyak peneliti yang tertarik menggunakan pembelajaran mesin untuk melakukan deteksi malware Android [4], [5].

Banyak ilmuwan telah mencoba-coba pembelajaran mesin selama bertahun-tahun untuk mendeteksi malware android, salah satunya adalah algoritme *Support Vector Machine* (SVM). Algoritme SVM adalah Supervised Learning yang sangat cocok untuk mendeteksi atau mengklasifikasikan data yang memiliki label kelas [6], [7] seperti data malware android [3]. Kumpulan data malware memiliki dimensi yang tinggi, dari beberapa kelebihan algoritme SVM, SVM juga sangat cocok digunakan pada data yang berdimensi tinggi [8] SVM memiliki konsep yang lebih komprehensif dan lebih eksplisit secara matematis dibandingkan dengan metode klasifikasi lainnya [7], [9]. Namun SVM memiliki keterbatasan dari segi waktu komputasi dan penggunaan memori untuk data dalam jumlah besar, data memiliki jumlah sampel yang besar yaitu lebih dari 10.000 sampel [10], dataset malware android proyek DREBIN memiliki jumlah sampel 15.036 [3], karena banyaknya vektor pelatihan sehingga membutuhkan solusi dari masalah pengoptimalan *Quadratic Programming* (QP), proses untuk memecahkan masalah optimisasi matematika tertentu yang melibatkan fungsi kuadrat [4], [11]. Studi ini mengusulkan penerapan klasifikasi pembelajaran mesin paralel untuk pemrosesan data skala besar untuk menghemat waktu komputasi [4], [12].

Studi ini melakukan pendekatan metode *Sequential Minimal Optimization* (SMO) untuk memecahkan masalah *Quadratic Programming* (QP) pada saat pelatihan dan teknik dekomposisi untuk konsep komputasi paralel. Dalam melakukan dekomposisi, dengan mendistribusikan data ke sejumlah *subsets* untuk melakukan pelatihan secara independen, proses ini dapat disebut juga dengan *local training*. Setelah proses tersebut, setiap *local training* SMO memberikan *output* nilai berupa w (*weight*) parsial dan b (bias) parsial, lalu setiap *output* dihitung rata-rata dan digabungkan untuk mendapatkan nilai w (*weight*) global dan b (bias) global, nilai yang digunakan untuk melakukan klasifikasi model.

Correlation Coefficient juga diterapkan untuk melakukan analisis data malware android [13], melakukan seleksi fitur dengan memilih sejumlah fitur berkorelasi positif dengan seluruh fitur dataset sebagai parameter pengujian pada model klasifikasi, serta menggunakan dua kernel yaitu kernel linier dan gaussian.

Topik dan Batasannya

Penelitian ini berfokus pada kinerja paralel dalam melakukan deteksi malware android, menerapkan metode klasifikasi algoritme SVM untuk mendeteksi malware android, dan menerapkan dekomposisi SMO untuk menguji konsep SVM paralel, berikut beberapa batasan pada penelitian ini:

- 1) Dalam penelitian ini dataset malware android yang dipakai berasal dari proyek DREBIN [3] yaitu kumpulan data vektor fitur, dari 215 atribut yang diekstrak dari 15.036 aplikasi.
- 2) Melakukan analisis korelasi koefisien [13]–[15] pada pemilihan fitur untuk mendapatkan nilai fitur yang berkorelasi positif terhadap kelas.
- 3) Metode optimasi SMO [11] dengan dekomposisi untuk konsep paralel [4], [16].
- 4) Parameter pengujian yaitu jumlah fitur paling optimal dan dua kernel yaitu linier dan gaussian.

Tujuan

Terkait latar belakang didapatkan tujuan yang akan dicapai pada penelitian ini, sebagai berikut:

- 1) Membangun model deteksi malware menggunakan *Non-Parallel SVM* dan *Parallel SVM* dengan teknik SMO.
- 2) Menguji konsep paralel dengan teknik dekomposisi SMO (2 dekomposisi dan 4 dekomposisi).
- 3) Membandingkan hasil dan analisis untuk memperoleh kinerja terbaik pada model yang diusulkan berdasarkan parameter jumlah fitur dan kernel.