

BAB II

KAJIAN PUSTAKA DAN DASAR TEORI

Pada bab ini akan membahas mengenai teori-teori dan referensi yang berkaitan dan menunjang dalam penyelesaian skripsi.

2.1 Penelitian Terkait

Penelitian yang membahas tentang pola sekuensial pada *web access log* dilakukan oleh Ford Lumban Gaol yang berjudul “Exploring The Pattern of Habits of Users Using Web Log Sequential Pattern” pada tahun 2010. Untuk menggali pola sekuensial yang ada pada *web log* Ford Lumban Gaol menggunakan algoritma *aprioriAll*. Sebelum adanya proses penerapan algoritma harus dilakukan *preprocessing* terhadap data, setelah itu data dianalisis oleh sistem untuk mendapatkan pola sekuensial. Informasi yang akurat dan tepat sasaran mengenai kebiasaan pola navigasi pengguna dapat digunakan untuk melakukan desain *website* sesuai dengan pola kebiasaan pengguna. Hal ini akan lebih meningkatkan jumlah kunjungan ke situs *web* tersebut. Kesimpulan yang dihasilkan adalah dengan menggunakan penggalian pola *web log* secara sekuensial dapat lebih mengeksplorasi pola dari kebiasaan pengguna mengakses halaman *website* di *internet*. [GAO - 10]

Penelitian yang dilakukan oleh Ayudya Paramita berjudul “Analisis Pola Kunjungan Pengguna Situs Web IPB Menggunakan Algoritma *Totally Fuzzy*”, meneliti tentang tren pengguna situs *web* terkait dengan pengaksesannya. Konsep fuzzy yang diterapkan dalam teknik *web mining* dapat lebih baik dalam menangani data numerik, karena himpunan fuzzy ”memperhalus” batasan yang tegas. Data numerik diubah menjadi bentuk data fuzzy menggunakan teknik Fuzzy C-Means yang menerapkan konsep ketidakmiripan dalam mengelompokkan suatu objek. Pola sekuensial yang memiliki frekuensi cukup tinggi kemudian dipresentasikan agar mudah dipahami dan diinterpretasikan. Informasi tersebut dapat digunakan oleh *web master* sebagai dasar untuk perancangan ulang disain sebuah situs web, baik tatanan antarmuka maupun isi. [PAR - 09]

Rully Soelaiman dan Dhany Saputra melakukan analisis kinerja algoritma *PrefixSpan* dan *AprioriAll* pada penggalian pola sekuensial. Dalam penelitian

tersebut *PrefixSpan* akan disusun menggunakan teknik proyeksi basis data *Pseudoprojection* dan *AprioriAll* akan disusun sesuai empat fase prosedurnya, yaitu fase pengurutan, fase *itemset*, fase transformasi, dan fase sekuens. Berdasarkan uji coba dapat disimpulkan bahwa secara umum durasi eksekusi dan utilisasi memori *PrefixSpan* lebih kecil daripada *AprioriAll* dan secara grafis dapat dikatakan bahwa *PrefixSpan* lebih skalabel dan terpercaya daripada *AprioriAll*. [SOE - 06]

Penelitian lain yang membahas tentang pola sekuensial dilakukan oleh Stefano Gerry Sutrisno yang berjudul “Perancangan dan Pembuatan Aplikasi Sistem Rekomendasi Buku pada Pespustakaan Universitas Kristen Petra dengan Menggunakan Algoritma *PrefixSpan* dan Generalized Sequential Patterns”, aplikasi yang dibuat dapat memberikan informasi berupa aturan yang digunakan dalam sistem rekomendasi. Penelitian yang dilakukan oleh Stefano Gerry Sutrisno ini memberikan kesimpulan bahwa *PrefixSpan* membutuhkan waktu yang lebih sedikit dibandingkan dengan GSP, namun dalam pengolahannya *PrefixSpan* membutuhkan memori yang lebih besar. [SUT - 10]

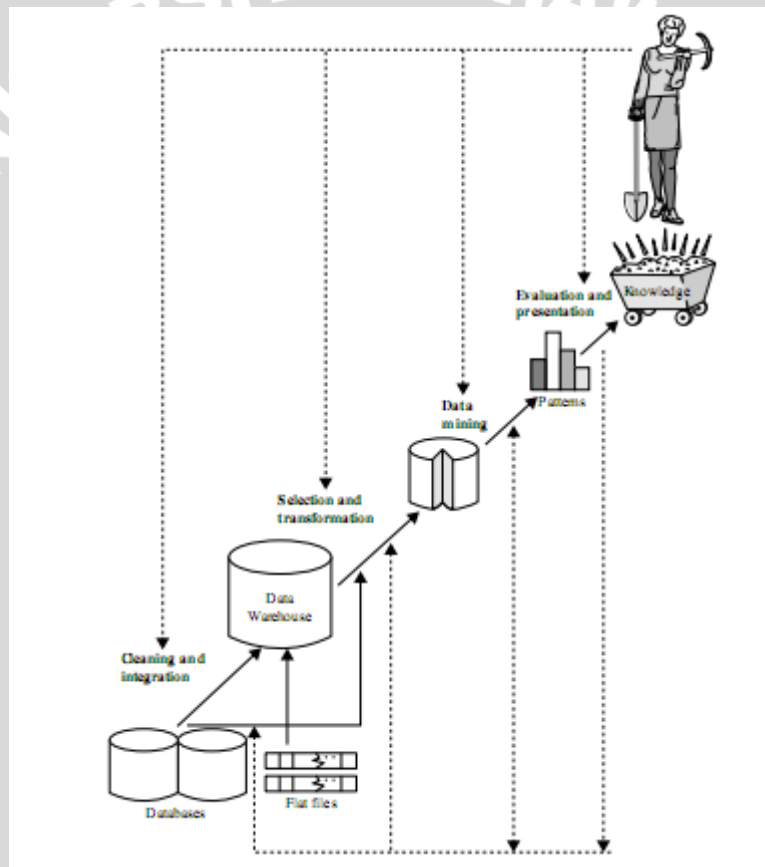
Dalam tulisan Navjot Kaur, Dr. Himanshu Aggarwal yang berjudul “Survey of Various Techniques for Web Usage Mining” dijelaskan bahwa algoritma berbasis *association rule mining* seperti *aprioriAll* dan GSP tidak efisien bila diterapkan pada pola sekuensial yang panjang, hal ini merupakan kelemahan yang sangat penting ketika diterapkan untuk *web log* [KAU – 11]. Penelitian yang dilakukan [PEI - 04] dengan membandingkan tiga algoritma yaitu GSP, *FreeSpan*, dan *PrefixSpan* didapatkan hasil bahwa performa dari algoritma *PrefixSpan* lebih baik dibandingkan dua algoritma yang lain (GSP dan *FreeSpan*). Performa yang dimaksud disini adalah dari segi penggunaan memori dan lama waktu pemrosesan.

Pada peneltian “Penggalian Pola Sekuensial pada Data Akses Pengguna Website Menggunakan Algoritma *PrefixSpan*” akan dibahas penggalian pola sekuensial pada *web access log* Program Teknologi Informasi dan Ilmu Komputer (PTIIK). algoritma pola sekuensial yang digunakan adalah algoritma *PrefixSpan*. Diharapkan hasil pengolahan *web access log* PTIIK dengan teknik Web Usage

Mining (WUM) menggunakan algoritma *PrefixSpan* dapat dimanfaatkan oleh web designer sebagai rekomendasi dalam menata ulang struktur website PTIIK.

2.2 Data Mining

Data mining merupakan proses untuk menemukan pengetahuan dari sejumlah data yang besar. Sumber data yang dimaksud dapat berupa basis data, data warehouse, web, atau media penyimpanan lainnya. Kebanyakan orang menganggap *data mining* sebagai sinonim dari *knowledge discovery from data* (KDD), sebagian orang yang lainnya melihat *data mining* hanya sebagai proses dalam menemukan pengetahuan. [HAN – 12]



Gambar 2. 1 *Data mining* sebagai tahapan dari proses *knowledge discovery*

Sumber [HAN – 12]

Gambar 2.1 merupakan tahap-tahap dari proses *knowledge discovery* yang dilakukan mulai dari pemrosesan data mentah hingga ditemukannya pengetahuan, penjelasan proses tersebut adalah sebagai berikut :

1. Pembersihan data

Pembersihan data digunakan untuk membersihkan *noise* dan menghilangkan data-data yang tidak relevan ataupun yang tidak konsisten.

2. Integrasi data

Integrasi data merupakan proses untuk mengkombinasikan beberapa sumber data.

3. Seleksi data

Data-data yang relevan berdasarkan kebutuhan analisis selanjutnya akan dipilih dan dipisahkan berdasarkan kriteria.

4. Transformasi data

Data relevan yang sudah melalui proses seleksi belum tentu bisa digunakan, transformasi data akan merubah data menjadi bentuk yang sesuai untuk proses pertambangan.

5. *Data mining*

Merupakan proses penting di mana metode cerdas diterapkan untuk mengekstrak pola.

6. Evolusi pola dan Presentasi pengetahuan

Proses mengidentifikasi pola yang didapat. Pola tersebut merupakan representasi dari pengetahuan yang nantinya akan dilakukan pemeriksaan apakah pola yang ditemukan sesuai dengan fakta. Untuk tahap presentasi pengetahuan dilakukan visualisasi / mempresentasikan pengetahuan dalam bentuk yang lebih mudah dimengerti oleh pengguna.

2.3 Web Mining

Menurut [HAN – 12] *web mining* merupakan pengaplikasian dari teknik data mining untuk menemukan pola, struktur, dan pengetahuan dari *website*. *Web mining* sendiri dapat dikategorikan menjadi tiga yaitu : *web content mining*, *web structure mining*, dan *web usage mining*.

Web content mining (WCM) menggali/menambang informasi atau dari konten *web*. Contohnya halaman *web* dapat diklasifikasikan dan di *cluster* berdasarkan topik yang diinginkan. Selain itu, dalam konten *web* dapat juga ditemukan pola untuk mengekstrak data yang berguna seperti deskripsi produk,

posting forum, dll, yang bisa digunakan untuk berbagai tujuan. Misalnya menemukan sentimen konsumen, kita dapat menambang ulasan pelanggan, dan posting forum. [LIU – 07]

Web structure mining (WSM) menemukan pengetahuan yang bermanfaat dari *hyperlink* (link), yang merupakan representasi dari struktur *web* [LIU – 07]. WSM dapat digunakan untuk mengkategorikan halaman *web* dan berguna untuk menghasilkan informasi seperti kesamaan dan hubungan antara situs *web* yang berbeda. [NIN - 09]

Web usage mining (WUM) mencoba untuk memahami data yang dihasilkan oleh sesi pengguna. Jika WCM dan WSM memanfaatkan data yang nyata atau primer di *web*, WUM menambang/menggali data sekunder yang berasal dari interaksi pengguna dengan *web* [NIN - 09].

2.4 Web Usage Mining

Web usage mining (WUM) merupakan penerapan teknik data mining untuk data repositori *web* yang besar dan bertujuan untuk memproduksi hasil yang dapat digunakan dalam merancang situs *web*[KAU – 11]. Dalam tulisan [HAN – 06] dijelaskan bahwa WUM merupakan pertambangan catatan *web* untuk menemukan pola pengguna mengakses halaman *web*. Beberapa algoritma data mining yang umum digunakan dalam WUM adalah *association rule*, *sequential pattern*, and *clustering* . [KAU - 11].

2.4.1 Proses-prose Web Usage Mining

Web Usage Mining memiliki tiga proses utama yang saling bergantung: data *pre-processing*, *pattern discovery*, dan *pattern analysis*. [LIU – 07]

a *Data Preprocessing*

Pada tahap *preprocessing* file *web access log* dibersihkan dan dibagi menjadi satu set transaksi dari pengguna (*sequence database*) [LIU-07]. Tahap *data preprocessing* terdiri dari pembersihan data, identifikasi pengguna, dan identifikasi sesi[NIN - 09].

- Pembersihan Data

Pemberihan data dilakukan untuk menghilangkan item yang tidak relevan[KAU - 11]. Penghapusan item yang tidak relevan dapat dilakukan dengan cara memeriksa akhiran URL, sehingga dapat diketahui jenis format filenya.[NIN - 09]

Item yang harus dihilangkan yaitu :

- Catatan gambar, video, dan informasi format dari dokumen. Catatan ini memiliki akhiran nama file GIF, JPEG, CSS, SWF dan sebagainya yang dapat ditemukan pada URL dari setiap catatan. [KAU – 11] Catatan tersebut dihapus karena format gambar dan video akan muncul bersamaan saat pengguna meminta sebuah halaman dan file ini bukan mewakili sebuah halaman. Sedangkan untuk format dokumen seperti .doc, .pdf, dan .txt merupakan format file yang di download sehingga file ini tidak bisa dijadikan *sequence*.
- Catatan kode status HTTP yang gagal, catatan dengan kode selain 200 akan dihapus[PAR - 09]. Kode status 200 menyatakan bahwa *request* berhasil dilakukan dan *server* berhasil memproses permintaan lalu mengirimkan halaman yang diminta kepada pengguna.

- Identifikasi Pengguna

Identifikasi pengguna digunakan untuk mengidentifikasi siapa yang mengakses *website* dan halaman apa yang diakses. Kesulitan yang terjadi pada tahap ini adalah pengguna yang berbeda mungkin memiliki alamat IP yang sama dalam log. Alamat IP merupakan representasi dari satu unit komputer yang sedang mengakses ke jaringan. Untuk mengidentifikasi pengguna digunakan beberapa aturan: [KAU – 11]

- IP yang berbeda menandakan pengguna yang berbeda
- Jika alamat IP sama, *agent* (browser dan sistem operasi) yang berbeda yang akan menunjukkan pengguna yang berbeda

Contoh identifikasi pengguna dapat dilihat pada tabel 2.1

Tabel 2. 1 Contoh Identifikasi Pengguna Menggunakan IP dan User Agent

Time	IP	URL	Ref	Agent
0:01	1.2.3.4	A	-	IE;Win2k
0:09	1.2.3.4	B	A	IE;Win2k
0:10	2.3.4.5	C	-	IE6;WinXP;SP1
0:12	2.3.4.5	B	C	IE6;WinXP;SP1
0:15	2.3.4.5	E	C	IE6;WinXP;SP1
0:19	1.2.3.4	C	A	IE;Win2k
0:22	2.3.4.5	D	B	IE6;WinXP;SP1
0:22	1.2.3.4	A	-	IE6;WinXP;SP1
0:25	1.2.3.4	E	C	IE;Win2k
0:25	1.2.3.4	C	A	IE6;WinXP;SP2
0:33	1.2.3.4	B	C	IE6;WinXP;SP2
0:58	1.2.3.4	D	B	IE6;WinXP;SP2
1:10	1.2.3.4	E	D	IE6;WinXP;SP2
1:15	1.2.3.4	A	-	IE;Win2k
1:16	1.2.3.4	C	A	IE;Win2k
1:17	1.2.3.4	F	C	IE6;WinXP;SP2
1:26	1.2.3.4	F	C	IE;Win2k
1:30	1.2.3.4	B	A	IE;Win2k
1:36	1.2.3.4	D	B	IE;Win2k

0:01	1.2.3.4	A	-
0:09	1.2.3.4	B	A
0:19	1.2.3.4	C	A
0:25	1.2.3.4	E	C
1:15	1.2.3.4	A	-
1:16	1.2.3.4	C	A
1:26	1.2.3.4	F	C
1:30	1.2.3.4	B	A
1:36	1.2.3.4	D	B

User 1

0:10	2.3.4.5	C	-
0:12	2.3.4.5	B	C
0:15	2.3.4.5	E	C
0:22	2.3.4.5	D	B

User 2

0:22	1.2.3.4	A	-
0:25	1.2.3.4	C	A
0:33	1.2.3.4	B	C
0:58	1.2.3.4	D	B
1:10	1.2.3.4	E	D
1:17	1.2.3.4	F	C

User 3

Sumber : [LIU - 07]

- Identifikasi Sesi

Tujuan dari identifikasi sesi adalah untuk membagi akses halaman dari setiap pengguna pada satu waktu ke sesi setiap pengguna masing-masing. Identifikasi sesi dapat dilakukan dengan cara menggunakan interval waktu antara

entri log berturut-turut. Jika dua akses dari pengguna yang sama dipisahkan oleh interval yang lebih lama dari *threshold* maka mereka dianggap sebagai sesi yang berbeda. *Threshold* yang biasanya digunakan adalah interval waktu 30 menit. [NIN - 09]

b *Pattern Discovery*

Setelah tahap data *preprocessing* metode penemuan pola (*pattern discovery*) harus diterapkan. Cara untuk menemukan pola dapat dilakukan dengan berbagai teknik antara lain analisis statistik, *asosiation rule*, analisis pola sekuensial, *clustering*, klasifikasi, dan sebagainya [NIN - 09].

c *Pattern Analysis*

Tantangan analisis pola adalah untuk memfilter informasi yang tidak menarik kemudian memvisualisasikan dan menginterpretasikan pola tersebut menjadi bentuk yang lebih menarik bagi pengguna.[KAU - 11]

2.4 *Aturan Asosiasi*

Aturan asosiasi adalah suatu prosedur yang mencari asosiasi antar item dalam suatu data set yang ditentukan[HAN - 06]. Aturan asosiasi memberikan informasi dalam bentuk *rule* "if – then" atau "jika – maka". Biasanya digunakan istilah antecedent untuk mewakili bagian "jika" dan consequent untuk mewakili bagian "maka" [SAR - 12].

Dalam penggalian pola sekuensial untuk mendapatkan *rule* dilakukan dengan cara mengambil semua himpunan pola sekuensial yang mempunyai length lebih besar sama dengan 2. Kemudian untuk length-2 maka *rule* yang terbentuk adalah item pertama menjadi antecedent dan item kedua menjadi consequent. Untuk pola yang memiliki length lebih dari 2 maka yang dijadikan consequent adalah item terakhir, kemudian semua item sebelum item terakhir menjadi antecedent.[ARD- 13]

Interestingness measure yang dapat digunakan adalah [HAN - 06]:

- a. *Support*, suatu ukuran yang menunjukkan seberapa besar tingkat dominasi suatu item atau itemset dari keseluruhan transaksi.

- b. *Confidence*, suatu ukuran yang menunjukkan hubungan antar dua item secara conditional. *Confidence* dapat juga disebut sebagai tingkat kebenaran dari aturan yang terbentuk

Rumus *support* adalah sebagai berikut:

$$\text{support}(A) = \frac{\text{jumlah transaksi yang mengandung } A}{\text{total transaksi}} \times 100\% \quad (2-1)$$

Sumber: [SAI - 10]

misal ditemukan *rule* $A \rightarrow B$ maka:

$$\text{Confidence } P(B|A) = \frac{\text{total transaksi mengandung } A \text{ dan } B}{\text{transaksi mengandung } A} \times 100\% \quad (2-2)$$

Sumber: [SAI - 10]

2.5 Lift Ratio

Lift ratio dapat digunakan untuk mengetahui kuat tidaknya sebuah *rule*. *Lift ratio* dapat dihitung dengan cara membandingkan nilai *confidence* dengan nilai *benchmark confidence*. Nilai dari *benchmark confidence* setiap *rule* didapatkan dari nilai jumlah transaksi consequent dibagi dengan jumlah transaksi [LEU-07]. Rumus *benchmark confidence* dan *lift ratio* dapat dilihat pada persamaan (2-3) dan (2-4).

$$\text{Benchmark Confidence} = \frac{\text{jumlah transaksi consequent}}{\text{jumlah keseluruhan transaksi}} \quad (2-3)$$

Sumber [SAR - 12]

$$\text{Liftratio} = \frac{\text{confidence}}{\text{benchmark confidence}} \quad (2-4)$$

Sumber [SAR - 12]

Nilai *lift ratio* yang lebih dari 1.0 menunjukkan bahwa adanya manfaat dari *rule* yang dihasilkan. Semakin besar nilai *lift ratio* yang dihasilkan, maka semakin besar kekuatan asosianya [SAR - 12].

2.6 Pola Sekuensial

Pola sekuensial adalah pola yang menggambarkan urutan waktu dari peristiwa yang sering terjadi [HAN - 06]. Diberikan basis data transaksi (*database sequence*) D dan *threshold minimum support* ε maka dapat didefinisikan

sequential pattern mining adalah mencari nilai *frequent sequence* yang maksimal diantara semua pola sekuensial yang mempunyai nilai *support* lebih besar atau sama dengan ϵ . [PAR - 09]

Contoh : Diberikan minimum *support* = 2, Pada tabel 2.2 *sequence database* memiliki 4 *sequence*. Perhatikan id 10 dengan *sequence* < a (abc) (ac) d (cf) >. *Sequence* ini mempunyai 5 *event* yaitu a, (abc), (ac), d, dan (cf) yang terjadi sesuai urutannya. Item a dan c masing-masing muncul lebih dari satu kali pada *event* yang berbeda dalam *sequence* tersebut. *Sequence* id 10 memiliki 9 *instance*, Oleh karena itu disebut *sequence* tersebut memiliki panjang 9 dan disebut 9-*sequence*. Item a terjadi tiga kali dalam id 10 dan memberikan kontribusi panjang 3 dari *sequence*. Perhatikan *subsequence* $s = (ab)c$. Jika dilihat pada tabel 2.2 terlihat bahwa id 10 dan id 30 adalah satu-satunya yang mengandung *subsequence* s . *support* dari s tersebut adalah 2, hal tersebut memenuhi minimum *support*.

Tabel 2. 2 Tabel *Sequence Database*

id	<i>Sequence</i>
10	< a (abc) (ac) d (cf) >
20	< (ad) c (bc) (ae) >
30	< (ef) (ab) (df) c b >
40	< e g (af) c b c >

Sumber : [PEI - 04]

2.7 Algoritma *PrefixSpan* (Prefix – Projected Sequential Pattern Growth)

PrefixSpan merupakan salah satu algoritma yang menggunakan pendekatan *Pattern growth* dan melakukan proyeksi suffix terhadap *sequence database* [SOE - 06]. *Pattern growth* merupakan metode penggalan pola yang tidak memerlukan pembangkitan kandidat. *Pattern growth* mengadopsi pendekatan *divide-and-conquer*, prinsip dari *pattern growth* adalah *sequence database* secara rekursif diproyeksikan ke sekumpulan basisdata yang lebih kecil berdasarkan pada pola berulang saat itu (s), dan kemudian proyeksi database tersebut digali untuk memperoleh polanya. [PEI – 2004]. *PrefixSpan* memproyeksi *sequence database* dengan berdasarkan hanya pada awalan yang sering muncul / *frequent prefixes* [SUT - 10]. *Pseudocode* dari algoritma *PrefixSpan* dapat dilihat pada gambar 2.2.

Algoritma (*PrefixSpan*)

Masukan : Basis data sekuen S, dukungan minimum min_support

Luaran : Himpunan lengkap pola sekuensial

Metode :

1) *PrefixSpan*(<>, 0, S)

procedure *PrefixSpan* (α , L, $S|_{\alpha}$)

α : pola sekuensial, L : panjang dari α , $S|_{\alpha}$: proyeksi database jika $\alpha \neq \langle \rangle$

- 1) Pindai $S|_{\alpha}$ satu kali, dapatkan semua item berulang b, sedemikian sehingga:
 - a) b dapat diletakkan di dalam elemen terakhir dari α untuk menentukan pola sekuensial.
 - b) $\langle b \rangle$ dapat diletakkan di belakang α untuk membentuk pola sekuensial.
- 2) Untuk setiap item berulang b, letakkan dibelakang α untuk membentuk pola sekuensial α' dan menghasilkan α' .
- 3) Untuk setiap α' , bangunlah proyeksi basis data α' $S|_{\alpha'}$.
- 4) *PrefixSpan* (α' , L+1, $S|_{\alpha'}$)

Gambar 2. 2 Algoritma *PrefixSpan*

Sumber : [SOE - 06]

2.8 Format Log Apache

Combined log format

Format "%h %l %u %t \"%r\" %>s %b %r %a"

Contoh: 127.0.0.1 - frank [10/Oct/2000:13:55:36 -0700] "GET

/apache_pb.gif HTTP/1.0" 200 2326

"http://www.example.com/start.html" "Mozilla/4.08 [en] (Win98; I;Nav)".

Gambar 2. 3 Gambar *Combined Log Format*

Sumber [ANO - 13]

Gambar 2.3 adalah gambar format dari log akses untuk keterangan dari format log apache ini dapat dilihat pada tabel 2.3

Tabel 2. 3 Keterangan *Combined Log Format*

Field	Deskripsi
127.0.0.1 (%h)	Alamat IP dari klien (<i>remote host</i>) yang membuat permintaan ke <i>server</i> .
- (%l)	"Tanda hubung" dalam output menunjukkan bahwa potongan informasi yang diminta tidak tersedia
Frank (%u)	User id yang melakukan permintaan
[10/Oct/2000:13:55:36 -0700] (%t)	Informasi tanggal dan waktu saat <i>server</i> selesai memproses permintaan
"GET /apache_pb.gif HTTP/1.0" ("%r\n")	Permintaan pengguna ke <i>server</i> diberikan tanda kutip "".
200 (%>s)	kode status HTTP yang dikirimkan kembali ke klien oleh <i>server</i>
2326 (%b)	Panjang byte dari data dikirimkan ke klien
http://www.example.com/start.html (%r)	merupakan "Referer" header permintaan HTTP.
"Mozilla/4.08 [en] (Win98; I ;Nav)" (%a)	<i>user agent</i> (tipe dan versi browser dan sistem operasi).