

**APLIKASI FUZZY TEMPORAL ASSOCIATION RULES
UNTUK MENGETAHUI KOMBINASI AKSES KUNJUNGAN
WEBSITE OLEH PENGGUNA INTERNET DI LINGKUNGAN
UNIVERSITAS BRAWIJAYA MALANG**

SKRIPSI

Sebagai salah satu syarat untuk memperoleh gelar
Sarjana dalam bidang Ilmu Komputer



Disusun oleh :

JIHAN RIZKI DEWANTI TREESEFF

NIM. 0810960011

**PROGRAM STUDI INFORMATIKA / ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG**

2013

LEMBAR PERSETUJUAN

APLIKASI FUZZY TEMPORAL ASSOCIATION RULES

**UNTUK MENGETAHUI KOMBINASI AKSES KUNJUNGAN
WEBSITE OLEH PENGGUNA INTERNET DI LINGKUNGAN**

UNIVERSITAS BRAWIJAYA MALANG

SKRIPSI



Disusun oleh :

JIHAN RIZKI DEWANTI TREESEFF

NIM. 0810960011

Telah diperiksa dan disetujui oleh :

Pembimbing I, Pembimbing II,

Candra Dewi, S.Kom., M.Sc

Drs.Muh Arif Rahman, M.Kom

NIP.197711142003122001

NIP.196604231991111001

LEMBAR PENGESAHAN

APLIKASI FUZZY TEMPORAL ASSOCIATION RULES

**UNTUK MENGETAHUI KOMBINASI AKSES KUNJUNGAN
WEBSITE OLEH PENGGUNA INTERNET DI LINGKUNGAN
UNIVERSITAS BRAWIJAYA MALANG**

SKRIPSI

Sebagai salah satu syarat untuk memperoleh gelar
Sarjana dalam bidang Ilmu Komputer

Disusun oleh :

JIHAN RIZKI DEWANTI TREESEFF

NIM. 0810960011

Skripsi ini telah diuji dan dinyatakan lulus tanggal 14 Januari 2013

Penguji I

Penguji II

Rekyan Regasari MP., S.T., M.T.

NIK. 77041406120253

Sabriansyah R.A, ST., M.Eng

NIK. 82080906110084

Penguji III

Himawat Aryadita, ST., MSc.

NIP. 198010182008011003

Mengetahui

Ketua Program Studi Informatika / Ilmu Komputer

Drs. Marji, MT.

NIP. 196708011992031001

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata didalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku. (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, Januari 2013

Mahasiswa,

Jihan Rizki Dewanti Treeseff

NIM 0810960011

**APLIKASI FUZZY TEMPORAL ASSOCIATION RULES
UNTUK MENGETAHUI KOMBINASI AKSES KUNJUNGAN
WEBSITE OLEH PENGGUNA INTERNET DI LINGKUNGAN
UNIVERSITAS BRAWIJAYA MALANG**

ABSTRAK

Akses kunjungan *website* yang dilakukan oleh para pengguna *internet* di lingkungan Universitas Brawijaya disimpan di dalam *file* yang dinamakan data *access log*. Pada penelitian ini dilakukan pencarian aturan asosiasi antara suatu kombinasi alamat *website* yang diakses oleh pengguna *internet* pada periode waktu tertentu dengan menggunakan metode *Fuzzy Temporal Association Rule*, dimana terdapat fungsi *fuzzy calendar* yang dapat mendeskripsikan kata awal, tengah dan akhir dari suatu periode waktu. Data yang digunakan adalah data *access log* mulai bulan Maret sampai Mei 2011. Untuk melihat keefektifan *fuzzy temporal association rules* digunakan *support threshold* sebesar 1%, 3% dan 5%, dan *confidence threshold* sebesar 10%, 30% dan 50%, dengan *fuzzy calendar* yang digunakan adalah awal, tengah dan akhir minggu. Sistem akan mencari aturan asosiasi dengan tahap awal mencari kandidat *n-itemset*, kemudian mencari *frequent itemset* dan terakhir menemukan aturan asosiasi. Pada tahap akhir, sistem akan menghitung kekuatan aturan asosiasi yang terbentuk dengan menggunakan *lift ratio*. Aturan asosiasi yang dihasilkan cukup bervariasi dengan *lift ratio* diatas 1, hal ini menunjukkan bahwa masing-masing aturan asosiasi yang dihasilkan memiliki kekuatan yang baik. Aturan asosiasi dengan nilai *weighted support* dan *weighted confidence* tertinggi diperoleh pada Tengah Minggu di bulan Maret 2011, yaitu “jika mengakses [<http://kh.google.com>] maka mengakses [<http://mw1.google.com>]”.

Kata Kunci : *internet*, kombinasi *website*, aturan asosiasi, *Fuzzy Temporal Association Rules*, *fuzzy calendar*, *lift ratio*

FUZZY TEMPORAL ASSOCIATION RULES APPLICATION TO INVESTIGATE THE WEBSITE VISIT ACCESS COMBINATIONS BY INTERNET USERS IN BRAWIJAYA UNIVERSITY OF MALANG

ABSTRACT

Website visit access by internet users in Brawijaya University stored in file called access log. In this research, the search rules of association between a combination of website addresses that are accessed by Internet users at a specific time period by using Temporal Fuzzy Rule Association, where there is a fuzzy calendar function that can describe the word beginning, middle and end of a time period. The data used is access log data from March 2011 to Mei 2011. In order to see the effectiveness of fuzzy temporal association rules, support threshold is used at 1%, 3%, and 5%, and confidence threshold is used at 10%, 30%, and 50% by using the first, middle, and last week of fuzzy calendar. As the first step, the system will find association rules by searching the candidate of n-itemset, find itemset frequent, and finally, find the association rule. In the final phase, the system will calculate the power of the association rule formed by using lift ratio. The resulting rule is quite varied with a lift ratio above 1, this suggests that each association rules produced has good strength. Rule with a value weighted support and weighted confidence obtained at the highest Middle of Week in the month of March 2011, that "if access [<http://kh.google.com>] then access [<http://mw1.google.com>]".

Key words: *internet, website combination, association rule, Fuzzy Temporal Association Rules, fuzzy calendar, lift ratio*

KATA PENGANTAR

Puji syukur kehadirat Allah SWT, hanya dengan rahmat dan karunia yang telah diberikan kepada penulis, sehingga dapat menyelesaikan skripsi yang berjudul “Aplikasi *Fuzzy Temporal Association Rules* untuk Mengetahui Kombinasi Akses Kunjungan *Website* oleh Pengguna *Internet* di Lingkungan Universitas Brawijaya Malang”.

Skripsi ini merupakan salah satu syarat untuk memenuhi persyaratan akademis untuk menyelesaikan studi di program Sarjana Ilmu Komputer Universitas Brawijaya.

Pada kesempatan ini penulis mengucapkan banyak terima kasih atas segala bantuan dan dedikasi moral maupun material dalam rangka penyusunan skripsi ini kepada :

1. Candra Dewi, S.Kom., M.Sc selaku dosen pembimbing I yang telah membimbing dengan bijaksana dan sabar dalam membimbing dengan baik penyusunan skripsi ini.
2. Drs.Muh Arif Rahman,M.Kom., selaku dosen pembimbing II yang telah membimbing dengan bijaksana dan sabar dalam membimbing dengan baik penyusunan skripsi ini.
3. Drs. Marji, M.T, selaku Ketua Program Studi Ilmu Komputer, PTIIK.
4. Nanang Yudi Setiawan,ST selaku dosen pembimbing akademik atas bimbingan yang diberikan selama penulis menuntut ilmu.
5. Segenap Bapak dan Ibu dosen yang telah mendidik dan mengajarkan ilmunya kepada penulis selama menempuh pendidikan.
6. Segenap staf dan karyawan Jurusan Matematika Universitas Brawijaya yang telah membantu penyusunan skripsi ini.
7. Ayahandaku Ir. Joni Suhartono,MM dan ibundaku Rr. Yeni Treeseff, S.H yang selalu memberi *support* baik materiil dan immateriil untukku, kupersembahkan kelulusanku ini kepada kalian berdua, walaupun hal ini belum cukup untuk membalas semua yang telah kalian berikan padaku.

8. Suamiku tercinta, Sandi Prayoga, ST yang selalu memberi cintanya untukku, sehingga aku memiliki motivasi besar dalam menyelesaikan setiap tahap dalam hidupku.
9. Eyangku RAY Treeseff yang tak pernah lelah menimangku sampai detik ini, adikku Johan Nur Rahman Treeseff yang masuk ITB bersamaan dengan kelulusanku, dan adikku Jesi Kurnia Dewi Treeseff yang sebentar lagi juga akan merasakan dunia perkuliahan, aku sayang kalian.
10. Anakku tercinta yang telah menemaniku sejak semester 4, Reza Rizki Aria Treeseff, mami sangat sangat sayang kamu.
11. Sahabat-sahabat terbaikku deRenboo yang kukenal sejak semester 1, siDul UlyTsulusia yang sudah seperti saudara kembarku dan selalu kemana-mana bersama, Cici yang selalu berusaha untuk gendut tapi selalu gagal dan bebebe Mila si itik dari Blitar yang berubah menjadi angsa paling cantik di kota Malang. Terima kasih sudah menjadi teman paling setia baik suka maupun duka. Ayo kita sukses sama-sama!
12. Teman-teman A ILKOM 08 yang unyu-unyu, terima kasih atas kekuatan doa kalian, semua kakak tingkat di ILKOM, khususnya mbak nini dan mas hendra, vina, yanu dan andri atas saran dan tips n trik menghadapi skripsi, thx juga buat dek Akbar yg udah bantuin di halaman iv.
13. Rekan-rekan di GANESHA OPERATION dan KUMON, yang memberiku kesempatan untuk merasakan bagaimana mengajar yang baik sembari menyelesaikan skripsi ini.
14. Seluruh pihak yang tidak dapat disebut secara langsung yang telah memberikan bantuan demi terselesaikannya skripsi ini.

Penulis menyadari bahwa masih banyak kekurangan dalam penulisan laporan ini yang disebabkan oleh keterbatasan kemampuan dan pengalaman. Oleh karena itu, Penulis sangat menghargai saran dan kritik yang sifatnya membangun demi perbaikan penulisan dan mutu isi penelitian ini.

Malang, Januari 2013

Penulis

DAFTAR ISI

SKRIPSI	1	
LEMBAR PERSETUJUAN	i	
LEMBAR PENGESAHAN	ii	
PERNYATAAN ORISINALITAS SKRIPSI	iii	
ABSTRAK	iv	
ABSTRACT	v	
KATA PENGANTAR	vi	
DAFTAR ISI	viii	
DAFTAR GAMBAR	x	
DAFTAR TABEL	xii	
DAFTAR SOURCECODE	xiv	
1	PENDAHULUAN	1
1.1	Latar Belakang.....	1
1.2	Rumusan Masalah	3
1.3	Batasan Masalah.....	3
1.4	Tujuan.....	3
1.5	Manfaat.....	4
1.6	Sistematika Penulisan.....	4
2	DASAR TEORI	5
2.1	Data.....	5
2.1.1	<i>Pengertian Data</i>	5
2.1.2	<i>Data Web Access</i>	5
2.1.3	<i>Website</i>	6
2.1.4	<i>URL (Uniform Resource Locator)</i>	7
2.1.5	<i>HTTP (HyperText Transfer Protocol)</i>	7
2.2	Logika Fuzzy	8
2.2.1	<i>Himpunan Fuzzy</i>	8
2.2.2	<i>Fuzzy Calendar Algebra</i>	8
2.3	Data Mining	11
2.3.1	<i>Pengertian Data Mining</i>	11
2.3.2	<i>Tahap – tahap Data Mining</i>	12
2.3.3	<i>Teknik Data Mining</i>	13
2.3.4	<i>Association rules Mining</i>	14
2.4	<i>Fuzzy Temporal Association rules</i>	16

2.5	<i>Lift Ratio</i>	20
3	METODE DAN PERANCANGAN SISTEM	21
3.1	Studi Literatur.....	22
3.2	Pengumpulan Data.....	22
3.3	Deskripsi Umum Sistem.....	23
3.4	Perancangan Sistem.....	23
3.4.1	<i>Preprocessing Data Access log</i>	24
3.4.2	<i>Proses Fuzzy Temporal Association rules</i>	28
3.5	Perhitungan Manual.....	39
3.6	Rancangan Antar Muka.....	46
3.7	Rancangan Uji Coba.....	47
4	IMPLEMENTASI DAN PEMBAHASAN	48
4.1	Lingkungan Implementasi.....	48
4.1.1	<i>Lingkungan Perangkat Keras</i>	48
4.1.2	<i>Lingkungan Perangkat Lunak</i>	48
4.2	Implementasi Program.....	48
4.2.1	<i>Implementasi Tahap preprocessing data access log</i>	49
4.2.2	<i>Implementasi Tahap pembentukan frequent itemset</i>	54
4.2.3	<i>Implementasi Tahap pembentukan aturan asosiasi</i>	59
4.3	Implementasi Antarmuka.....	60
4.4	Pengujian Sistem.....	64
4.5	Analisa Hasil.....	70
5	KESIMPULAN DAN SARAN	80
5.1	Kesimpulan.....	80
5.2	Saran.....	81
	DAFTAR PUSTAKA	82
	LAMPIRAN A	84
	LAMPIRAN B	89
	LAMPIRAN C	91
	LAMPIRAN D	95

DAFTAR GAMBAR

Gambar 2. 1 Data Mining sebagai tahapan dalam proses KDD.....	13
Gambar 3. 1 Diagram Alir Penelitian	21
Gambar 3. 2 Proses Perancangan Sistem	24
Gambar 3. 3 Proses <i>preprocessing</i> data <i>access log</i>	27
Gambar 3. 4 Proses pembentukan <i>frequent itemset</i> (L).....	29
Gambar 3. 5 Proses pembentukan data transaksi.....	30
Gambar 3. 6 Proses Pembagian Partisi	31
Gambar 3. 7 Proses perhitungan w_i , m_i dan M_{ij}	32
Gambar 3. 8 Proses pencarian kandidat <i>k-itemset</i> (C).....	35
Gambar 3. 9 Proses pembentukan <i>frequent itemset</i> (L).....	36
Gambar 3. 10 Proses pembentukan aturan Asosiasi	38
Gambar 3. 11 Tampilan Rancangan Antarmuka.....	46
Gambar 4. 1 Tampilan submenu Data.....	61
Gambar 4. 2 Tampilan submenu Kode Web.....	61
Gambar 4. 3 Tampilan submenu hasil <i>preprocessing</i>	62
Gambar 4. 4 Tampilan submenu <i>frequent itemset</i>	63
Gambar 4. 5 Tampilan submenu hasil aturan asosiasi	63
Gambar 4. 6 Hasil uji pengaruh nilai <i>support threshold</i> dan <i>confidence threshold</i> terhadap jumlah <i>rule</i> yang dihasilkan di bulan Maret.....	64
Gambar 4. 7 Hasil uji pengaruh nilai <i>support threshold</i> dan <i>confidence threshold</i> terhadap jumlah <i>rule</i> yang dihasilkan di bulan April.....	65
Gambar 4. 8 Hasil uji pengaruh nilai <i>support threshold</i> dan <i>confidence threshold</i> terhadap jumlah <i>rule</i> yang dihasilkan di bulan Mei.....	65
Gambar 4. 9 Grafik nilai <i>support rule</i> hasil pengujian.....	72
Gambar 4. 10 Grafik nilai <i>confidence rule</i> hasil pengujian.....	72
Gambar 4. 11 Grafik nilai <i>lift-ratio rule</i> hasil pengujian.....	72
Gambar 4. 12 Grafik nilai <i>support rule</i> hasil pengujian.....	74
Gambar 4. 13 Grafik nilai <i>confidence rule</i> hasil pengujian.....	74
Gambar 4. 14 Grafik nilai <i>lift-ratio rule</i> hasil pengujian.....	75
Gambar 4. 15 Grafik nilai <i>support rule</i> hasil pengujian.....	76

Gambar 4. 16 Grafik nilai *confidence rule* hasil pengujian..... 76

Gambar 4. 17 Grafik nilai *lift-ratio rule* hasil pengujian..... 77



DAFTAR TABEL

Tabel 2. 1 Bentuk Umum data <i>access log</i> . Sumber : (Peter, 2004)	5
Tabel 2. 2 <i>Fuzzy Calendar</i> dan derajat keanggotaannya.....	9
Tabel 2. 3 <i>Fuzzy Calendar</i> dan derajat keanggotaannya (μ).....	10
Tabel 3. 1 Contoh format data <i>access log</i>	22
Tabel 3. 2 Contoh 2 baris Data <i>Access log</i>	25
Tabel 3. 3 Elemen yang ada dalam setiap baris Data <i>Access log</i>	25
Tabel 3. 4 Isi elemen ke-3	26
Tabel 3. 5 Isi elemen ke-9	26
Tabel 3. 6 Data <i>Access log</i> setelah proses <i>filtering</i>	26
Tabel 3. 7 Sampel Data dalam basis data D.....	39
Tabel 3. 8 Perhitungan bobot (w_i)	40
Tabel 3. 9 Perhitungan m_i dan M_{ij}	40
Tabel 3. 10 Perhitungan 2- <i>itemset</i> di P_1	41
Tabel 3. 11 Kandidat 2- <i>itemset</i> (C_2) di P_1	41
Tabel 3. 12 Perhitungan 2- <i>itemset</i> di C_2	42
Tabel 3. 13 Perhitungan 2- <i>itemset</i> di P_2	42
Tabel 3. 14 Kandidat 2- <i>itemset</i> (C_2) di P_2	42
Tabel 3. 15 Perhitungan 2- <i>itemset</i> di C_2	43
Tabel 3. 16 Perhitungan 2- <i>itemset</i> di C_2	43
Tabel 3. 17 Perhitungan 2- <i>itemset</i> di P_3	43
Tabel 3. 18 Kandidat 2- <i>itemset</i> (C_2).....	43
Tabel 3. 19 Kandidat 3- <i>itemset</i> (C_3).....	44
Tabel 3. 20 Perhitungan <i>frequent itemset</i> (L) dari C.....	44
Tabel 3. 21 Pembentukan Aturan Asosiasi	45
Tabel 3. 22 <i>Lift Ratio</i> Aturan Asosiasi.....	45
Tabel 3. 23 Tabel Uji Pengaruh Nilai <i>Support threshold</i> dan <i>Confidence threshold</i> terhadap jumlah aturan asosiasi yang dihasilkan	47
Tabel 3. 24 Tabel Uji <i>Lift Ratio Association Rules</i>	47
Tabel 4. 1 Tabel Uji <i>Lift Ratio Association rule</i>	66
Tabel 4. 2 Keterangan <i>Rule</i> Hasil Uji Kekuatan <i>Lift Ratio</i>	66

Tabel 4. 3 Tabel Uji <i>Lift Ratio Association rule</i>	66
Tabel 4. 4 Keterangan <i>Rule</i> Hasil Uji Kekuatan <i>Lift Ratio</i>	67
Tabel 4. 5 Tabel Uji <i>Lift Ratio Association rule</i>	68
Tabel 4. 6 Keterangan <i>Rule</i> Hasil Uji Kekuatan <i>Lift Ratio</i>	69



DAFTAR SOURCECODE

Source Code 4. 1 Implementasi Pembacaan Data access log	50
Source Code 4. 2 Implementasi filtering data access log.....	50
Source Code 4. 3 Implementasi pembentukan transaksi data <i>access log</i>	52
Source Code 4. 4 Implementasi pembentukan partisi data <i>access log</i>	53
Source Code 4. 5 Implementasi penghitungan <i>Fuzzy Calendar</i>	55
Source Code 4. 6 Implementasi penghitungan nilai <i>Weighted count threshold</i> ...	55
Source Code 4. 7 Implementasi penghitungan nilai <i>cumulative weighted count threshold</i>	56
Source Code 4. 8 Implementasi <i>generate</i> kombinasi.....	57
Source Code 4. 9 Implementasi <i>filtering</i> kandidat.....	57
Source Code 4. 10 Implementasi pembentukan <i>frequent itemset</i>	58
Source Code 4. 11 Implementasi <i>filtering frequent itemset</i>	58
Source Code 4. 12 Implementasi Tahap pembentukan aturan asosiasi	59
Source Code 4. 13 Implementasi <i>filtering</i> aturan asosiasi.....	59

BAB I PENDAHULUAN

1.1 Latar Belakang

Sebagai salah satu perguruan tinggi di Indonesia, Universitas Brawijaya memberikan fasilitas *internet* yang dapat menunjang kegiatan belajar mengajar dan aktivitas selama berada di lingkungan kampus. Dimana pengguna *internet* terbesarnya adalah mahasiswa.

Biasanya, mahasiswa mengakses *internet* dengan tujuan mencari referensi tugas kuliah, *chatting*, berbagi informasi melalui media surat elektronik atau *email*, bermain *games online*, dan yang sedang marak saat ini adalah mengikuti jejaring sosial seperti *facebook* dan *twitter* (Apriantony, 2012.). Kegiatan mahasiswa tersebut akan berbeda untuk waktu – waktu tertentu.

Akses kunjungan *website* yang dilakukan oleh para pengguna *internet* tersebut disimpan di dalam *file* yang dinamakan data *access log*. Sebelumnya, pernah dilakukan penelitian mengenai pencarian pola *hyperlink* di lingkungan Universitas Brawijaya menggunakan data *access log* (Perdana,2008). Namun penelitian tersebut hanya sebatas pada pola *hyperlink* yang diakses oleh pengguna *internet*. Sedangkan dari data tersebut kita bisa memperoleh informasi mengenai kombinasi akses kunjungan *website* yang dapat menunjukkan kecenderungan seseorang dalam mengakses alamat *website* saat terkoneksi dengan *internet* pada periode waktu tertentu. Informasi tersebut selanjutnya disebut dengan aturan asosiasi / *rule*.

Untuk dapat memperoleh suatu informasi dari data *access log* yang merupakan data dalam skala besar, dibutuhkan suatu teknik khusus. Teknik tersebut adalah *Data Mining*. *Data Mining* merupakan ilmu yang digunakan untuk menggali informasi berharga yang terpendam atau tersembunyi pada suatu koleksi data (*database*) yang sangat besar sehingga ditemukan suatu pola yang menarik yang sebelumnya tidak diketahui (Hairiyonov, 2004).

Ada metode di dalam data *Mining* yang dapat digunakan untuk menemukan aturan asosiasi antara suatu kombinasi alamat *website* yang diakses oleh pengguna *internet*. Metode tersebut adalah *Association rules*. *Association rules* merupakan

metode untuk mencari hubungan antar *item* dalam suatu data set yang ditentukan. Dalam menentukan suatu *association rule*, terdapat suatu ukuran kepercayaan yang didapatkan dari hasil pengolahan data dengan perhitungan tertentu. *Association rules* memberikan informasi dalam bentuk "if – then" atau "jika – maka" (Santosa, 2007).

Metode *Association rules* digunakan oleh Perdana pada kasus data *access log* dalam pencarian pola *hyperlink* pada *website – website* yang sering diakses oleh pengguna *internet* di Universitas Brawijaya dengan menggabungkan metode *Hyperlink Clustering* untuk sistem rekomendasi pengaturan *hyperlink* pada halaman utama. Tetapi pada penelitian tersebut juga tidak mengaitkan waktu pengaksesan dengan alamat *website* yang diakses oleh pengguna *internet*.

Untuk melihat kecenderungan pengguna dalam mengakses alamat *website* dengan mengaitkan waktu pengaksesannya maka digunakan fungsi - fungsi yang terdapat pada *Fuzzy Calendar* (Suminar,2007). *Fuzzy Calendar* dapat menggambarkan kata awal, tengah dan akhir dari suatu periode waktu (misalkan mingguan, bulanan dan tahunan). Sebagai contoh, mahasiswa biasanya lebih sering mengakses *internet* di tengah atau akhir minggu untuk mencari referensi tugas kuliah.

Dalam memperoleh aturan asosiasi untuk mengetahui keterkaitan antar *item* pada periode waktu tertentu, digunakan metode *Fuzzy Temporal Association Rules* (Lee & Lee, 2004). Berdasarkan penelitian yang telah dilakukan oleh Suminar (2007) dalam skripsinya yang berjudul Pengembangan Aplikasi *Fuzzy Temporal Association rule Mining* (studi kasus : data transaksi pasar swalayan) diperoleh pola atau aturan asosiasi yang memperlihatkan keterkaitan jenis barang yang dibeli oleh pembeli pada waktu-waktu tertentu.

Sehingga untuk mendapatkan aturan asosiasi dari kecenderungan pengguna dalam mengakses *internet* pada periode waktu tertentu maka digunakan metode *Fuzzy Temporal Association Rules*. Berdasarkan latar belakang tersebut, maka judul skripsi ini adalah "Aplikasi *Fuzzy Temporal Association Rules* untuk Mengetahui Kombinasi Akses Kunjungan *Website* oleh Pengguna *Internet* di Lingkungan Universitas Brawijaya Malang".

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan diatas, maka rumusan masalah yang diambil adalah :

1. Bagaimana menerapkan metode *Fuzzy Temporal Association Rules* untuk mengetahui kombinasi akses kunjungan *website* yang dilakukan oleh pengguna *internet* pada periode waktu (*Fuzzy Calendar*) Awal Minggu, Tengah Minggu dan Akhir Minggu pada bulan Maret, April dan Mei 2011 di Universitas Brawijaya Malang.
2. Bagaimana tingkat kekuatan (*Lift-Ratio*) kombinasi akses kunjungan *website* hasil penerapan metode *Fuzzy Temporal Association Rules*.

1.3 Batasan Masalah

Berdasarkan rumusan masalah, penulisan skripsi ini dibatasi pada :

1. Data yang digunakan adalah data *access log* Universitas Brawijaya mulai Maret 2011 sampai dengan Mei 2011.
2. Input yang dibutuhkan adalah data *access log*, *support threshold*(s%), *confidence threshold*(c%) dan periode waktu (*Fuzzy Calendar*).
3. Satu transaksi didefinisikan sebagai akses *internet* yang dilakukan pengguna dengan 1 *IP address* dengan rentang waktu 30 menit (Perdana, 2008).

1.4 Tujuan

Tujuan yang ingin dicapai dalam penulisan skripsi ini adalah :

1. Menerapkan metode *Fuzzy Temporal Association Rules* untuk mengetahui kombinasi akses kunjungan *website* yang dilakukan oleh pengguna *internet* pada periode waktu (*Fuzzy Calendar*) Awal Minggu, Tengah Minggu dan Akhir Minggu pada bulan Maret, April dan Mei 2011 di Universitas Brawijaya Malang.
2. Mengetahui tingkat kekuatan (*Lift-Ratio*) kombinasi akses kunjungan *website* hasil penerapan metode *Fuzzy Temporal Association rules*.

1.5 Manfaat

Manfaat yang dapat diambil dari skripsi ini adalah :

1. Memberikan informasi mengenai kecenderungan pengguna dalam melakukan akses kunjungan *website* saat terkoneksi dengan jaringan *internet* di Universitas Brawijaya.
2. Untuk pemanfaatan lebih lanjut dapat digunakan dalam mendukung keputusan misalnya dalam pengalokasian *bandwith* pada *website* tertentu, pemblokiran *website* tertentu pada saat jam kerja dan *blacklist* untuk *website* tertentu

1.6 Sistematika Penulisan

Sistematika penulisan skripsi terdiri dari 5 bab, yaitu:

BAB I PENDAHULUAN

Pada bab ini membahas mengenai latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, serta sistematika penulisan skripsi.

BAB II DASAR TEORI

Pada bab ini menjelaskan tentang teori-teori dasar dari data *access log*, *Fuzzy*, data *Mining*, gambaran umum *Association rules* serta metode *Fuzzy Temporal Association rules*.

BAB III METODE DAN PERANCANGAN

Pada bab ini dijelaskan kebutuhan sistem, metode, perancangan dan langkah yang digunakan dalam analisa metode *Fuzzy Temporal Associatioan Rules*.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Pada bab ini dijelaskan implementasi aplikasi, uji coba dan analisa hasil.

BAB V PENUTUP

Pada bab ini berisi kesimpulan dari pembahasan dan saran yang diharapkan bermanfaat untuk pengembangan skripsi ini selanjutnya.

BAB II DASAR TEORI

2.1 Data

2.1.1 Pengertian Data

Data merupakan representasi dari fakta atau gambaran mengenai suatu objek atau kejadian. Contohnya fakta mengenai biodata mahasiswa yang meliputi nama, alamat, jenis kelamin, agama, dan lain-lain. Data dapat dinyatakan dengan nilai yang berbentuk angka, deretan karakter, atau simbol (Kusrini, 2007).

Data adalah kumpulan obyek dan atributnya. Atribut-atribut tersebut menunjukkan karakteristik sebuah obyek. Kualitas data yang akan di-*Mining* menentukan proses *Mining* (Kusrini, 2007).

2.1.2 Data Web Access

Data *web* adalah data yang berkaitan dengan *web*, mulai dari teks, halaman HTML, data *multimedia*, data transaksi, hingga data *web access log*, yaitu data yang mencatat dan menunjukkan *user session* (aktifitas pengguna) pada *website* (Peter, 2004). Bentuk umum data *access log* yang disimpan oleh *web servers* adalah *remotehost*, *rfc931*, *authuser*, *date*, *request*, *status*, *bytes*, *referrer*, *user_agent*

Tabel 2. 1 Bentuk Umum data *access log* . Sumber : (Peter, 2004)

Nama Field	Deskripsi field disertai contoh
<i>Remotehost</i>	<i>Remote hostname</i> (atau <i>IP number</i> jika tidak terdapat <i>DNS hostname</i>). Contoh : 127.0.0.1
<i>rfc931</i>	<i>Remote login</i> untuk sebuah <i>username</i> . Contoh : -
<i>Authuser</i>	<i>Username</i> yang melakukan otentifikasi terhadap dirinya. Contoh : -

<i>Date</i>	Waktu (tanggal dan jam) <i>request</i> berdasarkan zona waktu <i>web server's</i> Contoh : [29/May/2007:14:50:04 +0700]
<i>Request</i>	Untuk data <i>request</i> , adalah data yang sebenarnya diperoleh dari aktifitas pengguna. Data request ini dibagi menjadi 3 bagian, yaitu : <i>method</i> , alamat data yang diminta, dan jenis <i>protocol</i> yang digunakan. Contoh : "GET / HTTP/1.1"
<i>status</i>	<i>HTTP status code</i> yang dikembalikan ke pengguna. Contoh : 200
<i>bytes</i>	Besarnya dokumen yang ditransfer. Contoh : 12079
<i>referrer</i>	<i>URL</i> yang dikunjungi pengguna sebelum adanya permintaan terhadap <i>URL</i> yang baru. Contoh : "--"
<i>user_agent</i>	<i>Software</i> yang digunakan oleh pengguna. Contoh : "Mozilla/4.0 (compatible: MSIE 5.01: Windows NT 5.0)"

2.1.3 Website

Website atau situs juga dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, *video* dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).

Website bekerja berdasarkan pada tiga mekanisme berikut (Supriyanto, 2007):

- 1 *Protocol standard*, aturan yang di gunakan untuk berkomunikasi pada *computer networking*, dan *HTTP (Hypertext Transfer Protocol)* adalah *protocol* untuk *website*.

2 *Address*, *website* memiliki aturan penamaan alamat *web* yaitu URL (*Uniform Resource Locator*) yang digunakan sebagai standar alamat *internet*.

3 *HTML*, digunakan untuk membuat dokumen yang bisa diakses melalui *web*.

2.1.4 URL (*Uniform Resource Locator*)

URL menunjukkan sumber daya *Internet* atau alamat sebuah halaman *web* (*homepage*) yaitu halaman suatu dokumen atau program yang ingin ditampilkan atau digunakan. Secara umum tiga informasi yang penting untuk dimasukkan saat menuju ke alamat tertentu, yaitu :

- Protokol,
- Alamat *Server*,
- *Path File*.

Bagian pertama URL menunjukkan protokol misalnya `http://` atau `https://`. Protokol adalah persetujuan bersama yang digunakan untuk berkomunikasi dengan *Hypertext Transfer Protocol*. Bagian kedua URL menunjukkan alamat *server* dimana sumber daya tersebut terletak, misalnya `www.microsoft.com` untuk *website Microsoft Corporation*. Bagian ketiga URL adalah *path file* yaitu menunjukkan lokasi dan nama dokumen atau program dalam *server* tersebut, misalnya: `kb/deskapp/word/q1974.html`. Di mana `kb/deskapp/word/` adalah lokasi *file* dan `q1974.html` adalah nama berkas (Purwanti,2008).

2.1.5 HTTP (*HyperText Transfer Protocol*)

Hypertext Transfer Protokol atau HTTP adalah suatu *protocol* yang menentukan aturan yang perlu diikuti oleh *web browser* dalam meminta atau mengambil suatu dokumen. Protokol ini merupakan *protocol* standar yang digunakan untuk mengakses *html*. Jika menjelajahi *web* dan melihat tulisan seperti `http://www.google.co.id`, merupakan salah satu penggunaan *protokol* http dalam *web* (Supriyanto, 2007).

Ketika seseorang mengklik *link*, menulis URL atau *submit form*, *browser* mengirimkan permintaan ke *server*. Ketika *server* menerima permintaan tersebut, *server* akan mengirimkan kembali suatu *respon* yang berisi nomor kode, biasanya

dikenal sebagai kode status atau kode *respon*. Daftar kode status dapat dilihat di Lampiran A.

2.2 Logika Fuzzy

2.2.1 Himpunan Fuzzy

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu *item* x dalam suatu himpunan A (yang sering ditulis dengan $\mu_A(x)$), terdapat dua kemungkinan, yaitu satu (1) dan nol (0). Nilai satu (1) memiliki arti bahwa suatu *item* menjadi anggota dalam suatu himpunan. Sedangkan nilai nol (0) memiliki arti bahwa suatu *item* tidak menjadi anggota dalam suatu himpunan (Kusumadewi, 2010).

Pemakaian himpunan *crisp* (yang bernilai satu atau nol saja) mengakibatkan perbedaan katagori yang cukup signifikan, walaupun perubahan yang terjadi pada suatu nilai sangat kecil. Himpunan *fuzzy* digunakan untuk mengantisipasi hal tersebut. Suatu *item* dapat masuk dalam beberapa himpunan yang berbeda. Seberapa besar eksistensi suatu *item* dalam himpunan dapat dilihat pada nilai keanggotaan. Pada himpunan *fuzzy*, nilai keanggotaan terletak pada rentang 0 sampai dengan 1. Jika x memiliki nilai keanggotaan *fuzzy* $\mu_A(x)=0$ berarti x tidak menjadi anggota himpunan A . jika x memiliki nilai keanggotaan *fuzzy* $\mu_A(x)=1$ berarti x menjadi anggota penuh pada himpunan A (Kusumadewi, 2010).

Himpunan *fuzzy* memiliki 2 atribut yaitu linguistik dan numeris. Linguistik yaitu penamaan suatu grup yang mewakili suatu kondisi tertentu dengan menggunakan bahasa alami, seperti: muda, parobaya, tua. Sedangkan numeris yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel *fuzzy* seperti: 40, 25, 50 (Kusumadewi, 2010).

2.2.2 Fuzzy Calendar Algebra

Sebuah kalender merupakan koleksi terstruktur dari *interval* waktu. Teori himpunan *fuzzy* diadopsi untuk membangun kalender. Konsep dan operasi dari *fuzzy* digunakan untuk membantu pengguna dalam mengekspresikan kalender dengan baik dan mudah. Untuk membangun kalender, hirarki dari *time granularity*, seperti minggu, bulan, dan tahun, digunakan untuk mendeskripsikan

multiple time granularity (Lee & Lee 2004). Definisi-definisi untuk membangun *Fuzzy Calendar* akan dijelaskan di bawah ini (Lee & Lee 2004).

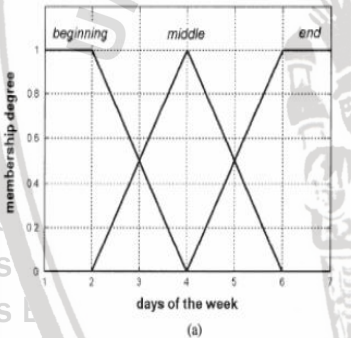
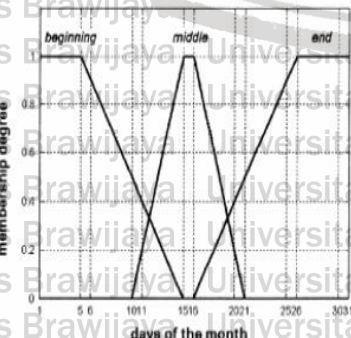
Definisi 1: Sebuah *basic Fuzzy Calendar*, A , mencirikan sebuah proposisi *fuzzy* tentang koleksi dari *interval* waktu pada sebuah *time granularity* U , dideskripsikan sebagai derajat keanggotaan μ_A , dengan

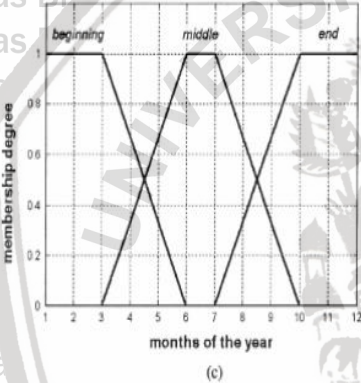
$$\mu_A : U \rightarrow [0,1]$$

Untuk setiap interval waktu $T_i \in U$. Nilai fungsi $\mu_A(T_i)$ mengidentifikasi pencocokan derajat dari T_i ke A . Contoh dari *Fuzzy Calendar* ditunjukkan pada

Tabel 2.2.

Tabel 2. 2 *Fuzzy Calendar* dan derajat keanggotaannya

Gambar <i>Fuzzy Calendar</i>	Derajat Keanggotaannya
 <p>(a)</p>	<p>Derajat Keanggotaan <i>Days of the week</i>, terbagi dalam 3 linguistik yaitu <i>Beginning</i>, <i>middle</i> dan <i>end of week</i>.</p> $\mu_{bw}[x] = \begin{cases} 1; & x < 2 \\ \frac{(4-x)}{2}; & 2 \leq x \leq 4 \\ 0; & x > 4 \end{cases}$ $\mu_{mw}[x] = \begin{cases} 0; & x < 2 \text{ atau } x > 6 \\ \frac{(x-2)}{2}; & 2 \leq x < 4 \\ 1; & x = 4 \\ \frac{(6-x)}{2}; & 4 < x \leq 6 \end{cases}$ $\mu_{ew}[x] = \begin{cases} 0; & x < 4 \\ \frac{(x-4)}{2}; & 4 \leq x \leq 6 \\ 1; & x > 6 \end{cases}$
 <p>(b)</p>	<p>Derajat Keanggotaan <i>Days of the month</i>, terbagi dalam 3 linguistik yaitu <i>Beginning</i>, <i>middle</i> dan <i>end of month</i>.</p> $\mu_{bm}[x] = \begin{cases} 1; & x < 5 \\ \frac{(15-x)}{10}; & 5 \leq x \leq 15 \\ 0; & x > 15 \end{cases}$

	$\mu_{mm}[x] = \begin{cases} 0; & x < 10 \text{ atau } x > 21 \\ \frac{(x-10)}{5}; & 10 \leq x < 15 \\ 1; & 15 \leq x \leq 16 \\ \frac{(21-x)}{5}; & 16 < x \leq 21 \\ 0; & x < 16 \\ \frac{(x-16)}{10}; & 16 \leq x \leq 26 \\ 1; & x > 26 \end{cases}$
 <p style="text-align: center;">(c)</p>	<p>Derajat Keanggotaan <i>Month of the year</i>, terbagi dalam 3 linguistik yaitu <i>Beginning</i>, <i>middle</i> dan <i>end of year</i>.</p> $\mu_{by}[x] = \begin{cases} 1; & x < 3 \\ \frac{(6-x)}{3}; & 3 \leq x \leq 6 \\ 0; & x > 6 \end{cases}$ $\mu_{my}[x] = \begin{cases} 0; & x < 3 \text{ atau } x > 10 \\ \frac{(x-3)}{3}; & 3 \leq x < 6 \\ 1; & 6 \leq x \leq 7 \\ \frac{(10-x)}{3}; & 7 < x \leq 10 \\ 0; & x < 7 \\ \frac{(x-7)}{3}; & 7 \leq x \leq 10 \\ 1; & x > 10 \end{cases}$

Definisi 2 : Sebuah *Fuzzy Calendar* didefinisikan sebagai sebuah *basic Fuzzy Calendar* adalah *Fuzzy Calendar*.

Tabel 2.3 menunjukkan *Fuzzy Calendar* beserta derajat keanggotaannya (μ) sesuai Tabel 2.2.

Tabel 2.3 *Fuzzy Calendar* dan derajat keanggotaannya (μ)

No	<i>Fuzzy Calendar</i>	μ
1	<i>Beginning of the week (bw)</i>	μ_{bw}
2	<i>Middle of the week (mw)</i>	μ_{mw}
3	<i>End of the week (ew)</i>	μ_{ew}

4	<i>Beginning of the month (bm)</i>	μ_{bm}
5	<i>Middle of the month (mm)</i>	μ_{mm}
6	<i>End of the month (em)</i>	μ_{em}
7	<i>Beginning of the year (by)</i>	μ_{by}
8	<i>Middle of the year (my)</i>	μ_{my}
9	<i>End of the year (ey)</i>	μ_{ey}

2.3 Data Mining

2.3.1 Pengertian Data Mining

Data Mining adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual (Pramudiono,2003).

Menurut Han dan Kamber (2006), data Mining merupakan solusi yang mampu menemukan kandungan informasi yang tersembunyi berupa pola dan aturan dari sekumpulan data yang besar. Informasi yang tersembunyi ini menguntungkan dari sudut pandang penelitian, bisnis dan lainnya. Data Mining diharapkan menghasilkan informasi yang sangat akurat dan mudah dipahami.

Menurut Gartner Group data Mining adalah proses untuk menemukan korelasi baru, pola dan tren dengan seleksi pada sejumlah data yang besar dengan menggunakan teknologi pengenalan pola, statistik, dan matematika (Larose, 2005).

Data Mining adalah kegiatan menemukan pola yang menarik dari data dalam jumlah besar, data dapat disimpan dalam *database*, *data warehouse*, atau penyimpanan informasi lainnya. Data Mining berkaitan dengan bidang ilmu – ilmu lain, seperti *database system*, *data warehousing*, statistik, *machine learning*, *information retrieval*, dan komputasi tingkat tinggi. Selain itu, data Mining didukung oleh ilmu lain seperti *neural network*, pengenalan pola, *spatial data analysis*, *image database*, *signal processing* (Han, 2006).

2.3.2 Tahap – tahap Data Mining

Menurut Han dan Kamber (2006), data *Mining* merupakan bagian dari proses *Knowledge Discovery in Database*. Tahap-tahapnya dapat dilihat pada

Gambar 2.1 yang terdiri dari :

1 *Data Cleaning* (Pembersihan data)

Proses *cleaning* antara lain membuang duplikasi data, memeriksa data yang tidak konsisten, dan memperbaiki kesalahan pada data, seperti kesalahan penulisan. Pembersihan data akan mempengaruhi performansi dari sistem data *Mining* karena data akan berkurang jumlah dan kompleksitasnya.

2 *Data IntegRation* (Integrasi Data)

Proses menambah data yang sudah ada dengan data atau informasi lain yang *relevan* dan dibutuhkan oleh KDD. Tidak jarang data yang diperlukan untuk data *Mining* tidak hanya berasal dari satu basis data tetapi juga berasal dari beberapa basis data atau *file* teks.

3 *Data Selection* (Pemilihan Data)

Pemilihan data yang *relevan* dan dapat dilakukan analisis dari data operasional. Data hasil pemilihan disimpan dalam basis data yang terpisah.

4 *Data Transformation* (Transformasi data)

Proses transformasi data ke dalam bentuk tertentu sehingga data tersebut sesuai untuk proses data *Mining*. Beberapa teknik data *Mining* membutuhkan format data yang khusus sebelum bisa diaplikasikan.

5 *Data Mining* (Penggalian Data)

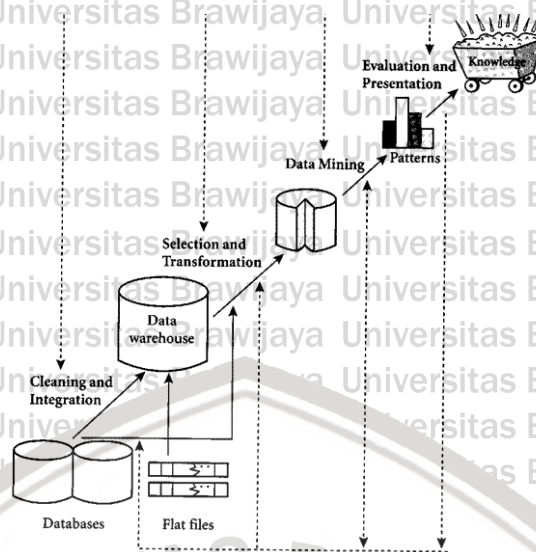
Proses mencari pola atau informasi menarik dengan menggunakan teknik, metode atau algoritma tertentu.

6 *Pattern Evaluation* (Evaluasi Pola)

Mengidentifikasi pola-pola yang benar-benar menarik dari hasil data *Mining*. Hasil dari data *Mining* dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai.

7 *Knowledge Presentation* (Pengetahuan)

Menampilkan pola informasi yang dihasilkan dari proses data *Mining* sehingga mudah dimengerti.



Gambar 2. 1 Data Mining sebagai tahapan dalam proses KDD

2.3.3 Teknik Data Mining

Menurut Pramudiono (2003), teknik – teknik data Mining yang paling populer antara lain :

1 Association rule Mining

Association rule Mining adalah teknik data Mining untuk menemukan aturan asosiasi antara suatu kombinasi item.

2 Classification

Classification adalah proses untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data dengan tujuan untuk dapat memperkirakan kelas dari suatu obyek yang labelnya tidak diketahui.

3 Clustering

Clustering adalah proses pengelompokkan data tanpa berdasarkan kelas data tertentu. *Clustering* dapat dipakai untuk memberikan label pada kelas data yang belum diketahui.

Pengelompokan data Mining berdasarkan tugas yang dapat dilakukan, antara lain (Larose, 2005) :

1 Deskripsi

Deskripsi adalah menggambarkan pola dan kecenderungan yang terdapat dalam data yang memungkinkan memberikan penjelasan dari suatu pola atau kecenderungan tersebut.

2. Estimasi

Estimasi hampir sama dengan klasifikasi, akan tetapi variabel target estimasi lebih ke arah numerik daripada ke arah kategori.

3. Prediksi

Dalam prediksi nilai dari hasil akan terwujud di masa yang akan datang.

4. Klasifikasi

Klasifikasi adalah proses untuk menemukan model atau fungsi yang menggambarkan dan membedakan kelas data atau konsep dengan tujuan memprediksikan kelas untuk data yang tidak diketahui kelasnya (Han dan Kamber, 2006).

5. Clustering

Clustering atau analisis *cluster* adalah proses pengelompokan satu set benda-benda fisik atau abstrak ke dalam kelas objek yang sama (Han dan Kamber, 2006).

6. Asosiasi

Asosiasi dalam data *Mining* adalah menemukan atribut yang muncul dalam satu waktu.

2.3.4 Association rules Mining

Association rule adalah suatu prosedur untuk mencari hubungan antar *item* dalam suatu data set yang ditentukan. Dalam menentukan suatu *association rule*, terdapat suatu ukuran kepercayaan yang didapatkan dari hasil pengolahan data dengan perhitungan tertentu. *Association rule* memberikan informasi dalam bentuk "if-then" atau "jika - maka". Biasanya digunakan istilah *antecedent* untuk mewakili bagian "jika" dan *consequent* untuk mewakili bagian "maka" (Santosa, 2007).

Association rule meliputi dua tahap, yaitu sebagai berikut (Kantardzic, 2003) :

- 1 Mencari kombinasi yang paling sering terjadi dari suatu *itemset*.
- 2 Membangkitkan *association rule* dari *frequent itemset* yang telah dibuat sebelumnya.

Association rule Mining adalah teknik *Mining* untuk menemukan aturan assosiatif antara suatu kombinasi *item*. Contoh dari aturan assosiatif dari analisa pembelian di suatu pasar swalayan adalah bisa diketahui berapa besar kemungkinan seorang pelanggan membeli roti bersamaan dengan susu. Dengan pengetahuan tersebut, pemilik pasar swalayan dapat mengatur penempatan barangnya atau merancang kampanye pemasaran untuk kombinasi barang tertentu. Penting tidaknya suatu aturan assosiatif dapat diketahui dengan dua parameter, *support* dan *confidence* (Pramudiono, 2003).

Misalkan $I = \{i_1; i_2, \dots; i_m\}$ adalah himpunan literal, yang disebut *item*, sedangkan T adalah himpunan transaksi, dimana setiap transaksi T adalah himpunan *item* sehingga $T \subseteq I$. Setiap transaksi mempunyai identifier yang unik. Kita mengatakan bahwa transaksi T mengandung X , himpunan beberapa *item* dalam I , jika $X \subseteq T$. *Association rule* (aturan asosiatif) adalah suatu implikasi, dengan bentuk umum pada rumus 2.12:

$$X \Rightarrow Y \quad (2.1)$$

dimana $X \subset I$, $Y \subset I$, dan $X \cap Y = \phi$. X disebut *antecedent* dan Y disebut *consequent* (Agrawal dkk.,1994).

Misalnya suatu *association rule* dengan bentuk :

$$\{\text{roti, mentega}\} \Rightarrow \{\text{susu}\} \{ \text{support} = 40\%, \text{confidence} = 50\% \}$$

hal ini berarti bahwa : “50% dari transaksi di *database* yang memuat *item* roti dan mentega juga memuat *item* susu. Sedangkan 40% dari seluruh transaksi yang ada di *database* memuat ketiga *item* itu.” Dapat juga diartikan : “Seorang konsumen yang membeli roti dan mentega punya kemungkinan 50 % untuk juga membeli susu. Aturan ini cukup signifikan karena mewakili 40 % dari catatan transaksi selama ini” (Kusrini,2007).

1 *Support*

Support adalah suatu ukuran yang menunjukkan seberapa besar tingkat dominasi suatu *itemset* dari keseluruhan transaksi (Han dan Kamber, 2006).

Menurut Rochmah (2010), *support* dari aturan asosiasi adalah proporsi transaksi yang mengandung *antecedent* dan *consequent*. Nilai *support* 1-*itemset* dapat

diperoleh dari persamaan 2.2 dan nilai *2-itemset* dapat diperoleh dari persamaan 2.3.

$$\text{Support}(A) = \frac{\text{Jumlah Transaksi yang Mengandung A}}{\text{Total Transaksi}} \quad (2.2)$$

$$\begin{aligned} \text{Support}(A; B) &= \frac{P(A \cap B)}{\text{Total Transaksi}} \\ &= \frac{\text{Jumlah Transaksi yang Mengandung A dan B}}{\text{Total Transaksi}} \end{aligned} \quad (2.3)$$

2 Confidence

Confidence adalah suatu ukuran yang menunjukkan hubungan antara dua atau lebih *item* secara *conditional* (Han dan Kamber, 2006). Menurut Rochmah (2010), *confidence* adalah *Ratio* antara jumlah transaksi yang meliputi semua *item* dalam *antecedent* dan *consequent* dengan jumlah transaksi meliputi semua *item* dalam *antecedent*. Nilai *confidence* dapat diperoleh dari persamaan 2.4.

$$\begin{aligned} \text{Confidence}(A \rightarrow B) &= \frac{\text{Support } A \cap B}{\text{Support } A} \\ &= \frac{\text{Jumlah Transaksi yang Mengandung A dan B}}{\text{Jumlah Transaksi yang Mengandung A}} \end{aligned} \quad (2.4)$$

2.4 Fuzzy Temporal Association rules

Itemset adalah himpunan dari *items*, dengan $I = \{i_1, i_2, i_3, \dots, i_m\}$. *D* adalah basis data yang menyimpan transaksi, dimana setiap transaksi *t* diidentifikasi oleh TID. Basis data *D* dibagi menjadi *n* partisi, yaitu $P_1, P_2, \dots, \text{ dan } P_n$.

Langkah – langkah yang digunakan dalam metode *Fuzzy Temporal Association Rules* ini adalah :

1. Bobot (w_i) dari tiap partisi P_i dihitung berdasarkan pencocokan derajat keanggotaan dari waktu transaksi dengan operasi *Fuzzy Calendar* yang telah ditentukan.
2. *Weighted count threshold* (m_i) dihitung untuk menyaring pola asosiasi yang penting di dalam sebuah partisi, dengan rumus :

$$m_i = w_i \times |P_i| \times s\% \quad (2.5)$$

dengan :

$|P_i|$ adalah jumlah transaksi pada partisi P_i

w_i adalah bobot pada partisi P_i

$s\%$ adalah *support threshold*.

Setelah itu, akan dicari *cumulative weighted count threshold*, M_j , dari partisi P_i ke partisi P_j atau P_i, P_{i+1}, \dots, P_j , sehingga

$$M_{ii} = m_i \quad (2.6)$$

$$M_{ij} = M_{i(j-1)} + m_j \quad (2.7)$$

dengan $1 \leq i \leq j \leq n$ dengan n adalah banyaknya partisi dalam basis data.

3. Proses selanjutnya mencari himpunan *candidate 2-itemset*, C_2 , kemudian mencari *candidate frequent itemset*, C . Dimulai dari partisi P_1 , partisi akan diproses satu demi satu. Inisialisasi C_2 adalah himpunan kosong. Kemungkinan kombinasi *2-itemset* I di dalam partisi P_j dicari, kemudian masing-masing *itemset* akan dihitung nilai *weighted count*-nya ($\sigma_{P_j}(I)$) dengan rumus :

$$\sigma_{P_j}(I) = w_j \times |P_j(I)| \quad (2.8)$$

$$U_{jj}(I) = \sigma_{P_j}(I) \quad (2.9)$$

Dengan :

$|P_j(I)|$ adalah jumlah transaksi yang mengandung *itemset* I pada partisi P_j

w_j adalah bobot pada partisi P_j

$U_{jj}(I)$ adalah *Cumulative weighted count* dari I pada partisi P_j

Misalkan akan menghitung di Partisi P_j . Jika I tidak terdapat pada C_2 dan $\sigma_{P_j}(I)$ lebih besar atau sama dengan m_j , maka I ditambahkan pada C_2 .

Tetapi jika I sudah ada dalam C_2 , maka hitung *Cumulative weighted count* dari I ($U_{ij}(I)$) yang dirumuskan :

$$U_{ij}(I) = U_{i(j-1)} + \sigma_{P_j}(I) \quad (2.10)$$

Dengan :

i adalah indeks untuk menunjukkan partisi awal dari sebuah *itemset* I (Partisi saat I tersebut ditemukan pertama kali).

j adalah indeks untuk menunjukkan sedang menghitung di Partisi P_j .

$\sigma_{P_j}(I)$ adalah nilai *weighted count* dari I .

Jika,

$$U_{ij}(I) \geq M_{ij} \quad (2.11)$$

maka I tetap disimpan dalam C_2 dan *cumulative weighted count* dari I diperbaharui menjadi $U_{i(j-1)} + \sigma_{P_j}(I)$, tetapi jika

$$U_{ij}(I) < M_{ij} \quad (2.12)$$

maka I dihapus dari C_2 dan semua informasi mengenai I juga dihapus.

4. C_2 digunakan untuk mencari *candidate k-itemsets*, C_k , dengan ketentuan sebagai berikut:

$$C_{k+1} = C_k * C_k, k \geq 2 \quad (2.13)$$

C adalah gabungan dari semua *candidate k-itemsets*, $k \geq 2$,

$$C = \bigcup_{k \geq 2} C_k \quad (2.14)$$

5. Proses selanjutnya adalah mencari *frequent itemset*, L , dengan memeriksa basis data sekali lagi. Untuk setiap I dalam C , *weighted count* dari *itemset* I pada basis data D , dinotasikan $\sigma_D(I)$ digunakan untuk menghitung seberapa sering *itemset* I muncul dalam basis data D , dengan

$$\sigma_D(I) = \sum_{i=1}^n (|P_i(I)| \times w_i) \quad (2.15)$$

Dengan:

$|P_i(I)|$ adalah jumlah transaksi yang mengandung *itemset* I pada partisi P_i

w_i adalah bobot pada partisi P_i

Itemset I adalah *frequent* dengan *support threshold*, $s\%$, jika

$$\sigma_D(I) \geq \left[\sum_{i=1}^n (|P_i| \times w_i) \right] \times s\% \quad (2.16)$$

Persamaan $\left[\sum_{i=1}^n (|P_i| \times w_i) \right] \times s\%$ disebut *weighted count threshold* dari D , atau disimbolkan menjadi M_{In} , dengan n adalah banyaknya partisi yang terbentuk pada basis data D . Jika nilai $\sigma_D(I)$ lebih besar atau sama dengan M_{In} , maka I adalah *frequent itemset* (L) dalam basis data D .

Sama seperti pencarian C , pencarian L , juga dimulai dari L_2 , kemudian dari L_2 diperluas menjadi L_3 sampai L_k . Himpunan *frequent itemset*, L , dari D kemudian mengandung

$$L = \bigcup_{k \geq 2} L_k \quad (2.17)$$

6. *Fuzzy Temporal association rule* dapat ditemukan setelah *frequent itemset*,

L , diperoleh. Sebuah *Fuzzy Temporal association rule* yang mengacu pada

Fuzzy Calendar, FC , adalah implikasi dari bentuk

$$X \Rightarrow^{FC} Y \quad (2.18)$$

dengan $X \subseteq I$, $Y \subseteq I$, dan $X \cap Y = \emptyset$, yang mana $I = \{i_1, i_2, i_3, \dots, i_m\}$.

Aturan asosiasi dari $X \Rightarrow^{FC} Y$ disebut memenuhi *weighted support*, $s\%$ dari basis data D , jika

$$\text{weighted support} = \frac{[\sum_{i=1}^n (|P_i(I)| \times w_i)]}{[\sum_{i=1}^n (|P_i| \times w_i)]} \quad (2.19)$$

Untuk sebuah *association rule* $X \Rightarrow^{FC} Y$, dengan

$$\text{weighted confidence} = \frac{\sigma_D(I)}{\sigma_D(X)} \quad (2.20)$$

dikatakan berada pada basis data D dengan *weighted confidence*, $c\%$. $X \Rightarrow^{FC} Y$ dikatakan aturan asosiasi jika *weighted confidence* $\geq c\%$, dimana $c\%$ adalah *confidence threshold* yang diinputkan pengguna (Suminar, 2007).

2.5 Lift Ratio

Lift Ratio digunakan untuk mengevaluasi kuat tidaknya sebuah aturan asosiasi. *Lift Ratio* adalah perbandingan antara *confidence* sebuah aturan dengan nilai benchmark *confidence*. Benchmark *confidence* adalah perbandingan antara jumlah semua *item consequent* terhadap total jumlah transaksi (Santosa, 2007).

Rumus benchmark *confidence* dan *Lift Ratio* dapat dilihat pada persamaan 2.21 dan persamaan 2.22.

$$\text{Benchmark Confidence} = \frac{Nc}{N} \quad (2.21)$$

Keterangan persamaan 2.21 :

Nc = jumlah transaksi dengan *item* dalam *consequent*

N = jumlah transaksi *database*

$$\text{Lift Ratio} = \frac{\text{Confidence}(A, C)}{\text{Benchmark Confidence}(A, C)} \quad (2.22)$$

Apabila nilai *Lift Ratio* lebih besar dari 1, maka menunjukkan adanya manfaat dari aturan tersebut. Lebih tinggi nilai *Lift Ratio* maka lebih besar kekuatan asosiasinya (Santosa, 2007).

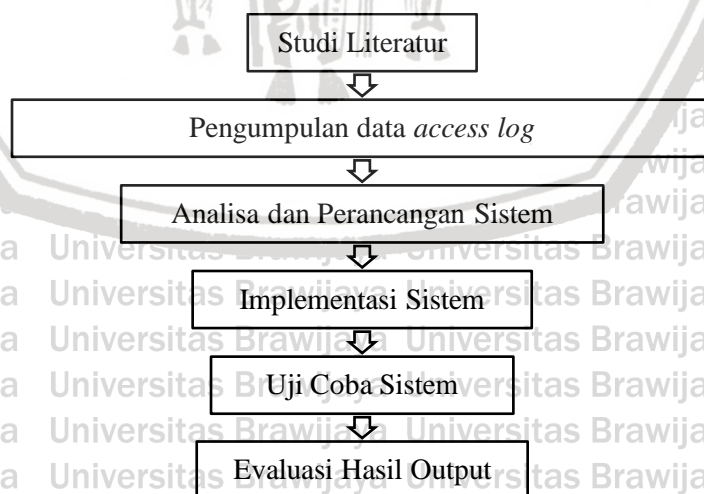
BAB III

METODE DAN PERANCANGAN SISTEM

Pada bab metodologi dan perancangan ini akan dibahas metode, rancangan yang digunakan, dan langkah-langkah yang dilakukan dalam penelitian tentang *Fuzzy Temporal Association Rules* untuk mengetahui kombinasi akses kunjungan *website* yang dilakukan oleh pengguna *internet* di lingkungan Universitas Brawijaya. Langkah-langkah yang dijalankan dalam penelitian ini, yaitu:

1. Mempelajari literatur yang berhubungan dengan metode *Fuzzy Temporal Association rules*.
2. Melakukan pengumpulan data *access log*.
3. Menganalisa dan melakukan perancangan sistem dengan metode *Fuzzy Temporal association rule*.
4. Membangun perangkat lunak berdasarkan analisis dan perancangan yang telah dilakukan (implementasi).
5. Melakukan uji coba terhadap perangkat lunak.
6. Mengevaluasi hasil *output* berupa aturan asosiasi yang dihasilkan oleh sistem.

Adapun langkah-langkah penelitian dapat digambarkan dalam bentuk diagram alir yang ditunjukkan pada gambar berikut.



Gambar 3. 1 Diagram Alir Penelitian

3.1 Studi Literatur

Dalam merealisasikan tujuan dan pemecahan masalah, penelitian ini dimulai dengan tahap awal yaitu mempelajari studi literatur atau sumber-sumber yang berkaitan. Teori-teori mengenai data *access log*, konsep data *Mining* dan *Fuzzy Temporal Association Rules* yang digunakan sebagai dasar penelitian. Studi literatur dilakukan dari sumber buku, jurnal maupun browsing *internet*.

3.2 Pengumpulan Data

Pengumpulan data merupakan usaha untuk memperoleh data atau dokumen yang dibutuhkan dalam penelitian dan untuk selanjutnya data tersebut akan diproses sesuai kebutuhan. Data yang digunakan untuk penelitian ini adalah data *access log*. Sumber data yang digunakan adalah data *access log* di Universitas Brawijaya. Data yang diperoleh adalah data *access log* dari bulan Maret sampai bulan Mei 2011. Tabel 3.1 menunjukkan format data *access log* yang terdapat di Universitas Brawijaya.

Tabel 3. 1 Contoh format data *access log*

Data Access log	
1298912544.768 658 172.17.105.26 TCP_MISS/200 226 POST http://www.bannersurvey.biz/ajax/widgets/gateway_iframe.php - DIRECT/72.32.87.240 text/html	
1298912545.654 475 172.17.61.67 TCP_MISS/200 1087 GET http://www.facebook.com/ajax/presence/reconnect.php? - DIRECT/69.63.189.39 application/x-javascript	
1298912545.937 297 172.17.62.45 TCP_MISS/200 2264 GET http://dnl- 17.geo.kaspersky.com/index/u0607g.xml.klz - DIRECT/38.117.98.196 application/octet-stream	
1298912546.177 228 172.17.62.39 TCP_MISS/200 208 GET http://1.38.channel.facebook.com/x/4099389245/2005355141/false/p_144580 6632=1 - DIRECT/66.220.145.35 text/plain	
1298912546.793 1645 172.17.53.114 TCP_MISS/200 24968 GET http://www.facebook.com/permalink.php? - DIRECT/69.63.189.39 text/html	

3.3 Deskripsi Umum Sistem

Secara umum sistem ini adalah suatu perangkat lunak yang dibangun untuk mengetahui kombinasi akses kunjungan *website* yang dilakukan oleh pengguna *internet* pada periode waktu tertentu di Universitas Brawijaya dengan menggunakan metode *Fuzzy Temporal Association rules*. Analisa dilakukan pada data *access log* Universitas Brawijaya yang didalamnya terdapat *website* yang diakses oleh pengguna saat berada di lingkungan Universitas Brawijaya. Pada sistem ini, 1 transaksi didefinisikan sebagai akses *internet* yang dilakukan pengguna dengan rentang 30 menit.

Pembentukan himpunan *fuzzy* pada sistem ini menggunakan *Fuzzy Calendar algebra*. Atribut yang digunakan dalam analisa adalah *support threshold* (s%), *confidence threshold* (c%) dan periode *Fuzzy Calendar*. Periode yang disediakan adalah Mingguan. Keanggotaan yang disediakan adalah Awal, Tengah dan Akhir. Hasil keluaran dari sistem adalah aturan – aturan asosiasi yang menunjukkan kombinasi akses kunjungan *website* yang dilakukan oleh pengguna *internet* di lingkungan Universitas Brawijaya pada periode waktu tertentu.

3.4 Perancangan Sistem

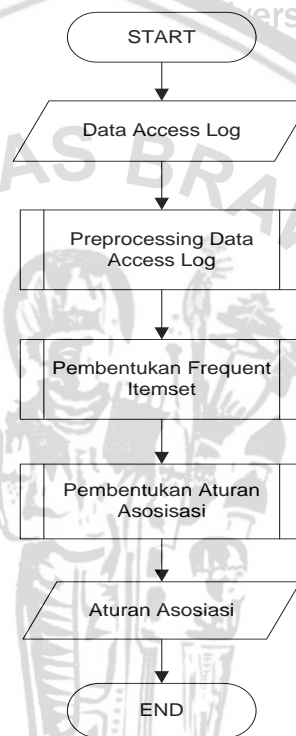
Tahapan pada perancangan sistem ialah sebagai berikut ;

- 1 Data *Access log* yang dicontohkan pada Tabel 3.1 merupakan data mentah yang belum dapat diolah menggunakan metode *Fuzzy Temporal Association rules*. Maka dari itu dibutuhkan sebuah proses yang dinamakan *Preprocessing Data Access log* untuk mengekstrak informasi yang dibutuhkan untuk proses selanjutnya.
- 2 Proses yang dilakukan setelah *Preprocessing Data Access log* adalah pembentukan *Frequent Itemset* (L) dari masukan data *access log* yang telah siap diolah, disebut dengan basis data D. Sebuah *itemset* (I) merupakan *frequent* dengan *support threshold* (s%) jika nilai *weighted count* dari *itemset* I ($\sigma_D(I)$) lebih besar dari *weighted count threshold* (M_{1n}) dari basis data D.
- 3 Dari semua *frequent itemset* yang didapat, dibangkitkan semua aturan asosiasi yang mungkin. Semua aturan asosiasi tersebut dihitung nilai

weighted confidence-nya. Aturan asosiasi yang memiliki nilai *weighted confidence* lebih besar dari *confidence threshold* ($c\%$) akan diambil sebagai aturan asosiasi yang telah memenuhi syarat, sedangkan lainnya dibuang.

- 4 Keluaran adalah aturan – aturan asosiasi dengan *support threshold* ($s\%$) dan *confidence threshold* ($c\%$).

Alur dari tahap perancangan sistem digambarkan pada **Gambar 3.2**.



Gambar 3. 2 Proses Perancangan Sistem

3.4.1 Preprocessing Data Access log

Data *access log* pada *server* yang mula – mula digunakan adalah seluruh data *access log* yang didapat dari pengumpulan data. Data tersebut adalah data mentah yang belum dapat diolah untuk proses selanjutnya. Contoh data yang belum dapat diolah tersebut sudah dipaparkan pada Tabel 3.1. Tahap *preprocessing data access log* dimulai dari proses pemecahan terhadap setiap baris data *access log*. Contoh :

Tabel 3. 2 Contoh 2 baris Data Access log

Baris ke-	Data Access log
1	1298912545.654 475 172.17.61.67 TCP_MISS/200 1087 GET http://www.facebook.com/ajax/presence/reconnect.php? DIRECT/69.63.189.39 application/x-javascript
2	1298912546.147 770 172.17.170.169 TCP_MISS/302 604 GET http://www.facebook.com/ai.php? - DIRECT/63.245.209.93 text/html

Dari setiap baris data *access log* tersebut, selanjutnya masing – masing baris akan dipecah menjadi sebuah *array* yang dipisahkan oleh spasi. Contoh :

Tabel 3. 3 Elemen yang ada dalam setiap baris Data Access log

Elemen ke-	Nama Elemen	Contoh Baris 1	Contoh Baris 2
0	<i>DateTime</i>	1298912545.654	1298912546.147
1	<i>Port</i>	475	770
2	<i>Host</i>	172.17.61.67	172.17.170.169
3	<i>Code/Status</i>	TCP_MISS/200	TCP_MISS/202
4	<i>Byte</i>	1087	604
5	<i>Method</i>	GET	GET
6	<i>URL</i>	http://www.facebook.com/ajax/presence/reconnect.php?	http://www.facebook.com/ai.php?
7	<i>Authuser</i>	-	-
8	<i>PeerStatus/PeerHost</i>	DIRECT/69.63.189.39	DIRECT/63.245.209.93
9	<i>Mime Type</i>	application/x-javascript	text/html

Data *access log* yang siap diolah adalah data yang terdiri dari waktu, IP *Address* dan alamat *website* yang diakses oleh pengguna. Maka dari itu, setiap baris yang sudah dipisahkan oleh spasi hanya akan diambil elemen ke-0, 2 dan 6 kemudian disimpan ke dalam basis data dengan syarat tertentu, karena tidak semua alamat *website* yang terdapat pada data *access log* adalah alamat yang penting untuk dianalisa (Suneetha dan Krishnamoorti, 2009). Syarat yang harus

dipenuhi agar alamat *website* yang diakses pengguna masuk ke dalam basis data adalah :

1. Pisahkan elemen ke-3 dan ke-9 berdasarkan garis miring (“/”).

Tabel 3. 4 Isi elemen ke-3

	Elemen3[0]	Elemen3[1]
Contoh Baris 1	TCP_MISS	200
Contoh Baris 2	TCP_MISS	202

Tabel 3. 5 Isi elemen ke-9

	Elemen9[0]	Elemen9[1]
Contoh Baris 1	Application	Javascript
Contoh Baris 2	Text	Html

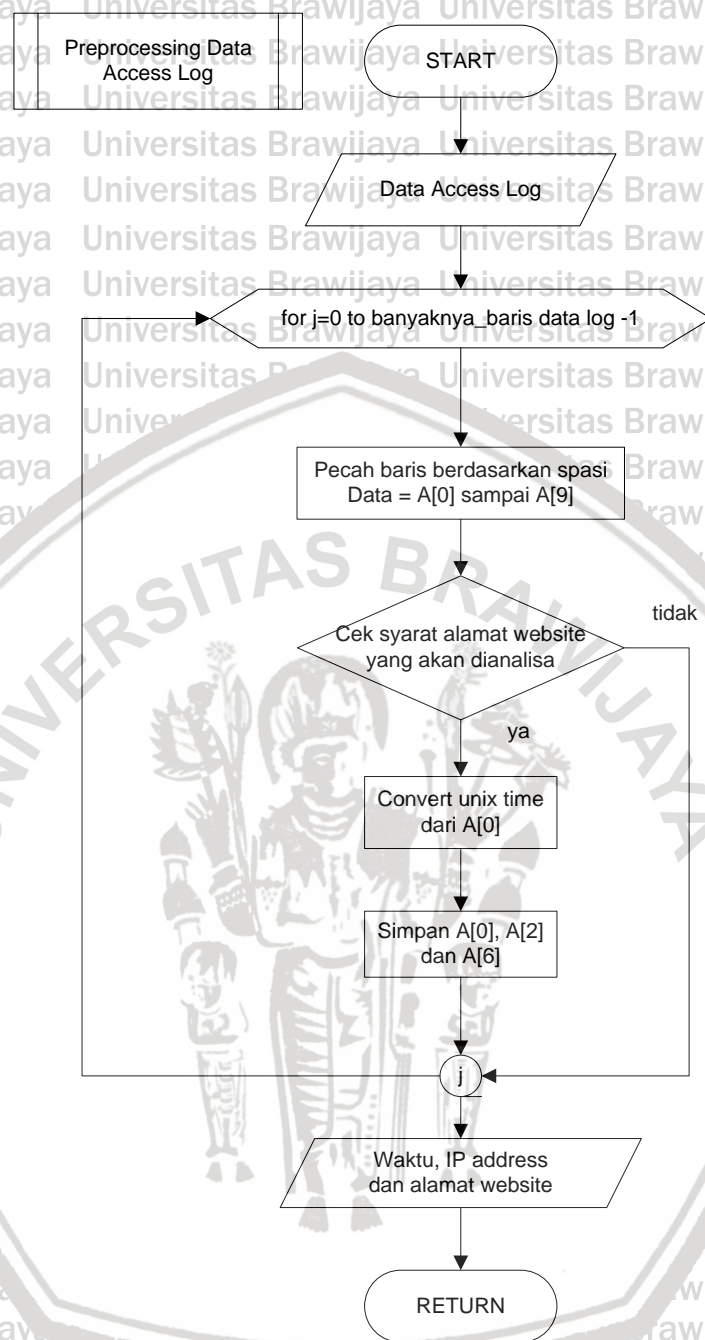
2. Pada elemen3[1] dimulai dengan angka “2” dan elemen9[0] adalah “*application*” dan elemen9[1] bukan “*javascript*”.
3. Pada elemen3[1] dimulai dengan angka “2” dan elemen9[1] adalah “*html*”.

Dari syarat tersebut, maka contoh diatas yang bisa masuk ke dalam basis data adalah Contoh Baris ke-2 dengan *format* sebagai berikut :

Tabel 3. 6 Data *Access log* setelah proses *filtering*

No	Data
1	09-03-2011 10:03:12 172.17.170.169 facebook.com

Dari Tabel 3.6 menunjukkan hanya elemen 0, 2 dan 6 yaitu waktu, IP *Address* dan alamat *website* yang disimpan ke dalam basis data. **Gambar 3.3** menunjukkan *flowchart* dari *preprocessing* data *access log*.



Gambar 3. 3 Proses *preprocessing* data access log

Langkah – langkah *preprocessing* data access log adalah :

1. Masukan awal adalah data mentah *access log*.
2. Lakukan pemecahan pada setiap baris data *access log* berdasarkan “spasi”
3. Simpan data dalam *array* dari $A[0] - A[9]$.

4. Dari array yang sudah terbentuk, cek elemen A[3] dan A[9] sesuai syarat yang telah ditentukan, jika Ya maka lanjutkan ke langkah 5. Jika Tidak kembali ke langkah 3 untuk baris berikutnya.
5. *Convert unix time* dari elemen A[0].
6. Simpan elemen A[0], A[2] dan A[6].
7. Keluaran dari proses *preprocessing* adalah waktu, *IP address* dan alamat *website* yang diakses pengguna.

3.4.2 Proses Fuzzy Temporal Association rules

Setelah proses *Preprocessing Data Access log*, maka tahap selanjutnya adalah proses *Fuzzy Temporal Association Rules* untuk mendapatkan aturan asosiasi dari keterkaitan antar *website* yang diakses oleh pengguna *internet* di lingkungan Universitas Brawijaya. Secara garis besar, proses ini dibagi menjadi 2 bagian, yaitu pembentukan *frequent itemset* (L) dan pembentukan aturan asosiasi.

3.4.2.1 Pembentukan Frequent Itemset (L)

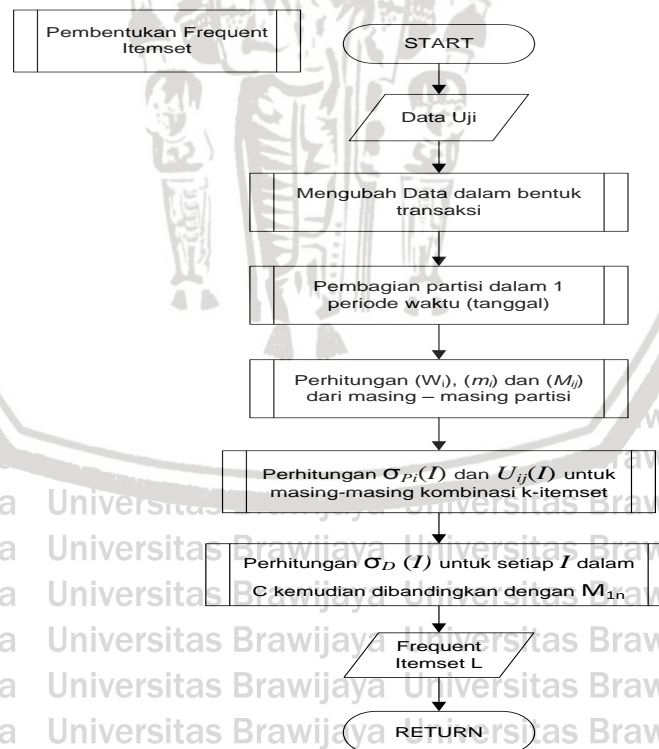
Proses *Fuzzy Temporal Association Rules* yang pertama adalah pembentukan *frequent itemset* (L), dimana L tersebut akan digunakan untuk membentuk aturan asosiasi.

Langkah – langkah dalam membentuk L adalah :

1. Masukan berupa data uji di basis data yang merupakan keluaran dari proses *preprocessing* data.
2. Data akan dibentuk dalam transaksi dengan ketentuan yang sudah disebutkan sebelumnya yaitu 1 transaksi adalah kumpulan alamat *website* dalam rentang waktu maksimal 30 menit.
3. Setelah membentuk data transaksi, dilakukan pembagian partisi berdasarkan tanggal akses yang dilakukan pengguna. 1 partisi adalah kumpulan transaksi dalam 1 hari.
4. Bobot tiap partisi dihitung berdasarkan pencocokan derajat keanggotaan dari waktu transaksi dengan *Fuzzy Calendar* yang telah ditentukan.

5. *Weighted count threshold* (m_i) dan *cumulative weighted count threshold* (M_{ij}) dihitung untuk menyaring pola asosiasi yang penting di dalam sebuah partisi.
6. Pada bagian ini, akan dicari kemungkinan kombinasi 2-*itemset* di tiap partisi yang kemudian masing-masing *itemset* akan dihitung *weighted count* ($\sigma_{P_i}(I)$) dan *cumulative weighted count* (U_{ij}) untuk dibandingkan dengan m_i dan M_{ij} . Kombinasi *itemset* yang lolos akan dimasukkan ke dalam himpunan C_2 . Dari C_2 akan dibentuk C_k dengan $k \geq 2$, sehingga diperoleh kandidat *frequent itemset* (C) yang merupakan gabungan C_2 sampai C_k .
7. Dari C yang sudah didapat, akan dicari *frequent itemset* (L) dengan membandingkan nilai *weighted count itemset* ($\sigma_D(I)$) dengan *weighted count threshold* (M_{1n}) dari basis data keseluruhan (basis data D).
8. Keluaran berupa himpunan *itemset* dalam L .

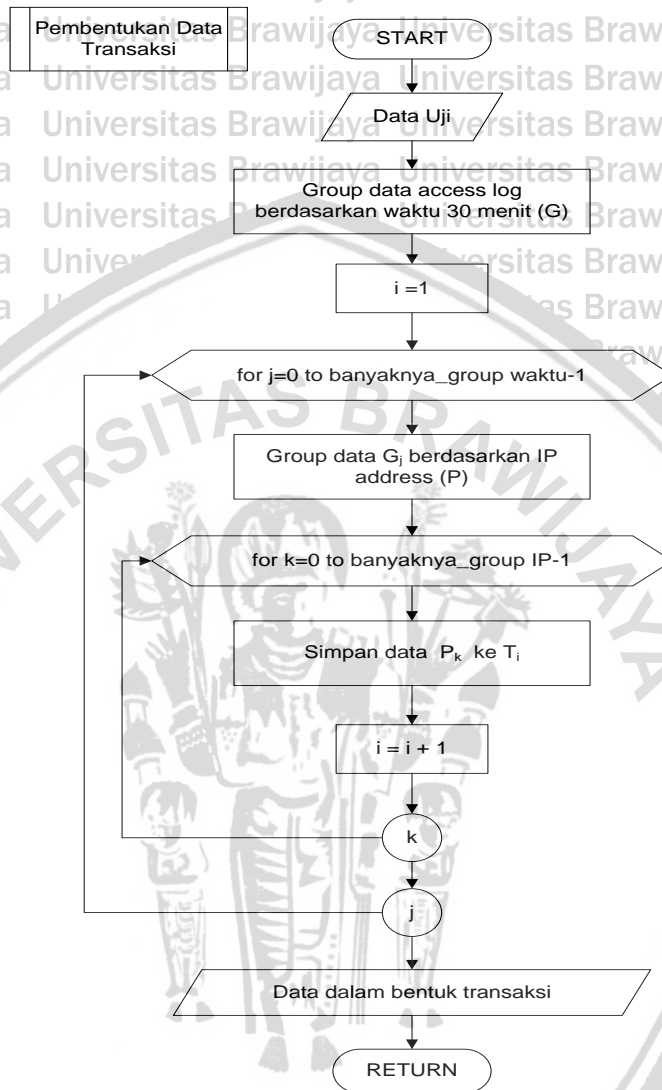
Tahapan pembentukan *frequent itemset* (L) ditunjukkan pada **Gambar 3.4**.



Gambar 3.4 Proses pembentukan *frequent itemset* (L)

3.4.2.1.1 Proses pembentukan data transaksi

Proses pembentukan data transaksi ditunjukkan oleh **Gambar 3.5**.



Gambar 3.5 Proses pembentukan data transaksi

Langkah – langkah pembentukan data transaksi adalah :

1. Masukan berupa data uji dari basis data.
2. Data tersebut dikelompokkan berdasarkan waktu transaksi 30 menit, disebut dengan data G.
3. Setiap data G dikelompokkan berdasarkan *IP_Address*, disebut dengan data P.

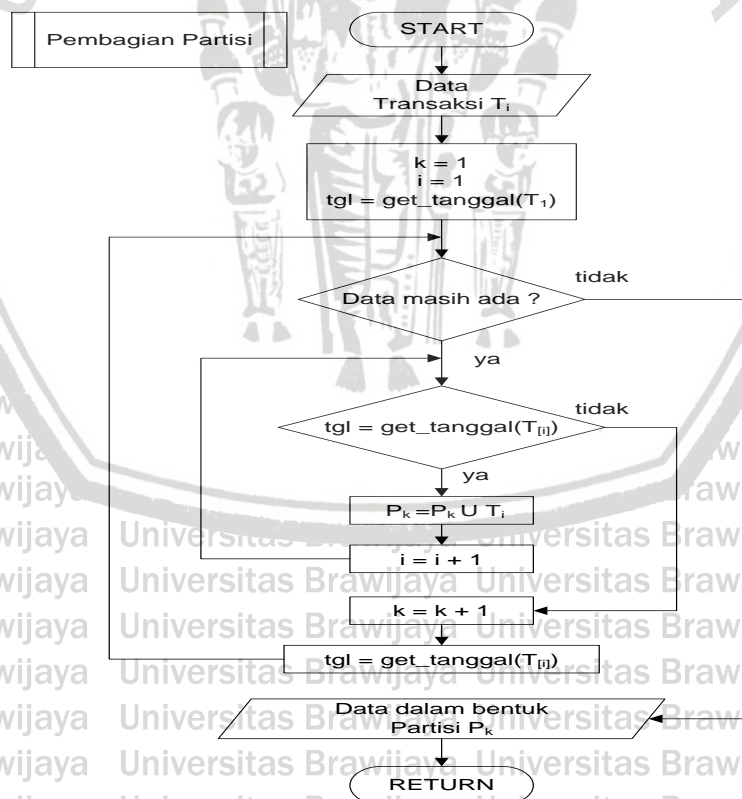
4. Setiap data yang ada di dalam P disebut dengan data transaksi T_i .
5. Keluaran merupakan data dalam bentuk transaksi, dimana di dalam 1 transaksi terdapat beberapa alamat *website* yang diakses oleh 1 *IP_Address* dalam rentang waktu 30 menit.

3.4.2.1.2 Proses Pembagian Partisi

Langkah – langkah pembentukan data transaksi adalah :

1. Masukan berupa data dalam bentuk transaksi.
2. Inisialisasi $i = 1$, $k = 1$ dan tgl adalah tanggal dari transaksi T_1 .
3. Jika data belum habis, cek apakah tgl sama dengan tanggal dari transaksi T_i .
Jika sama, maka T_i masuk ke dalam Partisi P_k , i ditambah 1. Jika tidak sama, maka k ditambah 1 dan set tgl = tanggal dari transaksi T_i .
4. Keluaran berupa data dalam bentuk Partisi, dimana dalam 1 Partisi terdiri dari kumpulan data transaksi dalam 1 hari.

Tahapan diatas ditunjukkan oleh *flowchart* pada **Gambar 3.6**.

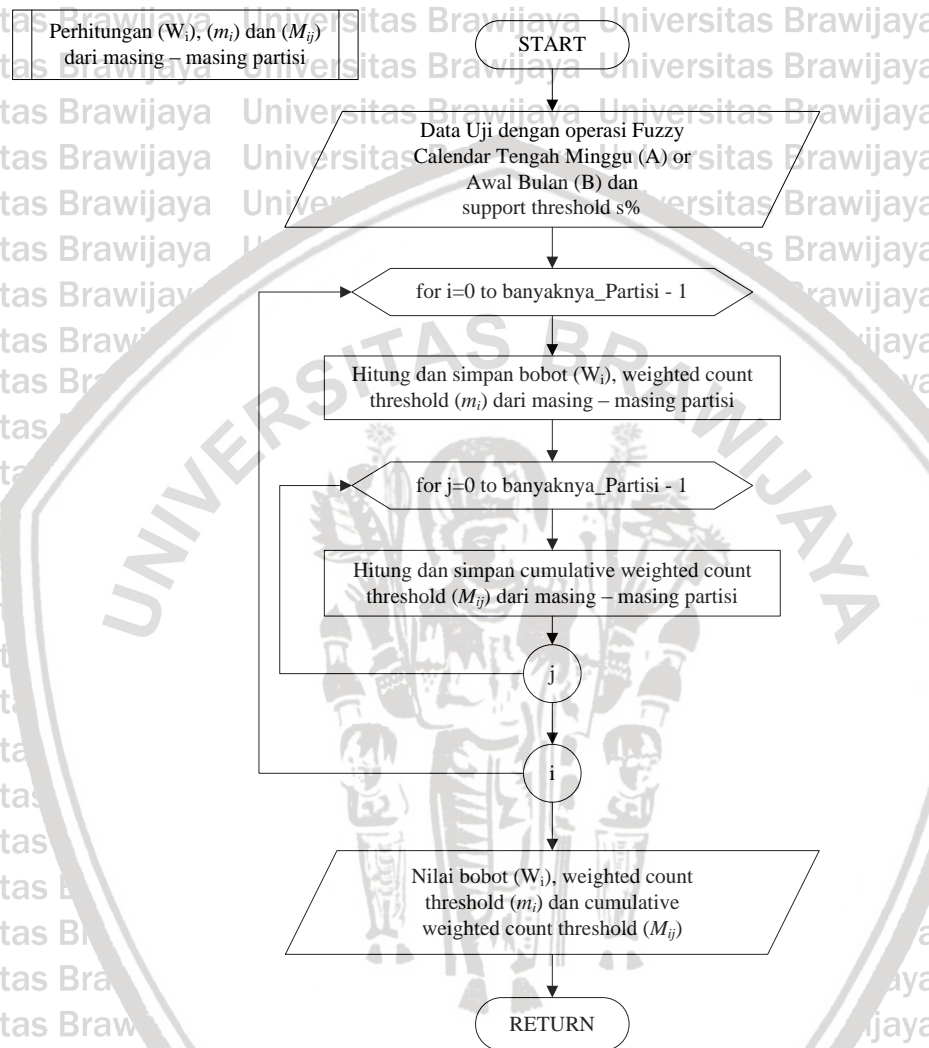


Gambar 3.6 Proses Pembagian Partisi

3.4.2.1.3 Perhitungan bobot, m_i dan M_{ij} dari setiap partisi

Proses perhitungan bobot, m_i dan M_{ij} dari setiap partisi ditunjukkan oleh

Gambar 3.7.



Gambar 3. 7 Proses perhitungan w_i , m_i dan M_{ij}

Tahapan yang dilalui pada proses ini adalah :

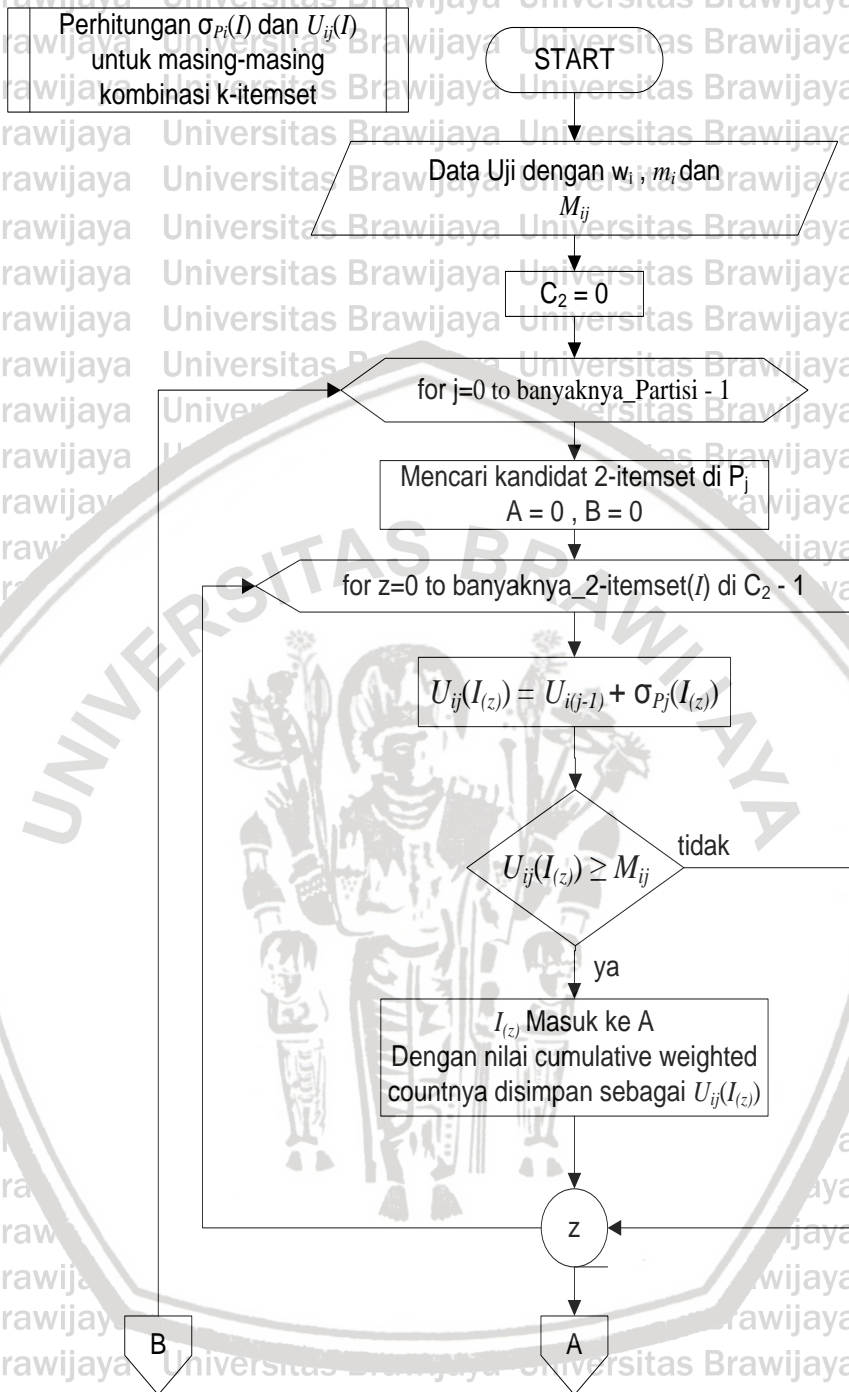
1. Masukan berupa data uji dari basis data D dengan operasi *Fuzzy Calendar* dan *support threshold s%*.
2. Hitung bobot dan *weighted count threshold* (m_i) sesuai persamaan 2.5.
3. Hitung dan *cumulative weighted count threshold* (M_{ij}) dengan persamaan 2.6 dan 2.7.
4. Simpan nilai – nilai tersebut untuk digunakan pada proses selanjutnya.

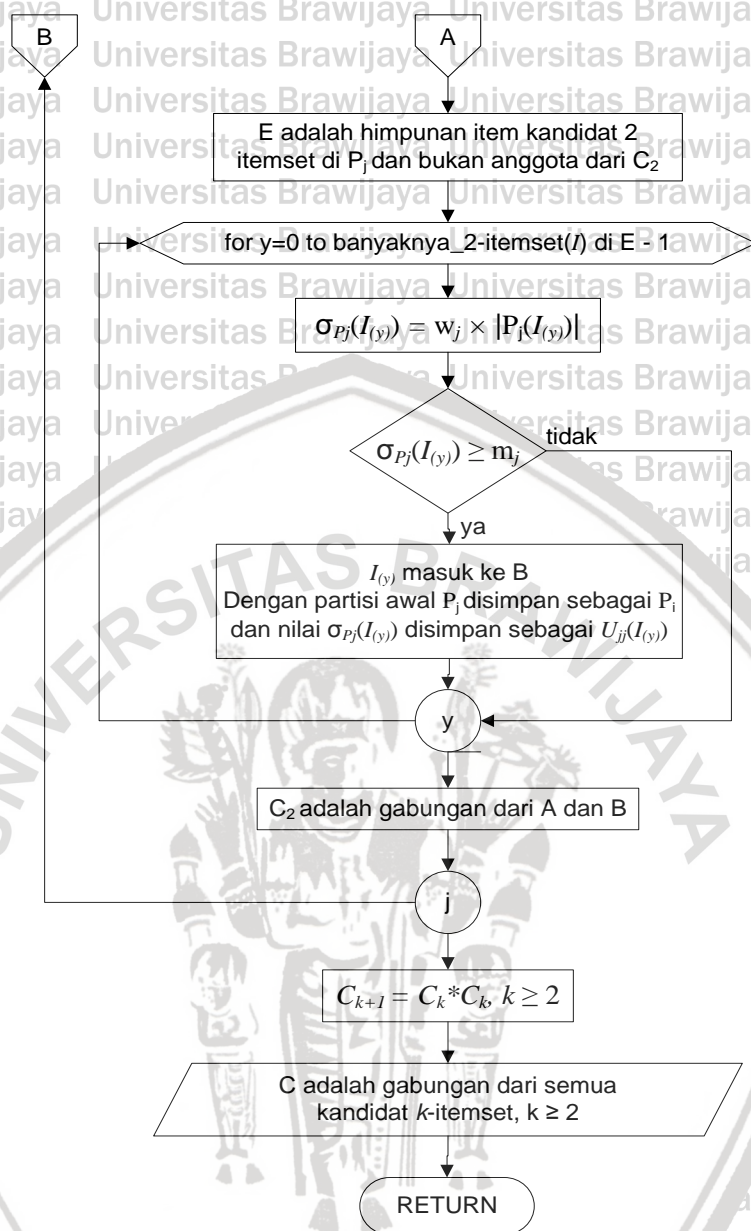
3.4.2.1.4 Perhitungan $\sigma_{P_j}(I)$ dan $U_{ij}(I)$ untuk masing-masing kombinasi k-itemset

Langkah-langkah untuk menemukan himpunan kandidat *frequent itemset* (C) adalah :

- 1 Masukan berupa data uji dan nilai bobot, m_i dan M_{ij} yang telah dihitung sebelumnya.
- 2 Inisialisasikan bahwa C_2 adalah himpunan kosong. Dimana C_2 adalah himpunan yang memuat kandidat 2-itemset.
- 3 Cari semua kemungkinan kombinasi 2-itemset yang ada di Partisi P_j . Inisialisasikan A dan B sebagai himpunan kosong.
- 4 Hitung *cumulative weighted count*(U_{ij}) untuk itemset yang ada di C^2 dengan persamaan 2.10.
- 5 Kemudian cek apakah $U_{ij} \geq M_{ij}$, jika ya maka itemset tersebut masuk ke A.
- 6 Inisialisasikan bahwa E adalah himpunan item kandidat 2-itemset yang ditemukan di Partisi P_j dan bukan anggota dari himpunan C_2 .
- 7 Hitung *weighted count*($\sigma_{P_j}(I)$) untuk itemset yang ada di E dengan persamaan 2.8 dan 2.9.
- 8 Kemudian cek apakah $\sigma_{P_j}(I) \geq m_j$, jika ya maka itemset tersebut masuk ke B, simpan *start* partisi itemset tersebut dengan P_i dan nilai $\sigma_{P_j}(I)$ disimpan sebagai $U_{ij}(I)$.
- 9 Proses penggabungan A dan B menjadi C_2 .
- 10 Dari C_2 , akan diperluas menjadi C_k dengan persamaan 2.13.
- 11 Keluaran dari proses ini adalah himpunan kandidat *frequent itemset* (C) yang didapat dari persamaan 2.14.

Proses pembentukan himpunan C ditunjukkan pada **Gambar 3.8**.

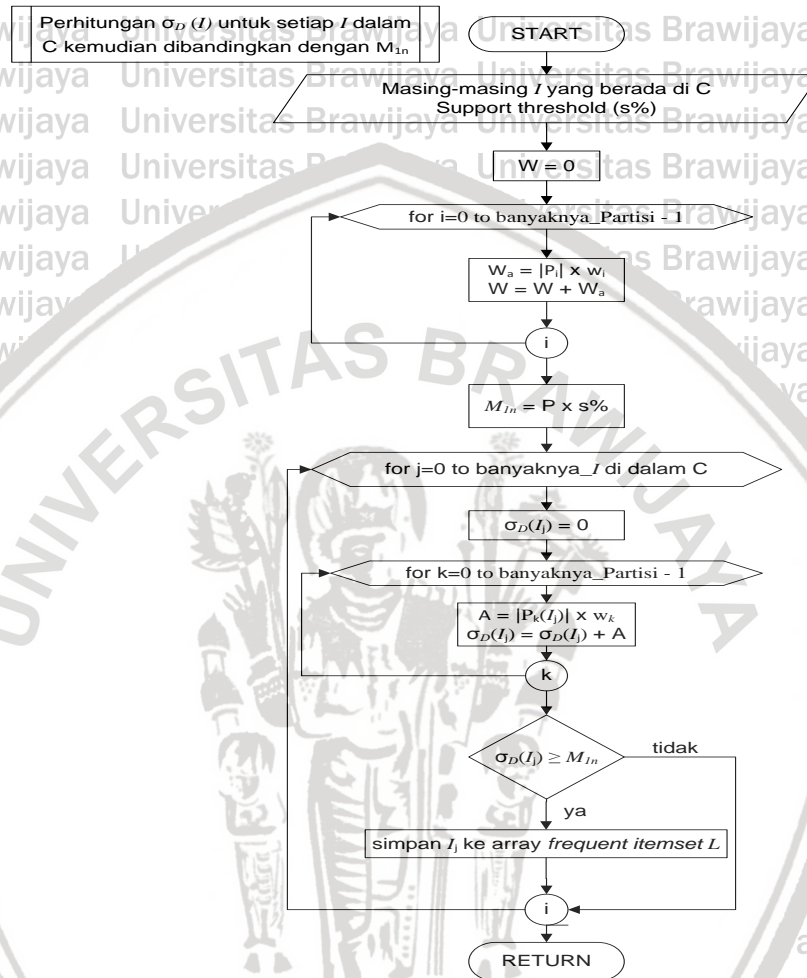




Gambar 3. 8 Proses pencarian kandidat k -itemset (C)

3.4.2.1.5 Perhitungan $\sigma_D(I)$ untuk setiap *itemset* I dalam C kemudian dibandingkan dengan M_{In}

Berikut adalah flowchart pembentukan *frequent itemset* (L) :



Gambar 3. 9 Proses pembentukan *frequent itemset* (L)

Langkah – langkah pembentukan *frequent itemset* (L) adalah :

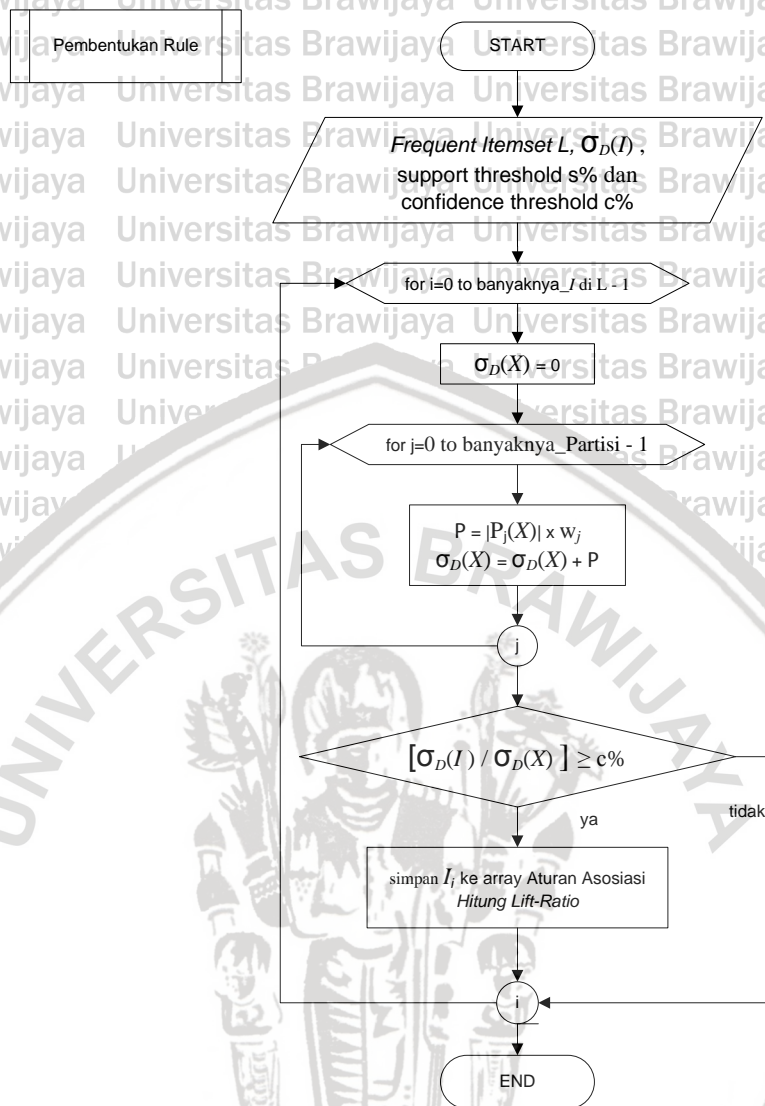
1. Masukan berupa himpunan kandidat *frequent itemset* (C) yang didapat dari proses sebelumnya dan *support threshold* $s\%$.
2. Hitung M_{In} dengan persamaan 2.16.
3. Hitung $\sigma_D(I)$ dengan persamaan 2.15.
4. Cek apakah $\sigma_D(I) \geq M_{In}$, jika ya simpan *itemset* tersebut ke dalam himpunan *frequent itemset* (L).

3.4.2.2 Pembentukan Aturan Asosiasi

Tahapan dalam pembentukan aturan asosiasi :

- 1 Masukkan berupa himpunan *frequent itemset* (L) yang didapat dari proses sebelumnya, *support threshold* $s\%$ dan *confidence threshold* $c\%$.
- 2 Misalkan *itemset* AB ada di L , maka *itemset* AB adalah XUY dan *itemset* A adalah X .
- 3 Hitung *weighted confidence* dengan persamaan 2.20.
- 4 Hitung $\sigma_D(X)$ dengan persamaan 2.15.
- 5 Lakukan pengecekan terhadap nilai *weighted confidence* apakah lebih besar atau sama dengan $c\%$, jika ya simpan *itemset* tersebut ke dalam himpunan aturan asosiasi.
- 6 Keluaran dari proses ini adalah semua aturan asosiasi yang menunjukkan keterkaitan antar alamat *website* pada periode waktu tertentu dengan nilai *Lift Ratio* masing-masing aturan asosiasi yang dihasilkan (gunakan persamaan 2.22).

Flowchart yang menunjukkan proses diatas ditunjukkan pada **Gambar 3.10**.



Gambar 3. 10 Proses pembentukan aturan Asosiasi

3.5 Perhitungan Manual

Perhitungan manual dilakukan pada sampel data yang diambil acak dari data *access log* yang sebelumnya sudah dilakukan *preprocessing* data dan dengan *support threshold* sebesar 40% dan *confidence threshold* 60%. Sampel data yang akan dihitung disebut dengan basis data D, yang akan ditunjukkan pada Tabel 3.7:

Tabel 3. 7 Sampel Data dalam basis data D

Pi	Hari, Tanggal	TID	Item
P1	9/3/2011 Rabu	T1	1,2,3,4
		T2	2,4,5,1,6
		T3	1,6,2,3
P2	10/3/2011 Kamis	T4	3,1,2,4
		T5	4,1,5
		T6	2,6,1
		T7	4,1,6,2
		T8	6,2,3,4
P3	11/3/2011 Jumat	T9	5,2,4,3
		T10	1,5,2,3
		T11	1,2,3,4
		T12	2,3,6,5

Dengan 1 = www.google.com

2 = www.facebook.com

3 = www.gmail.com

4 = www.twitter.com

5 = www.youtube.com

6 = www.yahoo.com

Setiap transaksi yang telah tercatat di dalam basis data D tersebut, akan dihitung bobot dari masing – masing partisi partisi berdasarkan pencocokan derajat keanggotaan dari tanggal transaksi dengan *Fuzzy Calendar* yang telah ditentukan misalnya, “Tengah Minggu(mw)”.

Tabel 3. 8 Perhitungan bobot (w_i)

w_1	μ_{mw} (rabu)	μ_{mw} (3)	$(x - 2) / 2 = (3 - 2) / 2$	0.5
w_2	μ_{mw} (kamis)	μ_{mw} (4)	1	1
w_3	μ_{mw} (jumat)	μ_{mw} (5)	$(6 - x) / 2 = (6 - 5) / 2$	0.5

Setelah bobot dari masing – masing partisi ditentukan, selanjutnya adalah menghitung *weighted count threshold*(m_i) dan *cumulative weighted count threshold* (M_{ij}) di masing – masing partisi dengan *support threshold* (s%) sebesar 40%, menggunakan persamaan 2.5 sampai 2.7.

Tabel 3. 9 Perhitungan m_i dan M_{ij}

m1	$w_1 * P1 * s\%$	0.6	$M_{11} = m_1$	0.6
m2	$w_2 * P2 * s\%$	2	$M_{12} = M_{11} + m_2$	2.6
			$M_{22} = m_2$	2
m3	$w_3 * P3 * s\%$	0.8	$M_{13} = M_{12} + m_3$	3.4
			$M_{23} = M_{22} + m_3$	2.8
			$M_{33} = m_3$	0.8

Langkah selanjutnya adalah mencari semua kemungkinan kombinasi 2-*itemset* di tiap partisi.

1 Kombinasi 2-*itemset* di partisi 1 dengan $w_1 = 0.5$ dan $M_{11} = 0.6$.

Pada tahap ini, C2 diinisialisasikan sebagai himpunan kosong, sehingga setiap kombinasi 2-*itemset* yang ditemukan dihitung dengan persamaan 2.8 dan 2.9. Kemudian bandingkan dengan M_{ij} .

Tabel 3. 10 Perhitungan 2-itemset di P_1

I	$ P_1(I) $	Start	$U_{11} = \sigma_{P_1}(I)$ $= w_1 \times P_1(I) $	CEK $\sigma_{P_1}(I) \geq m_1$
1,2	2	1	1	v
1,3	2	1	1	v
1,4	1	1	0.5	x
1,6	2	1	1	v
2,1	1	1	0.5	x
2,3	2	1	1	v
2,4	2	1	1	v
2,5	1	1	0.5	x
2,6	1	1	0.5	x
3,4	1	1	0.5	x
4,1	1	1	0.5	x
4,5	1	1	0.5	x
4,6	1	1	0.5	x
5,1	1	1	0.5	x
5,6	1	1	0.5	x
6,2	1	1	0.5	x
6,3	1	1	0.5	x

Dari semua kemungkinan kombinasi 2-itemset pada P_1 , ada 5 itemset (I) yang lolos ke- C_2 , yaitu :

Tabel 3. 11 Kandidat 2-itemset (C_2) di P_1

I	start	U_{11}
1,2	1	1
1,3	1	1
1,6	1	1
2,3	1	1
2,4	1	1

2 Dimulai dari partisi ini dan selanjutnya, setiap kombinasi 2-itemset yang ditemukan, akan dicari nilainya dengan 2 cara yaitu :

- Hitung *cumulative weighted count* ($U_{ij}(I)$) dari semua 2-itemset yang sudah ada di C_2 dengan menggunakan persamaan 2.10 dan bandingkan nilainya dengan persamaan 2.11.
- Hitung *weighted count* ($\sigma_{P_j}(I)$) dari semua 2-itemset yang ditemukan pada partisi ini dengan menggunakan persamaan 2.8 dan bandingkan nilainya dengan *weighted count threshold* (m_j).

Kombinasi 2-itemset di partisi 2 dengan $w_2 = 1$, $m_2 = 2$ dan $M_{12} = 2.6$.

Untuk 2-itemset yang sudah ada di C_2 :

Tabel 3. 12 Perhitungan 2-itemset di C_2

I	$ P_2(I) $	start	$U_{12} = U_{11} + \sigma_{P_2(I)}$ $= U_{11} + (w_2 \times P_2(I))$	CEK $U_{12} \geq M_{12}$
1,2	2	1	3	v
1,3	0	1	1	x
1,6	1	1	2	x
2,3	1	1	2	x
2,4	2	1	3	v

Untuk 2-itemset dari Partisi 2 yang belum ada di C_2 :

Tabel 3. 13 Perhitungan 2-itemset di P_2

I	$ P_2(I) $	start	$U_{22} = \sigma_{P_2(I)} = w_2 \times P_2(I) $	CEK $\sigma_{P_2(I)} \geq m_2$
1,4	1	2	1	x
1,5	1	2	1	x
2,1	1	2	1	x
2,6	1	2	1	x
3,1	1	2	1	x
3,2	1	2	1	x
3,4	2	2	2	v
4,1	2	2	2	v
4,2	1	2	1	x
4,5	1	2	1	x
4,6	1	2	1	x
6,1	1	2	1	x
6,2	2	2	2	v
6,3	1	2	1	x
6,4	1	2	1	x

Dari semua kemungkinan kombinasi 2-itemset pada P_2 , ada 5 itemset (I) yang lolos ke- C_2 , yaitu :

Tabel 3. 14 Kandidat 2-itemset (C_2) di P_2

I	start	U_{12}
1,2	1	3
2,4	1	3
I	start	U_{22}
3,4	2	2
4,1	2	2
6,2	2	2

3. Kombinasi 2-*itemset* di partisi 3 dengan $w_3 = 0.5$, $m_3 = 0.8$, $M_{13} = 3.4$ dan $M_{23} = 2.8$.

Untuk 2-*itemset* yang sudah ada di C_2 :

Tabel 3. 15 Perhitungan 2-*itemset* di C_2

I	$ P_3(I) $	start	$U_{13} = U_{12} + \sigma_{P_3(I)}$ $= U_{12} + (w_3 \times P_3(I))$	CEK $U_{13} \geq M_{13}$
1,2	2	1	4	v
2,4	2	1	4	v

Tabel 3. 16 Perhitungan 2-*itemset* di C_2

I	$ P_3(I) $	start	$U_{23} = U_{22} + \sigma_{P_3(I)}$ $= U_{22} + (w_3 \times P_3(I))$	CEK $U_{23} \geq M_{23}$
34	1	2	2.5	x
41	0	2	2	x
62	0	2	2	x

Untuk 2-*itemset* dari Partisi 3 yang belum ada di C_2 :

Tabel 3. 17 Perhitungan 2-*itemset* di P_3

I	$ P_3(I) $	start	$U_{33} = \sigma_{P_3(I)} = (w_3 \times P_3(I))$	CEK $U_{33} \geq m_3$
1,3	2	3	1	v
1,4	1	3	0.5	x
1,5	1	3	0.5	x
2,3	4	3	2	v
2,5	1	3	0.5	x
2,6	1	3	0.5	x
3,5	1	3	0.5	x
3,6	1	3	0.5	x
4,3	1	3	0.5	x
5,2	2	3	1	v
5,3	2	3	1	v
5,4	1	3	0.5	x
6,5	1	3	0.5	x

Dari semua kemungkinan kombinasi 2-*itemset* pada P_3 , ada 6 *itemset* (I) yang lolos ke- C_2 , yaitu :

Tabel 3. 18 Kandidat 2-*itemset* (C_2)

I	start	U_{13}
12	1	4
24	1	4

I	start	U ₃₃
13	3	1
23	3	2
52	3	1
53	3	1

Dari kandidat yang ada di C_2 akan dicari kandidat k -itemset (C_k) dengan persamaan 2.13.

Tabel 3. 19 Kandidat 3-itemset (C_3)

CANDIDATE 3 ITEMSET
1,2 - 2,4 = 1,2,4
1,2 - 2,3 = 1,2,3
5,2 - 5,3 = 5,2,3
5,2 - 5,4 = 5,2,4

Dari C_2 dan C_3 akan dibentuk himpunan C yang merupakan gabungan dari setiap kandidat k -itemset, sesuai dengan persamaan 2.14.

Penerus selanjutnya adalah mencari *frequent itemset*, L , dengan memeriksa basis data sekali lagi. Untuk setiap I dalam C , *weighted count* dari itemset I pada basis data D , dinotasikan $\sigma_D(I)$ digunakan untuk menghitung seberapa sering itemset I muncul dalam basis data D , dengan menggunakan persamaan 2.15.

Kemudian akan dibandingkan dengan *weighted count threshold* dari D , atau disimbolkan menjadi M_{1n} sesuai persamaan 2.16. Jika nilai $\sigma_D(I)$ lebih besar atau sama dengan M_{1n} , maka I adalah *frequent itemset* (L) dalam basis data D .

Tabel 3. 20 Perhitungan *frequent itemset* (L) dari C

I	P	$\sigma_D(I)$	cek $\sigma_D(I) \geq M_{1n}$
12	2,2,2	4	v
24	2,2,2	4	v
13	2,0,2	2	x
23	2,1,4	4	v
52	0,0,2	1	x
53	0,0,2	1	x
124	1,1,1	2	x
123	2,0,2	2	x
523	0,0,2	1	x
524	0,0,1	0.5	x

Dari tabel diatas, yang termasuk himpunan *frequent itemset* L adalah 1,2; 2,4 dan 2,3. Langkah terakhir, dari L akan dicari aturan asosiasi sesuai dengan

persamaan 2.19 dan 2.20, serta *itemset* tersebut dikatakan aturan asosiasi jika nilai *confidence*-nya lebih besar atau sama dengan nilai *confidence threshold*(c%).

Dimana pada ilustrasi ini, nilai *confidence threshold* sebesar 60%.

Tabel 3. 21 Pembentukan Aturan Asosiasi

I	$\sigma_D(I)$	$\sigma_D(X)$	C_I	cek $C_I \geq c\%$.
1,2	4	6.5	0.61538462	v
2,4	4	7.5	0.53333333	x
2,3	4	7.5	0.53333333	x

Dari tabel diatas dapat disimpulkan bahwa aturan asosiasi yang terbentuk adalah $1 \rightarrow 2$, yang dapat diartikan dengan operasi *Fuzzy Calendar* “Tengah Minggu(MW)” pengguna *internet* di lingkungan Brawijaya, jika mengakses *google.com* cenderung akan membuka *facebook.com*.

Selanjutnya *rule* yang telah lolos pada **Tabel 3.22** dilakukan uji kekuatan *rule* dengan menggunakan *Lift Ratio*. Hasil perhitungan nilai *Lift Ratio* dapat dilihat pada Tabel 3.22.

Tabel 3. 22 *Lift Ratio* Aturan Asosiasi

I	CI	$\sigma_D(Y)$	benchmark	$LR = CI / \text{benchmark}$
1,2	0.615385	7.5	0.8823529411764706	0.6974358974358975

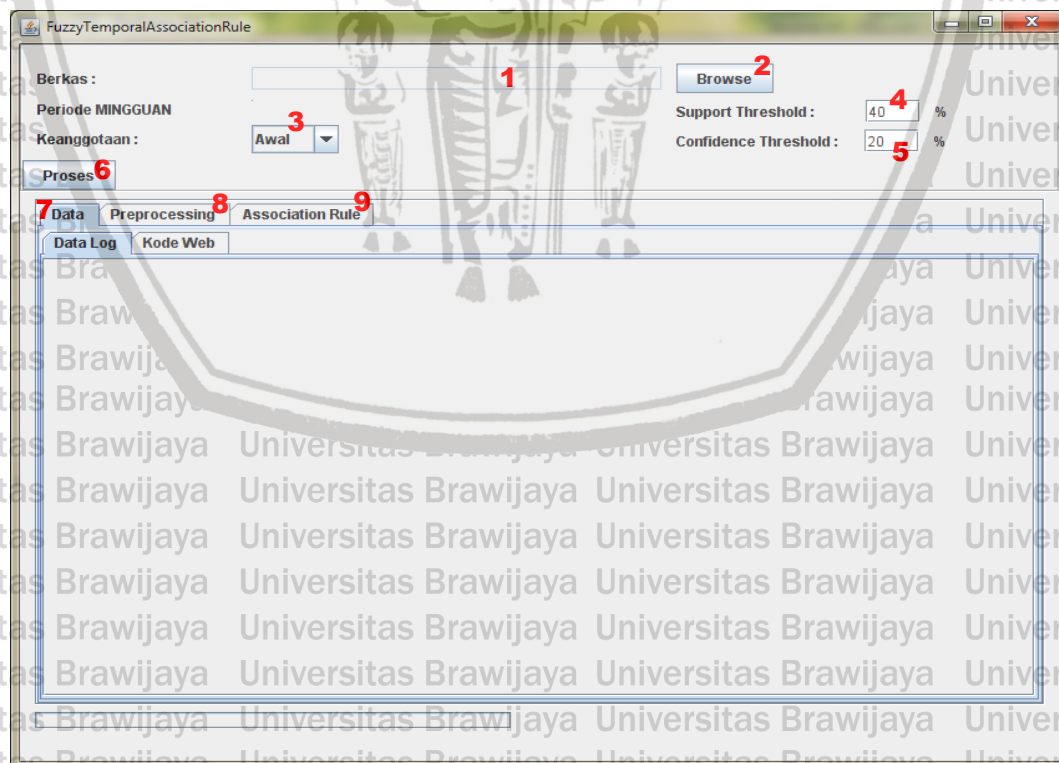
Dari tabel tersebut dapat dikatakan bahwa kekuatan *rule* $1 > 2$ sebesar 0,697 yang artinya *rule* tersebut kurang baik karena nilainya masih dibawah 1.

3.6 Rancangan Antar Muka

Pada bagian ini akan dijelaskan antarmuka sistem yang akan dibuat. Gambar

3.12 menunjukkan rancangan antarmuka proses *Fuzzy Temporal Association rules*. Berikut adalah penjelasan **Gambar 3.11** berdasarkan nomor :

1. Menampilkan nama *File data Access log* yang akan di-*input*-kan.
2. Tombol *browse* untuk menampilkan kotak *open file* yang digunakan untuk memilih *file input* data yang akan diproses.
3. Pilihan *fuzzy calendar* (Awal Minggu, Tengah Minggu dan Akhir Minggu)
4. Input *support threshold* sebagai batas nilai *support*.
5. Input *confidence threshold* sebagai batas nilai *confidence*.
6. Tombol Proses untuk melakukan proses pengujian data.
7. Submenu Data untuk menampilkan data mentah *access log*.
8. Submenu *preprocessing* untuk menampilkan data yang telah di-*preprocessing* dan sudah dalam bentuk transaksi.
9. Submenu *Association rule* untuk menampilkan aturan asosiasi yang telah telah dihasilkan.



Gambar 3. 11 Tampilan Rancangan Antarmuka

3.7 Rancangan Uji Coba

Uji coba sistem dilakukan untuk menganalisa keterkaitan antar *website* yang diakses pengguna *internet* di lingkungan Universitas Brawijaya pada periode waktu “Awal, Tengah atau Akhir Minggu”. Tujuan dari uji coba ini adalah untuk mengetahui pengaruh *support threshold* dan *confidence threshold* terhadap jumlah aturan asosiasi yang dihasilkan. Uji pengaruh nilai *threshold* dan *confidence threshold* dilakukan pada sejumlah data dengan periode bulan yang berbeda, dimulai dari bulan Maret 2011 sampai bulan Mei 2011.

Tabel 3. 23 Tabel Uji Pengaruh Nilai *Support threshold* dan *Confidence threshold* terhadap jumlah aturan asosiasi yang dihasilkan

Fuzzy Calendar	<i>Support threshold</i> (%)	<i>Confidence threshold</i> (%)	Jumlah Aturan Asosiasi

Tabel 3. 24 Tabel Uji *Lift Ratio Association Rules*

Fuzzy Calendar	Aturan (<i>Rule</i>)	<i>Confidence</i> (%)	<i>Lift Ratio</i>

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

4.1 Lingkungan Implementasi

Implementasi perangkat lunak ini berupa aplikasi *Fuzzy Temporal Association Rules* untuk mengetahui kombinasi akses kunjungan *website* oleh pengguna *internet* di lingkungan universitas Brawijaya. Adapun yang akan dijelaskan dalam subbab ini meliputi lingkungan implementasi perangkat keras dan perangkat lunak.

4.1.1 Lingkungan Perangkat Keras

Perangkat keras yang digunakan pada penelitian ini adalah sebuah PC (Personal Computer) dengan spesifikasi sebagai berikut :

- 1 AMD Athlon(tm) Neo Processor MV-40 1.60 GHz
- 2 Memori 2.00 GB
- 3 Harddisk dengan kapasitas 320 GB
- 4 Monitor 10.0"

4.1.2 Lingkungan Perangkat Lunak

Perangkat lunak yang digunakan pada penelitian ini adalah sebagai berikut :

- 1 Sistem operasi Windows 7 Ultimate
- 2 Editor Java (Eclipse Indigo) dan JDK 1.6 dalam mengembangkan aplikasi pencarian aturan asosiasi.

4.2 Implementasi Program

Berdasarkan rancangan pembuatan sistem pada bab 3, maka pada subbab ini akan dijelaskan implementasi proses – proses tersebut. Secara garis besar proses dikelompokkan menjadi, yaitu *preprocessing* data *access log*, pembentukan *frequent itemset* dan pembentukan aturan asosiasi.

4.2.1 Implementasi Tahap *preprocessing* data access log

Pada tahap ini dilakukan proses untuk menghasilkan data dalam bentuk transaksi, yang awalnya berasal dari data *access log*. Karena data tersebut adalah data mentah, maka diperlukan sebuah proses untuk mengekstrak informasi yang dibutuhkan untuk proses selanjutnya.

4.2.1.1 Implementasi Pembacaan Data *access log*

Pada tahap ini dilakukan proses untuk pembacaan data dari *file* yang bertipe *.txt*. Kemudian data tersebut dipecah berdasarkan spasi sehingga terbentuk 10 kolom.

```

1  public static ArrayList<DataLog> preprocessing(String filename)
2  throws Exception {
3
4  ArrayList<DataLog> dataLogs = new ArrayList<DataLog>();
5  File readerFileAccessLog = new File(filename);
6  ArrayList<Object> lineFile = readerFileAccessLog.read();
7
8  //Pre processing
9  for (Object line : lineFile) {
10     String [] lineSplit = line.toString().split(" ");
11
12     String urlAsli = lineSplit[6];
13
14     StringBuilder sb = new StringBuilder();
15     int b = 0;
16     for (int i = 0; i < urlAsli.length(); i++) {
17         String s = urlAsli.substring(i, i + 1);
18         if (s.equals("/")) {
19             b = b + 1;
20         }
21         if (b == 3) {
22             break;
23         } else {
24
25             sb.append(s);
26         }
27     }
28
29     DataLog dataLog = new DataLog(Double.parseDouble(lineSplit[0]),
30     lineSplit[2], lineSplit[3], sb.toString(), lineSplit[9]);
31
32     //unixtime - ikut waktunya indonesia (lebih siang dari gmt[+7])
33     SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
34     Long l = Long.parseLong(lineSplit[0].substring(0, 10));
35     Date date = new Date(l * 1000);
36     Calendar c = Calendar.getInstance();
37     c.setTime(date);
38     dataLog.setDayOfWeek(c.get(Calendar.DAY_OF_WEEK) == 1 ? 7 :
39     c.get(Calendar.DAY_OF_WEEK) - 1);

```



```

40 dataLog.setDateOfMonth(c.get(Calendar.DAY_OF_MONTH));
41
42 dataLog.setRealDateTime(sdf.format(date));
43 dataLogs.add(dataLog);
44
45 }
46     return dataLogs;
47 }
48 }

```

Source Code 4. 1 Implementasi Pembacaan Data access log

Method `preprocessing()` digunakan untuk melakukan pembacaan data dan melakukan pemecahan tiap-tiap baris data *access log* berdasarkan spasi. Dalam *method* ini juga dilakukan pemecahan terhadap *URL* berdasarkan "/" dan konversi *unixtime* menjadi data dengan format "yyyy-MM-dd HH:mm:ss". Setiap baris data yang telah dipecah berdasarkan spasi, disimpan dalam `ArrayList<DataLog>` dengan nama `dataLogs`.

4.2.1.2 Implementasi filtering data access log

Pada tahap ini dilakukan proses untuk menyaring data *access log* berdasarkan status, tipe *file*-nya dan *URL* yang berisi *IP address*.

```

1 public static ArrayList<DataLog> filtering(ArrayList<DataLog>
2 dataLogs) throws Exception {
3
4 ArrayList<DataLog> dataLogsFiltered = new ArrayList<DataLog>();
5
6 //Filtering
7 for (DataLog dataLog : dataLogs) {
8     //cek content
9     if (dataLog.getCode().contains("/2") &&
10        (dataLog.getType().contains("html") ||
11         (dataLog.getType().contains("application") &&
12          !dataLog.getType().contains("javascript"))) ) ) {
13
14 //cek ip
15 String pureHost = dataLog.getUrl().replace("http://", "");
16 pureHost = pureHost.replace("https://", "");
17 pureHost = pureHost.replace("ftp://", "");
18 if (!isIPAddress(pureHost)) {
19 dataLogsFiltered.add(dataLog);
20 }
21 }
22 }
23 }
24     return dataLogsFiltered;
25 }
26 }

```

Source Code 4. 2 Implementasi filtering data access log

`Method filtering()` ini melakukan perulangan sebanyak baris data *access log* untuk melakukan pemangkasan terhadap `dataLog.getCode()` yang berisi status, `dataLog.getType()` yang berisi *type* dan `dataLog.getUrl()` yang berisi *URL*.

4.2.1.3 Implementasi pembentukan transaksi data *access log*

Pada tahap ini dilakukan proses untuk membagi data berdasarkan waktu dan *IP address*. Sehingga dapat dikatakan bahwa 1 transaksi terdiri dari alamat-alamat *website* yang diakses oleh 1 *IP address* dalam waktu 30 menit.

```

1  public static ArrayList<Transaksi> generateTransaksi
2  (ArrayList<DataLog> dataLogs, double interval) throws Exception {
3
4  ArrayList<Transaksi> transaksies = new ArrayList<Transaksi>();
5  ArrayList<ArrayList<DataLog>> groupWaktu = new
6  ArrayList<ArrayList<DataLog>>();
7
8  ArrayList<DataLog> dataLogInTransaksi = new ArrayList<DataLog>();
9
10 DataLog start = dataLogs.get(0);
11 for (DataLog dataLog: dataLogs) {
12     if (dataLogInTransaksi.size() == 0) {
13         start = dataLog;
14         dataLogInTransaksi.add(dataLog);
15     } else {
16         if ( (start.getUnixTime() + interval) >
17             dataLog.getUnixTime()) {
18             dataLogInTransaksi.add(dataLog);
19         } else {
20             ArrayList<DataLog> addDataAkses = new ArrayList<DataLog>();
21
22             addDataAkses.addAll(dataLogInTransaksi);
23             groupWaktu.add(addDataAkses);
24             start = dataLog;
25             dataLogInTransaksi.clear();
26             dataLogInTransaksi.add(dataLog);
27         }
28     }
29 }
30
31 //add Last partition
32 if (dataLogInTransaksi.size() > 0) {
33     ArrayList<DataLog> addDataAkses = new ArrayList<DataLog>();
34     addDataAkses.addAll(dataLogInTransaksi);
35     groupWaktu.add(addDataAkses);
36 }
37
38 for (ArrayList<DataLog> groupWaktuItem : groupWaktu) {
39     ArrayList<String> hostUser = new ArrayList<String>();
40

```

```

41 //find IPs on group waktu item
42 for (DataLog dataLog : groupWaktuItem) {
43     if (!hostUser.contains(dataLog.getHost())) {
44         hostUser.add(dataLog.getHost());
45     }
46 }
47
48 //add to transaksi
49 for (String host : hostUser) {
50     ArrayList<DataLog> dataTransIP = new ArrayList<DataLog>();
51     for (DataLog dataLog : groupWaktuItem) {
52         if (dataLog.getHost().compareToIgnoreCase(host) == 0) {
53             dataTransIP.add(dataLog);
54         }
55     }
56
57     if (dataTransIP.size() >= 2) {
58         ArrayList<DataLog> distinctTrans = new ArrayList<DataLog>();
59         ArrayList<Integer> distinctKodeWeb = new ArrayList<Integer>();
60
61         for (DataLog dataLog : dataTransIP) {
62             if (!distinctKodeWeb.contains(dataLog.getKodeWeb())) {
63                 distinctKodeWeb.add(dataLog.getKodeWeb());
64                 distinctTrans.add(dataLog);
65             }
66         }
67         if (distinctTrans.size() >= 2)
68             transaksies.add(new Transaksi(distinctTrans, host));
69     }
70 }
71 }
72     return transaksies;
73 }

```

Source Code 4. 3 Implementasi pembentukan transaksi data *access log*

Method generateTransaksi() melakukan 3 aktifitas di dalam perulangan terhadap jumbal baris data *access log* yang telah disaring. Yang pertama adalah membagi data berdasarkan waktu (30 menit) kemudian disimpan ke dalam ArrayList<DataLog> dengan nama groupWaktu. Yang kedua membagi berdasarkan *IP address*-nya setelah data di-group berdasarkan waktu dan disimpan ke dalam ArrayList<String> dengan nama hostUser. Sehingga data yang telah di-filter dari proses sebelumnya sudah dalam bentuk transaksi. Yang ketiga melakukan filtrasi terhadap data transaksi, dengan aturan, 1 transaksi paling sedikit terdapat 2 *item web* yang diakses dan tidak boleh ada *item* yang sama dalam 1 transaksi.

4.2.1.4 Implementasi pembentukan partisi data *access log*

Pada tahap ini dilakukan proses untuk membentuk data transaksi dalam partisi. 1 partisi merupakan kumpulan transaksi yang ada dalam 1 hari.

```

1 public static ArrayList<ArrayList<Transaksi>> generatePartisi
2 (ArrayList<Transaksi> transaksi) throws Exception {
3
4     ArrayList<ArrayList<Transaksi>> partisi = new
5     ArrayList<ArrayList<Transaksi>>();
6
7     int i = 0;
8     ArrayList<Transaksi> transInPartisi = new ArrayList<Transaksi>();
9
10    while (i < transaksi.size()) {
11        if (transInPartisi.size() == 0) {
12            transInPartisi.add(transaksi.get(i));
13        } else {
14            if (transInPartisi.get(0).getDataAkses().get(0)
15                .getRealDateTIme().substring(0, 10)
16                .compareToIgnoreCase(transaksi.get(i).getDataAkses().get(0)
17                    .getRealDateTIme().substring(0, 10)) == 0) {
18
19                transInPartisi.add(transaksi.get(i));
20            } else {
21                ArrayList<Transaksi> addTransInPartisi = new
22                ArrayList<Transaksi>();
23                addTransInPartisi.addAll(transInPartisi);
24                partisi.add(addTransInPartisi);
25                transInPartisi.clear();
26                transInPartisi.add(transaksi.get(i));
27            }
28        }
29        i++;
30    }
31
32    //add Last partition
33    if (transInPartisi.size() > 0) {
34        ArrayList<Transaksi> addTransInPartisi = new
35        ArrayList<Transaksi>();
36        addTransInPartisi.addAll(transInPartisi);
37        partisi.add(addTransInPartisi);
38    }
39
40    return partisi;
41 }

```

Source Code 4. 4 Implementasi pembentukan partisi data *access log*

Method generatePartisi() digunakan untuk membentuk data yang sudah berbentuk transaksi ke dalam partisi-partisi. Dimana 1 partisi adalah kumpulan transaksi dalam 1 hari. Didalam *method* ini dilakukan pengecekan terhadap tanggal di tiap data transaksi, jika selisih tanggal transaksi dengan transaksi

sebelumnya bernilai 0, maka transaksi tersebut masuk dalam 1 partisi yang sama.

Jika selisihnya lebih dari 0, maka kedua transaksi tersebut berbeda partisi. Langkah tersebut terlaetak pada baris 10-30.

4.2.2 Implementasi Tahap pembentukan *frequent itemset*

Pada tahap ini dilakukan proses untuk membentuk *frequent itemset* dari kandidat *k-itemset* yang sudah terbentuk.

4.2.2.1 Implementasi *fuzzy* kalender

Pada tahap ini dilakukan proses untuk menghitung nilai bobot (w) tiap partisi yang dihitung berdasarkan pencocokan derajat keanggotaan dari waktu transaksi dengan *Fuzzy Calendar* yang telah ditentukan. *Weighted count threshold*(m_i) dan *cumulative weighted count threshold*(M_{ij}) dihitung untuk menyaring pola asosiasi yang penting di dalam sebuah partisi. Juga dihitung *weighted count threshold* dari $D(M_{In})$ untuk menyaring kandidat menjadi *frequent itemset*.

```

1  public void countW(Constant periode, Constant keanggotaan) {
2
3  double Ww = 0.0;
4  if (periode == Constant.MINGGUAN) {
5  if (keanggotaan == Constant.AWAL) {
6  if (this.transaksi.get(0).getDataAkses()
7  .get(0).getDayOfWeek() < 2) {
8  Ww = 1;
9  } else if (this.transaksi.get(0).getDataAkses()
10 .get(0).getDayOfWeek() > 4) {
11     Ww = 0;
12 } else {
13 Ww = (4.0 - this.transaksi.get(0).getDataAkses()
14 .get(0).getDayOfWeek()) / 2.0;
15 }
16 } else if (keanggotaan == Constant.TENGAH) {
17 if (this.transaksi.get(0).getDataAkses()
18 .get(0).getDayOfWeek() < 2 || this.transaksi.get(0)
19 .getDataAkses().get(0).getDayOfWeek() > 6) {
20     Ww = 0;
21 } else if (this.transaksi.get(0).getDataAkses()
22 .get(0).getDayOfWeek() == 2 || this.transaksi.get(0)
23 .getDataAkses().get(0).getDayOfWeek() == 3) {
24     Ww = (this.transaksi.get(0).getDataAkses()
25 .get(0).getDayOfWeek() - 2.0) / 2.0;
26 } else {
27     Ww = (6.0 - this.transaksi.get(0).getDataAkses()
28 .get(0).getDayOfWeek()) / 2.0;
29 }

```

```

30     }
31 }
32 } else if (keanggotaan == Constant.AKHIR) {
33     if (this.transaksi.get(0).getDataAkses()
34         .get(0).getDayOfWeek() < 4) {
35         Ww = 0.0;
36     } else if (this.transaksi.get(0).getDataAkses()
37         .get(0).getDayOfWeek() > 6) {
38         Ww = 1.0;
39     } else {
40         Ww = (this.transaksi.get(0).getDataAkses()
41             .get(0).getDayOfWeek() - 4.0) / 2.0;
42     }
43 }
44 }
45     this.w = Ww;
46 }

```

Source Code 4. 5 Implementasi penghitungan *Fuzzy Calendar*

Method `countW()` digunakan untuk menghitung bobot masing – masing partisi berdasarkan *Fuzzy Calendar* Awal Minggu, Tengah Minggu dan Akhir Minggu.

```

1 public void count_m(double supportThreshold) {
2     this.m = this.w * transaksi.size() * supportThreshold;
3 }

```

Source Code 4. 6 Implementasi penghitungan nilai *Weighted count threshold*

Method `count_m()` digunakan untuk menghitung nilai *Weighted count threshold*(m_i) dari masing-masing partisi, yang sebelumnya telah dihitung bobot dan *support threshold* dari masukan pengguna.

```

1 Double [][] M = new Double[partisiData.size() + 1]
2     [partisiData.size() + 1];
3
4 for (int p = 1; p <= partisiData.size(); p++) {
5     for (int q = p; q <= partisiData.size(); q++) {
6         if (q == p) {
7             M[p][p] = partisiData.get(p - 1).get_m();
8         } else {
9             M[p][q] = M[p][q - 1] + partisiData.get(q - 1).get_m();
10        }
11    }
12 }
13
14 Double M1n = 0.0;
15
16 for (Partisi ps : partisiData)
17     M1n += (ps.getTransaksi().size() * ps.getW());
18
19 M1n *= supportThreshold;

```

Source Code 4.7 Implementasi penghitungan nilai *cumulative weighted count threshold*

Source code 4.7 pada baris 1-12 melakukan penghitungan nilai *cumulative weighted count threshold* (M_{ij}) yang disimpan ke dalam array 2 dimensi dengan nama $M[i][j]$

Untuk baris 14-19 digunakan untuk menghitung nilai *weighted count threshold* dari $D(M_{in})$ yang disimpan ke dalam $M1n$ dengan tipe data Double.

4.2.2.2 Implementasi pembentukan kandidat itemset

Pada tahap ini dimulai dari proses untuk membentuk kandidat *itemset* (C) dengan membandingkan nilai *cumulative weighted count itemset* $I(U_{ij}(I))$ dengan *cumulative weighted count threshold* (M_{ij}).

```

1 public static ArrayList<Itemset> generateKombinasi
2     (Partisi partisi, int numKombinasi) {
3     ArrayList<Itemset> kombinasi = new ArrayList<Itemset>();
4
5     ArrayList<Integer> distinctKodeWeb = new ArrayList<Integer>();
6
7     ArrayList<ArrayList<Integer>> kodeWebInTransaksi = new
8     ArrayList<ArrayList<Integer>>();
9
10    for (Transaksi trans : partisi.getTransaksi()) {
11    ArrayList<Integer> kodeWebInTransaksiItem = new
12    ArrayList<Integer>();
13
14    for (DataLog dataLog : trans.getDataAkses()) {
15
16    kodeWebInTransaksiItem.add(dataLog.getKodeWeb());
17    if (!distinctKodeWeb.contains(dataLog.getKodeWeb())) {
18    distinctKodeWeb.add(dataLog.getKodeWeb());
19    }
20    }
21    kodeWebInTransaksi.add(kodeWebInTransaksiItem);
22    }
23
24    if (numKombinasi == 2) {
25    //generate kombinasi
26    ArrayList<ICombinatoricsVector<Integer>> combination =
27    FuzzyTAS.getKombinasiKodeWeb(distinctKodeWeb, numKombinasi);
28
29    for (ICombinatoricsVector<Integer> comb : combination) {
30    ArrayList<Integer> items = new ArrayList<Integer>();
31    for (Integer kw : comb.getVector()) {
32    items.add(kw);
33    }
34    Itemset newItemset = new Itemset(items);
35

```

```

36 //cek apakah ada pada transaksi
37 for (ArrayList<Integer> kodeWebInTransaksiItem :
38     kodeWebInTransaksi) {
39
40     ArrayList<Integer> checkKodeWebInTransaksiItem = new
41     ArrayList<Integer>();
42
43     checkKodeWebInTransaksiItem.addAll(kodeWebInTransaksiItem);
44     checkKodeWebInTransaksiItem.retainAll(newItemset.getItems());
45
46     boolean inTransaksi = false;
47     if (checkKodeWebInTransaksiItem.size() >= numKombinasi) {
48         for (int i = 0; i < numKombinasi; i++) {
49             if (newItemset.getItems().get(i) ==
50                 checkKodeWebInTransaksiItem.get(i)) {
51                 inTransaksi = true;
52             } else {
53                 inTransaksi = false;
54                 break;
55             }
56         }
57     }
58
59     if (inTransaksi) {
60         newItemset.setpOnTransaksi(newItemset.getpOnTransaksi() + 1);
61         newItemset.setU(0, partisi.getW());
62         newItemset.setOnKandidat(partisi.get_m());
63     }
64 }
65
66 if (newItemset.getpOnTransaksi() > 0) {
67     kombinasi.add(newItemset);
68 }
69 }
70 } else {
71 }
72 }
73
74 return kombinasi;
75 }
76

```

Source Code 4. 8 Implementasi *generate* kombinasi

```

1 public void setOnKandidat(double M) {
2     if (this.U >= M) {
3         this.isOnKandidat = true;
4     } else {
5         this.isOnKandidat = false;
6     }
7 }

```

Source Code 4. 9 Implementasi *filtering* kandidat

Method generateKombinasi() melakukan 2 proses utama. Proses pertama (baris 24-71) adalah melakukan *generate* kombinsi 2 *itemset* dari masing-

masing transaksi, dimana pada baris 36-64 melakukan pencarian frekuensi kemunculan 2 *itemset* hasil *generate* di dalam transaksi per partisi.

Kemudian di dalam *method* *setOnKandidat()* melakukan filtering terhadap 2 *item* hasil *generate* yang memiliki nilai *weighted count itemset* $I(U)$ kurang dari nilai *cumulative weighted count threshold* (M).

4.2.2.3 Implementasi pembentukan *frequent itemset*

Pada tahap ini dilakukan proses untuk mencari *frequent itemset* (L) dengan membandingkan nilai *weighted count itemset* $I(\sigma_D(I))$ dengan *weighted count threshold* (M_{in}).

```

1 //frequent itemset (L)
2 //hitung nilai bobot i
3 ArrayList<Itemset> itemsetUpdated = new ArrayList<Itemset>();
4
5 for (Itemset itsOnKand : kandidat.getItemsets()) {
6     Itemset newItemset = new Itemset(itsOnKand.getItems());
7     newItemset.countBobot_i(partisiData);
8     //cek on kandidat ?
9     newItemset.setOnKandidatFrequentItemset(M1n);
10    if (newItemset.isOnKandidat()) {
11        itemsetUpdated.add(newItemset);
12    }
13 }
14
15 kandidat.setItemsets(itemsetUpdated);

```

Source Code 4. 10 Implementasi pembentukan *frequent itemset*

```

1 public void setOnKandidatFrequentItemset(double M) {
2     if (this.bobot_i >= M) {
3         this.isOnKandidat = true;
4     } else {
5         this.isOnKandidat = false;
6     }
7 }

```

Source Code 4. 11 Implementasi *filtering frequent itemset*

Method *setOnKandidatFrequentItemset()* digunakan untuk menyaring kandidat *n-itemset* yang memiliki nilai *weighted count itemset* I (*bobot_i*) kurang dari nilai *weighted count threshold* (M). Proses tersebut dilakukan sebanyak *n-itemset* mulai baris 5-15 di dalam **Source Code 4.10**.

4.2.3 Implementasi Tahap pembentukan aturan asosiasi

Pada tahap ini dilakukan proses untuk menghasilkan semua aturan asosiasi yang menunjukkan keterkaitan antar alamat *website* pada periode waktu tertentu.

```

1 //pembentukan aturan asosiasi
2 itemsetUpdated.clear();
3 for (Itemset itsOnKand : kandidat.getItemsets()) {
4     Itemset newItemset = new Itemset(itsOnKand.getItems());
5     newItemset.setBobot_i(itsOnKand.getBobot_i());
6     newItemset.countBobot_x(partisiData);
7     newItemset.countC();
8
9     //cek on kandidat ?
10    newItemset.setOnKandidatAturanAsosiasi(confidenceThreshold);
11    if (newItemset.isOnKandidat()) {
12        itemsetUpdated.add(newItemset);
13    }
14 }
15
16    kandidat.setItems(itemsetUpdated);
17 //lift ratio
18 itemsetUpdated.clear();
19 for (Itemset asoItem : kandidat.getItemsets()) {
20     asoItem.countfreqY(partisiData);
21     asoItem.countBenchmark(partisiData);
22     asoItem.countLiftRatio();
23 }

```

Source Code 4. 12 Implementasi Tahap pembentukan aturan asosiasi

```

1 public void setOnKandidatAturanAsosiasi(double confidenceThreshold)
2 {
3     if (this.C >= confidenceThreshold) {
4         this.isOnKandidat = true;
5     } else {
6         this.isOnKandidat = false;
7     }
8 }
9
10
11 public void countLiftRatio() {
12     this.liftRatio = this.C / this.benchmark;
13 }

```

Source Code 4. 13 Implementasi *filtering* aturan asosiasi

Method `setOnKandidatAturanAsosiasi()` dilakukan untuk menyaring *frequent itemset* yang memiliki nilai lebih dari *confidence threshold* yang diinputkan pengguna (Source Code 4.13, baris 1-8), yang selanjutnya *frequent itemset* yang telah lolos dinamakan sebagai aturan asosiasi / *rule*. Dalam proses itu juga dilakukan penghitungan nilai *Lift Ratio* untuk setiap *rule* yang terbentuk

(Source Code 4.13, baris 11-13). Proses tersebut dilakukan sebanyak n -frequent *itemset* mulai baris 3-23 di dalam Source Code 4.12.

4.3 Implementasi Antarmuka

Implementasi antarmuka sistem seperti yang telah dijelaskan dalam rancangan antarmuka pada subbab 3.6, terdiri dari bagian input, bagian output.

Bagian input terdiri dari :

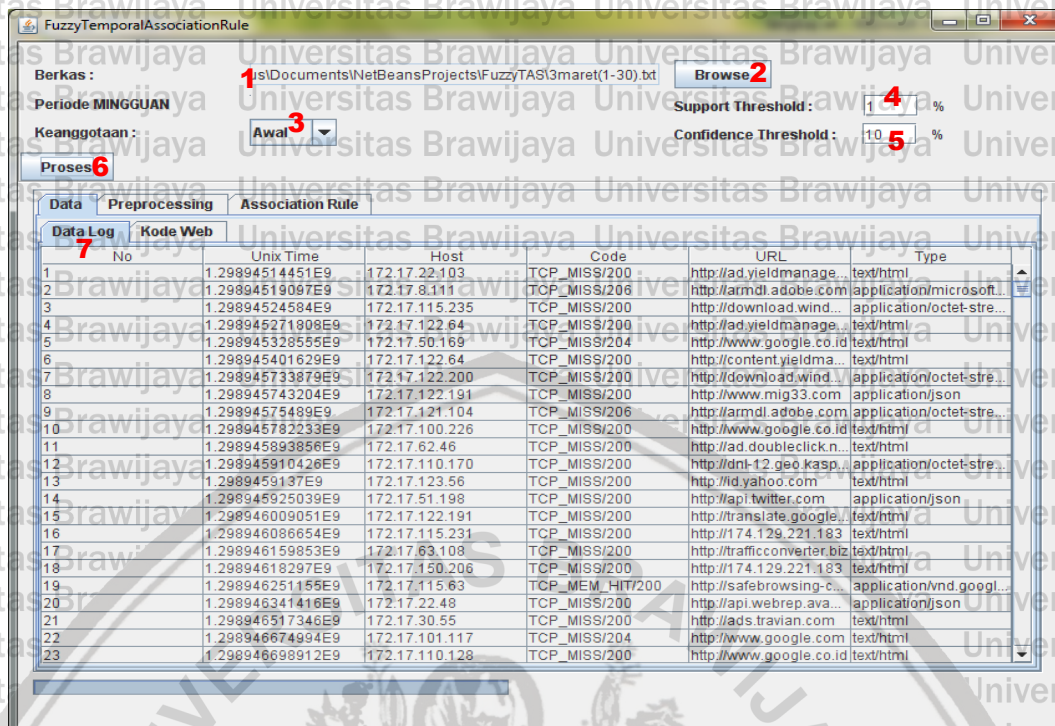
1. *File data Access log*
2. Pilihan periode waktu (Awal,tengah, akhir dari mingguan)
3. *support threshold* sebagai batas nilai *support*
4. *confidence threshold* sebagai batas nilai *confidence*

Bagian output terdiri dari tiga submenu yaitu submenu :

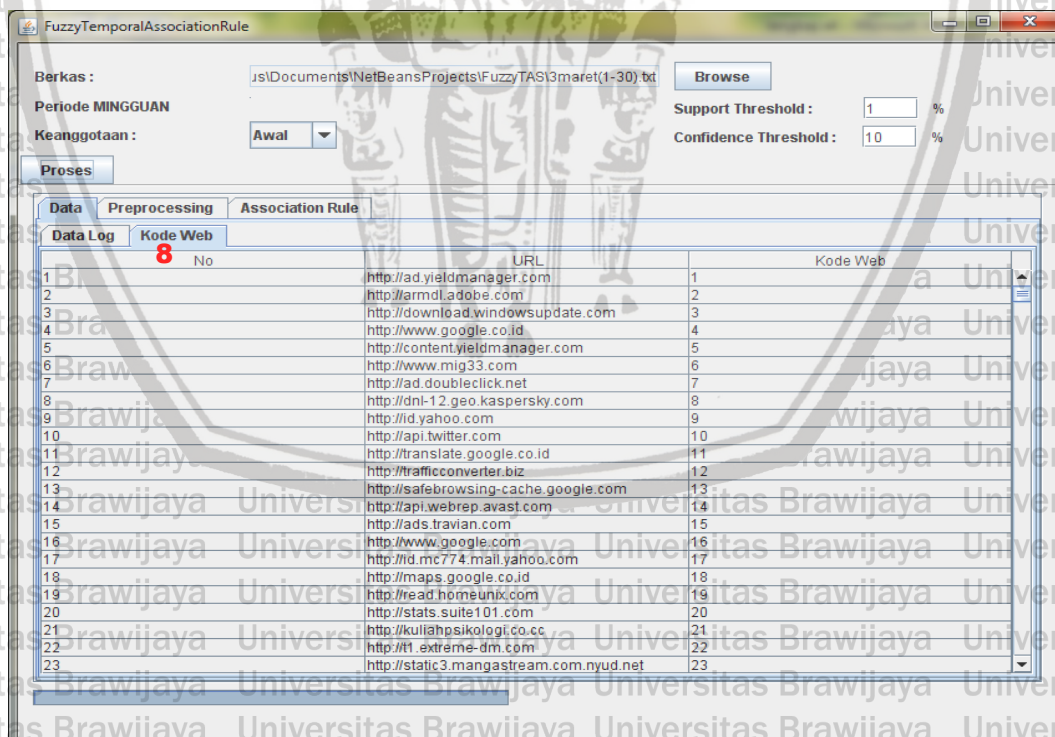
1. Submenu data untuk menampilkan seluruh data *Access log* dan kode *web*
2. Submenu *Preprocessing* untuk menampilkan hasil *filtering*, pembagian waktu dan *IP address* serta hasil *preprocessing* yang sudah dalam bentuk transaksi
3. Submenu *Association rule* untuk menampilkan *frequent itemset* dan aturan asosiasi yang telah dihasilkan.

Dimulai dengan menginputkan *file data access log* dalam bentuk .txt di bagian nomor 1 dan 2, kemudian memilih keanggotaan fungsi *fuzzy calendar* dari periode MINGGUAN di *combo box* nomor 3, yang dilanjutkan dengan menginputkan nilai *support threshold* dan *confidence threshold* di nomor 4 dan 5.

Untuk mengetahui hasil dari metode yang digunakan pada data *access log*, tekan tombol proses di nomor 6, maka yang muncul pertama kali adalah submenu data yang berisi data *access log*(sudah dipecah berdasarkan spasi) di nomor 7 dan kodeweb dari seluruh *website* dari data *access log* di nomor 8. Seperti yang ditunjukkan pada **Gambar 4.1 dan Gambar 4.2.**



Gambar 4. 1 Tampilan submenu Data



Gambar 4. 2 Tampilan submenu Kode Web

Di saat yang sama, sub menu preprocessing menampilkan hasil preprocessing dalam bentuk transaksi seperti pada nomor 9 yang terdiri dari 4 kolom, yaitu "Group Partisi" (menampilkan Partisi_1 sampai Partisi_akhir), "tanggal" (menampilkan tanggal dari masing-masing transaksi), "Kode transaksi", dan "Kode web" (menampilkan *website-website* yang diakses dalam 1 transaksi).

Filtering	Daftar Transaksi	TID	Item
P1	Pi	Hari, Tanggal	
		Selasa, 01 Maret 2011	
		T1	[1, 5]
		T2	[6, 11]
		T3	[9, 18]
		T4	[28, 39]
		T5	[49, 51]
		T6	[57, 65]
		T7	[105, 4]
P2		Rabu, 02 Maret 2011	[126, 127]
P3		Kamis, 03 Maret 2011	[133, 111]
		T9	[143, 93, 4]
		T10	[5, 165]
		T11	[177, 178]
P4		Sabtu, 05 Maret 2011	[126, 180]
		T12	[181, 78]
		T13	[183, 184]
		T14	[62, 28]
		T15	[192, 196]
		T16	[214, 217]
		T17	[220, 163]
P5		Minggu, 06 Maret 2011	[221, 1]
		T18	[224, 111]
		T19	[225, 229]
		T20	[226, 94]
		T21	
		T22	

Gambar 4. 3 Tampilan submenu hasil *preprocessing*

Pada submenu *Association rule* ditampilkan Fuzzy Kalender, proses perhitungan partisi, *Frequent*, hasil Asosiasi, Aturan Asosiasi dan *Lift Ratio*. Pada tab *Frequent* di nomor 10, ditampilkan *frequent itemset* yang dihitung nilai *confidence* dari masing-masing *itemset* dan dibandingkan dengan nilai *confidence threshold* yang diinputkan pengguna. Pada tab Aturan Asosiasi di nomor 11, ditampilkan seluruh aturan asosiasi dalam bentuk teks. Dan pada tab *Lift Ratio* di nomor 12, ditampilkan nilai *Lift Ratio* dari aturan asosiasi yang dihasilkan.

Fuzzy Kalender	Perhitungan Partisi	Frequent	Hasil Asosiasi	Aturan Asosiasi	Lift Ratio
[1, 5]		Items	10		Torsi
[6, 31]			1,0		
[9, 30]			1,0		
[2, 30]			1,0		
[49, 51]			1,0		
[57, 65]			1,0		
[126, 127]			1,0		
[127, 126]			1,0		
[126, 63]			1,0		
[63, 126]			0,5		
[94, 10]			0,5		
[113, 69]			0,5		
[126, 127]			2,5		
[57, 10]			0,5		
[105, 3]			0,5		
[127, 4]			0,5		
[12, 11]			1,0		
[3, 4]			1,0		
[126, 127, 4]			0,5		

Gambar 4. 4 Tampilan submenu *frequent itemset*

Fuzzy Kalender	Perhitungan Partisi	Frequent	Hasil Asosiasi	Aturan Asosiasi	Lift Ratio
Jika mengakses [http://ad.yieldmanager.com] maka mengakses [http://content.yieldmanager.com]					
Jika mengakses [http://safebrowsing.clients.google.com] maka mengakses [http://www.google.com]					
Jika mengakses [http://dss1.siteadvisor.com] maka mengakses [http://www.google.co.id]					
Jika mengakses [http://mw1.google.com] maka mengakses [http://kh.google.com]					
Jika mengakses [http://mw2.google.com] maka mengakses [http://kh.google.com]					
Jika mengakses [http://scr.klikksaya.com] maka mengakses [http://mw4.scribdassets.com]					
Jika mengakses [http://kh.google.com] maka mengakses [http://mw1.google.com]					
Jika mengakses [http://safebrowsing.clients.google.com] maka mengakses [http://api.twitter.com]					
Jika mengakses [http://dss1.siteadvisor.com] maka mengakses [http://download.windowsupdate.com]					
Jika mengakses [http://mw1.google.com] maka mengakses [http://www.google.co.id]					
Jika mengakses [http://download.windowsupdate.com] maka mengakses [http://twitter.com]					
Jika mengakses [http://download.windowsupdate.com] maka mengakses [http://www.google.co.id]					
Jika mengakses [http://kh.google.com] dan [http://mw1.google.com] maka mengakses [http://www.google.co.id]					

Gambar 4. 5 Tampilan submenu hasil aturan asosiasi

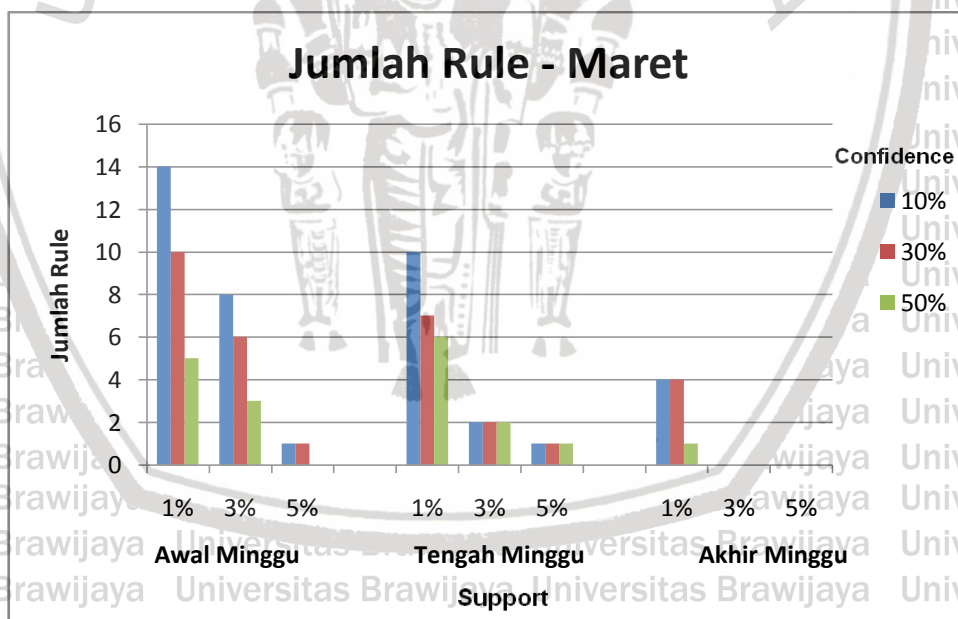
Fuzzy Kalender	Perhitungan Partisi	Frequent	Hasil Asosiasi	Aturan Asosiasi	Lift Ratio
[1, 5]		C	Frekuensi	Benchmark	Lift Ratio
[57, 65]	0.5666666666666666	2,0	2,0	0.019801980198019802	35,25
[05, 4]	0.5	11,0	11,0	0.010891089108910891	4.590909090909091
[127, 126]	0.2857142857142857	11,0	11,0	0.010891089108910891	2.623376623376623
[63, 126]	0.15304615304615305	4,0	4,0	0.039603960396039604	3.8946153046153046
[113, 69]	0.3333333333333333	1,0	1,0	0.009900990990990991	3.6666666666666664
[126, 127]	0.33846153846153844	7,0	7,0	0.0594059405940594	5.611111111111111
[57, 10]	0.3333333333333333	6,0	6,0	0.0594059405940594	5.611111111111111
[105, 3]	0.5	3,0	3,0	0.027027027027027	16.833333333333336
[127, 4]	0.4	11,0	11,0	0.0891089108910891	1.3116883116883116
[3, 4]	0.14285714285714285	1,0	1,0	0.009900990990990991	40,4
[3, 4]	0.4	11,0	11,0	0.010891089108910891	3.872727272727273
[126, 127, 4]	0.2	11,0	11,0	0.010891089108910891	1.8363636363636364

Gambar 4. 6 Tampilan submenu *Lift Ratio*

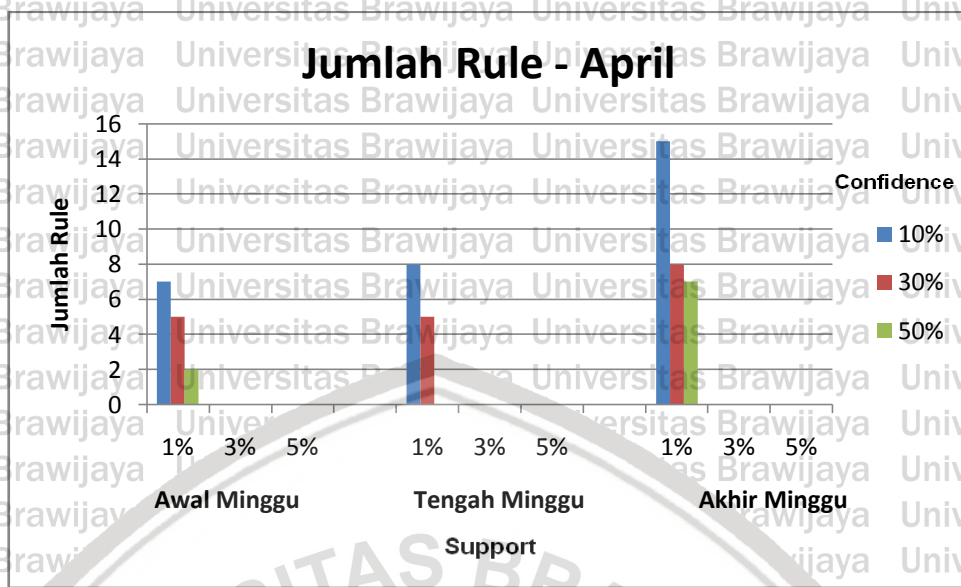
4.4 Pengujian Sistem

Dalam melakukan pengujian untuk mengetahui kombinasi akses kunjungan *website* yang dilakukan oleh pengguna *internet* di Universitas Brawijaya dengan menerapkan metode *Fuzzy Temporal Association rules*, masukan berupa data *Access log* mulai bulan Maret 2011 sampai Mei 2011. Data diuji untuk periode *Fuzzy Calendar* yaitu Awal Minggu, Tengah Minggu dan Akhir Minggu untuk masing-masing bulan.

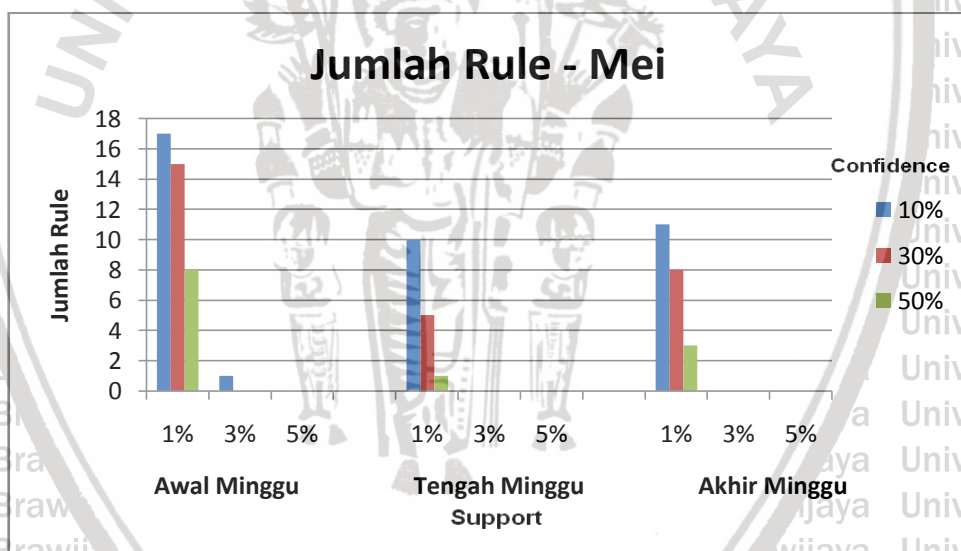
Pengujian pertama dilakukan untuk mengetahui pengaruh nilai minimum *support* dan nilai minimum *confidence* yang diberikan terhadap jumlah *rule* yang dihasilkan. Pada pengujian pertama ini, *support threshold* yang digunakan adalah 1%, 3% dan 5%. *Confidence threshold* yang digunakan adalah 10%, 30% dan 50%. Hasil uji pengaruh nilai *support threshold* dan *confidence threshold* terhadap jumlah *rule* yang dihasilkan dapat dilihat pada **Gambar 4.7** sampai **Gambar 4.9**.



Gambar 4.7 Hasil uji pengaruh nilai *support threshold* dan *confidence threshold* terhadap jumlah *rule* yang dihasilkan di bulan Maret



Gambar 4. 8 Hasil uji pengaruh nilai *support threshold* dan *confidence threshold* terhadap jumlah *rule* yang dihasilkan di bulan April



Gambar 4. 9 Hasil uji pengaruh nilai *support threshold* dan *confidence threshold* terhadap jumlah *rule* yang dihasilkan di bulan Mei

Pengujian kedua dilakukan untuk mengetahui tingkat kekuatan (*Lift Ratio*) dari aturan asosiasi yang dihasilkan. Pada pengujian ini, aturan asosiasi yang diuji adalah aturan asosiasi dengan *support threshold* tertinggi yang masih menghasilkan aturan asosiasi dengan *confidence threshold* sebesar 30%. Pada

setiap parameter, nilai minimum *support* tertinggi yang masih menghasilkan *rule* berbeda-beda. Hasil pengujian pada masing-masing bulan adalah sebagai berikut.

a. Bulan Maret

Tabel 4. 1 Tabel Uji *Lift Ratio Association rule*

<i>Fuzzy Calendar</i>	<i>Rules</i>	<i>Confidence (%)</i>	<i>Lift Ratio</i>
Awal minggu	[126, 127]	38.46%	2.97
Tengah minggu	[126, 127]	71.43%	5.20
Akhir minggu	[62, 28]	66.67%	16.17
	[62, 55]	33.33%	8.08
	[94, 113]	33.33%	16.17
	[94, 55]	33.33%	8.08

Tabel 4. 2 Keterangan *Rule* Hasil Uji Kekuatan *Lift Ratio*

<i>Rule</i>	Keterangan
[126, 127]	Jika mengakses [http://kh.google.com] maka mengakses [http://mw1.google.com]
[62, 28]	Jika mengakses [http://ad.kaskus.us] maka mengakses [http://platform0.twitter.com]
[62, 55]	Jika mengakses [http://ad.kaskus.us] maka mengakses [http://www.blogger.com]
[94, 113]	Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://scr.kliksaya.com]
[94, 55]	Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://www.blogger.com]

b. Bulan April

Tabel 4. 3 Tabel Uji *Lift Ratio Association rule*

<i>Fuzzy Calendar</i>	<i>Rules</i>	<i>Confidence (%)</i>	<i>Lift Ratio</i>
Awal minggu	[97, 46]	66.67%	28.33
	[97, 12]	33.33%	9.44

	[69, 59]	33.33%	5.67
	[12, 8]	66.67%	2.58
	[59, 8]	40.00%	1.55
Tengah minggu	[56, 8]	40.00%	5.52
	[15, 69]	33.33%	23
	[15, 59]	33.33%	11.5
	[21, 82]	33.33%	23
	[12, 13]	33.33%	11.5
	[91, 85]	50.00%	16.50
Akhir minggu	[85, 98]	50.00%	22.00
	[85, 65]	50.00%	22.00
	[98, 65]	66.67%	29.33
	[101, 102]	50.00%	33.00
	[6, 31]	50.00%	33.00
	[91, 84]	50.00%	6.00
	[97, 69]	33.33%	11.00

Tabel 4. 4 Keterangan *Rule* Hasil Uji Kekuatan *Lift Ratio*

<i>Rule</i>	Keterangan
[97, 46]	Jika mengakses [http://r1rk9np7bpcsf0eek10khkd2juj27q30-a-fc-opensocial.googleusercontent.com] maka mengakses [http://search.twitter.com]
[97, 12]	Jika mengakses [http://r1rk9np7bpcsf0eek10khkd2juj27q30-a-fc-opensocial.googleusercontent.com] maka mengakses [http://clients1.google.co.id]
[69, 59]	Jika mengakses [http://urls.api.twitter.com] maka mengakses [http://content.yieldmanager.com]
[12, 8]	Jika mengakses [http://clients1.google.co.id] maka mengakses [http://www.google.co.id]
[59, 8]	Jika mengakses [http://content.yieldmanager.com] maka mengakses [http://www.google.co.id]
[56, 8]	Jika mengakses [http://safebrowsing-cache.google.com] maka mengakses [http://www.google.co.id]
[15, 69]	Jika mengakses [http://armdl.adobe.com] maka mengakses [http://urls.api.twitter.com]
[15, 59]	Jika mengakses [http://armdl.adobe.com] maka mengakses [http://content.yieldmanager.com]

[21, 82]	Jika mengakses [http://api.twitter.com] maka mengakses [http://platform0.twitter.com]
[12, 13]	Jika mengakses [http://clients1.google.co.id] maka mengakses [http://toolbarqueries.google.co.id]
[91, 85]	Jika mengakses [http://s.youtube.com] maka mengakses [http://www.youtube.com]
[85, 98]	Jika mengakses [http://www.youtube.com] maka mengakses [http://geo.predictad.com]
[85, 65]	Jika mengakses [http://www.youtube.com] maka mengakses [http://ad.kaskus.us]
[98, 65]	Jika mengakses [http://geo.predictad.com] maka mengakses [http://ad.kaskus.us]
[101, 102]	Jika mengakses [http://block.nawala.org] maka mengakses [http://sp.ltassrv.com]
[6, 31]	Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://www.mangareader.net]
[91, 84]	Jika mengakses [http://s.youtube.com] maka mengakses [http://pubads.g.doubleclick.net]
[97, 69]	Jika mengakses [http://r1rk9np7bpcsf0eek10khkd2juj27q3o-a-fc-opensocial.googleusercontent.com] maka mengakses [http://urls.api.twitter.com]

c. Bulan Mei

Tabel 4. 5 Tabel Uji Lift Ratio Association rule

<i>Fuzzy Calendar</i>	<i>Rules</i>	<i>Confidence (%)</i>	<i>Lift Ratio</i>
Awal minggu	[4, 5]	50.00%	31.75
	[56, 2]	50.00%	2.89
	[19, 71]	33.33%	21.17
	[19, 34]	33.33%	3.26
	[19, 47]	33.33%	2.49
	[20, 47]	50.00%	3.74
	[1, 47]	40.00%	2.99
	[19, 11]	33.33%	21.17
	[19, 20]	33.33%	10.58
	[80, 2]	33.33%	1.92
	[21, 56]	50.00%	15.88
	[9, 1]	66.67%	16.93
	[4, 111]	50.00%	15.88
	[8, 34]	66.67%	6.51
	[27, 49]	66.67%	42.33
[1, 17]	33.33%	30.00	

Tengah minggu	[16, 82]	33.33%	10.00
	[69, 101]	66.67%	15.00
	[11, 2]	33.33%	2.73
	[16, 20]	33.33%	15.00
Akhir minggu	[38, 11]	50.00%	2.97
	[16, 75]	33.33%	14.83
	[21, 63]	50.00%	1.53
	[27, 2]	50.00%	4.94
	[19, 11]	40.00%	2.37
	[47, 11]	40.00%	2.37
	[47, 63]	40.00%	1.23
	[4, 10]	33.33%	14.83

Tabel 4. 6 Keterangan *Rule* Hasil Uji Kekuatan *Lift Ratio*

<i>Rule</i>	Keterangan
[4, 5]	Jika mengakses [http://r1rk9np7bpcsf0eek10khkd2juj27q3o-a-fc-opensocial.googleusercontent.com] maka mengakses [http://i377.photobucket.com]
[56, 2]	Jika mengakses [http://www.mangafox.com] maka mengakses [http://www.google.co.id]
[19, 71]	Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://ad.yieldmanager.com]
[19, 34]	Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://api.twitter.com]
[19, 47]	Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://safebrowsing-cache.google.com]
[20, 47]	Jika mengakses [http://content.yieldmanager.com] maka mengakses [http://safebrowsing-cache.google.com]
[1, 47]	Jika mengakses [http://www.google.com] maka mengakses [http://safebrowsing-cache.google.com]
[19, 11]	Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://clients1.google.co.id]
[19, 20]	Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://content.yieldmanager.com]
[80, 2]	Jika mengakses [http://ad.doubleclick.net] maka mengakses [http://www.google.co.id]
[21, 56]	Jika mengakses [http://pubads.g.doubleclick.net] maka mengakses [http://www.mangafox.com]
[9, 1]	Jika mengakses [http://ads.yimg.com] maka mengakses [http://www.google.com]

[4, 111]	Jika mengakses [http://r1rk9np7bpcsf0eek10khkd2juj27q3o-a-fc-opensocial.googleusercontent.com] maka mengakses [http://id.mg60.mail.yahoo.com]
[8, 34]	Jika mengakses [http://search.twitter.com] maka mengakses [http://api.twitter.com]
[27, 49]	Jika mengakses [http://twitter.com] maka mengakses [http://poll daddy.com]
[1, 17]	Jika mengakses [http://www.google.com] maka mengakses [http://s102.filesonic.com]
[16, 82]	Jika mengakses [http://googleads.g.doubleclick.net] maka mengakses [http://platform0.twitter.com]
[69, 101]	Jika mengakses [http://openx.detik.com] maka mengakses [http://block.opendns.com]
[11, 2]	Jika mengakses [http://clients1.google.co.id] maka mengakses [http://www.google.co.id]
[16, 20]	Jika mengakses [http://googleads.g.doubleclick.net] maka mengakses [http://content.yieldmanager.com]
[38, 11]	Jika mengakses [http://scr.klik saya.com] maka mengakses [http://clients1.google.co.id]
[16, 75]	Jika mengakses [http://googleads.g.doubleclick.net] maka mengakses [http://ocsp.verisign.com]
[21, 63]	Jika mengakses [http://pubads.g.doubleclick.net] maka mengakses [http://www.facebook.com]
[27, 2]	Jika mengakses [http://twitter.com] maka mengakses [http://www.google.co.id]
[47, 11]	Jika mengakses [http://safebrowsing-cache.google.com] maka mengakses [http://clients1.google.co.id]
[47, 63]	Jika mengakses [http://safebrowsing-cache.google.com] maka mengakses [http://www.facebook.com]
[4, 10]	Jika mengakses [http://r1rk9np7bpcsf0eek10khkd2juj27q3o-a-fc-opensocial.googleusercontent.com] maka mengakses [http://www.blogger.com]

4.5 Analisa Hasil

Berdasarkan hasil pengujian pengaruh nilai *support threshold* dan *confidence threshold* didapatkan bahwa jumlah *rule* yang dihasilkan berbanding terbalik. Semakin tinggi *support threshold* dan nilai *confidence threshold* yang ditentukan maka semakin sedikit jumlah *association rule* yang dihasilkan. Hal ini

digambarkan pada **Gambar 4.7** sampai **Gambar 4.9** pada masing-masing *Fuzzy Calendar* dan masing-masing bulan.

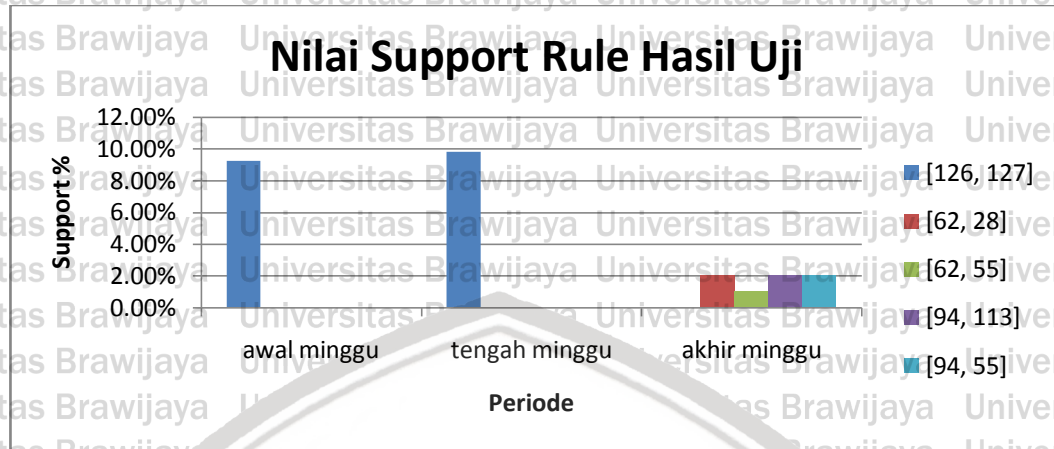
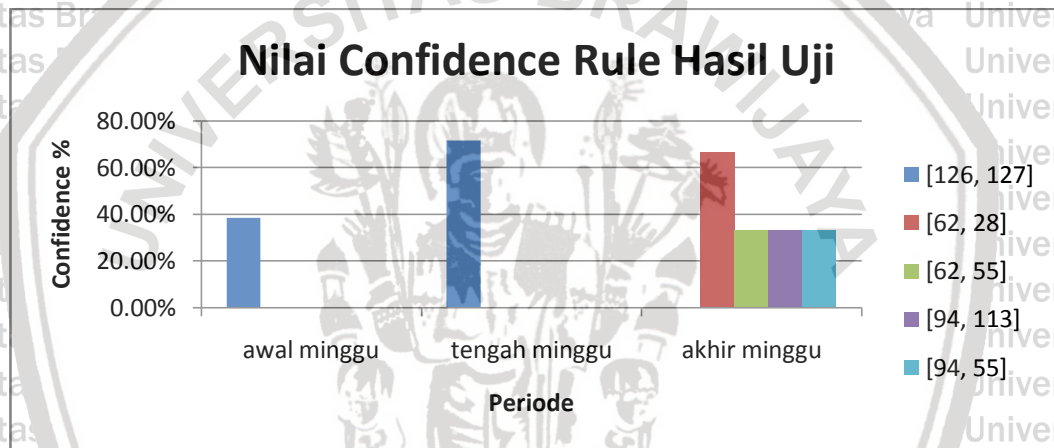
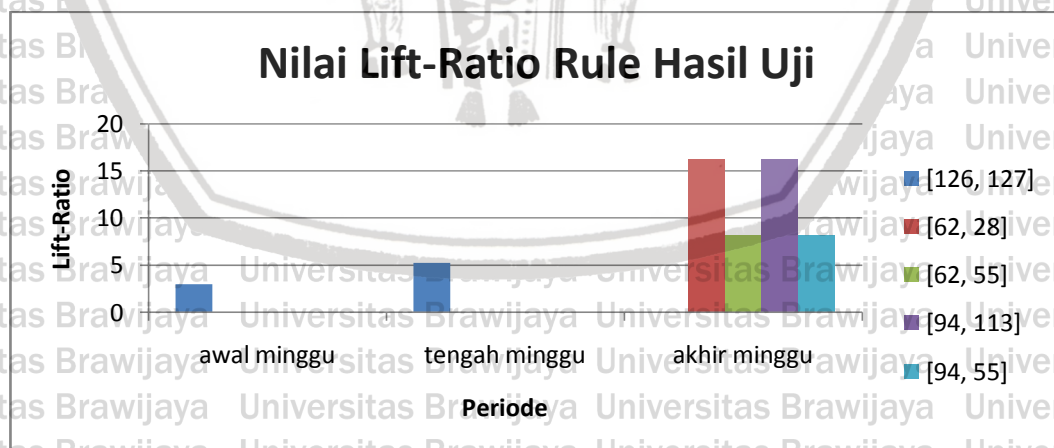
Nilai *support threshold* dan nilai *confidence threshold* berbanding terbalik terhadap jumlah *rule* tersebut dikarenakan semakin tinggi nilai *support threshold* yang digunakan berarti semakin tinggi nilai batas *support* atau prosentase ketersediaan kombinasi data yang harus dicapai sehingga semakin sedikit yang dapat mencapai batasan untuk menjadi *frequent itemset*. Bila semakin tinggi nilai *confidence threshold* yang digunakan maka semakin tinggi nilai batas *confidence* yang harus dicapai sehingga semakin sedikit *frequent itemset* yang bisa memenuhinya sehingga semakin sedikit jumlah *rule* yang dihasilkan.

Namun pada Bulan Maret terdapat jumlah aturan yang tidak selalu menurun, melainkan menghasilkan jumlah aturan yang sama. Hal ini disebabkan aturan-aturan yang dihasilkan memiliki nilai diatas *support threshold* dan *confidence threshold* yang diinputkan. Sedangkan untuk beberapa Grafik juga menunjukkan tidak adanya aturan yang dihasilkan, hal ini disebabkan ada beberapa aturan yang memiliki nilai dibawah *support threshold* dan *confidence threshold* yang diinputkan.

Jika dilihat dalam dari tiap-tiap grafik, misalkan ingin melihat pola jumlah *rule* dari periode Awal Minggu pada bulan Maret, April dan Mei, ternyata menghasilkan pola yang tidak selalu menurun atau naik. Hal ini disebabkan masing-masing data memiliki kekarakteristikan sendiri. Sedangkan jika ingin melihat pola jumlah *rule* yang dihasilkan dalam 1 bulan dengan periode dari Awal Minggu, Tengah Minggu dan Akhir Minggu, juga menghasilkan pola yang tidak selalu menurun atau naik. Hal ini disebabkan karena adanya keefektifan dari *Fuzzy Calendar*.

Parameter yang digunakan pada aplikasi ini adalah *Fuzzy Calendar* (yang selanjutnya diistilahkan dengan periode), *support threshold* dan *confidence threshold*. Ketiga parameter tersebut memberikan pengaruh terhadap variasi, nilai *support* dan *confidence* dari aturan yang dihasilkan. Hasil pengujian pada **Tabel 4.1** sampai **Tabel 4.6** akan dianalisa sebagai berikut :

a. Bulan Maret

Gambar 4. 10 Grafik nilai *support rule* hasil pengujianGambar 4. 11 Grafik nilai *confidence rule* hasil pengujianGambar 4. 12 Grafik nilai *lift-ratio rule* hasil pengujian

Pada Gambar 4.10 dan Gambar 4.11 menunjukkan bahwa rule [126,127] memiliki nilai *weighted support* dan *weighted confidence* tertinggi di periode

Tengah Minggu. *Rule* [126,127] memiliki nilai *weighted support* sebesar 9.80% dan *weighted confidence* sebesar 71.43%. Hal ini menunjukkan bahwa, di Bulan Maret para pengguna *internet* di Universitas Brawijaya Malang pada periode Tengah Minggu yang mengakses item [126] punya kemungkinan 71.43% untuk juga mengakses item [127]. Aturan ini didukung dengan kemunculannya sebesar 9.80% dari catatan transaksi di bulan Maret 2011.

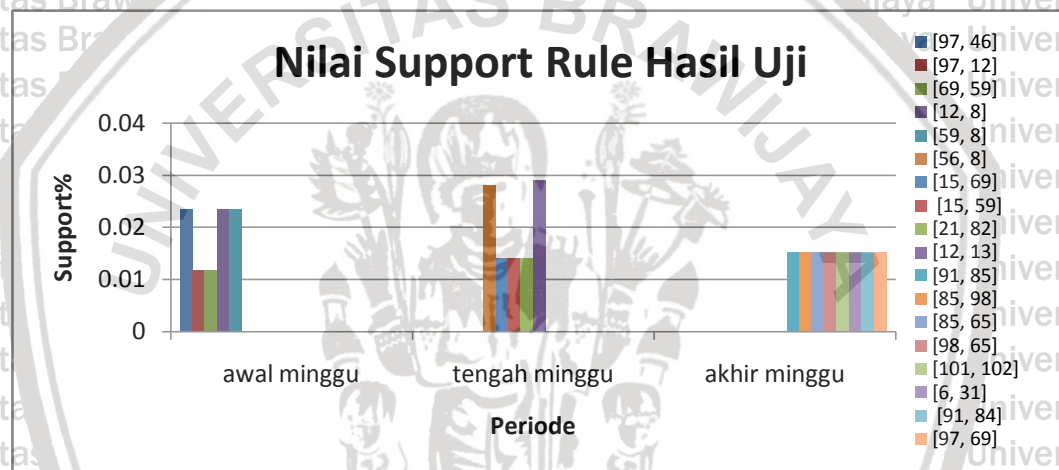
Untuk *rule* [126,127] pada **Gambar 4.10** muncul pada periode awal minggu dan tengah minggu dengan nilai *support* yang meningkat. Hal ini menunjukkan bahwa *rule* tersebut memiliki nilai *weighted count* ($\sigma_D(I)$) yang tinggi dalam transaksi. Nilai *support* yang meningkat menunjukkan bahwa peluang kemunculan *rule* pada periode tengah minggu lebih besar dibandingkan pada periode awal minggu. Begitu pula pada **Gambar 4.11**, *rule* [126,127] mengalami peningkatan nilai *confidence* dari periode awal minggu ke periode tengah minggu. Hal ini disebabkan karena nilai *weighted count item* jika ($\sigma_D(X)$), yaitu *item* [126] pada periode awal minggu lebih besar dari ($\sigma_D(X)$) *item* [126] pada periode tengah minggu. Sehingga saat dibandingkan dengan ($\sigma_D(I)$) dari [126,127] menghasilkan nilai *confidence* [126,127] pada periode tengah minggu lebih besar dari periode awal minggu. Yang dapat diartikan bahwa pada periode tengah minggu *rule* [126,127] diakses secara bersama-sama meingkat dengan tingkat kepercayaan lebih dari 70% dari *item* [126] yang diakses.

Untuk *rule* [62,28],[62,25],[94,113] dan [94,115] baru muncul pada periode tengah minggu. Dari **Gambar 4.10** dapat diartikan bahwa *rule-rule* tersebut menjadi lebih sering diakses pada periode akhir minggu. Sedangkan dari **Gambar 4.11**, dapat dilihat bahwa *rule* [62,28] yang awalnya memiliki nilai *support* sama dengan *rule* [94,113] dan [94,115] ternyata memiliki nilai *Confidence* yang berbeda. Hal ini disebabkan karena nilai ($\sigma_D(X)$) *item* [62] lebih kecil daripada nilai ($\sigma_D(X)$) *item* [94], yang awalnya memiliki nilai ($\sigma_D(I)$) sama (penyebab nilai *support* dari ketiga *rule* sama). Untuk *rule* [62,25] yang awalnya memiliki nilai *support* berbeda dari *rule* lainnya di periode akhir minggu (karena nilai ($\sigma_D(I)$) *item* [62,25] lebih kecil dibanding ketiga *rule* lainnya), nilai *confidence*'nya menjadi sama dengan *rule* [94,113] dan [94,115]. Hal ini disebabkan karena nilai

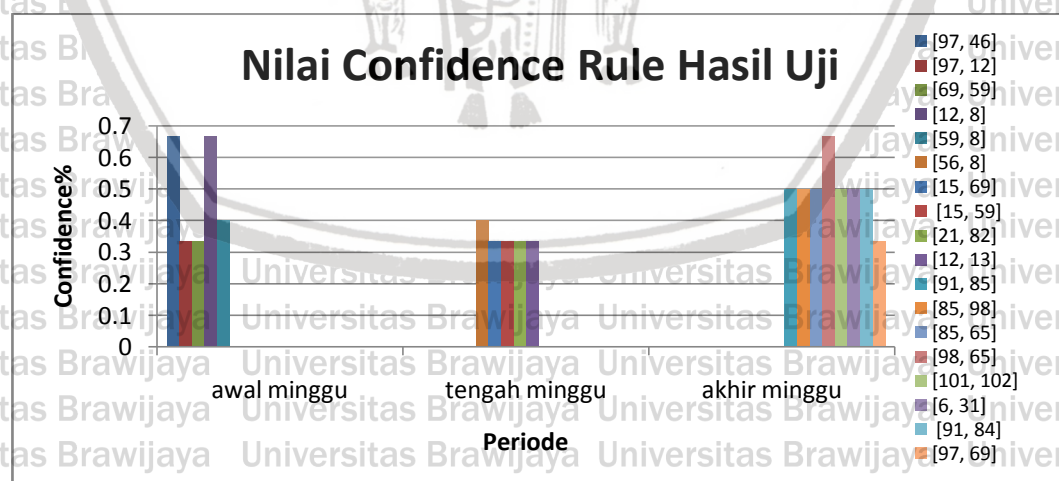
$(\sigma_D(I) / \sigma_D(X))$ item [62,25] sebanding dengan nilai $(\sigma_D(I) / \sigma_D(X))$ item [94,113] dan [94,115].

Nilai *Lift Ratio* tertinggi berasal dari *rule* [62,28] dan [94,113]. Hal ini menjelaskan bahwa untuk *rule* [62,28] pada periode akhir minggu *item* [62] dan *item* [28] diakses secara bersama-sama memiliki tingkat kepercayaan yang cukup tinggi terhadap jumlah *item* [28] yang diakses. Artinya, dari jumlah *item* [28] yang diakses, jika dihubungkan dengan nilai *confidence* dari **Gambar 4.11**, maka lebih dari 60% diakses bersama *item* [62]. Begitu pula untuk *rule* [94,113].

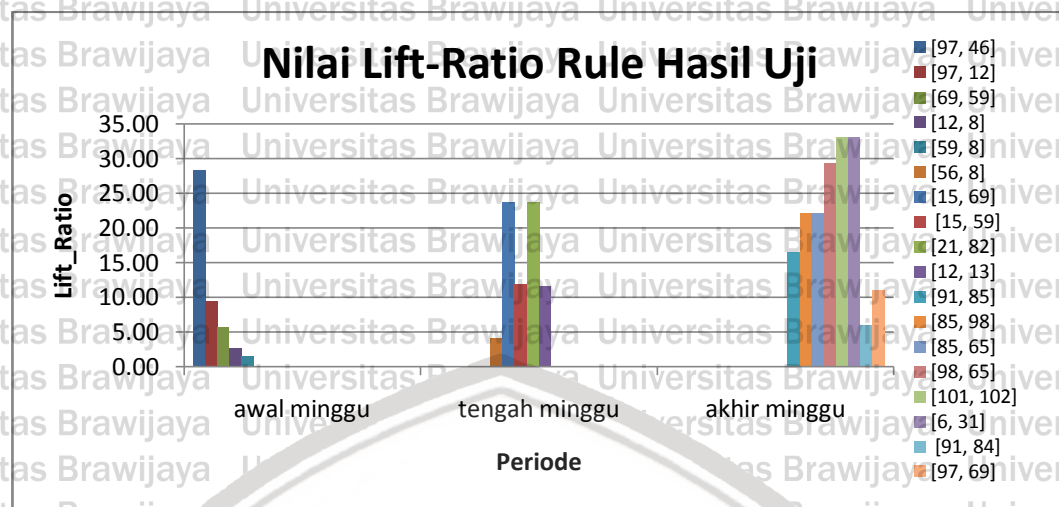
b. Bulan April



Gambar 4. 13 Grafik nilai *support rule* hasil pengujian



Gambar 4. 14 Grafik nilai *confidence rule* hasil pengujian



Gambar 4. 15 Grafik nilai *lift-ratio rule* hasil pengujian

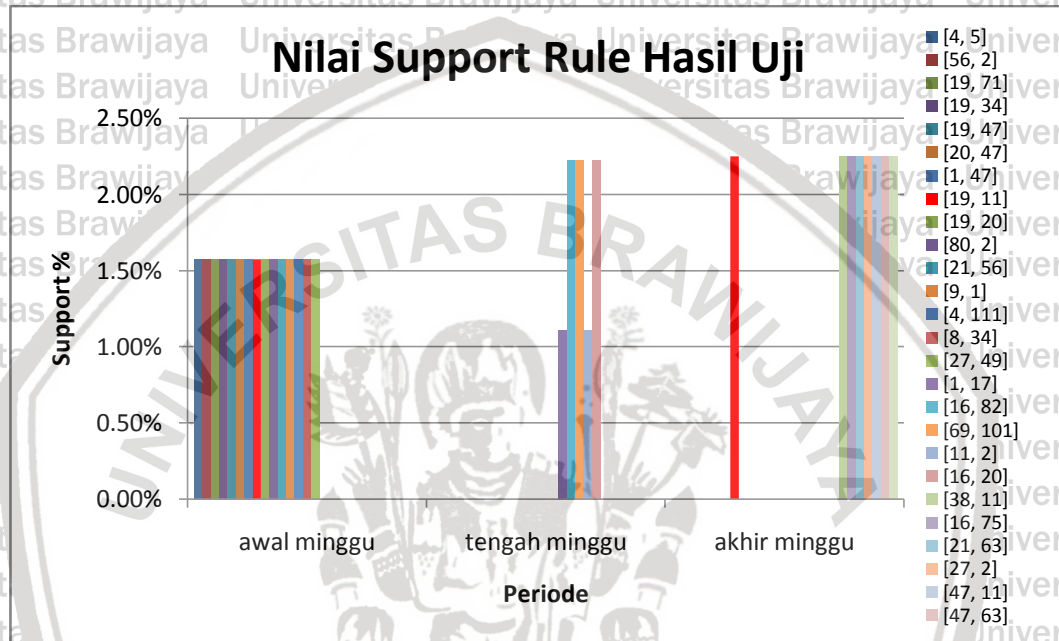
Pada bulan April, *rule* yang dihasilkan bervariasi. Hal ini menunjukkan bahwa *rule-rule* tersebut memiliki nilai $\sigma_D(I)$ yang tinggi di masing-masing periode dengan *confidence threshold* sebesar 30%. Pada **Gambar 4.13** dan **Gambar 4.14** menunjukkan bahwa *rule* [12,13] memiliki nilai *weighted support* tertinggi di periode Tengah Minggu, tetapi nilai *weighted confidence rule* tersebut bukan yang paling tinggi di antara *rule* lainnya. *Rule* [12,13] memiliki nilai *weighted support* sebesar 2.89% dan *weighted confidence* sebesar 33.33%. Hal ini menunjukkan bahwa, di Bulan April para pengguna *internet* di Universitas Brawijaya Malang pada periode Tengah Minggu yang mengakses item [12] punya kemungkinan 33.33% untuk juga mengakses item [13], dengan peluang kemunculan *rule* tersebut secara bersama-sama sebesar 2.89% dari catatan transaksi di bulan Maret 2011.

Dari **Gambar 4.13** pada periode akhir minggu, setiap *rule* memiliki nilai *support* yang sama. Kemudian pada **Gambar 4.14**, nilai *confidence* masing-masing *rule* ada yang mengalami perubahan, seperti *rule* [98,65] yang memiliki *confidence* lebih tinggi dari *rule* lainnya. Hal ini disebabkan nilai $\sigma_D(X)$ item [98] lebih kecil dibandingkan nilai $\sigma_D(X)$ *rule* lainnya. Sebaliknya, pada *rule* [97,69] memiliki *confidence* lebih rendah dari *rule* lainnya. Hal ini disebabkan nilai $\sigma_D(X)$ item [97] lebih besar dibandingkan nilai $\sigma_D(X)$ *rule* lainnya.

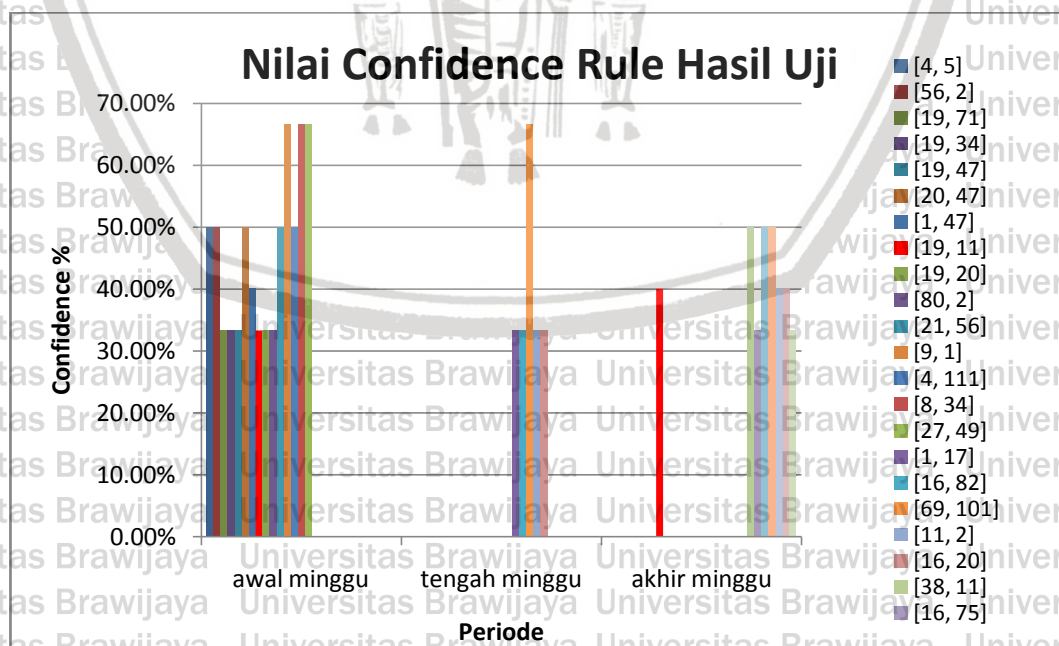
Nilai *Lift Ratio* tertinggi berasal dari *rule* [101,102] dan [6,31]. Hal ini menjelaskan bahwa untuk *rule* [101,102] pada periode akhir minggu item [101]

dan *item* [102] diakses secara bersama-sama memiliki tingkat kepercayaan yang cukup tinggi terhadap jumlah *item* [102] yang diakses. Artinya, dari jumlah *item* [102] yang diakses, jika dihubungkan dengan nilai *confidence* dari **Gambar 4.14**, maka 50% diakses bersama *item* [101]. Begitu pula untuk *rule* [6,31].

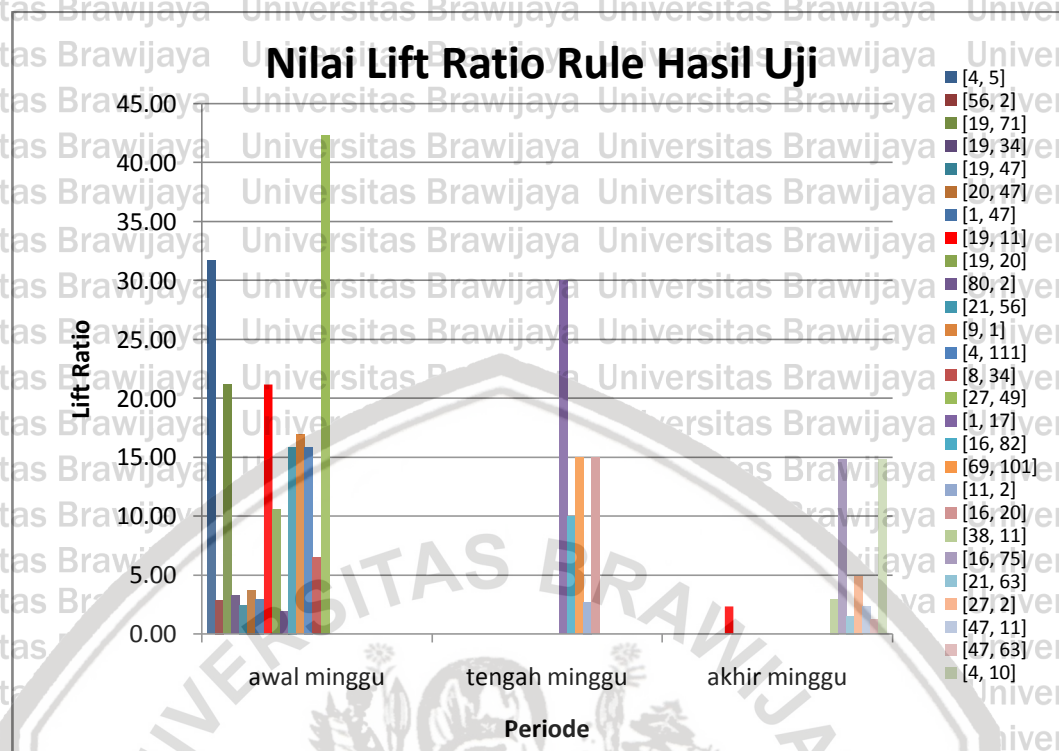
c. Bulan Mei



Gambar 4. 16 Grafik nilai *support rule* hasil pengujian



Gambar 4. 17 Grafik nilai *confidence rule* hasil pengujian



Gambar 4. 18 Grafik nilai *lift-ratio rule* hasil pengujian

Pada **Gambar 4.16** dan **Gambar 4.17** menunjukkan bahwa *rule* [19,11] memiliki nilai *weighted support* yang tinggi di periode Akhir Minggu, tetapi nilai *weighted confidence rule* tersebut bukan yang paling tinggi diantara *rule* lainnya.

Rule [19,11] memiliki nilai *weighted support* sebesar 2.25% dan *weighted confidence* sebesar 40%. Hal ini menunjukkan bahwa, di Bulan Mei para pengguna *internet* di Universitas Brawijaya Malang pada periode Akhir Minggu yang mengakses item [19] punya kemungkinan 40% untuk juga mengakses item [11], dengan peluang kemunculan *rule* tersebut secara bersama-sama sebesar 2.25% dari catatan transaksi di bulan Maret 2011.

Untuk *rule* [19,11] pada **Gambar 4.16** muncul pada periode awal minggu dan akhir minggu dengan nilai *support* yang meningkat. Hal ini menunjukkan bahwa *rule* tersebut memiliki nilai $(\sigma_D(I))$ yang tinggi dalam transaksi. Nilai *support* yang meningkat menunjukkan bahwa peluang kemunculan *rule* pada periode akhir minggu lebih besar dibandingkan pada periode awal minggu. Begitu pula pada **Gambar 4.17**, *rule* [19,11] mengalami peningkatan nilai *confidence* dari periode awal minggu ke periode tengah minggu. Hal ini disebabkan karena nilai *weighted count item* jika $(\sigma_D(X))$, yaitu *item* [19] pada periode awal minggu

lebih besar dari $(\sigma_D(X))$ item [19] pada periode tengah minggu. Sehingga saat dibandingkan dengan $(\sigma_D(I))$ dari [19,11] menghasilkan nilai *confidence* [19,11] pada periode akhir minggu lebih besar dari periode awal minggu. Yang dapat diartikan bahwa pada periode akhir minggu *rule* [19,11] diakses secara bersama-sama meningkat dengan tingkat kepercayaan sebesar 40% dari *item* [19] yang diakses.

Dari **Gambar 4.16** pada periode akhir minggu, setiap *rule* memiliki nilai *support* yang sama. Kemudian pada **Gambar 4.17**, nilai *confidence* masing-masing *rule* ada yang mengalami perubahan, seperti *rule* [38,11] yang memiliki *confidence* lebih tinggi dari *rule* lainnya. Hal ini disebabkan nilai $\sigma_D(X)$ item [38] lebih kecil dibandingkan nilai $\sigma_D(X)$ *rule* lainnya. Sebaliknya, pada *rule* [16,75] memiliki *confidence* lebih rendah dari *rule* lainnya. Hal ini disebabkan nilai $\sigma_D(X)$ item [16] lebih besar dibandingkan nilai $\sigma_D(X)$ *rule* lainnya.

Nilai *Lift Ratio* tertinggi berasal dari *rule* [27,49]. Hal ini menjelaskan bahwa untuk *rule* [27,49] pada periode akhir minggu *item* [27] dan *item* [49] diakses secara bersama-sama memiliki tingkat kepercayaan yang cukup tinggi terhadap jumlah *item* [49] yang diakses. Artinya, dari jumlah *item* [49] yang diakses, jika dihubungkan dengan nilai *confidence* dari **Gambar 4.17**, maka lebih dari 60% diakses bersama *item* [27].

Berdasarkan hasil perhitungan *Lift Ratio* dari bulan Maret, April dan Mei, dapat dilihat bahwa semua *rule* yang dihasilkan memiliki kekuatan yang bagus karena nilai *Lift Ratio* semua *rule* lebih besar dari 1 dan memiliki kekuatan asosiasi yang tinggi sehingga dapat menunjukkan adanya manfaat. Selain itu, dapat dilihat juga bahwa nilai *confidence* yang tinggi belum dapat menjadi acuan bahwa *rule* tersebut memiliki kekuatan asosiasi yang tinggi karena dengan *Lift Ratio*, peluang munculnya *item* yang diakses bersamaan akan dibandingkan dengan peluang munculnya masing – masing *item* tersebut muncul secara independen. Sehingga apabila nilai *confidence* dari suatu *rule* tinggi sedangkan nilai peluang munculnya masing – masing *item* yang diakses tersebut muncul secara independen juga tinggi maka kekuatan dari *rule* tersebut menjadi rendah. Seperti yang terjadi pada *rule* [19,11] di **Gambar 4.17**, dimana nilai *confidence* *rule* [19,11] di Akhir Minggu lebih tinggi dari Awal Minggu.

Namun karena peluang munculnya *item* [11] secara independen pada Akhir Minggu juga tinggi, maka *rule* [19,11] memiliki nilai *lift-ratio* yang lebih rendah ketika berada di Akhir Minggu dibandingkan di Awal Minggu (**Gambar 4.18**).

Pengujian ini juga dapat digunakan untuk melihat keefektifan dari *Fuzzy Calendar*, yaitu bisa dilihat dari 3 penggunaan *Fuzzy Calendar* yang berbeda (Awal Minggu, Tengah Minggu dan Akhir Minggu). Penggunaan *Fuzzy Calendar* ini, menunjukkan adanya perubahan *item* yang diakses pengguna. Lebih jelasnya bisa dilihat dari Tabel di Lampiran C. Hal ini menunjukkan bahwa metode *Fuzzy Temporal association rule* efektif untuk diterapkan pada data *access log* Universitas Brawijaya.



BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisa uji coba dapat diambil kesimpulan:

1. Penerapan metode *Fuzzy Temporal Association Rule* pada data *access log* mulai bulan Maret, April sampai Mei 2011, meliputi 2 tahap utama yaitu :

- a. Tahap *preprocessing*, merupakan tahapan untuk mengekstrak informasi yang dibutuhkan pada tahapan berikutnya.
- b. Tahap algoritma *Fuzzy Temporal Association Rule*, terdiri dari dua proses utama, yaitu proses pencarian *frequent itemset* dan proses pembentukan aturan asosiasi.

Kedua tahapan diatas menghasilkan aturan asosiasi yang bervariasi mulai Bulan Maret, April dan Mei 2011 dengan periode waktu Awal Minggu, Tengah Minggu dan Akhir Minggu. Aturan asosiasi yg memiliki nilai *weighted support* dan *weighted confidence* tertinggi diperoleh pada Tengah Minggu di bulan Maret 2011, yaitu “jika mengakses [<http://kh.google.com>] maka mengakses [<http://mw1.google.com>]”.

Aturan asosiasi tersebut menunjukkan bahwa pada bulan Maret di Tengah Minggu, ketika mengakses [<http://kh.google.com>] para pengguna *internet* di lingkungan Universitas Brawijaya Malang memiliki kecenderungan sebesar 71.42% untuk juga mengakses [<http://mw1.google.com>], dengan didukung oleh peluang kemunculannya sebesar 9.80% dari total transaksi yang ada di bulan Maret 2011.

2. Pengujian terhadap nilai *Lift Ratio* dari setiap *rule* memiliki nilai diatas 1, yang berarti bahwa *rule-rule* yang ditemukan tersebut memiliki kekuatan yang baik, dimana didapatkan nilai tertinggi *lift ratio* dari aturan asosiasi “jika mengakses [<http://twitter.com>] maka mengakses [<http://poll daddy.com>]” pada bulan Mei di Awal Minggu dan nilai *lift ratio* terendah dari aturan asosiasi

“Jika mengakses [<http://safebrowsing-cache.google.com>] maka mengakses [<http://www.facebook.com>]” pada bulan Mei di Akhir Minggu.

5.2 Saran

Saran yang dapat dijadikan bahan pertimbangan untuk penelitian selanjutnya adalah :

1. Pada proses *preprocessing* di bagian *filtering*, untuk alamat *website* yang akan diteliti sebaiknya diambil bagian domainnya saja. Misalkan “<http://safebrowsing-cache.google.com>”, maka yang diambil hanya “google.com” saja.
2. Pada proses diatas juga bisa dilakukan pengkhususan situs yang akan diteliti. Contohnya, ingin meneliti situs di bidang kesehatan / teknologi / bisnis, dan lain-lain.



DAFTAR PUSTAKA

- Agrawal, Rakesh dan Ramakrishnan Srikant. 1994. *Fast Algorithm for Mining Association rules*. In Proceedings of the International Conference on Very Large Data Bases.
- Apriantony, Lerry. 2012. *Manfaat Internet Bagi Pelajar, Pendidikan dan Masyarakat*. <http://www.camsh.com/internet/manfaat-internet-bagi-pelajar-pendidikan-dan-masyarakat.html>. Tanggal akses 12 Maret 2012.
- Hairiyonov, Muhammad. 2004. *Program Magister Teknik Elektro Bidang Khusus Teknologi Informasi*. Institut Teknologi Bandung.
<http://www.cert.or.id/~budi/courses/ec7010/dikmenjur2004/hairiyonov-report.pdf>.
Tanggal akses 12 Maret 2012.
- Han, Jiawei dan Micheline Kamber. 2006. *Data Mining : Concepts and Techniques 2nd Edition*. Morgan Kaufmann, California.
- Kantardzic, Mehmed. 2003. *Data Mining : Concepts, Models, Methods, and Algorithms*. John Willey & Sons, Inc. New Jersey.
- Kusumadewi, Sri. 2010. *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Graha Ilmu. Jakarta.
- Kusrini, M.Kom. 2007. *Konsep dan Aplikasi Sistem Pendukung Keputusan*. CV Andi Offset. Yogyakarta.
- Larose, Daniel T. 2005. *Discovering Knowledge in Data : An Introduction to Data Mining*. John Willey & Sons, Inc New Jersey.
- Lee WJ dan Lee SJ. 2004. *Discovery of Fuzzy Temporal Association Rules*. IEEE Transactions On Systems, Man, and Cybernetics, Vol. 34, No. 6.
- Nixon, Benny. 2010. *Pengembangan Program Penyaringan Data Weblog Untuk Analisis Pola Akses Pengunjung Webserver*. Jurusan Teknik Elektro Universitas Indonesia. Depok.
- Perdana, Fredy. 2008. *System rekomendasi untuk pengaturan hyperlink pada halaman utama website menggunakan metode Association rule dan Hyperlink Clustering*. Universitas Brawijaya. Malang.
- Peter I.hofgesang. 2004. *Web usage Mining structuring semantically enriched click stream data*. Vrije universiteit Amsterdam. The Netherland.

Pramudiono, Iko. 2003. *Pengantar Data Mining : Menambang Permata Pengetahuan di Gudang Data*. [http:// www.ilmukomputer.com](http://www.ilmukomputer.com). Tanggal akses 22 Maret 2012.

Purwanti, Dian. 2008. *Pengertian Website*. <http://deeyaan.blogspot.com/2008/03/pengertian-website.html>. Tanggal akses 22 Maret 2012.

Rochmah, Affriantari. 2010. *Perancangan Fitur Rekomendasi Film Website Solo Movie dengan Menggunakan Metode Algoritma Apriori*. Universitas Sebelas Maret. Surakarta.

Santosa, Budi. 2007. *Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis*. Graha Ilmu. Yogyakarta.

Suminar, Handayani. 2007. *Pengembangan Aplikasi Fuzzy Temporal Association rule Mining (studi kasus : data transaksi pasar swalayan)*. Institut Pertanian Bogor. Bogor.

Suneetha, K.R. dan Krishnamoorti, R. 2009. *Identifying User Behavior by Analyzing Web Server Access log File*. *IJCSNS International Journal of Computer Science and Network Security*(Vol.9 No.4.April).

Supriyanto, Aji. 2007. *Web dengan HTML dan XML*. Graha Ilmu. Yogyakarta.

LAMPIRAN A

Daftar kode status dari *server* ketika *browser* mengirimkan permintaan ke *server* (Nixon, 2010) :

1. *Informational*

Kode status ini menunjukkan respon sementara, hanya terdiri dari *status-line* dan *optional header*.

▪ **100 -Continue (Melanjutkan)**

Memberitahukan kepada *client* bahwa permintaan pertama telah sudah diterima dan seharusnya boleh melanjutkan ke sisa permintaan berikutnya. Dan mengabaikan bila permintaan telah terpenuhi.

▪ **101 -Switching Protocols (Berpindah Protokol)**

Memberitahukan *client* bahwa *server* akan beralih ke protokol yang ditentukan dalam header pesan pada saat *upgrade* selama sambungan saat ini.

2. *Successful*

Kode status ini menunjukkan bahwa permintaan *client* berhasil diterima, dipahami.

▪ **200 - OK**

Merupakan *respon* standar dari *server* yang menunjukkan bahwa permintaan (*request*) yang dilakukan oleh *client* telah berhasil diterima, dimengerti, disetujui, dan diproses dengan sukses (berhasil). Hasilnya yang akan ditampilkan tergantung pada jenis *request* yang dilakukan oleh *client*.

▪ **201-Created**

Permintaan berhasil dan sumber daya yang baru telah dibuat.

▪ **202 -Accepted**

Permintaan tersebut telah diterima untuk diproses, namun belum diproses.

Dalam *section* ini, permintaan mungkin tetap atau mungkin juga tidak akan ditindak lanjuti. Karena bila ditindak lanjuti akan menjadi batasan saat proses sebenarnya terjadi.

▪ **203 -NonAuthoritative Information (since HTTP/1.1)**

Server sukses dalam memproses *request*, namun meta informasi yang dikembalikan dalam isi *header* ditetapkan bukan berasal dari *server* asal.

- **204 -No Content**

Permintaan berhasil, namun tidak memerlukan isi dari permintaan.

- **205 -Reset Content**

Permintaan berhasil tetapi *user agent* harus melihat ulang dokumen yang menyebabkan permintaan tersebut.

- **206 -Partial Content**

Sebagian *request* telah diterima oleh *client*. Ini biasanya terjadi bila kita menggunakan perintah *Wget* untuk *enableresume* dalam mengirim data atau dalam mengunduh *file* dalam berbagai bagian secara simultan.

- **207 -Multi-Status (WebDAV) (RFC 4918)**

Isi pesan yang berikut adalah pesan XML (*Extensible Markup Language*) dan dapat berisi sejumlah kode respon terpisah, tergantung pada berapa banyak sub permintaan dibuat.

3. *Redirection*

Kode status ini menunjukkan bahwa tindakan lebih lanjut harus dilakukan oleh *user agent* untuk memenuhi permintaan tersebut.

- **300 -Multiple Choices**

Sumber daya yang diminta memiliki beberapa kemungkinan, masing-masing dengan lokasi yang berbeda.

- **301 -Moved Permanently**

Sumber daya yang diminta telah secara permanen dipindahkan ke lokasi yang lain (URL yang lain).

- **302 -Found**

Sumber daya yang diminta telah ditemukan di bawah URI (*Uniform Resources Identifier*) yang berbeda namun *client* harus terus menggunakan URI yang asli.

- **303 -See Other (since HTTP/1.1)**

Respon dari *request* telah ditemukan dibawah URL (*Uniform Resources Locator*) yang berbeda dan harus dapat diakses dengan menggunakan perintah GET.

- **304 -Not Modified**

Sumber daya belum diubah sejak terakhir diminta. Ini dapat menghemat *bandwith* dalam proses antara *server* dan *client*.

- **305 -Use Proxy (since HTTP/1.1)**

Sumber daya yang diminta hanya dapat diakses melalui *proxy* di lokasi yang ditentukan. Banyak HTTP *client* pada saat tulisan ini dibuat (seperti *browser Mozilla Firefox* ataupun *Internet Explorer*) tidak menanggapi status kode seperti ini terutama dengan alasan untuk keamanan.

- **306 -Switch Proxy**

Digunakan untuk masa yang akan datang (cadangan).

- **307 -Temporary Redirect (since HTTP/1.1)**

Sumber daya/alamat sementara sudah dipindahkan ke URI yang berbeda. *Client* harus menggunakan URI asli untuk mengakses sumber daya tersebut di masa depan.

4. Client Error

Kode status ini dimaksudkan untuk kasus di mana *client* tampaknya telah keliru.

- **400 -Bad Request**

Syntax yang digunakan untuk meminta data tidak dipahami oleh *server*.

- **401 -Unauthorized**

Permintaan membutuhkan otentikasi dari pengguna.

- **402 -Payment Required**

Digunakan untuk masa yang akan datang (cadangan).

- **403 -Forbidden**

Server menolak untuk memenuhi permintaan tersebut.

- **404 -Not Found**

Dokumen / *file* yang diminta oleh *client* tidak ditemukan.

- **405 -Method Not Allowed**

Metode yang diterapkan dalam permintaan tidak diperbolehkan atau tidak didukung untuk sumber daya ini.

- **406 -Not Acceptable**

Sumber daya yang diminta hanya mampu menghasilkan respon entitas yang tidak memiliki karakteristik konten seperti yang diminta.

- **407 -Proxy Authentication Required**

Permintaan pertama memerlukan otentikasi dengan *proxy*.

- **408 -Request Timeout**

Client gagal mengirimkan permintaan dalam waktu yang dibolehkan oleh server. Atau menurut W3 HTTP specifications: "The client did not produce a request within the time that the server was prepared to wait. The client MAY repeat the request without modifications at any later time."

- **409 -Conflict**

Permintaan tidak berhasil karena adanya konflik dari sumber daya.

- **410 -Gone**

Menunjukkan bahwa resource atau sumber daya yang diminta oleh client tidak terdapat lagi adanya. Namun berbeda dengan HTTP status kode 404, client tidak diperbolehkan lagi untuk meminta data lagi. Client seperti halnya mesin pencari harus menghapus resource (sumber daya) tersebut dari daftar index mereka.

- **411 -Length Required**

Server tidak akan menerima permintaan yang sah tanpa mengikuti ketentuan pada *content-length header*.

- **412-Precondition Failed**

Prasyarat yang ditetapkan dalam meminta satu atau lebih *header* adalah palsu.

- **413-Request Entity Too Large**

Permintaan tersebut tidak berhasil karena permintaan melebihi besar dari yang diizinkan oleh server.

- **414 -Request-URL Too Long**

Permintaan tidak berhasil dilakukan, karena proses untuk mengakses URL terlalu panjang.

- **415 -Unsupported Media Type**

Permintaan tersebut tidak berhasil karena konten yang diminta tidak didukung oleh sumber daya.

- **416 -Requested Range Not Satisfiable**

Client telah meminta sebagian *file*, tetapi *server* tidak dapat memasok bagian itu. Sebagai contoh, jika *client* meminta bagian dari *file* yang terletak di luar akhir *file*.

- **417 -Expectation Failed**

Harapan dalam permintaan *expect header* tidak dapat dipenuhi oleh *server*.

5. Server Error

Kode status ini menunjukkan kasus-kasus di mana *server* sadar bahwa telah keliru atau tidak mampu melakukan permintaan.

- **500 -Internal Server Error**

Permintaan tersebut tidak berhasil karena mengalami kondisi yang tidak diharapkan oleh server.

- **501 -Not Implemented**

Permintaan gagal karena *server* tidak mendukung fungsi yang diperlukan untuk memenuhi permintaan tersebut.

- **502 -Bad Gateway**

Server menerima respon yang tidak *valid* dari *upstreamserver* ketika mencoba untuk memenuhi permintaan tersebut.

- **503 -Service Unavailable**

Permintaan tidak berhasil karena *server* sedang lemah atau kelebihan beban.

- **504 -Gateway Timeout**

Upstreamserver yang gagal mengirimkan permintaan dalam waktu yang dibolehkan oleh *server*.

- **505 -HTTP Version Not Supported**

Server tidak mendukung atau tidak mengizinkan versi protokol HTTP dalam permintaan tersebut.

- **530 -User access denied**

Akses yang dilakukan user ditolak.

LAMPIRAN B

BULAN	Periode	Support	Confidence	Jumlah Rule
MARET	Awal Minggu	1%	10%	14
			30%	10
			50%	5
		3%	10%	8
			30%	6
			50%	3
		5%	10%	1
			30%	1
			50%	0
	Tengah Minggu	1%	10%	10
			30%	7
			50%	6
		3%	10%	2
			30%	2
			50%	2
		5%	10%	1
			30%	1
			50%	1
	Akhir Minggu	1%	10%	4
			30%	4
			50%	1
3%		10%	0	
		30%	0	
		50%	0	
5%		10%	0	
		30%	0	
		50%	0	
APRIL	Awal Minggu	1%	10%	7
			30%	5
			50%	2
		3%	10%	0
			30%	0
			50%	0
	Tengah Minggu	1%	10%	9
			30%	4
			50%	0
		3%	10%	0

			30%	0		
			50%	0		
			10%	0		
			5%	30%	0	
			50%	0		
			10%	15		
		Akhir Minggu	1%	30%	8	
			50%	7		
			10%	0		
			3%	30%	0	
			50%	0		
			10%	0		
		5%	10%	0		
			30%	0		
			50%	0		
		MEI	Awal Minggu	1%	10%	17
					30%	15
					50%	8
3%	10%			1		
	30%			0		
	50%			0		
5%	10%		0			
	30%		0			
	50%		0			
Tengah Minggu	1%		10%	10		
			30%	5		
			50%	1		
	3%	10%	0			
		30%	0			
		50%	0			
5%	10%	0				
	30%	0				
	50%	0				
Akhir Minggu	1%	10%	11			
		30%	8			
		50%	3			
	3%	10%	0			
		30%	0			
		50%	0			
5%	10%	0				
	30%	0				
	50%	0				

LAMPIRAN C

Ket : Awal Minggu
Tengah Minggu
 (No Fill) Akhir Minggu

MARET	APRIL	MEI
INFO : Jika mengakses [http://kh.google.com] maka mengakses [http://mw1.google.com]	INFO :Jika mengakses [http://r1rk9np7bpcsf0eekl0khkd2juj27q3o-a-fc-opensocial.googleusercontent.com] maka mengakses [http://search.twitter.com]	INFO : Jika mengakses [http://r1rk9np7bpcsf0eekl0khkd2juj27q3o-a-fc-opensocial.googleusercontent.com] maka mengakses [http://i377.photobucket.com]
	INFO :Jika mengakses [http://r1rk9np7bpcsf0eekl0khkd2juj27q3o-a-fc-opensocial.googleusercontent.com] maka mengakses [http://clients1.google.co.id]	INFO : Jika mengakses [http://www.mangafox.com] maka mengakses [http://www.google.co.id]
	INFO :Jika mengakses [http://urls.api.twitter.com] maka mengakses [http://content.yieldmanager.com]	INFO : Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://ad.yieldmanager.com]
	INFO :Jika mengakses [http://clients1.google.co.id] maka mengakses [http://www.google.co.id]	INFO : Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://api.twitter.com]
	INFO :Jika mengakses [http://content.yieldmanager.com] maka mengakses [http://www.google.co.id]	INFO : Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://safebrowsing-cache.google.com]

		INFO : Jika mengakses [http://content.yieldmanager.com] maka mengakses [http://safebrowsing-cache.google.com]
		INFO : Jika mengakses [http://www.google.com] maka mengakses [http://safebrowsing-cache.google.com]
		INFO : Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://clients1.google.co.id]
		INFO : Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://content.yieldmanager.com]
		INFO : Jika mengakses [http://ad.doubleclick.net] maka mengakses [http://www.google.co.id]
		INFO : Jika mengakses [http://pubads.g.doubleclick.net] maka mengakses [http://www.mangafox.com]
		INFO : Jika mengakses [http://ads.yimg.com] maka mengakses [http://www.google.com]
		INFO : Jika mengakses [http://r1rk9np7bpcsf0eekl0khkd2juj27q30-a-fc-opensocial.googleusercontent.com] maka mengakses [http://id.mg60.mail.yahoo.com]
		INFO : Jika mengakses [http://search.twitter.com] maka mengakses [http://api.twitter.com]
		INFO : Jika mengakses [http://twitter.com] maka mengakses [http://polldaddy.com]
INFO : Jika mengakses [http://kh.google.com] maka mengakses [http://mw1.google.com]	INFO : Jika mengakses [http://safebrowsing-cache.google.com] maka mengakses [http://www.google.co.id]	INFO : Jika mengakses [http://www.google.com] maka mengakses [http://s102.filesonic.com]

	INFO : Jika mengakses [http://armdl.adobe.com] maka mengakses [http://urls.api.twitter.com]	INFO : Jika mengakses [http://googleads.g.doubleclick.net] maka mengakses [http://platform0.twitter.com]
	INFO : Jika mengakses [http://armdl.adobe.com] maka mengakses [http://content.yieldmanager.com]	INFO : Jika mengakses [http://openx.detik.com] maka mengakses [http://block.opendns.com]
	INFO : Jika mengakses [http://api.twitter.com] maka mengakses [http://platform0.twitter.com]	INFO : Jika mengakses [http://clients1.google.co.id] maka mengakses [http://www.google.co.id]
	INFO :Jika mengakses [http://clients1.google.co.id] maka mengakses [http://toolbarqueries.google.co.id]	INFO : Jika mengakses [http://googleads.g.doubleclick.net] maka mengakses [http://content.yieldmanager.com]
INFO : Jika mengakses [http://ad.kaskus.us] maka mengakses [http://platform0.twitter.com]	INFO : Jika mengakses [http://s.youtube.com] maka mengakses [http://www.youtube.com]	INFO : Jika mengakses [http://scr.kliksaya.com] maka mengakses [http://clients1.google.co.id]
INFO : Jika mengakses [http://ad.kaskus.us] maka mengakses [http://www.blogger.com]	INFO : Jika mengakses [http://www.youtube.com] maka mengakses [http://geo.predictad.com]	INFO : Jika mengakses [http://googleads.g.doubleclick.net] maka mengakses [http://ocsp.verisign.com]
INFO : Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://scr.kliksaya.com]	INFO : Jika mengakses [http://www.youtube.com] maka mengakses [http://ad.kaskus.us]	INFO : Jika mengakses [http://pubads.g.doubleclick.net] maka mengakses [http://www.facebook.com]
INFO : Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://www.blogger.com]	INFO : Jika mengakses [http://geo.predictad.com] maka mengakses [http://ad.kaskus.us]	INFO : Jika mengakses [http://twitter.com] maka mengakses [http://www.google.co.id]
	INFO : Jika mengakses [http://block.nawala.org] maka mengakses	INFO : Jika mengakses [http://au.download.windowsupdate.com] maka

	[http://sp.ltassrv.com]	mengakses [http://clients1.google.co.id]
	INFO : Jika mengakses [http://au.download.windowsupdate.com] maka mengakses [http://www.mangareader.net]	INFO : Jika mengakses [http://safebrowsing-cache.google.com] maka mengakses [http://clients1.google.co.id]
	INFO : Jika mengakses [http://s.youtube.com] maka mengakses [http://pubads.g.doubleclick.net]	INFO : Jika mengakses [http://safebrowsing-cache.google.com] maka mengakses [http://www.facebook.com]
	INFO : Jika mengakses [http://r1rk9np7bpcsf0eekl0khkd2juj27q30-a-fc-opensocial.googleusercontent.com] maka mengakses [http://urls.api.twitter.com]	INFO : Jika mengakses [http://r1rk9np7bpcsf0eekl0khkd2juj27q30-a-fc-opensocial.googleusercontent.com] maka mengakses [http://www.blogger.com]

LAMPIRAN D

BULAN MARET

RULE	$\sigma_D(I)$	Support	$\sigma_D(X)$	C	$\sigma_D(Y)$	Benchmark	LiftRatio	W
[126, 127]	2.5	9.26%	6.5	38.46%	3.5	0.1296	2.9670	27
[126, 127]	2.5	9.80%	3.5	71.43%	3.5	0.1373	5.2041	25.5
[62, 28]	1	2.06%	1.5	66.67%	2	0.0412	16.1667	48.5
[62, 55]	0.5	1.03%	1.5	33.33%	2	0.0412	8.0833	
[94, 113]	1	2.06%	3	33.33%	1	0.0206	16.1667	
[94, 55]	1	2.06%	3	33.33%	2	0.0412	8.0833	

BULAN APRIL

RULE	$\sigma_D(I)$	Support	$\sigma_D(X)$	C	$\sigma_D(Y)$	benchmark	LiftRatio	W
[97, 46]	1	2.35%	1.5	66.67%	1	0.0235	28.33	42.5
[97, 12]	0.5	1.18%	1.5	33.33%	1.5	0.0353	9.44	
[69, 59]	0.5	1.18%	1.5	33.33%	2.5	0.0588	5.67	
[12, 8]	1	2.35%	1.5	66.67%	11	0.2588	2.58	66
[59, 8]	1	2.35%	2.5	40.00%	11	0.2588	1.55	
[56, 8]	1	2.82%	2.5	40.00%	3.5	0.0986	4.06	34.5
[15, 69]	0.5	1.41%	1.5	33.33%	0.5	0.0141	23.67	
[15, 59]	0.5	1.41%	1.5	33.33%	1	0.0282	11.83	
[21, 82]	0.5	1.41%	1.5	33.33%	0.5	0.0141	23.67	
[12, 13]	1	2.90%	3	33.33%	1	0.0290	11.5	66
[91, 85]	1	1.52%	2	50.00%	2	0.0303	16.50	
[85, 98]	1	1.52%	2	50.00%	1.5	0.0227	22.00	
[85, 65]	1	1.52%	2	50.00%	1.5	0.0227	22.00	
[98, 65]	1	1.52%	1.5	66.67%	1.5	0.0227	29.33	
[101, 102]	1	1.52%	2	50.00%	1	0.0152	33.00	
[6, 31]	1	1.52%	2	50.00%	1	0.0152	33.00	
[91, 84]	1	1.52%	2	50.00%	5.5	0.0833	6.00	
[97, 69]	1	1.52%	3	33.33%	2	0.0303	11.00	

BULAN MEI

RULE	$\sigma D(I)$	Support	$\sigma D(X)$	C	$\sigma D(Y)$	benchmark	LiftRatio	W
[4, 5]	1	1.57%	2	50.00%	1	0.0157	31.75	
[56, 2]	1	1.57%	2	50.00%	11	0.1732	2.89	
[19, 71]	1	1.57%	3	33.33%	1	0.0157	21.17	
[19, 34]	1	1.57%	3	33.33%	6.5	0.1024	3.26	
[19, 47]	1	1.57%	3	33.33%	8.5	0.1339	2.49	
[20, 47]	1	1.57%	2	50.00%	8.5	0.1339	3.74	
[1, 47]	1	1.57%	2.5	40.00%	8.5	0.1339	2.99	
[19, 11]	1	1.57%	3	33.33%	1	0.0157	21.17	63.5
[19, 20]	1	1.57%	3	33.33%	2	0.0315	10.58	
[80, 2]	1	1.57%	3	33.33%	11	0.1732	1.92	
[21, 56]	1	1.57%	2	50.00%	2	0.0315	15.88	
[9, 1]	1	1.57%	1.5	66.67%	2.5	0.0394	16.93	
[4, 111]	1	1.57%	2	50.00%	2	0.0315	15.88	
[8, 34]	1	1.57%	1.5	66.67%	6.5	0.1024	6.51	
[27, 49]	1	1.57%	1.5	66.67%	1	0.0157	42.33	
[1, 17]	0.5	1.11%	1.5	33.33%	0.5	0.0111	30.00	
[16, 82]	1	2.22%	3	33.33%	1.5	0.0333	10.00	
[69, 101]	1	2.22%	1.5	66.67%	2	0.0444	15.00	45
[11, 2]	0.5	1.11%	1.5	33.33%	5.5	0.1222	2.73	
[16, 20]	1	2.22%	3	33.33%	1	0.0222	15.00	
[38, 11]	1	2.25%	2	50.00%	7.5	0.1685	2.97	
[16, 75]	1	2.25%	3	33.33%	1	0.0225	14.83	
[21, 63]	1	2.25%	2	50.00%	14.5	0.3258	1.53	
[27, 2]	1	2.25%	2	50.00%	4.5	0.1011	4.94	44.5
[19, 11]	1	2.25%	2.5	40.00%	7.5	0.1685	2.37	
[47, 11]	1	2.25%	2.5	40.00%	7.5	0.1685	2.37	
[47, 63]	1	2.25%	2.5	40.00%	14.5	0.3258	1.23	
[4, 10]	1	2.25%	3	33.33%	1	0.0225	14.83	