

KLASIFIKASI PENGGUNAAN PROTOKOL KOMUNIKASI PADA *NETWORK TRAFFIC* MENGGUNAKAN *NAÏVE BAYES* SEBAGAI PENENTUAN *QoS*

Made Sudarma

Staff Pengajar Magister Teknik Elektro, Manajemen
Sistem Informasi dan Komputer Universitas Udayana.
Email : debronxs@yahoo.co.id

Dandy Pramana Hostiadi

Mahasiswa Magister Teknik Elektro, Manajemen
Sistem Informasi dan Komputer, Universitas Udayana .
Email : dan_ganx_cil@yahoo.co.id

Abstrak — Penggunaan model komunikasi berbasis teknologi jaringan komputer merupakan teknologi yang sudah banyak digunakan. Model komunikasi pada jaringan komputer, menggunakan aturan komunikasi yang sesuai dengan standar tipe komunikasi yang sering dikenal dengan nama *network protocol*. Jumlah dari *port* yang mengidentifikasi *network protocol* menurut IANA adalah sejumlah 65.536 *port*. Penggunaan *network protocol* dalam komunikasi di jaringan komputer, terkadang menuntut adanya prioritas komunikasi yang sering dikenal dengan *QoS (Quality of Service)*. Prioritas komunikasi yang dimaksud atas dasar banyaknya penggunaan model komunikasi yang menggunakan *network protocol* seperti sering user menggunakan *protocol HTTP*. Analisa terhadap suatu trafik jaringan komputer merupakan salah satu cara mengetahui penggunaan *protocol* komunikasi jaringan komputer, sehingga dapat menjadi dasar penentuan prioritas (*QoS*). Penggunaan *Naïve Bayes* digunakan untuk melakukan klasifikasi terhadap *protocol* komunikasi pada jaringan komputer. Penggunaan aplikasi *wireshark* digunakan sebagai *tools network capture* dan hasil yang dicapai adalah mengklasifikasikan *protocol* jaringan yang digunakan sebagai acuan penentuan *QoS*.

Kata kunci — *Network protocol, QoS, network capture*

I. PENDAHULUAN

Klasifikasi yang tepat terhadap sebuah trafik internet sangat penting dilakukan terutama dalam hal desain perancangan arsitektur jaringan, manajemen jaringan dan keamanan jaringan [1]. Klasifikasi yang dilakukan adalah berdasarkan atas banyaknya tipe aktifitas komunikasi. Aktifitas komunikasi dalam jaringan komputer diatur dalam proses komunikasi menggunakan *network protocol*. Perkembangan *network protocol* yang diidentifikasi ke dalam *port number* dikembangkan berdasarkan penggunaan model komunikasi yang terus berkembang dan terstandarisasi secara internasional. Jumlah *port number* yang ditentukan dalam proses komunikasi adalah 65.536 [2].

Banyaknya penggunaan *network protocol* dalam suatu komunikasi terkadang menuntut adanya penggunaan prioritas komunikasi seperti kualitas *throughput, delay time, realability* dan keamanan komunikasi [3]. Penggunaan *service* prioritas sering disebut dengan istilah *QoS*. Dasar pemberian prioritas *QoS* adalah dengan penganalisaan terhadap *Network traffic*. *Network traffic* atau *Internet Traffic* adalah lalu lintas komunikasi data dalam jaringan yang ditandai dengan satu set aliran statistik dengan penerapan pola terstruktur [4]. Pola terstruktur yang dimaksud adalah informasi dari header data informasi komunikasi. Aplikasi *wireshark* merupakan aplikasi yang handal dalam hal *capture network traffic* [5]. Hasil *capture network traffic* terdiri atas record lalu lintas (*network traffic*) dari transaksi komunikasi yang berjalan dalam jaringan komputer.

Penggunaan metode *Naïve Bayes* dalam penelitian yang dilakukan digunakan sebagai metode pengklasifiasian terhadap *network traffic*. Hasil proses klasifikasi akan dijadikan sebagai acuan dalam penentuan pemberian *QoS* terhadap *Network protocol* yang sering digunakan dalam komunikasi jaringan.

II. LANDASAN TEORI

A. *Naïve Bayes Classification*

Metode Bayes merupakan pendekatan statistik untuk melakukan inferensi induksi pada persoalan klasifikasi. Metode ini menggunakan probabilitas bersyarat sebagai dasarnya. Metode Klasifikasi *Naïve Bayes* didasarkan pada teorema Bayes, dengan asumsi bahwa efek dari nilai atribut pada kelas tertentu tidak tergantung pada nilai dari atribut lainnya. asumsi ini sering dikatakan sebagai “model fitur independen”.

Probabilitas untuk klasifikasi model kondisional adalah sebagai berikut:

$$P(C|F_1, \dots, F_n) \quad (1)$$

Kondisional di atas merupakan variabel kelas dependen C dengan sejumlah kecil hasil atau kelas, tergantung pada beberapa variabel fitur F_1 sampai F_n . Sehingga penulisan teori Bayes adalah :

$$P(C|F_1, \dots, F_n) = \frac{p(C)p(F_1, \dots, F_n|C)}{p(F_1, \dots, F_n)} \quad (2)$$

Dalam bahasa Inggris persamaan di atas dapat ditulis sebagai :

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}} \quad (3)$$

Asumsi kemandirian bersyarat yang "naif" memegang peranan. Menganggap bahwa setiap fitur F_i adalah secara kondisi independen terhadap setiap fitur lainnya F_j untuk $j \neq i$. Ini berarti bahwa :

$$P(F_i | C, F_j) = p(F_i | C) \quad (4)$$

untuk $i \neq j$, sehingga joint model dapat dinyatakan sebagai :

$$\begin{aligned} p(C|F_1, \dots, F_n) &= p(C) p(F_1|C) p(F_2|C) p(F_3|C) \dots \\ &= p(C) \prod_{i=1}^n p(F_i|C) \end{aligned}$$

Ini berarti bahwa di bawah asumsi independen di atas, distribusi bersyarat dari variabel kelas C dapat dinyatakan seperti ini :

$$P(C|F_1, \dots, F_n) = \frac{1}{Z} p(C) \prod_{i=1}^n p(F_i|C) \quad (5)$$

dimana Z (bukti) adalah faktor skala tergantung hanya pada F_1, \dots, F_n , yaitu, sebuah konstanta jika nilai dari variabel fitur diketahui.

Model dari bentuk ini jauh lebih mudah dikelola, karena mereka memecah menjadi class prior $p(C)$ dan distribusi probabilitas independen $p(F_i|C)$. Jika ada k kelas dan jika model untuk masing-masing $p(F_i|C = c)$ dapat dinyatakan dalam bentuk parameter, maka model naif Bayes yang sesuai memiliki $(k - 1) + n \times k$ parameter. Dalam prakteknya, sering $k = 2$ (klasifikasi biner) dan $r = 1$ (variabel Bernoulli sebagai fitur) yang umum, sehingga jumlah parameter model *Naive Bayes* adalah $2n + 1$, dimana n adalah jumlah fitur biner yang digunakan untuk klasifikasi dan prediksi.

B. QoS

QoS dalam Jaringan Komputer merujuk kepada pemberian kualitas prioritas terhadap proses komunikasi yang berlangsung dalam jaringan komputer. Pemberian kualitas berdasarkan penentuan prioritas. Penggunaan QoS sering diimplementasikan pada *protocol* komunikasi dalam jaringan komputer yang sering disebut dengan *Network protocol*. Umumnya tujuan akhir dari QoS adalah memberikan network service yang lebih baik dan terencana dengan

dedicated bandwidth, *jitter* dan *latency* yang terkontrol dan meningkatkan *loss* karakteristik

C. Network traffic

Pengukuran dan analisis lalu lintas jaringan penting dilakukan untuk mendapatkan pengetahuan tentang karakteristik lalu lintas jaringan [6]. Umumnya sebuah data *network traffic* memiliki informasi seperti :

- *IP Address*

IP Address sering dikenal sebagai alamat komputer. Alamat komputer ini berfungsi sebagai identitas komputer dalam sebuah komunikasi jaringan. Alamat ini terbagi menjadi dua bagian yaitu sebagai identitas *source* dan *destination*. *IP Source Address* adalah alamat asal yang dapat diidentifikasi dengan si pengirim pada proses komunikasi data berlangsung. Sedangkan *IP Destination Address* adalah alamat tujuan yang dapat diidentifikasi dengan si penerima data pada proses komunikasi data.

- *Protocol*

Protocol adalah aturan yang diberlakukan dalam proses komunikasi data yang berjalannya diidentifikasi berdasarkan jenis layanannya. Setiap *protocol* yang berjalan akan dinamai sesuai dengan proses yang dilakukan dalam proses komunikasi di jaringan komputer. Contoh *protocol* komunikasi adalah *protocol tcp, udp, http, ftp, icmp*, dll

- *Length*

Length adalah ukuran besar data yang berjalan dalam jaringan komputer. Ukuran yang umumnya digunakan dalam trafik di jaringan adalah dalam ukuran *byte*.

Network traffic sendiri dapat ditampilkan dalam bentuk data mentah (data dalam bentuk *record* trafik seperti hasil) atau dalam bentuk jadi (sudah dalam bentuk grafik).

D. Wireshark

Wireshark adalah salah satu dari sekian banyak *tools Network Analyzer* yang banyak digunakan oleh *Network administrator* untuk menganalisa kinerja jaringannya. *Wireshark* banyak disukai karena *interface*-nya yang menggunakan *Graphical User Interface (GUI)* atau tampilan grafis. *Wireshark* digunakan untuk *troubleshooting* jaringan, analisis, perangkat lunak dan pengembangan protokol komunikasi, dan pendidikan. *Wireshark* banyak digunakan oleh *network administrator* untuk menganalisa kinerja jaringannya. *Wireshark* mampu menangkap data/informasi yang melewati suatu jaringan yang kita amati dalam bentuk *network traffic*. Manfaat dari penggunaan aplikasi *Wireshark* ini yaitu sebagai berikut :

- Menangkap informasi atau data paket yang dikirim dan diterima dalam jaringan komputer

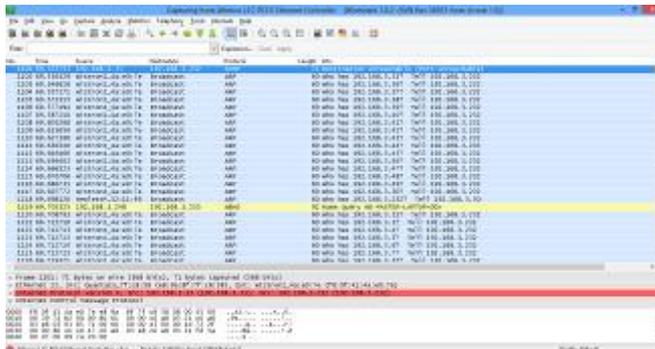
- Mengetahui aktivitas yang terjadi dalam jaringan komputer
- Mengetahui dan menganalisa kinerja jaringan komputer yang kita miliki seperti kecepatan akses/share data dan koneksi jaringan ke internet
- Mengamati keamanan dari jaringan komputer yang kita miliki

Beberapa informasi yang dapat di *capture* oleh *tools wireshark* sebagai informasi *network traffic* antara lain *time elapse* (adalah waktu yang dicatat dalam periode tertentu), *source address* (alamat asal dari pengirim data, dapat berupa *IP Address* ataupun *mac address*), *destination address* (alamat tujuan dari pengiriman data, dapat berupa *IP Address* ataupun *mac address*), *protocol* (layanan atau service yang berjalan dalam jaringan komputer), *length* (adalah ukuran data yang dikirimkan), dan *info* (adalah informasi tambahan dari tiap layanan yang berjalan dalam jaringan komputer).

III. METODELOGI

A. Network Capturing

Pengambilan *Network traffic* menggunakan aplikasi *wireshark*. *Capture Network traffic* dilakukan dengan meng-*capture traffic*. Pengambilan *network traffic* yang dilakukan menghasilkan kurang lebih hingga puluhan juta *record traffic*. Namun jumlah *record* yang dihasilkan tiap harinya tidak sama. Ketidak samaan jumlah *record traffic* disebabkan karena ketidaksamaan model komunikasi dalam jaringan komputer yang dilakukan oleh user. Model *capture Network traffic* dari *wireshark* adalah seperti pada gambar berikut :



Gambar 1. Capturing Network traffic

B. Data Filtering

Data Filter dilakukan dengan memilih data yang akan digunakan sebagai proses penghitungan klasifikasi. Format *record* dari hasil *network capture* berupa data mentah diolah dan difilter menggunakan Microsoft SQL Server 2008. *Counting* dilakukan untuk menghitung *record traffic* yang sama. Pada data filter, *field* yang dihitung nanti menggunakan metode *Naive Bayes* hanya menggunakan field *Protocol*,

Length dan jumlah *counting*. Dari informasi tiga field diatas, dapat dijadikan acuan dalam penentuan *QoS* dengan menganalisa jumlah *counting*.

C. Klasifikasi Naive Bayes

Data *network traffic* yang dipetakan dalam metode *Naive Bayes* sebagai kelas klasifikasi adalah sebagai berikut :

- Jenis Protokol dalam label kelas adalah jenis *ARP, DHCP, DHCPv6, DNS, HTTP, ICMP, ICMPv6, IGMP, MDNS, MNDP, NBNS, NTP, SSDP, SSHv2, TCP*.
- *Length Range* adalah 0 - 32, 33 - 64, 65 - 128, 129 - 256, 257 - 512, 513 - 1024, 1025 - 2048
- *Counting Range* adalah 0 - 500, 501 - 1000, 1001 - 1500, 1501 - 2000, 2001 - 2500, 2501 - 3000, x > 3000.
- Prioritas *QoS* sebagai penentuan hasil *network traffic* adalah prioritas rendah, menengah dan Tinggi.

Jumlah klasifikasi yang merepresentasikan klasifikasi berdasarkan prioritas adalah sebanyak 105 kelas. Berikut merupakan contoh kelas label berdasarkan prioritas

TABEL I. TABEL CONTOH KELAS LABEL

Protocol	Lenth Range (byte)	Counting	Prioritas
ARP	0 - 32	0 - 500	Menengah
ARP	33 - 64	501 - 1000	Menengah
DHCP	257 - 512	2001 - 2500	Menengah
DHCPv6	513 - 1024	2501 - 3000	Menengah
DHCPv6	1025 - 2048	x > 3000	Menengah
DNS	0 - 32	0 - 500	Menengah
HTTP	257 - 512	2001 - 2500	Tinggi
HTTP	513 - 1024	2501 - 3000	Tinggi
HTTP	1025 - 2048	x > 3000	Tinggi
ICMP	0 - 32	0 - 500	Menengah
ICMPv6	513 - 1024	2501 - 3000	Rendah
ICMPv6	1025 - 2048	x > 3000	Rendah
IGMP	0 - 32	0 - 500	Menengah
IGMP	33 - 64	501 - 1000	Menengah
MDNS	1025 - 2048	x > 3000	Menengah
MNDP	0 - 32	0 - 500	Rendah
NBNS	1025 - 2048	x > 3000	Menengah
NTP	0 - 32	0 - 500	Menengah
SSHv2	513 - 1024	2501 - 3000	Menengah
TCP	513 - 1024	2501 - 3000	Tinggi
TCP	1025 - 2048	x > 3000	Tinggi

Dari Kelas klasifikasi diatas dilakukan penghitungan *Naive Bayes* terhadap data sample contoh *record Network traffic*

TABEL II. TABEL SAMPLE DATA NETWORK TRAFFIC

Nama Protocol	Length	Counting	Prioritas
HTTP	1430	4788	?

Penghitungan data sample terhadap P adalah sebagai berikut :

$P(\text{Rendah}) : 33/105 = 0.3142857$

$P(\text{Menengah}) : 60/105 = 0.57142857$

$P(\text{Tinggi}) : 12/105 = 0.11428571$

Perhitungan P terhadap Nama Protocol :

Nilai P (HTTP | Rendah) : $2/33 = 0.06060606$

Nilai P (HTTP | Menengah) : $2/60 = 0.03333333$

Nilai P (HTTP | Tinggi) $3/12 = 0.25$

Perhitungan P terhadap Length :

Nilai P (1025 – 2048 | Rendah) : $4/33 = 0.12121212$

Nilai P (1025 – 2048 | Menengah) : $6/60 = 0.1$

Nilai P (1025 – 2048 | Tinggi) : $5/12 = 0.41666667$

Perhitungan P terhadap Counting :

Nilai P ($x > 3000$ | Rendah) : $4/33 = 0.12121212$

Nilai P ($x > 3000$ | Menengah) : $6/60 = 0.12121212$

Nilai P ($x > 3000$ | Tinggi) : $5/12 = 0.41666667$

Perhitungan Posterior Rendah

$0.3142857 \times 0.06060606 \times 0.12121212 \times 0.12121212 =$

0.000279854804193053

Perhitungan Posterior Menengah

$0.57142857 \times 0.03333333 \times 0.1 \times 0.12121212 =$

0.000230880204906205

Perhitungan Posterior Tinggi

$0.11428571 \times 0.25 \times 0.41666667 \times 0.41666667 =$

0.00496031735367063

Berdasarkan perhitungan nilai Posterior masing masing kelas prioritas terlihat bahwa Posterior Tinggi yang memiliki nilai tertinggi, maka hasil Perhitungan dengan metode *Naive Bayes* menghasilkan klasifikasi prioritas Tinggi untuk Data *Network traffic* pada table 2.

IV. PERCOBAAN DAN HASIL

Percobaan penghitungan dalam penelitian yang dilakukan adalah melakukan perhitungan terhadap 91 record *network traffic* yang sudah difilterisasi pada tahap *filtering* Data dimana pada data mentah jumlah record traffic yang belum di filter (menghilangkan duplikasi record) mencapai jutaan record. Hasil dari Klasifikasi ditunjukkan pada table berikut :

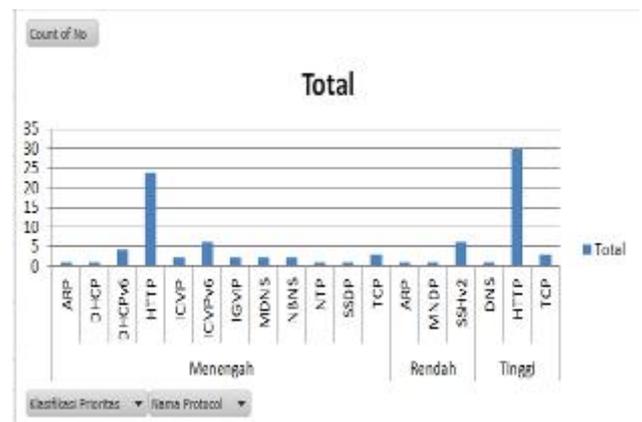
TABEL III. TABEL HASIL KLASIFIKASI

No	Nama Protocol	Length	Counting	Label Length	Label Counting	Klasifikasi Prioritas
1	ARP	60	130	33 - 64	0 - 500	Rendah
2	ARP	142	123	129 - 256	0 - 500	Menengah
3	DHCP	948	1844	513 - 1024	1501 - 2000	Menengah
4	DHCPv6	759	4770	513 - 1024	$x > 3000$	Menengah
5	DHCPv6	775	1040	513 - 1024	1001 - 1500	Menengah
6	DHCPv6	1315	519	1025 - 2048	501 - 1000	Menengah
7	DHCPv6	1530	2086	1025 - 2048	2001 - 2500	Menengah
8	DNS	1605	2027	1025 - 2048	2001 - 2500	Tinggi
9	HTTP	1084	2196	1025 - 2048	2001 - 2500	Tinggi
10	HTTP	769	2581	513 - 1024	2501 - 3000	Tinggi
11	HTTP	1041	4568	1025 - 2048	$x > 3000$	Tinggi
12	HTTP	1357	562	1025 - 2048	501 - 1000	Menengah
13	HTTP	635	4143	513 - 1024	$x > 3000$	Tinggi
14	HTTP	1646	1346	1025 - 2048	1001 - 1500	Menengah
15	HTTP	569	1038	513 - 1024	1001 - 1500	Menengah
16	HTTP	1430	4788	1025 - 2048	$x > 3000$	Tinggi
17	HTTP	1219	2405	1025 - 2048	2001 - 2500	Tinggi
18	HTTP	165	2536	129 - 256	2501 - 3000	Menengah
19	HTTP	479	1807	257 - 512	1501 - 2000	Menengah
20	HTTP	717	1044	513 - 1024	1001 - 1500	Menengah
21	HTTP	689	3812	513 - 1024	$x > 3000$	Tinggi
22	HTTP	1531	4635	1025 - 2048	$x > 3000$	Tinggi
23	HTTP	1588	4685	1025 - 2048	$x > 3000$	Tinggi
24	HTTP	59	3883	33 - 64	$x > 3000$	Menengah
25	HTTP	483	2136	257 - 512	2001 - 2500	Tinggi
26	HTTP	1439	2720	1025 - 2048	2501 - 3000	Tinggi
27	HTTP	1450	435	1025 - 2048	0 - 500	Menengah
28	HTTP	40	3120	33 - 64	$x > 3000$	Menengah
29	HTTP	1104	3224	1025 - 2048	$x > 3000$	Tinggi
30	HTTP	34	4403	33 - 64	$x > 3000$	Menengah

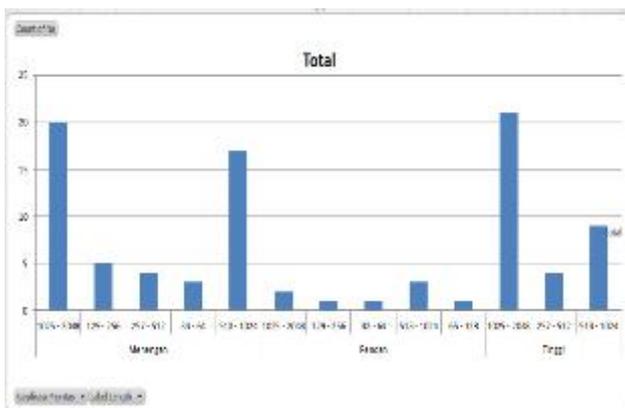
31	HTTP	558	333	513 - 1024	0 - 500	Menengah
32	HTTP	858	3429	513 - 1024	x > 3000	Tinggi
33	HTTP	550	3249	513 - 1024	x > 3000	Tinggi
34	HTTP	702	1876	513 - 1024	1501 - 2000	Menengah
35	HTTP	1590	2966	1025 - 2048	2501 - 3000	Tinggi
36	HTTP	157	3132	129 - 256	x > 3000	Menengah
37	HTTP	815	1180	513 - 1024	1001 - 1500	Menengah
38	HTTP	1289	1204	1025 - 2048	1001 - 1500	Menengah
39	HTTP	914	3353	513 - 1024	x > 3000	Tinggi
40	HTTP	1344	1637	1025 - 2048	1501 - 2000	Menengah
41	HTTP	1027	2997	1025 - 2048	2501 - 3000	Tinggi
42	HTTP	1334	1282	1025 - 2048	1001 - 1500	Menengah
43	HTTP	1675	893	1025 - 2048	501 - 1000	Menengah
44	HTTP	976	4362	513 - 1024	x > 3000	Tinggi
45	HTTP	153	525	129 - 256	501 - 1000	Menengah
46	HTTP	679	2252	513 - 1024	2001 - 2500	Tinggi
47	HTTP	1318	212	1025 - 2048	0 - 500	Menengah
48	HTTP	1659	2062	1025 - 2048	2001 - 2500	Tinggi
49	HTTP	379	3637	257 - 512	x > 3000	Tinggi
50	HTTP	1327	3089	1025 - 2048	x > 3000	Tinggi
51	HTTP	1633	512	1025 - 2048	501 - 1000	Menengah
52	HTTP	264	3711	257 - 512	x > 3000	Tinggi
53	HTTP	1474	799	1025 - 2048	501 - 1000	Menengah
54	HTTP	1529	3592	1025 - 2048	x > 3000	Tinggi
55	HTTP	768	4797	513 - 1024	x > 3000	Tinggi
56	HTTP	1588	2907	1025 - 2048	2501 - 3000	Tinggi
57	HTTP	631	848	513 - 1024	501 - 1000	Menengah
58	HTTP	1170	4863	1025 - 2048	x > 3000	Tinggi
59	HTTP	1086	2466	1025 - 2048	2001 - 2500	Tinggi
60	HTTP	949	1626	513 - 1024	1501 - 2000	Menengah
61	HTTP	1360	4746	1025 - 2048	x > 3000	Tinggi
62	HTTP	1542	4343	1025 - 2048	x > 3000	Tinggi
63	ICMP	577	870	513 - 1024	501 - 1000	Menengah
64	ICMP	1210	1828	1025 - 2048	1501 - 2000	Menengah
65	ICMPv6	236	2417	129 - 256	2001 - 2500	Menengah
66	ICMPv6	1202	984	1025 - 2048	501 - 1000	Menengah
67	ICMPv6	1521	1928	1025 - 2048	1501 - 2000	Menengah

68	ICMPv6	1379	4073	1025 - 2048	x > 3000	Menengah
69	ICMPv6	1116	2534	1025 - 2048	2501 - 3000	Menengah
70	ICMPv6	569	2866	513 - 1024	2501 - 3000	Menengah
71	IGMP	970	4839	513 - 1024	x > 3000	Menengah
72	IGMP	921	4967	513 - 1024	x > 3000	Menengah
73	MDNS	1144	1876	1025 - 2048	1501 - 2000	Menengah
74	MDNS	797	2090	513 - 1024	2001 - 2500	Menengah
75	MNDP	201	3500	129 - 256	x > 3000	Rendah
76	NBNS	330	4680	257 - 512	x > 3000	Menengah
77	NBNS	690	1536	513 - 1024	1501 - 2000	Menengah
78	NTP	1667	3345	1025 - 2048	x > 3000	Menengah
79	SSDP	379	2521	257 - 512	2501 - 3000	Menengah
80	SSHv2	735	3970	513 - 1024	x > 3000	Rendah
81	SSHv2	586	3650	513 - 1024	x > 3000	Rendah
82	SSHv2	114	2515	65 - 128	2501 - 3000	Rendah
83	SSHv2	627	1295	513 - 1024	1001 - 1500	Rendah
84	SSHv2	1179	1182	1025 - 2048	1001 - 1500	Rendah
85	SSHv2	1195	2522	1025 - 2048	2501 - 3000	Rendah
86	TCP	378	1424	257 - 512	1001 - 1500	Menengah
87	TCP	1694	896	1025 - 2048	501 - 1000	Menengah
88	TCP	1441	2668	1025 - 2048	2501 - 3000	Tinggi
89	TCP	1211	4167	1025 - 2048	x > 3000	Tinggi
90	TCP	982	1551	513 - 1024	1501 - 2000	Menengah
91	TCP	398	3160	257 - 512	x > 3000	Tinggi

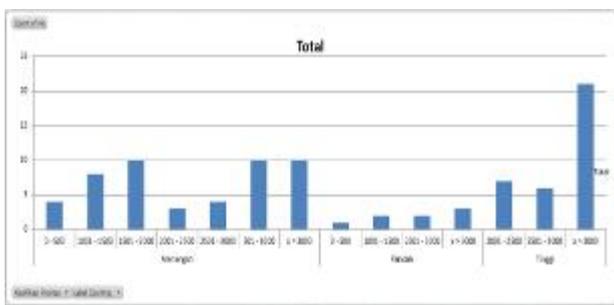
Dari table diatas, di petakan ke dalam grafik prioritas sebagai berikut :



Gambar 2. Banyak Protocol Berdasarkan Prioritas



Gambar 3. Banyak Length Range Berdasarkan Prioritas



Gambar 4. Banyak Counting Range Berdasarkan Prioritas

Pada gambar 2, dapat dilihat bahwa Hasil klasifikasi *network traffic* dari sisi *protocol* terhadap prioritas menjelaskan bahwa *protocol* HTTP memiliki prioritas menengah hingga tinggi dalam penentuan *QoS*. Gambar 3 menunjukkan bahwa Length yang dapat diberikan prioritas menengah adalah terhadap komunikasi yang memiliki ukuran data dari 1025 -2048 byte dan ukuran 513 – 1024. Sedangkan untuk prioritas *QoS* tinggi adalah komunikasi pengiriman data pada ukuran 1025 – 2048 byte. Gambar 3 menunjukkan bahwa prioritas *QoS* dapat diberikan terhadap proses komunikasi yang sama yang memiliki banyak kemunculan di atas 3000 record. Sehingga dalam implementasi manajemen jaringan, kebijakan *QoS* dapat dilakukan berdasarkan prioritas terhadap *protocol*, ukuran length maupun counting (banyaknya data record aktifitas yang sama).

V. KESIMPULAN

Berdasarkan dari hasil pengujian maka dapat disimpulkan bahwa penggunaan metode *Naive Bayes* mampu menunjukkan hasil klasifikasi berdasarkan penggunaan *Protocol*, ukuran komunikasi data yang ditransmisikan dalam bentuk length byte, dan juga berdasarkan counting (record yang sering muncul sebagai aktifitas). Klasifikasi yang dihasilkan merupakan acuan kebijakan penggunaan *QoS* oleh seorang administrator Jaringan.

Pengembangan dari penelitian yang telah dilakukan dalam penelitian ini, dapat dimaksimalkan dengan penggunaan *Scheduling Process*. Sehingga hasil dari dari klasifikasi yang didapatkan adalah berdasarkan keperluan prioritas *QoS* berdasarkan pola waktu aktifitas komunikasi

DAFTAR PUSTAKA

- [1] Yuhai Liu, Zhiqiang Li, Shanqing Guo, Taiming Feng: Efficient, Accurate Internet Traffic Classification using Discretization in Naive Bayes. ICNSC 2008: 1589-1592
- [2] IANA *Port Numbers*, [http://www.iana.org/ assignments/port/numbers](http://www.iana.org/assignments/port/numbers). Diakses tanggal 24 September 2013
- [3] Stephen S. Yau, Yin Yin: *QoS*-Based Service Ranking and Selection for Service-Based Systems. IEEE SCC 2011: 56-63
- [4] Jun Zhang, Chao Chen, Yang Xiang, Wanlei Zhou, Yong Xiang: Internet Traffic Classification by Aggregating Correlated Naive Bayes Predictions. 5-15
- [5] *SecTools.Org: Top 125 Network Security Tools*, <http://sectools.org>. Diakses tanggal 24 September 2013
- [6] Jiqing Liu, Jinhua Huang.2010.Broadband *Network traffic* Analysis and Study In Various Types Of Application.IEEE : 978-1-4244-7050-1110.