Optimality of Feature set for Intrusion Detection System

Lucas Valentino Nainggolan*
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
Email: {*13218042}@students.itb.ac.id

***Abstract*—Sistem deteksi intrusi (IDS) menjadi tidak terpisahkan dengan kebutuhan keamanan yang muncul dalam sistem jaringan komputer. Teknik deteksi konvensional, seperti tanda tangan dan deteksi intrusi berbasis aturan, memerlukan intervensi manusia secara teratur atau membiarkan instrusi tidak terdeteksi. Untungnya, deteksi melalui *Machine Learning* (ML) bebas dari kekurangan tersebut. Namun, pemilihan fitur yang paling signifikan dan prediktif merupakan tantangan. Komunitas riset cukup aktif dalam pemilihan subset fitur terbaik di IDS. Namun, ada kekurangan prosedur pemilihan terstruktur dan daftar fitur yang berurutan. Kami berusaha memberikan daftar fitur yang lebih konkret mengenai signifikansinya dalam memprediksi intrusi. Kami melakukan survei dan mengikuti metodologi terstruktur dalam pemilihan fitur dari kumpulan data NSL-KDD yang dipublikasikan. Prosedur pemilihan fitur terdiri dari lima langkah. Tiga langkah pertama didedikasikan untuk menghilangkan fitur-fitur sepele, sedangkan dua langkah terakhir dilakukan untuk mengidentifikasi fitur-fitur yang berguna. Pembuatan model dilakukan dengan menggunakan *Support Vector Machine* (SVM) agar dapat diterima secara luas di komunitas riset IDS. Hasilnya terdiri dari daftar fitur yang diurutkan. Fitur diurutkan sesuai kemampuan prediktifnya dalam mengklasifikasikan lalu lintas jaringan berbahaya dan jinak.**

Keywords*—Pilihan subset fitur, sistem berbasis *Host*, *Intrusion detection system*, *Machine learning

I. Introduction

Intrusion Detection System (IDS) merupakan komponen penting dalam memastikan infrastruktur jaringan. Selama dekade terakhir, masalah keamanan mendapatkan fokus yang signifikan merujuk pada intrusi jaringan yang berlebihan seperti yang dilaporkan oleh *Computer Emergency Response Team* (CERT). Intrusi ini melahirkan bencana dan secara ekstensif melanggar keamanan, yaitu, *Confidentiality*, *Integrity*, dan *Availability* (CIA). Komunitas riset cukup aktif dalam menangani isu-isu terkait di IDS. Dengan munculnya persyaratan keamanan dalam sistem jaringan komputer, isu-isu yang berkaitan dengan mengatasi ancaman terhadap keamanan jaringan dan informasi bahkan lebih kandidat untuk ditangani. Meskipun ada sejumlah literatur yang ada untuk mensurvei IDS dan taksonominya [**?**], [**?**], [**?**], [**?**], [**?**], [**?**].

Seperti yang didefinisikan oleh *National Institute of Standards and Technology* (NIST) [**?**] Intrusi adalah upaya untuk kompromi salah satu atau semua faktor CIA. Sedangkan deteksi bertujuan untuk memantau adanya penyusupan. Hal yang sama berlaku untuk komunikasi kabel dan nirkabel. [**?**], [**?**]. IDS adalah perangkat lunak atau perangkat keras atau keduanya untuk secara otomatis melakukan deteksi intrusi dalam lalu lintas jaringan [**?**], [**?**]. Selain IDS, *intrusion prevision system* (IPS) berupaya mencegah terjadinya peristiwa intrusi. IPS bekerja dengan IDS dan komunitas menggunakannya secara bergantian. Namun, IPS berada di luar cakupan penelitian ini.

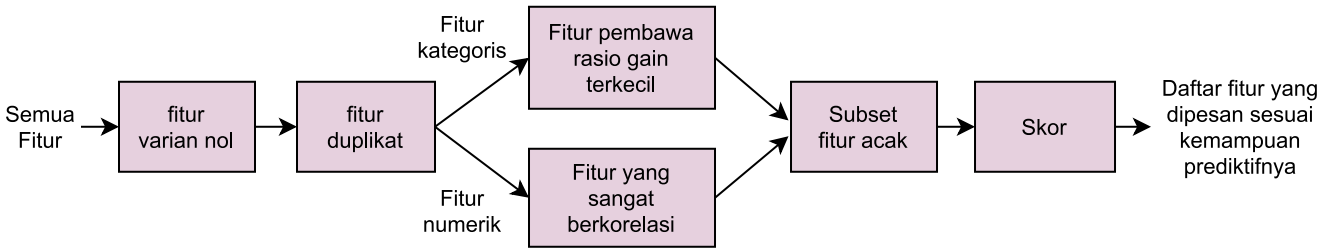
IDS secara luas dapat diklasifikasikan menjadi dua kelas; *Signature-based Detection* (SD), *Machine Learning-based Detection* (MLD). SD adalah metode yang relatif sederhana dan sama efektifnya untuk mendeteksi serangan atau ancaman yang diketahui. Proses membandingkan pola yang disimpan dengan kejadian yang sedang diuji. Namun, teknik ini tidak mampu mendeteksi serangan yang tidak diketahui, dengan persyaratan tambahan untuk menyimpan tanda tangan. Padahal, MLD cukup efektif dalam mengidentifikasi intrusi yang tidak diketahui, namun teknik ini tidak rentan terhadap *false positive* [**?**], [**?**], [**?**], [**?**].

Klasifikasi ortogonal lain dari IDS adalah deteksi intrusi berbasis *host* (HIDS) dan deteksi intrusi berbasis jaringan (NIDS). HIDS memvisualisasikan perilaku *host* yang berisi informasi sensitif. Sedangkan, NIDS menganalisis aktivitas aplikasi/protokol dan mengevaluasi segmen jaringan tertentu untuk mengidentifikasi kejadian yang tidak dapat dipercaya.

Pencegahan positif palsu (FP) dan negatif palsu (FN) sambil menjaga tingkat akurasi yang optimal adalah tantangan utama dari setiap sistem IDS. FP terjadi ketika IDS salah mengklasifikasikan aktivitas yang sah sebagai berbahaya, sedangkan FN terjadi ketika IDS tidak dapat mengenali kejadian buruk. [**?**], [**?**], [**?**]. Baru-baru ini, Ho dkk. [**?**] kumpulkan kasus *False Positive* (FP) dan *False Negative* (FN) dari lalu lintas dunia nyata.

Sayangnya, SD mencakup serangan yang jauh lebih sedikit daripada jumlah alarm palsu yang muncul. Hal ini memungkinkan intrusi nyata tidak terdeteksi. Selain itu, kebisingan tambahan dapat sangat menurunkan kinerja SD. Pemeliharaan dan pembaruan selalu diperlukan untuk SD agar tanda tangan selalu terbarui dengan ancaman baru. Selain itu serangan berbasis Protokol dapat menyebabkan IDS berbasis SD gagal.

Untungnya, di sinilah IDS berbasis ML berperan. IDS berbasis ML memungkinkan komputer untuk berlatih dan bertindak tanpa bantuan manusia. Ini ”mempelajari” pola spesifik jaringan dan dapat menentukan adanya intrusi. Ia mampu



Gambar 1. Metodologi pemilihan fitur

mengidentifikasi jenis intrusi yang tidak diketahui sebelumnya, sehingga rentan terhadap alarm palsu juga. Kami mengajarkan model ML untuk membedakan antara paket jinak dan paket berbahaya dalam lalu lintas. Selain itu, ia dapat menangani serangan yang dipelajari sebelumnya dengan cukup efektif dan juga dapat mengenali mutan dari serangan yang diketahui ini. Dengan memperhatikan kelebihan di atas, banyak penelitian telah melakukan deteksi intrusi menggunakan ML, oleh karena itu mendukung kelayakan IDS berbasis ML. Namun, pemilihan fitur yang paling signifikan dan prediktif merupakan tantangan. Yang merupakan masalah utama yang dibahas dalam makalah ini.

Sisa makalah ini disusun sebagai berikut: Bagian II menyajikan tinjauan pustaka bidang ini diikuti oleh Bagian III yang menguraikan proses pemilihan dan penyaringan fitur. Akhirnya, kesimpulan dan kemungkinan masalah yang akan dibahas di masa depan diuraikan dalam Bagian IV.

II. Literature Review

Komunitas riset cukup aktif dalam mengidentifikasi/memilih fitur terbaik untuk meningkatkan IDS. Namun, ada kekurangan prosedur seleksi terstruktur dan daftar fitur yang teratur. Di sini kami membahas secara singkat beberapa penelitian yang berfokus pada pemilihan subset fitur terbaik di IDS.

Jain dkk. [**?**] melaporkan bahwa dalam kumpulan data besar, pendekatan berurutan maju berkinerja lebih baik. Padahal, penelitian lain yang dilakukan oleh Kudo dan Sklansky [**?**] menemukan efektivitas metode acak. Ketika datang ke pemilihan fitur skala besar, metode ini cukup mahal secara komputasi. Untuk kriteria evaluasi, *Support Vector Machine* (SVM) dihargai karena mengatasi minimal lokal. Weston [**?**] menerapkan SVM untuk pemilihan fitur berdasarkan penemuan fitur yang meminimalkan batasan pada kesalahan *leave-one-out*. Setelah itu, Guyon dkk. [**?**] menggunakan SVM berdasarkan Penghapusan Fitur Rekursif. Secara khusus, di IDS, Vapnik [**?**] melaporkan kinerja yang lebih baik dari SVM untuk akselerasinya dalam pembuatan model dan pelaporan. [**?**] telah memperkenalkan pendekatan berbasis *wrapper* untuk pemilihan fitur untuk membuat IDS yang ringan. Mereka telah memodifikasi mutasi acak *hill climbing* untuk mencari subset dan memodifikasi SVM sebagai pendekatan *wrapper* untuk mengekstrak subset fitur yang dioptimalkan. Mereka telah melakukan eksperimen pada dataset KDD Cup '99. [**?**] menggunakan Algoritma Genetika dan SVM untuk seleksi fitur. Tujuan mereka adalah untuk mengurangi dimensi data, meningkatkan tingkat positif benar, dan secara bersamaan mengurangi tingkat positif palsu. Algoritme genetika memberikan fitur *fitness score* berdasarkan subset fitur optimal yang dibangun. [**?**] juga telah menggunakan SVM untuk mengidentifikasi fungsi diskriminan untuk pengurangan fitur. [**?**] mengusulkan IDS berbasis *gain* yang dinormalisasi yang terdiri dari dua modul. Modul pertama merupakan pemilihan fitur optimal menggunakan *Particle Swarm Optimization* (PSO) dan modul kedua menggunakan *support vector machine* untuk deteksi dan klasifikasi ancaman. Modul pertama melakukan pemeringkatan fitur menggunakan *gain* yang dinormalisasi dan kemudian mengekstraksi fitur yang paling signifikan menggunakan pengelompokan semi-diawasi.

[**?**] telah memperkenalkan algoritma berbasis informasi timbal balik untuk memilih fitur yang optimal. Fitur utama dari makalah ini adalah bahwa ia berurusan dengan fitur data yang berkorelasi linier dan non-linier. Mereka telah melakukan evaluasi pada dataset KDDCup'99, NSL-KDD, dan CIC-IDS2017 [**?**] telah mengusulkan metode pemilihan fitur berbobot untuk meningkatkan kinerja IDS. Pendekatan yang diusulkan menggunakan SVM, ANN, dan DT untuk seleksi fitur dan ANN untuk klasifikasi. Algoritme ML menetapkan bobot heuristik ke fitur berdasarkan kontribusinya dan subset fitur yang optimal tercapai. [**?**] juga menggunakan pendekatan yang sama seperti [**?**] menggunakan algoritma genetika dan SVM. Senthilnayaki dkk. [**?**] memberikan wawasan terperinci tentang pekerjaan sebelumnya yang dilakukan pada pemilihan fitur dan pendekatannya. Makalah ini membahas teknik pemilihan fitur berbasis filter dan teknik pemilihan fitur berbasis *wrapper* dan hibrida. Penulis juga mengidentifikasi masalah kumpulan data KDD yang digunakan dalam 26 dari 28 makalah yang mereka periksa dan mendesak untuk menggunakan kumpulan data baru untuk serangan modern.

Alazab dkk. [**?**] memberi peringkat fitur pada teknik berbasis filter dengan perolehan informasi. Menggunakan pohon keputusan untuk klasifikasi mereka telah mengurangi jumlah fitur menjadi 12 dari 41 dan karenanya mencapai peningkatan kinerja yang luar biasa.

Khor dkk. [**?**] melakukan dua jenis teknik pemilihan fitur yang berbeda menggunakan dataset KDD 99. Mereka menggunakan

Consistency Subset Evaluator dan *Correlation* untuk *Feature Selection Subset Evaluator* untuk mengurangi fitur dari 41 menjadi 8.

Zhou dkk. [**?**] mengusulkan *Correlation-based-Feature-Selection-Bat-Algorithm* untuk mengatasi tantangan data yang tidak relevan dan berlebihan. Ini adalah algoritma heuristik yang mengevaluasi korelasi antara fitur dan mengekstrak subset fitur yang optimal. Penulis menggunakan set data KDD-Cup'99, NSL-KDD, dan CIC-IDS2017 untuk evaluasi.

Mengingat studi di atas, disimpulkan bahwa ada kekurangan prosedur seleksi terstruktur dan daftar fitur yang teratur. Di bagian selanjutnya kami melakukan prosedur pemilihan fitur yang terstruktur dengan tepat dan akhirnya memberikan subset fitur terbaik untuk IDS.

III. Experimentation

A. NSL KDD

Kami memilih dataset NSL KDD untuk percobaan kami [**?**]. Dataset NSL KDD dirancang untuk mengatasi masalah yang ditemukan dalam dataset KDD'99. Seperti dalam studi besar penelitian akademis, kami menggunakan dataset NSL-KDD yang sekarang menjadi data *benchmark standar de facto*. Dataset umumnya digunakan untuk meningkatkan efektivitas tingkat deteksi intrusi. Melalui tujuan dataset adalah untuk membangun model yang efisien dan efektif untuk deteksi intrusi, namun dataset digunakan oleh studi yang sering untuk DL, dan komunitas data mining untuk tujuan validasi juga.

Ini dapat dengan aman digunakan sebagai tolok ukur untuk pelatihan dan evaluasi model IDS. Juga, ada cukup catatan yang tersedia untuk pelatihan dan pengujian sehingga tidak perlu menggunakan teknik untuk menghasilkan catatan yang disintesis yang entah bagaimana mempengaruhi hasil. Dataset sering digunakan oleh komunitas riset [**?**], [**?**], [**?**], [**?**], [**?**], [**?**], [**?**], [**?**], [**?**], [**?**].

Dataset terdiri dari 185559 *record* dengan 92904 *instance* anomali. Oleh karena itu, dataset hampir seimbang. Dataset memiliki 42 fitur sekaligus, 11 adalah kategorikal dan sisanya semuanya numerik. Di antara kategoris, lima fitur adalah biner-nominal. Semantik dataset dapat ditemukan melalui dokumentasinya.

B. Metodologi seleksi

Metodologi pemilihan fitur terdiri dari enam fase yang ditunjukkan pada Gambar 1.

1) First phase:: Pada fase pertama, kami menghapus fitur yang tidak memiliki perubahan di seluruh instans 185559. Kami menemukan satu fitur tersebut yaitu Tidak ada perintah keluar.

2) Second phase:: Pada fase kedua kami mencari fitur duplikat. Meskipun fase ini tidak menghilangkan fitur lebih lanjut, namun fase ini layak untuk dilakukan sebelum menerapkan teknik ML apa pun.

3) Third phase:: Dalam fase ini kami melakukan dua eksperimen, satu khusus untuk fitur kategoris dari dataset sementara yang lain khusus untuk fitur numerik dari dataset. Oleh karena itu, kami membagi dataset menjadi dua. Yang pertama terdiri dari fitur kategoris sedangkan yang kedua terdiri dari fitur numerik. Kedua set data memiliki label yang sesuai juga.

Dalam kumpulan data pembawa fitur kategorikal, kami menemukan kumpulan fitur kategoris yang paling tidak penting dengan menghitung *rasio-gain* terhadap Label. Enam fitur tersebut ditemukan yaitu sebagai berikut;

- *land*
- *root shell*
- *is host login*
- *is guest login*
- *su attempted*
- *urgent*

Fitur-fitur numerik dataset pembawa dianalisis dengan korelasi satu sama lain. Kami melakukan korelasi peringkat *Pearson* dan mengidentifikasi fitur yang memiliki korelasi 0,9 ke atas. Tabel I menunjukkan daftar fitur yang berkorelasi kuat.

Tabel I
FITUR YANG SANGAT BERKORELASI

Fitur Pertama	Fitur Kedua	Korelasi
No of compromised	No of root	1
server rate	server server rate	0.99
server rate	destination host server rate	0.97
server rate	destination host server server rate	0.97
server server rate	destination host server rate	0.97
server server rate	destination host server server rate	0.98
error rate	server error rate	0.98
error rate	destination host error rate	0.91
error rate	destination host server error rate	0.95
server error rate	destination host server error rate	0.96
destination host server rate	destination host server server rate	0.98
destination host error rate	destination host server error rate	0.91

Dengan memperhatikan hal ini, kami menurunkan tujuh fitur berikut;

- *No of compromised*
- *server rate*
- *server server rate*
- *error rate*
- *server error rate*
- *destination host server rate*
- *destination host error rate*

Setelah menyelesaikan proses di atas, kami menggabungkan kembali kedua kumpulan data yang telah dibagi pada awal fase ketiga.

Setelah menyelesaikan tiga fase di atas, kita memiliki 28 fitur, di mana lima adalah fitur kategoris, sedangkan 23 fitur lainnya bertipe numerik. Deskripsi statistik singkat dari fitur numerik dan kategoris masing-masing ditunjukkan pada Tabel II dan III.

Tabel II: Deskripsi statistik kumpulan fitur numerik dalam kumpulan data NSL-KDD

Fitur	mean	std	min	25%	50%	75%	max	Std
duration	289.5	2462.2	0	0	0	0	57715	2462.2
source bytes	4.E+4	5.E+6	0	0	44	276	1.E+9	5.E+6
destination bytes	14230	3.E+6	0	0	0	480	1.E+8	3.E+6
hot	0.2	2	0	0	0	0	101	2
No of failed logins	0	0.1	0	0	0	0	5	0.1
No of root	0.3	20.9	0	0	0	0	7468	20.9
No of file creations	0	0.6	0	0	0	0	100	0.6
No of shells	0	0	0	0	0	0	5	0
No of access files	0	0.1	0	0	0	0	9	0.1
No of outbound commands	0	0	0	0	0	0	0	0
count	84.2	119.2	0	2	13	141	511	119.2
server count	29.5	78.7	0	2	7	17	511	78.7
same server rate	0.7	0.4	0	0.1	1	1	1	0.4
diff server rate	0.1	0.2	0	0	0	0.06	1	0.2
server diff host rate	0.1	0.3	0	0	0	0	1	0.3
destination host count	185.7	97.9	0	92	255	255	255	97.9
destination host server count	118.7	110.9	0	11	70	255	255	110.9
destination host same server rate	0.5	0.4	0	0.5	0.58	255	255	0.4
destination host diff server rate	0.1	0.2	0	0	0.02	0.07	1	0.2
destination host same source port rate	0.2	0.3	0	0	0	0.06	1	0.3
destination host server diff host rate	0	0.1	0	0	0	0.01	1	0.1
destination host server server rate	0.2	0.4	0	0	0	0.31	1	0.4
destination host server error rate	0.1	0.3	0	0	0	0	1	0.3

Tabel III: Deskripsi statistik dari kumpulan fitur kategori dalam kumpulan data NSL-KDD

Fitur	Unik	Top	Frekuensi
logged in	2	0	112794
protocol type	3	tcp	150727
wrong fragment	3	0	184045
flag	11	SF	112071
service	70	http	57594

4) Pilihan subset fitur acak: Untuk menemukan subset terbaik dari fitur, *wrapper* bisa menjadi pilihan. Namun, terkadang tidak praktis karena biaya komputasi yang tinggi [**?**]. Dalam kasus kami ketika kami hanya memiliki 23 fitur, kami perlu mengevaluasi 8388608 himpunan bagian, yang tidak layak. Oleh karena itu, kami secara acak menghasilkan 5000 subset dan pindah ke fase berikutnya.

5) Pembuatan model dan evaluasi hasil: SVM digunakan untuk membangun model dan model dievaluasi menggunakan AuC. 5000 himpunan bagian yang dihasilkan pada fase terakhir, digunakan sebagai variabel independen. Setelah itu kita filter subset yang berhasil mencapai AuC sebesar 0.75. Gambar 2 menunjukkan daftar subset fitur terbaik.

C. Komputasi skor

Bagian akhir terdiri dari penilaian setiap fitur. Skoring dilakukan dengan menggunakan persamaan 1.

$$Score(F) = \sum_{i=1}^{n} \frac{AccompaningF_i}{AUC_i} \text{ x } PresenceOfF_i \quad (1)$$

Gambar 3 menunjukkan daftar fitur yang diurutkan berdasarkan urutan signifikansinya dalam prediksi anomali dan lalu lintas jaringan normal.

IV. Kesimpulan Dan Pekerjaan Masa Depan

Dalam makalah ini, kami memasukkan banyak penelitian tentang IDS saat ini. Lebih khusus lagi, pendekatan ML dan pemilihan fitur untuk meningkatkan pendekatan ML di IDS. Kami juga menguraikan pendekatan yang berbeda memiliki manfaat dan tantangan, sehingga kami harus berhati-hati dalam pemilihan pendekatan ML. Namun, masih ada kerangka kerja yang optimal untuk meningkatkan pemilihan fitur sistem IDS. Oleh karena itu, kami mengusulkan kerangka baru pemilihan fitur. Kerangka yang diusulkan sederhana untuk diimplementasikan. Kami secara empiris mengevaluasi kerangka kerja yang diusulkan pada dataset NSL-KDD. Hasil menunjukkan bahwa *No of failed logins* adalah fitur terbaik diikuti oleh *logged in* dan *flag*. Selain itu, model komputasi dan *time persist* (waktu bertahan) dapat dibangun dengan menambahkan lebih banyak fitur sesuai dengan urutan kepentingannya. Sekarang fitur yang dipilih dapat dimanfaatkan oleh komunitas pengembangan IDS untuk meningkatkan deteksi intrusi melalui ML.

Pada artikel ini, kami memberikan tren dan kecenderungan keseluruhan komunitas IDS saat ini menggunakan ML. Na-

Fitur	Kehadiran fitur yang sesuai dalam daftar variabel independen												Frek.
logged in		■			■					■		■	4
protocol type								■					1
wrong fragment							■						1
flag		■		■								■	3
service			■				■						2
duration						■							1
Source bytes	■								■				2
destination bytes		■		■								■	3
hot							■						1
No failed logins			■	■			■			■			4
No root			■				■						2
No file creations		■	■					■					3
No shells					■		■				■		3
No access files					■					■			2
No outbound cmds					■					■		■	3
count				■									1
server count						■			■		■		3
same server rate	■									■			2
diff server rate			■									■	2
server diff host rate						■		■					2
destination host count									■			■	2
destination host server count				■								■	2
destination host same server rate												■	1
destination host diff server rate		■				■					■		3
destination host same source port rate	■										■		2
destination host server diff host rate									■		■	■	3
destination host server error rate				■									1
destination host server rerror rate								■					1
Number of features	3	5	5	6	4	4	6	4	4	5	5	9	
AuC	**0.9**	**0.95**	**0.92**	**0.95**	**0.9**	**0.89**	**0.93**	**0.75**	**0.923**	**0.91**	**0.93**	**0.9**	
	0.30	0.32	0.31	0.32	0.30	0.30	0.31	0.25	0.31	0.30	0.31	0.30	

Gambar 2: AUC dihitung di seluruh subset fitur bersama dengan frekuensi fitur yang sesuai

Features	Score
No failed logins	1.237
logged in	1.220
flag	0.933
destination bytes	0.933
destination host diff server rat	0.923
No shells	0.920
destination host server diff ho	0.918
server count	0.914
No outbound cmds	0.903
No file creations	0.873
service	0.617
No root	0.617
destination host server count	0.617
destination host same Source	0.610
Source bytes	0.608
destination host count	0.608
diff server rate	0.607
No access files	0.603
same server rate	0.603
server diff host rate	0.547
count	0.317
destination host server error ra	0.317
wrong fragment	0.310
hot	0.310
destination host same server ra	0.300
duration	0.297
protocol type	0.250
destination host server rerror r	0.250

Gambar 3: Daftar fitur dengan urutan penurunan kepentingannya

mun, masih banyak masalah dan tantangan terbuka yang harus diatasi di masa depan. Mempertimbangkan kumpulan data lain dengan variasi pengklasifikasi mungkin merupakan pekerjaan yang bijaksana untuk dilakukan.