PENGELAS HIERARKI MESIN SOKONGAN VEKTOR UNTUK SISTEM PENGESANAN PENCEROBOHAN

Warhamni Jani@Mokhtar[1] dan Azizi Abdullah[2]

Pusat Keselamatan Siber, Fakulti Teknologi & Sains Maklumat,
Universiti Kebangsaan Malaysia

[1]warhamni@gmail.com, [2]azizia@ukm.edu.my

Abstrak: Pembangunan model sistem pengesanan pencerobohan (SPP) bagi mengelaskan pelbagai jenis serangan dicadangkan dalam kajian ini menggunakan Mesin Sokongan Vektor (MSV) bersama pokok binari dengan susunan hierarki. Model SPP ini dibangunkan bagi mengelaskan data set NSL-KDD kepada salah satu daripada lima kelas utama iaitu Normal, Probe, DoS, U2R dan R2L. Kemampuan SPP untuk mengelaskan kelas serangan yang berbilang merupakan isu utama bagi penentuan prestasi di samping mencapai kadar amaran palsu yang rendah. Kaedah satu-lawan-satu atau (OVO) merupakan kaedah yang popular dalam menyelesaikan masalah berbilang kelas. Namun begitu, masalah utama teknik OVO berbilang kelas adalah tahap kesamaran iaitu kemungkinan menerima jumlah undi yang sama disamping pengesanan secara terus tanpa pertimbangan kondisi lain. Kaedah hierarki ini menguji pada setiap tingkat menggunakan pengelas MSV berbeza. MSV yang asalnya merupakan pengelas binari dikembangkan kepada pengelas berbilang kelas. Kaedah pra-pemprosesan dilaksanakan melibatkan pemetaan atribut, penyediaan data dalam format MSV, penormalan serta pemilihan fitur. Kaedah hierarki MSV dicadangkan berdasarkan nilai tertinggi model pengujian menggunakan kaedah kelas binari OVO, kelas binari satu-lawan-semua (OVA) serta berbilang kelas OVO. Susunan model ditentukan dengan mengambil kira susunan keutamaan tinggi ke rendah mengikut tahap pengesanan yang diperolehi dari eksperimen-eksperimen tersebut. Pada setiap tingkat hierarki, satu kelas disingkirkan dan pengelasan dijalankan semula bagi baki kelas menggunakan pengelas seterusnya. Perbandingan dilakukan membuktikan bahawa penggunaan model berbilang kelas hierarki mampu memberikan purata ketepatan sehingga 90.98% berbanding 41.01% dengan penggunaan model berbilang kelas piawai (OVO). Jumlah amaran palsu juga berkurangan iaitu 5.77 bagi model berbilang kelas piawai berbanding 0.5 bagi model berbilang kelas hierarki.

***Kata kuncis*:** Mesin Sokongan Vektor, Sistem Pengesanan Pencerobohan, pengelas hierarki, NSL-KDD.

1.0 PENGENALAN

Sistem Pengesanan Pencerobohan (SPP) merupakan satu alat penting yang digunakan bersama komponen lain dalam pertahanan rangkaian. Kepentingan SPP adalah mampu mengenal pasti ancaman dalam rangkaian sama ada risiko ancaman dari luar atau dari dalam rangkaian itu sendiri. Menurut Veal (2005), aktiviti yang diperhatikan oleh SPP meliputi aktiviti mencurigakan contohnya cubaan capaian secara tidak sah, manipulasi dan gangguan keupayaan sistem komputer yang dilakukan oleh penceroboh seperti virus, cacing, probes, serangan, penyalahgunaan dan menyalahguna kelemahan atur cara system (Nguyen et al. 2012). Lee dan Stolfo (2000) juga menekankan bahawa SPP perlu menjadi sistem yang tepat, mampu beradaptasi dan dapat dikembangkan penggunaannya bagi mengawal selia rangkaian secara sistematik dan automatik. Menurut Panetta (2017) pakar keselamatan rangkaian bersetuju bahawa fokus utama dari mengelakkan ancaman kepada pengesanan dan respons terhadap pencerobohan.

Secara amnya, SPP dibina menggunakan prinsip pengesanan tanda tangan atau anomali (Lee dan Stolfo 2000; Alma 2012; Cheng & Syu 2015). Pengesanan secara tanda tangan mengenal pasti melalui corak serangan yang terdapat dalam pangkalan data SPP manakala pengesanan secara anomali pula berdasarkan profil kelakuan serangan yang dibina. Sebarang padanan akan mengaktifkan amaran oleh SPP dan tindakan bersesuaian akan dilaksanakan berdasarkan tetapan yang dilakukan. Model SPP tanda tangan berfungsi sama seperti program

pengesan virus yang mengenal pasti aktiviti mencurigakan berdasarkan padanan dalam pangkalan data. Model SPP tanda tangan mampu mengenal pasti serangan dengan tepat dan cepat namun mengalami kesukaran untuk mengesan aktiviti selain dari padanan sedia ada yang akan meningkatkan kadar positif palsu. Sementara itu, Model SPP anomali mempunyai kelebihan dalam mengenal pasti serangan yang belum diketahui namun mengalami kesukaran untuk membina model yang sesuai untuk aktiviti sah berikutan peningkatan tahap amaran palsu terutama dari aktiviti sah yang unik.

Terdapat beberapa kaedah yang digunakan dalam pembangunan SPP contohnya berdasarkan maklumat hos rangkaian seperti masa pengguna mengakses sistem dan sumber yang dicapai. Kaedah statistik yang ringkas dilaksanakan bagi menyemak aktiviti pengguna sama ada mempunyai padanan dengan model dalam pangkalan data. Kelemahan kaedah ini adalah aktiviti manusia berubah dan unik. Fokus ditukar dari berdasarkan jenis pengguna kepada set kelakuan. Manakala SPP berdasarkan maklumat rangkaian lebih fokus kepada paket yang dihantar dalam rangkaian berbanding set kelakuan manusia. Maklumat yang dihantar lebih ringkas dan melibatkan antara hubungan hos dan server contohnya aliran rangkaian seperti jumlah paket yang dihantar, jumlah bit yang ditukar dan sebagainya.

Disebabkan kemampuannya untuk mengenal pasti serangan yang belum pernah diketahui, model SPP anomali menjadi pilihan para penyelidik. Pada tahun 1980, James P. Anderson telah mengkaji mengenai cara-cara meningkatkan keselamatan komputer dan pemantauan di lokasi pengguna (Bruneau 2001). Kajian beliau menggunakan fail audit akaun untuk mengesan akses tidak sah. Seterusnya beliau mencadangkan satu model dibina dari statistik kelakuan normal pengguna agar ‘penyamar’ yang mempunyai perilaku berbeza dari profil normal dapat dikesan. Kajian ini telah merintis langkah awal pembinaan pengesanan pencerobohan dan mengembangkan idea asal pengesanan anomali. Pembangunannya berkembang rancak dengan gabungan pelbagai teknik seperti statistik termasuk analisis Bayesian, serta perlombongan data (Lee & Stolfo 2000). Lee & Stolfo (2000) membina kerangka SPP menggunakan algoritma perlombongan data bagi mengira corak aktiviti dari data sistem audit dan mengekstrak fitur jangkaan berdasarkan corak tersebut. Bagi meningkatkan kemampuan pembelajaran SPP, teknik pembelajaran mesin (PM) digunakan bagi memindahkan peranan pengesanan daripada manusia kepada sistem.

Bagi mendapatkan hasil yang baik, kajian ini menumpukan kepada kaedah pengelasan berbilang kelas bagi data rangkaian yang mempunyai pelbagai jenis serangan serentak dalam satu masa. SPP yang berkualiti mampu mengesan jenis serangan yang pelbagai dan tidak hanya tertumpu pada jenis serangan yang biasa dan popular. Di samping itu, walaupun jenis serangan adalah dalam jumlah yang kecil namun SPP yang berkualiti seharusnya berkebolehan untuk mengesan jenis serangan ini contohnya serangan berbahaya seperti U2R. Terdapat dua kaedah piawai pengujian berbilang kelas iaitu satu-lawan-satu (OVO) dan satu-lawan-semua (OVA). Dua kaedah utama dalam pengelasan berbilang kelas ini adalah (a) mengambil kira kesemua data dalam satu pengoptimuman contohnya OVA atau (b) membina beberapa pengelas binari contohnya OVO (Vural & Dy 2004). Isu utama bagi data yang berbilang kelas ialah bilangan data yang tidak sekata mampu mempengaruhi ketepatan pengesanan kerana jumlah data yang lebih besar mampu mendominasi keputusan akhir. Memandangkan masalah masih lagi berlarutan, kajian ini akan menyambung usaha untuk mendapatkan kaedah pengesananan serangan yang lebih baik bagi berbilang kelas.

Kajian ini memfokuskan kepada SPP berasaskan anomali. Pengesanan yang dilakukan mengambil kira kesemua kelas serangan dalam set data NSL-KDD. Bagi mendapatkan sistem

yang mempunyai kebergantungan sifar kepada manusia, SPP secara anomali memberikan banyak kelebihan berbanding secara tanda tangan (Singh & Nene 2013). Terdapat banyak kaedah telah dicadangkan untuk SPP anomali namun menurut Horng et al. (2011) pokok keputusan telah dibuktikan mempunyai prestasi yang baik. Namun begitu ramai penyelidik menyatakan MSV merupakan kaedah PM yang efektif dan mampu memberikan keputusan tepat berbanding kaedah lain (Li et al. 2011). MSV juga mudah untuk digunakan berbanding rangkaian neural (Hsu et al. 2010).

Walaupun kaedah OVA merupakan kaedah popular, ia mengalami beberapa masalah heuristik (Bishop 2006). Pertama, nilai ukuran kepercayaan yang diperolehi mungkin berbeza di antara pengelas binari. Kedua, sekiranya pembahagian di antara kelas seimbang bagi data latihan, pembelajaran pengelasan binari masih mampu melihat pembahagian yang tidak seimbang kerana pembahagian set negatif biasanya lebih besar dari set positif. Kaedah OVA juga mempunyai kelemahan terutama bagi data yang mempunyai bilangan yang kecil seperti U2R. Bilangan data yang besar akan mendominasi keputusan.Sementara itu, melalui kaedah OVO pula bilangan kelas yang banyak akan mengambil masa kerana banyak pengujian silang perlu dilakukan sebelum model yang tepat diperolehi. Di samping itu, pengesanan yang dilakukan adalah secara terus iaitu pada bahagian permukaan sahaja dan tidak terlalu mendalam kerana tiada penglibatan sebarang kondisi lain untuk dipertimbangkan. Kaedah OVO juga mengalami masalah kesamaran iaitu beberapa kawasan ruang input berkemungkinan menerima undian yang sama (Bishop 2006). Bilangan undian yang sama menyukarkan proses membuat keputusan.

Seterusnya, kajian ini memfokuskan untuk mengatasi masalah pengelasan piawai OVO melalui pembinaan pengelasan berbilang kelas hierarki. Gabungan pokok binari dan MSV dalam setiap tingkat hierarki berpandukan strategi bagi mendapatkan keputusan dengan menapis setiap kelas secara satu per satu pada setiap tingkat. Tapisan yang dilakukan dalam setiap tingkat mampu membantu proses pengelasan bagi mendapatkan hasil yang lebih tepat. Setiap kelas serangan akan diuji dengan pengelas MSV mengikut kelas serangan berdasarkan susunan keutamaan hierarki.

2.0 Aplikasi Kaedah Pembelajaran Mesin Dalam Pengelasan

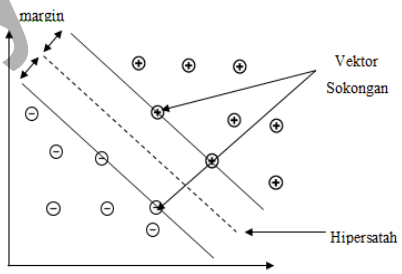
Terdapat pelbagai teknik yang diaplikasi dalam pembinaan model SPP antaranya Mesin Sokongan Vektor(MSV), Random Forest dan Algoritma Bat (BA). Enache dan Sgarciu (2015) mencadangkan satu model SPP berasaskan anomali yang mempunyai fasa pra-pemprosesan bagi pemilihan fitur menggunakan maklumat yang diperoleh dan pengesanan menggunakan pengelas MSV. Kajian ini menggunakan kelebihan algoritma Swarm Intelligence (SI), iaitu algoritma Bat (BA). Model yang dihasilkan diuji ke atas set data NSL-KDD iaitu sebanyak 9566 rekod dan dibahagikan kepada dua fail iaitu latihan dan pengujian. Hasil yang lebih baik diperolehi dengan perbandingan kaedah lain iaitu 99.15% dengan kadar amaran palsu sebanyak 0.019. Kajian ini turut menyatakan defisit algoritma MSV ialah kebergantungan kepada input parameter yang betul dari pengguna.

Seterusnya Hasan et al. (2014) membina dua jenis model pengelasan iaitu yang pertama berasaskan MSV dan yang kedua berasaskan Random Forests (RF). Hasil ujian ekperimen menunjukkan kedua-dua model adalah efektif. MSV memberikan hasil pengelasan lebih tepat berbanding RF namun mengambil masa. Manakala RF pula mampu memberikan hasil yang hampir sama dengan MSV namun lebih pantas sekiranya parameter model dibekalkan. Data

set yang digunakan adalah KDD'99 yang telah dibersihkan dari data berulang supaya pengelas tidak condong kepada rekod yang kerap. Teknik RF menghasilkan banyak pokok pengelasan. Setiap pokok dibina dengan sampel yang berbeza dari data asal menggunakan algoritma pokok pengelasan. Selepas hutan (forest) dihasilkan, satu objek yang ingin dikelaskan akan diletakkan bagi setiap pokok. Setiap pokok kemudianya akan mengundi kelas bagi objek tersebut. Undi tertinggi menjadi hasil akhir. Bagi model MSV, kernel RBF dipilih dan teknik pencarian grid digunakan bagi mendapatkan model terbaik. Hasil ketepatan bagi model MSV adalah 92.99 berbanding 91.41 bagi RF. Manakala masa yang diambil oleh RF adalah 10.62 minit berbanding 44.14 minit untuk MSV.

2.1 Mesin Sokongan Vektor

Bagi tujuan kajian ini, teknik PM yang dipilih adalah MSV. Kaedah PM menggunakan Mesin Sokongan Vektor (MSV) telah dipilih berdasarkan kemampuannya untuk melaksanakan proses pengesanan serangan dengan tepat dan betul. MSV adalah algoritma pembelajaran yang diperolehi dari teori pembelajaran statistik (Calix & Sankaran 2013; Schwenker 2000). MSV merupakan salah satu kaedah PM yang popular dan berguna bagi pengelasan data (Hsu et al. 2010) dibangunkan oleh Cortes dan Vapnik (1995) bagi kegunaan untuk menyelesaikan masalah pengesanan corak selain pengelas jiran terdekat (*nearest neighbor*). MSV telah mendapat jolokan *State-of-The-Art* iaitu satu kaedah moden, tercanggih dan terkini dalam membuat pengelasan pada pelbagai aplikasi dalam bidang pengecaman corak (Mohd Rizal Kadis 2016; Azizi Abdullah 2010; Boswell 2002; Cortes & Vapnik 1995), pengelasan imej dan teks, pengecaman tulisan tangan dan analisis bioinformatik (Pervez & Farid, 2014). Algoritma ini digunakan untuk melaksanakan pengelasan secara binari atau pengelasan dua kelas MSV, namun mampu dikembangkan dengan mudah bagi pengelasan berbilang kelas. MSV merupakan teknik pengelasan yang melibatkan pembahagian data kepada dua set data iaitu latihan dan pengujian (Azizi Abdullah 2010). Idea utama MSV adalah untuk menentukan ruang pemisahan hipersatah paling optimal sebagai garis pemisah yang mana memisahkan kelas +1 dari kelas -1 dengan memaksimumkan margin terbesar diantara titik terdekat keduanya (Calix & Sankaran 2013; Azizi Abdullah 2010). Hipersatah dibina dengan penentuan sempadan data yang dimasukkan. Titik-titik yang berada di sempadan dikenal sebagai vektor sokongan dan garis tengah diantara margin merupakan garis optimal hipersatah. Rajah 1.1 menujukkan kedudukan margin, hipersatah dan vektor sokongan.



Rajah 1.1 Kedudukan margin, hipersatah dan vektor sokongan dalam MSV

Konsep asal MSV adalah untuk memisahkan hipersatah di antara dua kelas yang terpisah secara garis lurus (linear) di mana satu kelas berlabel negatif (-1) manakala kelas berlawanannya mempunyai label positif (+1). Hipersatah yang terbaik adalah dengan mendapatkan ketebalan maksimum margin iaitu jarak batas sempadan antara dua kelas tersebut. Titik data yang terletak dengan tepat pada batas sempadan dikenali sebagai vektor sokongan (*support vector).* Schwenker (2000) menyatakan semakin besar margin, semakin tinggi kebolehan generalisasi untuk pemisahan hipersatah.

2.2 Pengoptimuman Parameter

Proses pengoptimuman parameter mampu meningkatkan prestasi pengelasan. Terdapat dua kaedah yang digunakan bagi mendapatkan pengoptimuman parameter iaitu keadah pencarian grid atau *grid-search* dan penentusahan bersilang atau *cross-validation*. Terdapat dua parameter bagi kernel RBF iaitu nilai C dan γ . Nilai keduanya tidak diketahui sebelum pengujian dijalankan maka terdapat cara bagi mendapatkan nilai terbaik bagi kedua-dua parameter ini. Penggunaan pencarian grid atau grid-search kepada nilai C dan γ menggunakan teknik pengesahan bersilang adalah disarankan. Dalam teknik pengesahan bersilang k-pusingan, data latihan dibahagikan kepada subset k yang sama saiz (Hsu et al. 2010). Seterusnya, satu subset diuji dengan menggunakan pengelas yang telah diuji kepada baki subset k-1. Oleh itu, jangkaan bagi setiap data bagi keseluruhan data latihan dilakukan dan peratusan pengesahan bersilang merupakan data yang telah dikelaskan dengan tepat. Proses pengesahan bersilang mampu mengelakkan masalah *overfitting* iaitu ralat model yang berlaku apabila sesuatu model cuba membuat jangkaan seberapa tepat kepada set poin data yang terhad. Penalaan parameter C merupakan perkara yang paling penting bagi memastikan langkah terbaik dalam MSV yang dapat meminimakan risiko struktur. Pencarian Grid atau Grid-search merupakan kaedah tradisional dalam penentuan pengoptimuman parameter yang melaksanakan pencarian satu persatu hingga selesai mengikut subset parameter yang telah ditetapkan bagi algoritma pembelajaran yang dipilih. Bagi pengelas MSV yang menggunakan kernel RBF terdapat dua parameter utama yang perlu ditalakan bagi menghasilkan prestasi yang baik bagi data yang tidak diketahui iaitu parameter C dan γ. Grid search kemudiannya melatih MSV dengan padanan C dan γ sehingga memperoleh prestasi pengelasan yang terbaik.

Penentusahan bersilang digunakan untuk mendapatkan jangkaan prestasi generalisasi sesebuah model dengan pemilihan parameter terbaik. Antara tujuan utama penentusahan bersilang adalah (a) sebagai teknik pengujian yang akan memberikan hasil yang tidak memihak kepada mana-mana jangkaan generalisasi yang boleh mengakibatkan overfitting. Seterusnya, ia juga (b) merupakan satu langkah bagi memilih model yang bersesuaian. Parameter yang diperolehi ini (nilai C dan γ terbaik) akan digunakan semula untuk mendapatkan model data latihan. Seterusnya, model yang diperolehi akan digunakan ke atas data ujian. Dalam penentusahan bersilang, set data dibahagikan kepada bilangan k-lipatan secara rawak dengan jumlah yang sama. Sekiranya nilai bagi k=10. Data latihan dipecahkan secara rawak kepada 10 subset. Satu subset ditetapkan sebagai set ujian manakala baki sembilan subset dianggap sebagai data latihan. Proses penentusahan bersilang diulang sebanyak sepuluh kali dan ketepatan pengelasan diukur dengan purata hasil ujian tersebut (Li et al. 2012). Bagi LIBSVM, terdapat program grid.py yang melaksanakan pencarian grid bagi parameter terbaik latihan untuk set fitur vektor yang dibekalkan. Program ini juga menggunakan teknik penentusahan bersilang untuk menjangka ketepatan setiap kombinasi parameter dalam skala tertentu dan seterusnya membantu pemilihan parameter terbaik.

2.3 Pengelasan menggunakan MSV

Bagi pengelasan menggunakan MSV, terdapat dua jenis pengelasan iaitu pengelasan binari dan pengelasan berbilang kelas.

a. Pengelasan Binari

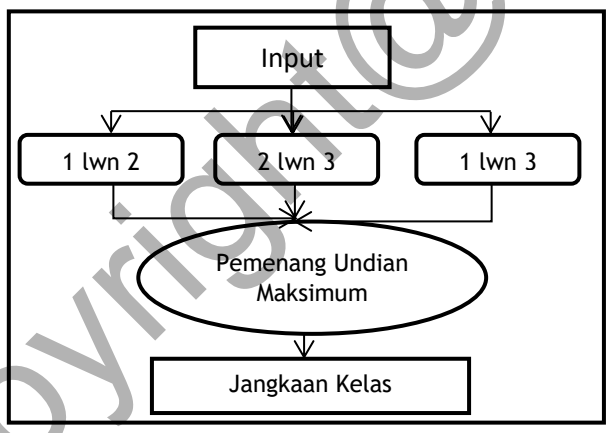
Kaedah ini digunakan apabila hanya terdapat dua kelas bagi data yang ingin diuji. Pengelas cuba mengelaskan data yang tidak diketahui kepada dua kumpulan. Namun begitu, pengelasan binari boleh dikembangkan kepada berbilang kelas iaitu dengan pengujian satu-

lawan-satu atau satu-lawan-semua sekiranya terdapat lebih daripada dua kelas yang wujud dalam set data (Azizi Abdullah 2010).

b. Pengelasan Berbilang Kelas

Jika terdapat berbilang kelas dalam sesebuah set data, tujuan yang ingin dicapai adalah untuk mengelaskan N kelas data kepada kelas yang betul. Terdapat empat kaedah dikenal pasti untuk pengelasan ini iaitu:

i. **Satu-lawan-Satu (OVO)** - Bagi pendekatan Satu-lawan-Satu, ia menggunakan kemenangan undian maksimum dan setiap satu dibeza dengan dua jenis kelas (Azizi Abdullah 2010). Jumlah kelas dikira berdasarkan N (N-1)/2 model kelas. Sebagai contoh, jika N=5, maka jumlah model kelas adalah 10. Setiap model dilatih dengan +1 bagi kelas sebenar dan -1 bagi kelas selainnya. Set data diuji kepada setiap model dan kelas yang kerap memenangi dianggap sebagai pemenang. Perbezaan dengan model satu-lawan-semua adalah lebih banyak model perlu dibina dan sukatan prestasi adalah melalui undian maksimum dengan mengambil kira hasil dari semua model. Namun jumlah rekod yang dipilih hanya bagi kelas yang terlibat dan tidak memerlukan kesemua kelas bagi setiap pengujian binari. Menurut Li et al. (2008) OVO memberikan prestasi yang lebih baik sekiranya pengelasan tepat dihasilkan. Kelemahan kaedah ini adalah apabila jumlah kelas terlalu besar. Contohnya jika N=20 maka jumlah kelas binari yang perlu dilatih adalah N (N-1)/2 = 190. Rajah 1.2 berikut menunjukkan konsep satu-lawan-satu bagi berbilang kelas.



Rajah 1.2 Konsep satu-lawan-satu.

Sumber: Gu et al. (2016)

ii. **Satu-lawan-Semua (OVA)** - Berbeza dengan pendekatan Satu-lawan-Satu, kaedah ini menggunakan strategi "*winner-takes-all*" (Azizi Abdullah 2010). Ini bermakna, jika N=5, maka jumlah model kelas adalah lima iaitu satu model bagi setiap kelas (Li et al. 2008). Setiap model akan diuji dengan set data ujian dan kelas yang memberikan keputusan pengelasan tertinggi dianggap sebagai pemenang. Kaedah OVA mengambil masa latihan yang lama dan kerapkali kadar ketepatan yang dihasilkannya lebih rendah dari OVO. Rajah 1.3 memberikan gambaran konsep satu-lawan-semua.
Pseudokod bagi algoritma pembelajaran bagi OVA yang dibina dari pengelasan binari L adalah seperti berikut:

Input:

- *L*, merupakan *learner* (algoritma pembelajaran pengelas binari)
- sampel *X*

- label y dimana $y_i \in \{1, \ldots K\}$ adalah label bagi sampel X_i

Output:

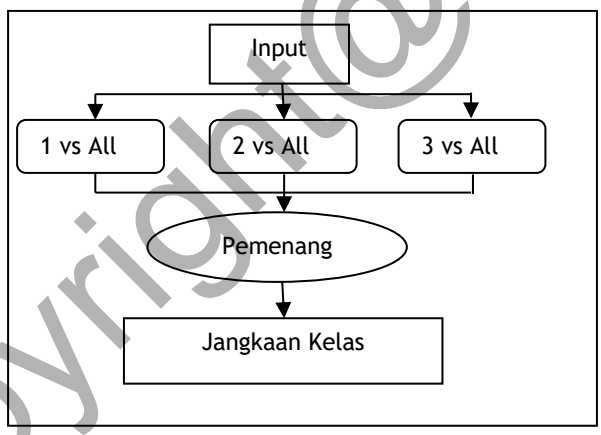
- senarai pengelas f_k bagi $k \in \{1, \ldots, K\}$

Prosedur:

- Bagi setiap k dalam $\{1, \ldots, K\}$
  - Bina label vektor yang baru, z dimana $z_i = 1$ jika $y_i = k$ dan $z_i = 0$ atau
  - Gunakan L kepada X, z untuk mendapatkan f_k

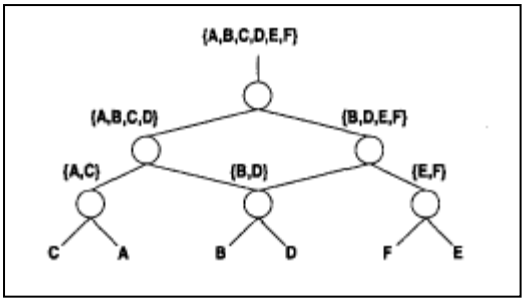
  Membuat keputusan bermakna memadankan semua pengelas kepada sampel baru x dan menjangka bagi label k yang mana bagi setiap pengelas menyatakan nilai tertinggi kepercayaan:

$$\hat{y} = \underset{k \in \{1 \ldots K\}}{\operatorname{argmax}} f_k(x)$$



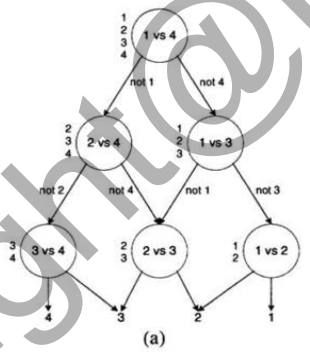
Rajah 1.3 Konsep satu-lawan-semua.
Sumber: Gu et al. (2016)

iii. **Hierarki atau pokok pengelasan binari MSV –** merupakan satu kaedah berbeza bagi menyelesaikan N-kelas masalah adalah dengan pembinaan hierarki atau pokok pengelasan binari (Schwenker 2000). Menggunakan kaedah ini, masalah pengelasan berbilang kelas dipecahkan kepada beberapa siri pengelas binari MSV yang disusun secara hierarki. Kaedah susunan adalah nod akar berada di bahagian atas manakala nod terminal (daun) berada di bahagian bawah. Setiap kelas dipersembahkan menggunakan daun dan setiap nod dikelaskan menggunakan pengelasan binari. Li et al. (2008) menyatakan hierarki yang dibina mestilah direka dengan betul sebelum latihan pengelasan dijalankan. Rajah 1.4 menunjukkan kaedah am pengelasan hierarki.



Rajah 1.4 Kaedah am Pengelasan Hierarki
Sumber: Schwenker (2000)

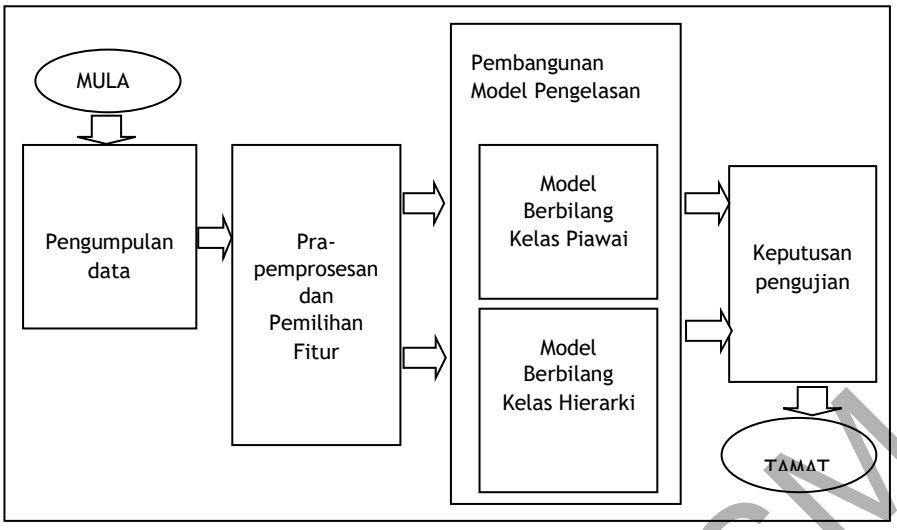
iv. **Directed acyclic graph SVM (DAGSVM) atau graf terbuka tanpa kitaran MSV:-** merupakan seni bina binari hierarki yang mana DAG digunakan untuk menggabungkan hasil yang diperolehi dari pengelas berbeza satu-lawan-satu diperkenalkan oleh Platt et. al (2000). Bagi masalah *N* kelas, sejumlah *N(N-1)/2* pengelas binari dilatih. DAGSVM bergantung kepada akar binari DAG untuk membuat keputusan. Apabila sampel ujian telah menghampiri nod daun, keputusan akhir dilakukan seperti Rajah 1.5. Pengujian binari bergantung kepada jumlah nod yang terkandung dalam laluan keputusan. Menurut Wang dan Casasent (2006), pada setiap nod, satu kelas disisihkan dari senarai.



Rajah 1.5 DAG membuat keputusan bagi 4 kelas dimana pengelas binari (MSV) digunakan dalam setiap nod
Sumber:Platt et al. (2000)

3.0 METODOLOGI

Dalam kajian ini, terdapat empat aktiviti utama yang akan dijalankan. Rajah 1.6 berikut menunjukkan aktiviti tersebut.



Rajah **Error! No text of specified style in document.**.6 Ringkasan Metodologi kajian

a. Langkah 1: Pengumpulan Data

Data yang akan diuji adalah set data NSL-KDD. Data ini digunakan untuk mengesan pencerobohan dan disediakan dari data awal yang telah diperakui iaitu dari KDD Cup' 99 (Chen & Syu 2015). Data NSL-KDD ini telah dinaiktaraf dari data asal yang mana beberapa penambahbaikan seperti pembersihan data berulang. Data ini mengandungi 41 fitur dan 1 label. Struktur data dan ciri adalah serupa dengan set data KDD Cup 1999. Terdapat 5 kelas utama data iaitu 1 normal dan 4 selebihnya data serangan.

b. Langkah 2: Pra-pemprosesan dan Pemilihan Fitur

Data tersebut kemudiannya akan menjalani proses awal bagi persediaan data kepada format yang sesuai. Proses pemetaan atribut akan dijalankan bagi menukarkan data dalam bentuk abjad kepada bentuk nombor. Di samping itu, penukaran data kepada format MSV dan penormalan akan dilaksanakan dalam langkah ini. Seterusnya bagi pengujian fitur menonjol, data akan melalui pengujian MSV untuk mendapatkan jumlah fitur yang bersesuaian dalam eksperimen selanjutnya. Tiga set fitur iaitu 13 (fitur rangkaian), 15 (fitur hos) dan 41 (keseluruhan) fitur disediakan bagi pengujian ini. Seterusnya, jumlah fitur dengan hasil ketepatan tertinggi akan digunakan.

c. Langkah 3: Pembangunan Model Pengelasan

Terdapat beberapa eksperimen dilakukan bagi pembangunan model pengelasan. Eksperimen dilakukan menerusi program LIBSVM menggunakan kernel RBF. Model berbilang kelas piawai iaitu OVO dan OVA akan dibangunkan dan diuji menerusi beberapa eksperimen bagi mendapatkan susunan keutamaan kelas serangan. Penerangan lanjut mengenai proses yang dijalankan dalam eksperimen tersebut akan diterangkan dalam langkah seterusnya. Berdasarkan susunan keutamaan, model pengelasan hierarki berbilang kelas dibina. Proses pembangunan model berbilang kelas hierarki akan dilakukan selepas ujian menggunakan LIBSVM berbanding kaedah yang digunakan oleh Horng et al. (2011) iaitu menggunakan algoritma hierarki sebelum pengujian dengan MSV.

d. Langkah 4: Keputusan Pengujian

Seterusnya, pengujian perbandingan antara pengelasan berbilang kelas piawai (OVO sahaja) dan berbilang kelas hierarki akan dilaksanakan untuk mengenal pasti kaedah yang lebih tepat

untuk pengesanan serta tahap amaran palsu yang lebih rendah. Pengelasan berbilang kelas hierarki yang dijalankan diuji bagi mendapatkan kesimpulan samada teknik tersebut mempengaruhi tahap ketepatan terutama untuk meningkatkan prestasi pengesanan. Proses pengiraan dan perbandingan disertakan bagi kedua-dua model. Kesimpulan dibuat bagi merumuskan dapatan yang diperolehi semasa kajian.

3.1 Sukatan Prestasi

Bagi mengukur tahap pencapaian prestasi model yang dibangunkan dalam kajian ini, sukatan prestasi perlu digunakan. Antara sebab utama penggunaan pengukuran adalah bagi mendapatkan hasil yang seragam dan dapat membuat perbandingan bagi algoritma pembelajaran yang dibangunkan dengan kaedah yang digunakan oleh penyelidik lain (Azizi Abdullah 2010). Bagi tujuan kajian ini, prestasi model diuji melalui tahap ketepatan (K), tahap pengesanan (P) dan amaran palsu (AP) yang dicapai (Mohd Rizal Kardis 2016, Parsaei et al 2016). Model-model yang dibangunkan dibentuk menggunakan kebarangkalian yang sesuai bagi memastikan semua faktor diambil kira. Model yang memberikan hasil ketepatan pengesanan yang tinggi dianggap model yang lebih baik dari yang lain. Namun begitu, AP perlu lebih rendah sebelum model dianggap baik dan sesuai.

Jadual 1.2 menunjukkan matriks kekeliruan yang menjadi asas pembinaan pengiraan bagi mendapatkan K, tahap P dan AP. Prestasi model dipersembahkan secara visual melalui matriks kekeliruan. Matriks kekeliruan adalah matriks empat segi dan nombor yang dipaparkan secara pepenjuru adalah jumlah pengelasan tepat dan selain dari itu adalah pengelasan yang salah. Pembacaan matriks kekeliruan adalah melalui lajur dan baris iaitu, setiap lajur adalah jangkaan manakala baris pula mewakili kategori sebenar data. Melaluinya, menurut Azizi Abdullah (2010), salah satu daripada faedah penggunaan matriks kekeliruan adalah mudah untuk melihat kelas mana yang dikesan secara tepat dan sebaliknya oleh pengelas.

Jadual 1.1 Matriks Kekeliruan

Kategori		**Jangkaan**	
		Normal	**Serangan**
Sebenar	Normal	TP	FP
	Serangan	FN	TN

Bagi maksud singkatan dalam Jadual 1.1 adalah seperti berikut:

- Positif Benar (TP) adalah nilai asal adalah benar dan berjaya dikesan sebagai benar
- Negatif Benar (TN) adalah nilai asal adalah salah dan berjaya dikesan sebagai salah
- Positif Palsu (FP) nilai asal adalah benar namun dikesan sebagai salah
- Negatif Palsu (FN) nilai asal adalah salah namun dikesan sebagai benar.

$$\text{Tahap Ketepatan (K)} = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.1)$$

$$\text{Tahap Pengesanan (P)} = \frac{TP}{TP + FP} \quad (3.2)$$

$$\text{Amaran Palsu (AP)} = \frac{FP}{FP + TN} \quad (3.3)$$

Berdasarkan persamaan tahap ketepatan (K), ianya diperolehi berdasarkan jumlah pengesanan betul bagi setiap kelas dan dibahagikan dengan jumlah data. Sementara itu, formula bagi tahap pengesanan pula diperolehi melalui jumlah tepat TP yang berjaya dikesan yang dibahagikan dengan keseluruhan pengesanan untuk serangan iaitu jumlah FP dan TP. Akhir sekali, Amaran Palsu (AP) diperolehi melalui jumlah data normal yang dikesan sebagai serangan dibahagikan dengan jumlah TN dan FP.

3.2 Pemilihan Fitur

Pemilihan fitur merupakan antara langkah penting dalam pra-pemprosesan. Bagi memastikan kajian mampu memberikan tahap pengesanan yang tinggi, beberapa ujian bagi mendapatkan nilai fitur yang menonjol telah dilaksanakan. Ujikaji yang dijalankan adalah mengambil kira jumlah fitur yang terlibat. Kang dan Kim (2016) menyatakan bahawa prestasi sistem pengesanan pencerobohan sangat bergantung kepada jumlah fitur yang dipilih dalam konteks ketepatan dan efisiensi. Lebih banyak fitur terlibat maka proses pengesanan akan mengambil masa dan sebaliknya. Walau bagaimanapun, objektif utama masih menumpukan kepada tahap pengesanan yang lebih baik antara jumlah fitur yang dipilih.

Bagi mendapatkan fitur yang menonjol atau terpenting dalam kajian ini, terdapat beberapa kaedah pemilihan fitur digunakan seperti yang telah dibincangkan. Bagi tujuan tersebut, kajian ini mengguna pakai penemuan dari kajian Staudemeyer & Omlin (2014) yang menggunakan kaedah histogram pengedaran, plot beselerak dan pokok keputusan bagi mendapatkan fitur yang benar-benar kuat dan mewakili setiap jenis kategori serangan. Melalui kaedah tersebut, kumpulan fitur yang benar-benar penting dan berguna bagi setiap jenis kategori serangan dapat ditentukan. Jadual 1.2 berikut menyenaraikan fitur yang relevan bagi setiap jenis serangan.

Jadual 1.2 Senarai fitur relevan bagi setiap kategori serangan.

Bil	Kategori Serangan	Fitur yang paling relevan dalam data set
1	DoS (Rangkaian)	3, 4, 5, 6, 8, 23, 29, 36, 38, 39, 40
2	Probe (Rangkaian)	2, 5, 29, 33, 34, 35
3	R2l (Hos)	1, 3, 5, 6, 10, 24, 32, 33, 35, 36, 37, 38, 39, 41
4	U2R (Hos)	5, 6, 10, 14, 17, 33

Dalam kajian ini, fitur ditentukan mengikut kumpulan sama ada kumpulan rangkaian atau kumpulan hos mengikut jenis serangan. DoS dan Probe merupakan kategori rangkaian manakala R2L dan U2R pula merupakan serangan kategori hos. Fitur yang telah dikenalpasti seperti jadual di atas kemudiannya digabungkan dalam satu kelas mengikut kategori sama ada serangan hos atau rangkaian. Di samping itu, setiap fitur yang berulang hanya akan dikira

sekali bagi memudahkan proses pengenalpastian mengikut kumpulan seperti Jadual 1.3 berikut.

Jadual 1.3 Senarai fitur relevan mengikut kategori serangan rangkaian dan hos.

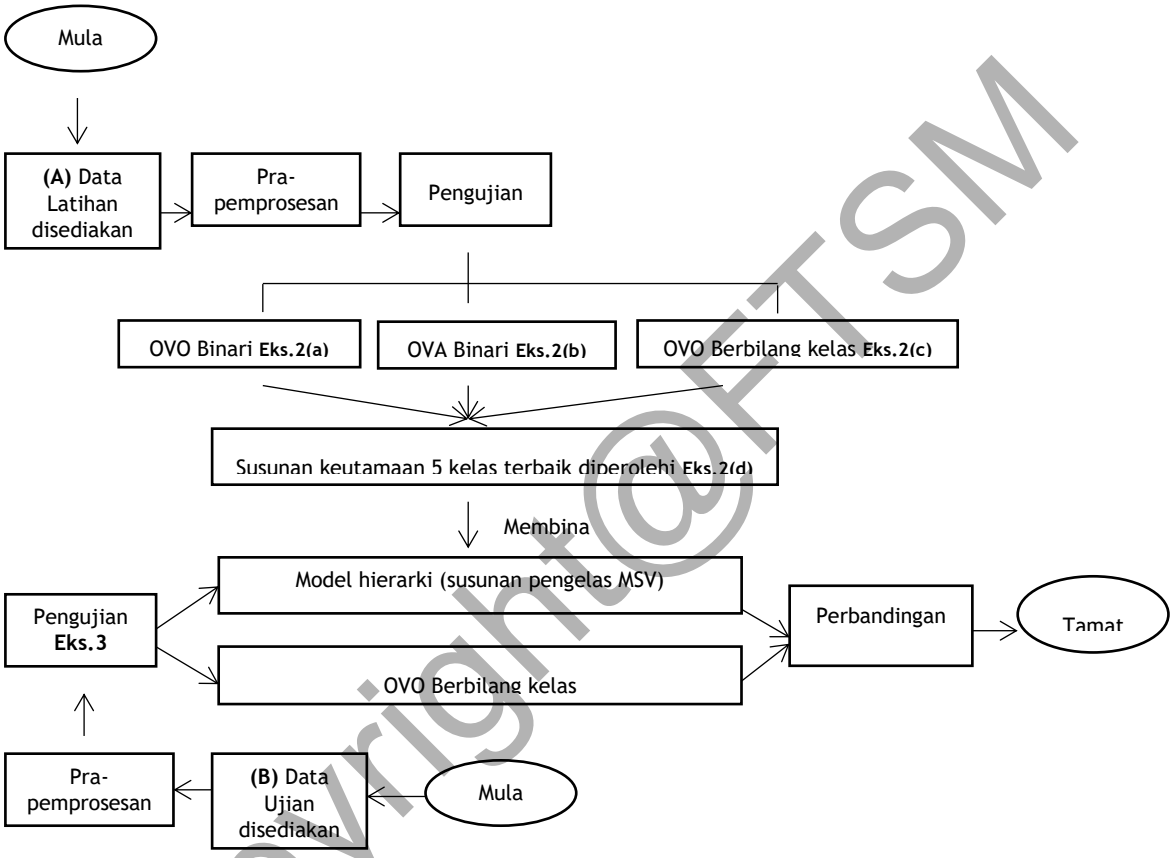
Bil	Serangan Rangkaian	Serangan Hos
1	protocol_type (2)	duration (1)
2	service (3)	service (3)
3	flag (4)	src_bytes(5)
4	src_bytes (5)	dst_bytes (6)
5	dst_bytes (6)	hot (10)
6	same_srv_rate (29)	num_file_creations (17)
7	dst_host_srv_count (33)	srv_count (24)
8	dst_host_same_srv_rate (34)	dst_host_count (32)
9	dst_host_diff_srv_rate (35)	dst_host_srv_count (33)
10	dst_host_same_src_port_rate (36)	dst_host_diff_srv_rate (35)
11	dst_host_serror_rate (38)	dst_host_same_src_port_rate (36)
12	dst_host_srv_serror_rate (39)	dst_host_srv_diff_host_rate (37)
13	dst_host_rerror_rate (40)	dst_host_serror_rate (38)
14		dst_host_srv_serror_rate (39)
15		dst_host_srv_serror_rate (41)

Bagi langkah seterusnya, set data disediakan mengikut fitur kumpulan iaitu dengan bilangan fitur 13 untuk jenis serangan rangkaian, 15 untuk serangan hos dan 41 untuk keseluruhan fitur. Set data tersebut akan melalui proses eksperimen dalam MSV. Set fitur yang memberikan nilai pengelasan tertinggi akan dipilih untuk eksperimen seterusnya.

3.3 Pengelasan Berbilang Kelas Menggunakan Pengelas Hierarki Mesin Sokongan Vektor

Teknik pengelasan menggunakan hierarki diguakan oleh beberapa penyelidik contohnya Nashat & Abdullah (2010) memberikan perincian pembinaan hierarki berbilang kelas dalam kajian mengenai pemeriksaan warna makanan menggunakan analisis Wilk's λ dan MSV. Sementara itu, Xiao dan Cheng (2015) menggunakan kaedah OVA dan OVO untuk membangunkan hierarki MSV berdasarkan pengelasan berbilang kelas bagi pengelasan berdasarkan status bas. Kajian tersebut menggunakan data pintar trafik GPS Guandong dan diproses oleh PCA serta fungsi RBF kernel untuk menguji sampel data. Data juga dikira menggunakan jarak Euclidean antara kelas. Hassan dan Damper (2010) menggunakan kaedah pengelasan binari MSV yang dipanjangkan kepada pengelasan berbilang kelas bagi mengenal pasti emosi berdasarkan ucapan. Kajian tersebut mengaplikasi dua pengelasan piawai iaitu satu-lawan-satu dan satu-lawan-semua untuk membangunkan model pengelasan hierarki yang mana setiap pengelasan memberikan pengelasan terhadap ahli bagi setiap kelas untuk tiga jenis set data awam. Set data yang digunakan adalah set data popular bagi jenis acted adalah EMO-DB, DES dan Serbian. Kesemua set data tersebut diuji menggunakan kaedah pengelasan binari iaitu satu-lawan-satu (OVO), satu-lawan-semua (OVA), Directed Acyclic Graph (DAG) dan Unbalanced Decision Tree (UDT).

Model kajian ini pula dibangunkan menerusi beberapa eksperimen yang dijalankan bagi mengenal pasti kaedah terbaik dalam pembinaan keseluruhan model dan penentuan pengelas bagi setiap tingkat hierarki. Data akan diuji menggunakan teknik OVA dan OVA Binari serta OVO Berbilang kelas terlebih dahulu dan hasil terbaik pengesanan yang diperolehi dari ujian tersebut akan menentukan susunan keutamaan pengelas bagi pembinaan hierarki. Rajah 1.7 berikut memberikan perincian penentuan model pengelas bagi setiap tingkat dan seterusnya pembinaan lengkap hierarki.



Rajah **Error! No text of specified style in document.**.7 Carta Alir Model Hierarki Berbilang kelas.

4.0 REKA BENTUK EKSPERIMEN DAN KEPUTUSAN

Bagi kajian ini, sebanyak tiga eksperimen dijalankan. Eksperimen I adalah untuk mendapatkan nilai fitur yang sesuai bagi data. Eksperimen II pula memberikan gambaran jelas mengenai data serangan berdasarkan ujian satu-lawan-semua dan satu-lawan-satu bagi model binari serta pembinaan model hierarki berbilang kelas. Bagi Eksperimen III, ujian dilaksanakan menggunakan data ujian bagi membandingkan prestasi model OVO berbilang kelas dengan model hierarki berbilang kelas.

4.1 Eksperimen 1

Objektif utama ujian adalah untuk memerhatikan tahap pengesanan bagi input fitur yang berbeza dan mendapatkan jumlah fitur yang memberikan ketepatan purata yang tinggi. Rekod yang digunakan dalam uji kaji ini adalah set data latihan. Ujian yang dijalankan pada

peringkat ini akan menggunakan tiga kumpulan fitur iaitu bilangan fitur 13 untuk jenis serangan rangkaian, 15 untuk serangan hos dan 41 untuk keseluruhan fiturSeterusnya, setiap set fitur akan diuji secara berulang iaitu sebelum dan selepas penggunaan parameter terbaik yang menghasilkan sebanyak enam pengelas. Teknik pengelasan yang digunakan adalah OVO berbilang kelas menggunakan LIBSVM dengan kernel RBF.

Jadual 1.4 Pecahan peratusan tahap ketepatan mengikut jenis serangan berdasarkan kumpulan fitur.

Bil	**Jenis serangan**	**Tahap Ketepatan (K)**		
		13 fitur	**15 fitur**	**41 fitur**
1	Normal	30.77%	31.05%	31.03%
2	U2R	0.23%	0.1%	0.255%
3	R2L	6.02%	6.16%	6.113%
4	DoS	31.10%	31.15%	31.16%
5	Probe	31.10%	31.07%	31.13%
	Purata Tahap Ketepatan %	**99.22%**	**99.53%**	**99.69%**

Jadual 1.4 menyenaraikan pecahan peratusan tahap ketepatan diperolehi dari tiga kumpulan fitur bagi setiap jenis serangan. Jumlah fitur 13 merupakan fitur jenis serangan rangkaian dan berdasarkan pemerhatian, nilai bagi Probe dan DoS adalah sebanyak 31.10% bagi kedua-duanya manakala bagi jumlah fitur 15 pula adalah sebanyak 31.15% dan 31.07%. Manakala bagi fitur 15 pula merupakan jenis fitur serangan hos dan boleh dilihat bahawa nilai pengesanan U2R kurang sedikit berbanding penggunaan 13 fitur iaitu 0.23% berbanding 0.1%. Namun terdapat sedikit kenaikan peratusan bagi nilai pengesanan yang diperolehi bagi kelas R2L dari 6.02% kepada 6.16%. Berdasarkan Jadual 1.6, didapati peratusan tertinggi pengesanan adalah dengan jumlah fitur 41 diikuti dengan 15 fitur dan terakhir dengan jumlah fitur 13. Oleh itu, jumlah fitur bagi set data latihan dan ujian untuk kesemua eksperimen seterusnya akan menggunakan 41 fitur berdasarkan dapatan ini.

4.2 Eksperimen 2

Objektif utama ujian peringkat kedua adalah untuk menilai prestasi model pengelasan iaitu diantara model pengelasan berbilang kelas piawai (OVO dan OVA) dan pembinaan model pengelasan berbilang kelas hierarki. Terdapat tiga jenis pengujian yang akan dijalankan iaitu OVO binari, OVO berbilang kelas serta OVA binari sebelum pembinaan hierarki berbilang kelas. Data disediakan mengikut ujian yang dinyatakan. Rekod yang digunakan adalah daripada set data latihan. Jumlah model pengelasan yang diuji pula adalah sebanyak 10 pengelas bagi OVO dan lima pengelas bagi OVA menggunakan kumpulan 41 fitur (hasil dari Eksperimen I). Hasil daripada pengujian adalah susunan keutamaan berdasarkan jenis serangan bagi pembinaan model pengelasan hierarki bagi Eksperimen 3.

Eksperimen 2(a)

Dalam eksperimen ini, data dipecahkan kepada 5 kelas utama. Setiap kelas akan diuji secara binari dengan kelas yang lain secara bersilang sehingga terhasil sebanyak 20 pasangan pengujian seperti Jadual 1.5. Seterusnya setiap pasangan kelas ini diuji menggunakan LIBSVM dengan kernel RBF. Kaedah ini merupakan kaedah pengujian secara OVO namun dihasilkan secara manual. Nilai tertinggi pengelasan bagi setiap kelas utama yang diuji akan dipilih (dihitamkan).

Jadual 1.5 Peratusan ketepatan pengesanan yang diperolehi daripada pengujian kelas Binari OVO.

Bil	**Set Gabungan Data**		**Jumlah Data**	**Jumlah Pengesanan Betul**		**Parameter terbaik**		**Peratus Pengesanan**
	A	**B**		**A**	**B**	C	γ	
1	DoS lawan Probe		10 000	5000	5000	32 768	0.008	**100%**
2	DoS lawan U2R		5052	4999	52	2048	0.031	99.98%
3	DoS lawan R2L		5995	4998	992	2048	0.0005	99.92%
4	DoS lawan Normal		10 000	4997	4998	32	0.5	99.95%
5	NORMAL lawan U2R		5052	4999	35	2048	0.008	99.64%
6	NORMAL lawan DoS		10 000	4998	4997	32	0.5	**99.95%**
7	NORMAL lawan Probe		10 000	4995	4992	128	0.125	99.87%
8	NORMAL lawan R2L		5995	4994	994	512	2	97.15%
9	PROBE lawan DoS		10 000	5000	5000	32768	0.008	**100%**
10	PROBE lawan R2L		5995	5000	995	128	0.031	**100%**
11	PROBE lawan U2R		5052	4999	45	512	0.0001	99.84%
12	PROBE lawan Normal		10 000	4990	4995	128	0.125	99.85%
13	U2R lawan Dos		5052	52	4999	2048	0.031	**99.98%**
14	U2R lawan Probe		5052	45	4999	512	0.0001	99.84%
15	U2R lawan Normal		5052	35	4999	2048	0.008	99.64%
16	U2R lawan R2L		1047	25	995	32 768	0.0005	97.42%
17	R2L lawan Probe		5995	995	5000	128	0.031	**100%**
18	R2L lawan U2R		1047	995	25	32 768	0.0005	97.42%
19	R2L lawan Normal		5995	994	4994	512	2	99.88%
20	R2L lawan DoS		5995	992	4998	2048	0.0005	99.93%

Didapati model DoS lawan Probe dan Probe lawan R2L menghasilkan jumlah peratus tertinggi iaitu masing-masing 100%. Ini menunjukkan model tersebut mampu mengelas dengan tepat. Disamping itu, dapat juga diperhatikan bahawa model yang memberikan peratusan paling rendah adalah model U2R lawan R2L iaitu 97.42%.

Eksperimen 2(b)

Pengujian ini adalah pengujian kelas binari satu-lawan-semua. Dalam eksperimen ini, data dipecahkan kepada 5 kelas utama dan pengujian dilakukan antara satu kelas dengan baki kelas yg lain. Sebagai contoh Dos lawan gabungan (Probe+U2R+R2L+Normal). Set-set yang terhasil ini diuji menggunakan LIBSVM bersama kernel RBF dan data ditanda dengan tetapan 0 bagi kelas utama dan 1 bagi kelas gabungan. Dalam keaedah ini, data diuji secara OVA. Jadual 1.6 memberikan perbandingan keputusan tahap ketepatan yang diperolehi dengan penggunaan parameter terbaik. Berdasarkan jadual tersebut, didapati kategori serangan DoS memberikan hasil tertinggi ketepatan menggunakan nilai terbaik parameter C dan γ iaitu 99.99%. Oleh itu, DoS mendapat susunan keutamaan tertinggi. Probe pula memberikan hasil ketepatan sebanyak 99.913% dan R2L sebanyak 99.907%. Seterusnya U2R memperolehi sebanyak 99.83% dan terakhir Normal sebanyak 99.58%.

Jadual 1.6 Peratusan ketepatan pengesanan bagi pengujian kelas binari OVA.

Kelas	**Nilai C terbaik**	**Nilai γ terbaik**	**Tahap Ketepatan (K) dengan nilai terbaik parameter C dan γ**
Normal	128	0.125	**99.58%**
U2R	512	0.00195	**99.83%**
R2L	128	2	**99.91%**
DoS	512	0.125	**99.99%**
Probe	128	0.125	**99.91%**

Eksperimen 2(c)

Bagi eksperimen ini, data dipecahkan kepada 5 kelas utama dan ditanda menggunakan nilai 0 hingga 4 bagi setiap kelas. Pengujian menggunakan LIBSVM dengan kernel RBF dan dijalankan secara OVO berbilang kelas. Jadual 1.7 menunjukkan matriks kekeliruan yang diperolehi dari pengelasan OVO. Manakala Jadual 1.8 pula memberikan tahap ketepatan bagi setiap kelas.

Jadual 1.7 Matriks kekeliruan ketepatan pengesanan yang diperolehi daripada pengelasan OVO.

Kategori	**Normal**	**U2R**	**R2L**	**DoS**	**Probe**
Normal	**4980**	1	12	1	6
U2R	7	**41**	4	0	0
R2L	14	0	**981**	0	0
DoS	0	0	0	**5000**	0
Probe	4	0	0	0	**4996**

Jadual 1.8 Tahap ketepatan bagi pengujian OVO berbilang kelas

Kategori	**Ketepatan**	**Peratus Ketepatan**
Normal	4980/5000	**99.60%**
U2R	41/52	**78.85%**
R2L	981/995	**98.59%**

DoS	5000/5000	**100.00%**
Probe	4996/5000	**99.92%**

Didapati model DoS menghasilkan jumlah peratus tertinggi iaitu 100% (5000/5000) tepat. Model kedua tertinggi adalah Probe iaitu 99.92% (4996/5000). Ini menunjukkan kedua-dua model tersebut mampu mengelas dengan tepat. Disamping itu, dapat juga diperhatikan bahawa model yang memberikan peratusan paling rendah adalah model U2R iaitu sekitar 78.85% (41/52).

Susunan Keutamaan 2(d)

Berdasarkan hasil yang diperolehi dari ketiga-tiga ujian iaitu (a) kelas binari OVO, (b) kelas binari OVA dan (c) berbilang kelas OVO tersebut, penyusunan keutamaan dilakukan bagi mendapatkan susunan mengikut tahap ketepatan tertinggi disusuli dengan ketepatan seterusnya seperti Jadual 1.9.

Jadual 1.9 Rumusan Perbandingan keputusan tahap ketepatan antara kelas binari OVO, kelas binari OVA dan berbilang kelas OVO

Kelas	**Kelas Binari OVO**	**Susunan keutamaan**	**Kelas Binari OVA**	**Susunan keutamaan**	**Berbilang Kelas OVO**	**Susunan keutamaan**
Normal	99.95%	3	99.58%	5	99.60%	3
U2R	99.98%	2	99.83%	4	78.85%	5
R2L	100%	1	99.91%	3	98.59%	4
DoS	100%	1	99.99%	1	100.00%	1
Probe	100%	1	99.91%	2	99.92%	2

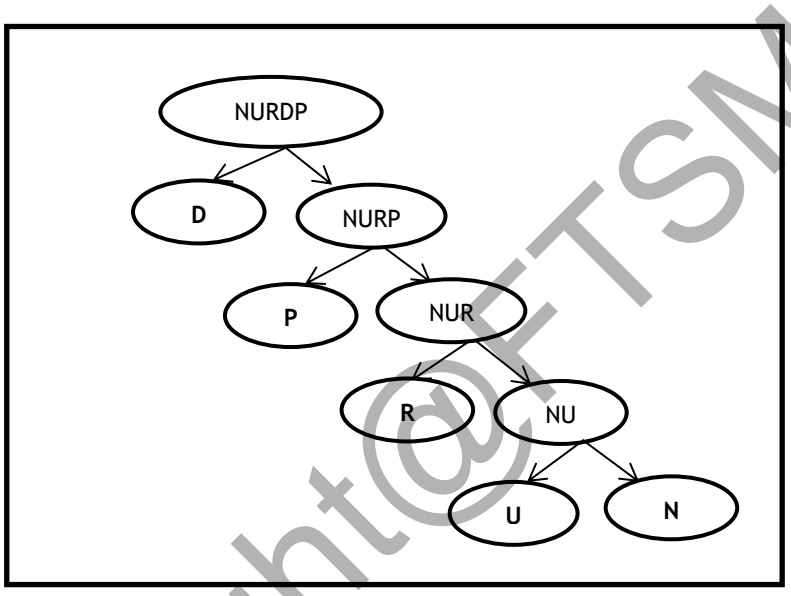
Seterusnya, dapat disimpulkan bahawa model pengelas DoS memberikan peratusan tertinggi berdasarkan ujian dari kelas binari OVO, kelas binari OVA dan berbilang kelas OVO diikuti pengelas Probe. Penentuan tingkat ketiga adalah R2L berdasarkan ketepatan tertinggi yang diperolehi semasa kelas binari OVO dan ketiga tertinggi bagi kelas binari OVA. Seterusnya U2R dan Normal merupakan kelas terbawah bagi susunan keutamaan nilai pengesanan berdasarkan kelas binari OVO dan kelas binari OVA. Berdasarkan nilai ketepatan pengesanan yang diperolehi daripada kesemua ujian tersebut maka rumusan susunan keutamaan terbaik adalah seperti Jadual 1.10 berikut.

Jadual 1.10 Susunan keutamaan akhir tahap pengesanan mengikut kelas berdasarkan susunan terbaik

Kelas	**Susunan keutamaan**
DoS	1
Probe	2
R2L	3
U2R	4
Normal	5

4.3 Eksperimen 3

Objektif utama adalah untuk menilai prestasi model pengelasan berbilang kelas piawai (OVO) dan hierarki terhadap rekod dalam set data ujian. Ujian ini adalah bagi melihat kemampuan model pengelasan untuk mengesan jenis serangan baru yang tiada dalam data latihan sebelum ini. Rekod yang digunakan adalah set data ujian. Model yang digunakan adalah model pengelasan piawai (OVO) serta model berbilang kelas hierarki yang dibangunkan dari Eksperimen II. Perbandingan dilakukan dengan hasil yang diperolehi daripada kaedah pengelasan berbilang kelas OVO dengan berbilang kelas hierarki terhadap set data ujian. Seterusnya, Rajah 1.8 merupakan cadangan pembinaan model pengelasan berbilang kelas hierarki dengan menyisihkan satu kelas pada setiap tingkat dimulakan mengikut susunan keutamaan dalam Jadual 1.10.



Rajah **Error! No text of specified style in document.**.8 Cadangan Model Hierarki Berbilang

Jadual 1.11 Perbandingan tahap pengesanan Model Berbilang Kelas OVO dan Model Berbilang Kelas Hierarki mengikut kelas menggunakan Data Latihan dan Ujian

Kelas	**Model Berbilang Kelas OVO**		**Model Berbilang Kelas Hierarki**	
	Data Latihan	**Data Ujian**	**Data Latihan**	**Data Ujian**
Normal	99.60%	96.89%	99.64%	99.38%
U2r	78.85%	0.00%	99.90%	99.38%
R2l	98.59%	13.72%	99.99%	77.17%
Dos	100%	0.00%	99.64%	88.73%
Probe	99.92%	94.42%	99.85%	90.28%
Purata %	**95.39%**	**41.01%**	**99.80%**	**90.98%**

Berdasarkan Jadual 1.11 di atas, Model berbilang kelas hierarki memberikan hasil lebih tinggi dalam pengujian menggunakan data latihan iaitu 99.80% berbanding 95.39% bagi Model berbilang kelas OVO. Seterusnya pengujian Model berbilang kelas hierarki menggunakan data ujian memberikan keputusan Purata Ketepatan (P) yang lebih baik berbanding Model berbilang kelas OVO iaitu sebanyak 90.98% berbanding 41.01%. Bagi model berbilang kelas OVO, terdapat dua kelas yang memberikan hasil 0% semasa pengujian dengan Data Ujian iaitu U2R dan DoS. Ini merupakan antara kelemahan OVO yang mana kaedah ini gagal untuk memberikan generalisasi yang tepat terutama bagi data serangan yang mempunyai jumlah yang sangat kecil seperti U2R atau data yang terlalu besar iaitu DoS. Ini kerana dalam set data latihan, data DoS merupakan jumlah terbesar bagi kelas serangan dan U2R pula merupakan jumlah yang paling kecil. Pengelasan Model berbilang kelas OVO yang dilakukan adalah serentak bagi kelima-lima model berbanding satu per satu bagi model hierarki. Melalui penggunaan Model berbilang kelas hierarki, generalisasi yang lebih baik mampu dihasilkan kerana jumlah kelas semakin berkurangan pada setiap tingkat menurun.

5.0 KESIMPULAN

Setiap pengelas MSV hanya mampu menguruskan pengelasan secara binari. Bagi tujuan pengelasan berbilang kelas, gabungan beberapa strategi MSV seperti OVA, OVO dan pokok binari digunakan. Matlamat kajian ini adalah untuk menguji tahap pengesanan yang lebih baik di antara Model berbilang kelas OVO dan Model berbilang kelas hierarki pokok binari. Berdasarkan eksperimen yang dijalankan ke atas set data NSL-KDD, model yang dicadangkan ini mampu mencapai ketepatan sebanyak 90.98% dengan kadar amaran palsu sebanyak 0.5. Model ini juga menunjukkan peningkatan bagi kelas U2R dan R2L walaupun tidak pada model DoS dan Probe. Kajian ini dilaksanakan menggunakan data latihan sebanyak 16 047 rekod. Jumlah data ujian adalah 22 544 dan terdapat jenis serangan baru yang tiada dalam set data latihan. Oleh itu, kajian ini mencadangkan penggunaan hierarki MSV pokok binari condong bagi pengelasan berbilang kelas yang mana ianya memberikan hasil lebih baik berbanding strategi OVO. Ujian melalui eksperimen menunjukkan kesimpulan adalah menyakinkan.

6.0 CADANGAN PERLUASAN KAJIAN

Sebagai kesinambungan bagi memastikan kajian yang berterusan, terdapat beberapa aspek yang boleh diberikan perhatian bagi memaksimumkan dapatan dan memantapkan lagi aspek kajian. Berdasarkan kajian yang telah dijalankan, dapat dirumuskan bahawa kaedah pengelasan hierarki MSV pokok binari berbilang kelas mampu memberikan hasil yang lebih baik berbanding kaedah pengelasan OVO dan OVA bagi pengelasan lima jenis serangan

untuk set data NSL-KDD. Ini kerana penyisihan satu kelas pada setiap tingkat mampu membantu mempercepatkan proses pengelasan serta menghasilkan keputusan yang lebih baik.Kajian ini juga menyumbangkan kepada kaedah susunan keutamaan bagi setiap tingkat hierarki. Dalam bidang pengesanan pencerobohan terdapat pelbagai kaedah yang digunakan untuk menentukan susunan hierarki contohnya kaedah Wilk's Analisis, kluster dan DAG. Namun, dalam kajian ini melalui beberapa eksperimen yang dijalankan, susunan tingkat hierarki ditentukan dengan hasil tertinggi kepada yang terendah diperolehi dari setiap eksperimen. Ini membantu perkembangan konsep susunan tingkat hierarki bagi kajian masa hadapan.Antara cadangan kajian yang boleh digunakan untuk mengembangkan kajian adalah pemilihan fitur gabungan antara fitur, penggunaan data yang seimbang dan menyeluruh bagi pembinaan model serta penggunaan kernel yang berbeza bagi mendapatkan hasil yang lebih baik. Secara keseluruhannya kajian ini membuktikan bahawa kaedah pengelasan Model berbilang kelas hierarki MSV pokok binari condong mampu memberikan keputusan yang jauh lebih baik berbanding Model pengelasan piawai berbilang kelas OVO dan OVA.

7.0 RUJUKAN

Azizi Abdullah. 2010. Supervised Learning Algorithms for Visual Object Categorization. Tesis PhD, Universiteit Utrecht.

Benabdeslem, K. 2006. Descendant Hierarchical Support Vector Machine for Multi-Class Problems. *International Joint Conference on Neural Networks (IJCNN)*. doi: 10.1109/IJCNN.2006.246868

Bishop, C. M. 2006. Pattern Recognition and Machine Learning. Springer-Verlag New York. Softcover ISBN 978-1-4939-3843-8. http://users.isr.ist.utl.pt/~wurmd/Livros/school/Bishop%20-%20Pattern%20Recognition%20And%20Machine%20Learning%20-%20Springer%20%202006.pdf

Bruneau, G. 2001. The History and Evolution of Intrusion Detection. SANS Institute InfoSec Reading Room. https://www.sans.org/reading-room/whitepapers/detection/history-evolution-intrusion-detection-344.

Calix, R. A. & Sankaran R. 2013. Feature Ranking and Support Vector Machines Classification Analysis of the NSL-KDD Intrusion Detection Corpus. *Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society (FLAIRS Conference) Conference.* Association for the Advancement of Artificial Intelligence (www.aaai.org)

Chen, L.-S. & Syu, J.-S. 2015. Feature Extraction based Approaches for Improving the Performance of Intrusion Detection Systems. *Proceedings of the International MultiConference of Engineers and Computer Scientists 2015 Vol I, IMECS 2015, March 18-20, 2015, Hong Kong.*

Cortes, C. & Vapnik, V. 1995. AT&T Bell Labs., Holmdel, NJ 07733, USA. *Machine Learning, 20, 273-297 (1995). Kluwer Academic Publishers, Boston.*

Denning, D. E. 1987. An Intrusion-Detection Model. *IEEE Transactions On Software Engineering, Vol. Se-13, No. 2, February 1987*

Eid, H. F., Hassanien, A. E., Kim, T.-H. & Banerjee, S. 2010. Linear Correlation-Based Feature Selection for Network Intrusion Detection Model. *Scientific Research Group in Egypt (SRGE)*.http://www.egyptscience.net

Enache, A.-C. & Sgârciu, V. 2015. Anomaly Intrusions Detection Based On Support Vector Machines with an Improved Bat Algorithm. 2015 20th International Conference on Control Systems and Computer Science. doi: 10.1109/CSCS.2015.12

Fischer, M. 2014. Resilient Networking: Intrusion Detection. https://www.tk.informatik.tu-darmstadt.de/fileadmin/user_upload/Group_TK/08_IDS_01.pdf. Technische Universitat Darmstadt. [21 September 2017]

Ghose, A. 2017. Support Vector Machine (SVM) Tutorial. https://blog.statsbot.co/support-vector-machines-tutorial-c1618e635e93

Gu, C., Zhang, B., Wan, X., Huang, M. & Zou, G. 2016. The Modularity-based Hierarchical Tree Algorithm for Multi-class Classification. Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2016 17th IEEE/ACIS International Conference on 30 May-1 June 2016.

Hasan, M. A. M., Nasser, M., Pal, B., & Ahmad, S. 2014. Support Vector Machine and Random Forest Modeling for Intrusion Detection System (IDS). *Journal of Intelligent Learning Systems and Applications. Vol. 6, No. 1(2014) 45-52*. doi: 10.4236/jilsa.2014.61005

Hassan, A. & Damper, R. I. 2010. Multi-class and Hierarchical SVMs for Emotion Recognition. School of Electronics and Computer Science, University of Southampton, SO17 1BJ, UK.

Horng, S.H., Su, M.-Y., Chen, Y.-H., Kao, T.-W., Chen, R.-J., Lai, J.-L. & Perkasa, C.D. 2011. A Novel Intrusion Detection System Based On Hierarchical Clustering And Support Vector Machines. *Expert Systems with Applications 38 (2011) 306-313.* https://doi.org/10.1016/j.eswa.2010.06.066

Hsu, C.-W., Chang, C.-C. & Lin, C.-J. 2010. A Practical Guide to Support Vector Classification. http://www.csie.ntu.edu.tw/~cjlin.

Kang, S.-H. & Kim, K. J. 2016. A Feature Selection Approach To Find Optimal Feature Subsets For The Network Intrusion Detection System. Springer Science+Business Media New York 2016.

Lee, W. & Stolfo, S. J. 2000. A Framework for Constructing Features and Models for Intrusion Detection Systems. *ACM Transactions on Information and System Security, Vol. 3, No. 4, November 2000, Pages 227-261*.

Li, H., Jiao R. & Fan J. 2008. Precision of Multi-class Classification Methods for Support Vector Machines. *Signal Processing, 2008. ICSP 2008. 9th International Conference on 26-29 Oct. 2008*.

Li, Y., Xia, J., Zhang, S., Yan, J. Ai, X. & Dai, K. 2012. An efficient intrusion detection system based on support vector machines and gradually feature removal method. *Expert Systems with Applications, 39(1):424-430.* doi: 10.1016/j.eswa.2011.07.032.

Liao, H.-J., Lin, C.-H. R., Lin, Y.-C. & Tung, K.-Y. 2013. Intrusion Detection System: A Comprehensive Review. *Journal of Network and Computer Applications 36 (2013):16-24.*

Limthong, K. 2013. Real-Time Computer Network Anomaly Detection Using Machine Learning Techniques. *Journal of Advances in Computer Networks, Vol. 1, No. 1, March 2013.*

Mohd Rizal Kadis. 2016. Umpukan Lembut Kluster Sejagat dan Setempat untuk Sistem Pengesanan Pencerobohan: Satu Kajian Perbandingan. Tesis Sarjana Keselamatan Siber. Universiti Kebangsaan Malaysia.

Nashat, S. & Abdullah, M.Z, 2010. Multi-Class Colour Inspection of Baked Foods Featuring Support Vector Machine and Wilk's λ Analysis. *Journal of Food Engineering 101 (2010) 370–380.* doi: 0.1016/j.jfoodeng.2010.07.022

Nashat, S., Abdullah, A., Aramvith, S. & Abdullah, M. Z. 2011. Support Vector Machine Approach to Real-Time Inspection of Biscuits on Moving Conveyor Belt. *Computers and Electronics in Agriculture, 75(1), 147-158.* doi: 10.1016/j.compag.2010.10.010

Nguyen, H. T., Franke, K. & Petrovic, S. 2012. Feature Extraction Methods for Intrusion Detections System. https://www.researchgate.net/publication/231175349_Feature_Extraction_Methods_for_Intrusion_Detection_Systems.

Panetta, K. 2017. 5 trends in cybersecurity for 2017 and 2018. http://www.gartner.com/smarterwithgartner/5-trends-in-cybersecurity-for-2017-and-2018/.

Parsaei, M. R., Rostami S. M. & Javidan, R. 2016. A Hybrid Data Mining Approach for Intrusion Detection on Imbalanced NSL-KDD Dataset. *International Journal of Advanced Computer Science and Applications, Vol. 7, No. 6, 2016*.

Pervez, M. S. & Farid, D. M. 2014. Feature Selection and Intrusion Classification in NSL-KDD Cup 99 Dataset Employing SVMs. *8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA).* doi: 10.1109/SKIMA.2014.7083539.

Platt, J. C., Cristianini, N. & Shawe-Taylor, J. 2000. Large Margin DAGs for Multiclass Classification. *In Advances in Neural Information Processing Systems (pp. 547-553).*doi: 10.1.1.158.4557.

Sahu, S. K., Sarangi, S. & Jena, S. K. 2014. A Detail Analysis on Intrusion Detection Datasets. *Souvenir of the 2014 IEEE International Advance Computing Conference, IACC 2014.* doi: 10.1109/IAdCC.2014.6779523

Sasan, H. P. S., & Sharma, M. 2016. Intrusion Detection Using Feature Selection and Machine Learning Algorithm with Misuse Detection. *International Journal of Computer Science & Information Technology (IJCSIT) Vol 8, No 1, February 2016.*

Schwenker, F. 2000. Hierarchical Support Vector Machines for Multi-Class Pattern Recognition. *Fourth International Conference on Knowledge-Based Intelligent Engineering Systems and Allied Technologies.*doi: 10.1109/KES.2000.884111

Singh, J. & Nene, M. J. 2013. A Survey on Machine Learning Techniques for Intrusion Detection Systems. *International Journal of Advanced Research in Computer and Communication Engineering. 2013, Nov,2(11).*

Sridhar, M. S. 2017 . Research Methodology Part 1:Introduction to Research & Research Methodology. ISRO Satellite Centre.

https://www.researchgate.net/publication/39168208_Research_Methodology_Part_1_Introduction_to_Research_Research_Methodology

Staudemeyer, R. C. & Omlin, C.W. 2014. Extracting Salient Features for Network Intrusion Detection using Machine Learning Methods. *South African Computer Journal Research Article-SACJ, 53, July 2014*. doi: 10.18489/sacj.v52i0.200.

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. 2009. A Detailed Analysis of the KDD CUP 99 Data Set. *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA 2009).* doi: 10.1109/CISDA.2009.5356528

The UCI KDD Archive 1999. KDD CUP 1999 Data. University of California, Irvine. http://kdd.ics.uci.edu/databases/kddcup99/task.html

Xiao, L. & Cheng, L. 2015. State Classification Algorithm for Bus Based on Hierarchical Support Vector Machine. *2015 8th International Symposium on Computational Intelligence and Design (ISCID).*doi: 10.1109/ISCID.2015.259

Xue, S., Jing, X., Sun, S. & Huang, H. 2014. Binary-Decision-Tree-Based Multiclass Support Vector Machines. *2014 14th International Symposium on Communications and Information Technologies (ISCIT).* doi: 10.1109/ISCIT.2014.7011875.

Wang, Y.-C. F. & Casasent, D. 2006. Hierarchical K-means Clustering Using New Support Vector Machines for Multi-class Classification. *2006 International Joint Conference on Neural Networks Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada July 16-21, 2006.*

Wu, T. 2009. Practical Guide to Support Vector Machines. MPLAB, UCSD. Retrieved from : http://tdlc.ucsd.edu/events/boot_camp_2009/tingfansvm.pdf

Zisserman, A. 2015. Lecture 2: The SVM classifier. Information Engineering, Department of Engineering Science, University of Oxford. http://www.robots.ox.ac.uk/~az/lectures/ml/lect2.pdf