Ilkka Lilleberg

Logging Web Behaviour for Association Rule Mining

Helsinki Metropolia University of Applied Sciences

Master of Engineering

Master's Thesis

2 October 2015

Author(s) Title Number of Pages Date	Ilkka Lilleberg Logging Web Behaviour for Association Rule Mining 53 pages 2 October 2015
Degree	Master of Engineering
Degree Programme	Information Technology
Specialisation option	
Instructor(s)	Vesa Ollikainen, PhD, Senior Lecturer Pasi Hiltunen, Vice President Nitro Group

The aim of this study was to suggest improvements to web server logs of existing web services. The ultimate goal was to study how logging process could be developed to enhance using log data with data mining tools to get better results. Better results could be, for example, understanding which actions during web browsing session indicate interest to buy certain products in an online store.

Data sets studied consist of web server data from Nasa web service, Sonera Joulukampanja webserver data and web server data from liandersson.fi web site. Liandersson.fi and Sonera Joulukampanja are based on Linux, Apache or Nginx, PHP and MySQL technologies.

For the study a data mining environment using association rule analysis to mine new information from web server log data was developed. The emphasis was on doing research on what kind of data mining results can be achieved with different kinds of log data and how developing the logging of web server data can affect the possibilities of data mining.

The two stated goals of the study were to generate a tool set that can automate the analysis of chosen log files and produce association rules of the log data and to analyse the capabilities of this data mining process, with the data available and to create a list of actions points that can be used to develop data logging.

Yet another objective was to develop the logging of web server data and thus improve analysing this data in the future. Alongside developing a data mining environment, the key outcome of the study is instructions of how to plan web server logging in a way that takes into account the requirements of data mining of the specific data in question in order to use the information created by data mining to improve existing web services.

Keywords	data mining, association analysis, machine learning

Tekijä(t) Otsikko Sivumäärä Aika	Ilkka Lilleberg Web-käyttäytymisen kirjaus palvelinlokeihin assosiaatio-analyysiä varten 53 sivua 2.10.2015
Tutkinto	Master of Engineering
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	
Ohjaaja(t)	FT, lehtori Vesa Ollikainen Pasi Hiltunen, Vice President Nitro Group

Tämän työn tarkoitus on tutkia olemassa olevien WWW-sovellusten ja -palveluiden lokitietoja ja selvittää, miten niiden keräystä tulisi kehittää, jotta tiedonjalostamistekniikoilla saataisiin paras mahdollinen hyöty saatavilla olevasta datasta. Hyöty voi olla esimerkiksi tieto siitä, mitkä tapahtumat ennakoivat kiinnostusta ostaa tietyn ryhmän tuotteita verkkokaupasta.

Aineistot joita tutkitaan ovat Nasan verkkopalvelun palvelinlokidataa, Sonera Joulukampanja - palvelun palvelinlokidataa ja liandersson.fi-palvelun palvelinlokidataa. Palvelut liandersson.fi ja Sonera Joulukampanja on toteutettu Linux-, Apache- tai Nginx-, PHP- ja MySQL tekniikoilla.

Tässä työssä kehitetään tiedonjalostusympäristö, jolla assosiaatiosääntöanalyysiä käyttäen pyritään jalostamaan tietoa WWW-palvelinten lokidatasta. Tässä tutkimuksessa selvitetään minkälaisia tuloksia erilaisella datalla saadaan ja miten hyödyllistä tietoa voitaisiin saada aikaan, mikäli lokien keräyksessä olisi paremmin otettu huomioon tiedon jalostamisen vaatimukset.

Tutkimuksen ensimmäisenä tavoitteena on luoda työkalut, joilla voi automatisoida valittujen lokitiedostojen analysoinnin ja luoda assosiaatiosääntöjä lokidatasta. Toisena tavoitteena on analysoida tämän tiedonlouhintaprosessin mahdollisuuksia tuotantopalvelimilta saamallamme datalla ja luoda lista ohjeita, joilla kehittää lokitietojen keräämistä.

Lisäksi tavoitteena on kehittää palvelinten lokien keräämistä ja siten mahdollistaa lokien sisältämän datan parempaa analysointia tulevaisuudessa. Tiedonjalostusympäristön kehityksen ohella työn keskeisiä tuloksia on ohjeistus, jonka avulla verkkopalvelun tuotantoon siirryttäessä voidaan ottaa paremmin analytiikan vaatimuksia huomioon.

Avainsanat	Tiedon jalostaminen, assosiaatiosääntöanalyysi, koneoppiminen

Contents

List of Abbreviations

ARFF	Attribute-Relation File Format is file format used by Weka machine learning software.
API	Abbreviation for application programming interface. Application programming interface is set of tools and protocols to create interface in building software applications.
CPU	Abbreviation for Central processing unit. Central processing unit is also called computer processor. It executes instructions given by computer programs.
CRISP-DM	Cross Industry Standard Process for Data Mining is process model used in data mining.
CSS	Cascading Style Sheets is a style sheet language used to describe visual outlook of documents using markup language.
CSV	Abbreviation for Comma-separated values. Text based file format to deliver data rows, which have data columns separated be delimiter, most commonly comma.
GPS	Abbreviation for Global Positioning System. GPS is satellite based positioning system widely used world wide in location based applications.
HPC	Abbreviation for High-performance computing. Term High-performance computing is used to describe either very high-level computational capacity or high capacity distributed network of computers.
HTML5	Abbreviation for Hypertext Markup Language's version 5.
HTTP	The Hypertext Transfer Protocol is a protocol for client-server computing. World Wide Web is based on HTTP protocol.

IP	The Internet Protocol is protocoll used to deliver packets in IP based networks.
IT	Abbreviation for Information Technology. Word is used to describe use of computers and communication equipment to send, retrieve and process data.
PDF	Abbreviation for Portable Document Format. Portable Document Format is used to present documents independently of operating systems and software vendors.
PHP	PHP: Hypertext Preprocessor is a programming language that is very commonly used in web content management systems.
RAM	Abbreviation for Random-access memory. Random-access memory is used in computer hardware as a data storage.
RSS	Abbreviation for Rich Site Summary. Rich Site Summary is group of standards used to publish data from web based services.
SAN	Abbreviation for Storage area network. Storage area network is network dedicated to distributed data storage.
URL	Abbreviation for Uniform Resource Identifier. String used to describe where certain piece of information is located.
WIFI	Wi-Fi is a local area wireless networking technology used to network devices wirelessly using wireless local area network.

1 Introduction

The field of data science has been growing rapidly in recent years. There are more and more applications that combine statistical analysis with computer science. These applications have also become a growing business in private market and therefore a growing area of interest.

The interest in this field has been fuelled by a growing capacity to store and process data, which has created new possibilities for processing growing amounts of data with decreasing costs. This development is making it cost efficient and financially lucrative to use methods of data science and data mining to produce information on existing data in new areas. During the first years of data science, focus was on online retailers, but the same methods can be used in other fields as well. In the case of this study the methods are used to learn from web site users online behaviour.

More specifically this thesis uses three different data sets from web server logs to test an association rule analysis tool developed partially in this study, and to gather information on what kind of data is needed for the analysis to be potentially useful. Association rule analysis, which is a central tool in data mining, aims at finding association rules that are relationships between items on events in the data.

Even more important than knowing which data mining methods can be used in different circumstances to mine data for new information, is to know exactly what kind of data has to be collected for the data mining to be fruitful.

To be able to see which data was possible to mine for usable association rules, there was a need to have tool a set that covered all different phases of the analysis process. Therefore this study concentrates on what kind of tools can be used and how these tools could be scaled to bigger data sets and faster iterations of analysis than were needed for the example data sets in this study.

In this thesis the aim is to have two end products as goals of the study.

1. To generate a tool set that can automate the analysis of chosen log files and produce association rules of the log data. This creates a possibility to have such analysis executed on high frequency to gather information of user behaviour in almost real time. This thesis covers saving of data, storing of data, analysis of data and using results of these analysis to develop web services further. Most of the theory section is used to describe different methods and tools to execute the previously mentioned tasks. The first part describes in short what is Data analysis in web development in general and the later parts go deeper into tools and methods.

2. To analyse the capabilities of this data mining process, with the data available and to create a list of action points that can be used to develop data logging. This list of actions points can produce richer log data than the data available for data mining in this thesis. This development in data can lead to better possibilities of data mining web server log data with association rule analysis.

Preprocessing is needed to read data from log files, create sessions of user events and discard sessions that would not contribute in finding usage patterns. Post processing is needed to find the rules that lead in to events that the researcher is interested in. Association rule analysis can create large sets of rules, so it is important to process rule sets to reduce rule sets to only the potentially interesting rules.

The priority in this kind of association rule analysis is in finding which behaviour leads in to the behaviour either wanted from web service users or which specifically is not wanted, so that this information can be used to design web sites in such a way that they more effectively guide to the desired behaviour, prevent unwanted behaviour or detect when it is likely that an event of interest will take place.

Firstly data is preprocessed with a preprocessing script developed for this thesis and described in detail in Chapter 5.2. Secondly, Weka data mining tools are used to execute data mining using association rule analysis with this preprocessed data. Thirdly, postprocessing script is used to refine association rules created by the association rule analysis.

Finally a list of action points is suggested for useful data logging in a web service. It is written in four steps to consider when planning an infrastructure of a web service and when configuring the server software used in a web service. It was possible to create these conclusions by analysing how potentially useful rules could be data mined with selected log files for web services in various application areas.

To scientifically verify the results that can be achieved with action points listed in this study, a new web server logging configuration should be made for the services in question. Also, the new data would be need to be produced by the web servers, that could be reviewed to verify that these actions points produce measurable results. Unfortunately this is out of the scope of the present project. So to configure web servers with the instructions and analyse the data produced would need to be another project.

2 Analysing Web Usage

In this section the focus is placed on giving a general understanding of the field of gathering data for web analytics purposes.

2.1 Introduction to Web Usage Analytics

Web analytics create reports and statistics from the use of web service, using different statistical methods. In brief it has information about users of a service: who they are, where they are coming from, what do they do when they use the service and how they use it. [1,5]

Information gathered is mainly used for answering the following questions: where do the users come from, how long do they spend time using the service, where do they go when they leave the service, how do they use the service, is the service achieving its goals and are there problem or error cases users face while using the service. [1,7]

The most typical use case of web analytics is to set certain goals for the service and follow how well these goals are achieved by web analytics. The common term to describe success to get a user to behave in a manner service provider has planned the user to behave, is conversion. Conversion means for example the percentage of users

who buy a product from an online store or watch a certain video. Conversion can be viewed and calculated as any other number in analytics. It differs from most other numbers by measuring the achieving target of the service and therefore often defines if service is reaching the service's goals. [2,34]

There are four general categories of techniques to gather data for web analytics: JavaScript based tags, server logs, web beacons and packet sniffing. JavaScript based tags and server logs are the most commonly used. [3]

Server logs are created on the web server and can exist for different purposes, for example to debug error cases or to gather data for analytics, while JavaScript based tags are used only for analytics. JavaScript tags are also capable to gather information of user behaviour, that doesn't create server logs. For example hovering of an image can be configured to send a JavaScript notification to the analytics server.

Server logs often have more detail on them, because everything that is being downloaded from webserver can leave logs and logs are generated even if server error occurs that prevents JavaScript from loading or web client can disable JavaScript completely or from specific analytics and tracking servers. [4]

Web analytics is used as methodology to improve web services, not as a technology, but as a process to understand and find development objectives and goals need to be revised constantly. [3]

2.1 Data in Server Log Files

In this study web usage mining is used to create new information from data collected in server logs. This chapter describes what data mining of server logs means and the methods it can be accomplished with.

When writing about web usage mining, the term web mining is often used. This can mean one of the following things: web content mining, web usage mining or web structure mining. This chapter presents web usage mining using web server logs as basis for analysis. [5]

Online services save and store large quantities of data related to user behaviour. Data mining based on this data can give organizations important views to how their services are used, how they can be optimized and developed. This data mining includes looking for patterns in usage data. Web server logs is one of the most important sources of this data to be analysed for patterns. [6,101]

Server logs are files, which include events that server software does, and records in server log files. This is done to have logs files that can be used to debug error situations or analyse server behaviour. Web servers have different log files. Most common log file is access log that has logs of every request processed by web server. This log consists of a list of log entries each having a timestamp, identifier of client host, URL requested, referrer, user agent, status code and size of reply in bytes. [7,75]

Below in Listing 1 there is a web server log entry example for NASA web service data.

```
slppp6.intermind.net - - [01/Aug/1995:00:00:12 -0400] "GET
/images/ksclogosmall.gif HTTP/1.0" 200 3635
```

Listing 1. Web server log entry. [8]

It consists of the address from which the client is using the server, this is the first string in the row. In this case it is “slppp6.intermind.net”. The second is timestamp of the time request has been made “[01/Aug/1995:00:00:12 -0400]”. The third piece of information is the fact that the request was a GET request, which requested ksclogosmall.gif image file. It was done in HTTP/1.0 protocol. The request received status code 200, which means the request was successfully processed. The last piece of data is the number of bytes that was the size of the response. [7,75]

Other important logs on web server are referrer logs that have collected information on from which web sites and pages links are followed to the web server in question. [9,560]

Client behaviour can be followed, to some extent, by studying the server logs. Server logs show where clients came from and provide the means for tracking the users. Being able to track users and sessions is a fundamental part of web usage mining. [6,1]

There are two different ways of following the client. One is to identify and the other to track users. Identification means to know their real name or similar implicit identifier. Tracking means to know which actions were done by the same client, but doesn't necessarily include knowing who this client is. Identification most often needs logging in or similar method of authenticating, while tracking only needs enough data to know for reasonable probability that which requests came from same browser. [5,1]

There are two common ways to track clients of a web service, with cookies or URL rewriting. [5,1]

Cookies can be separated to two categories. Server cookies are cookies that the server assigns to a client that makes a request to the server and that does not have a server cookie in the sent request. After the server has replied to a client, which did not send cookie information with its request, the server responds by assigning a cookie and delivering it with the response. The web client can then decide if it will start using this cookie and be tracked, or decide to not cooperate with tracking by plainly not using the cookie assigned. Usually this is defined in web browser settings. [5,1]

Web applications can and often do create their own cookies for authentication and tracking of users. Server cookies can be delivered to users in the event the server is only serving static files. [5,1]

A common way to have a capability to track users in web server logs, is to assign a tracking id to them and attach this id the query string of the URL the client sends. This dos not work with static html pages, but needs a dynamic application. [5,2]

To be able to track a user, the observant must be able to see when user starts a visit to the service and define somehow when that visit has ended. These visits are called sessions, which are defined as a sequence of page loads and action by user during a period of time. This period usually has expiration time, so that after a period of inactivity, for example 30 minutes, it is considered that the session has expired and a new page load from the same user is considered a new session. In most cases services that have authentication offer possibility to log out and end session by users own activity. These are only ways for a web service to find out if user has ended session by looking at the web log data. This is often enough, even thought it is possible to track more intensively with JavaScript that send requests to notify service of specific actions taken. [5,2]

If the web service cannot identify a user with a cookie or a query string, a common method is to combine the IP address with the User Agent string to have a pseudo unique identifier for the user. This method can solve most sessions in many cases, but is not very effective method, as shown in Table 1. [5,2]

Table 1. Statistics about a web site usage and how different methods of finding unique identifers from web usage data provide different levels of identifiability.

Total Page Views	92,000
Unique Cookie Ids	6,842
Unique Visitor IDs after re-heading	4,285
Unique IP addresses	4,548
Unique User Agents	950
Unique User Agent + IP Address	5,788

There is a very common case of misidentification that is called de-heading. It means that when browser makes its first request to the server, it does not have a cookie for tracking. After the first reply from the server, the user client is assigned a cookie that can be used to track the user client. Because the first request did not have this cookie, it can easily be counted as a separate session from the other requests that had the cookie. So if a user makes four requests, the first can be seen as an anonymous user that did one request and the next three requests can be seen as a separate session which was tracked by cookie. This same problem can be the case with URL rewriting or similar tracking methods as well. [5,3]

This problem not only causes sessions to be seen only partially, but also increases the number of sessions and makes it seem like there would be more users that do less. This also creates problems of counting conversion rates or similar statistics. [5,3]

Problem of de-heading can be solved by an action known as re-heading. It means connecting the first request of the session with the rest of the session. It is not always straightforward or 100% accurate. Below there are some methods for re-heading session data. [5,3]

User agent as a solution to de-heading

The browser sends the user agent information about the web client software as a string in every request. There is only a handful of the most common browsers, but the combination of browser versions and operating systems creates tens of thousands unique

user agent strings. By categorising web service clients by user agents, we can make some assumptions about them. Using this categorising together with IP addresses, which are not necessarily unique, but together with user agents can be used to make assumptions that same IP address, with same user agent in same time frame is the same user. Such assumptions can be accurate enough to make analysis based on the data. [5,3]

Looking at the time stamps makes sense in re-heading. For example often there are patterns, like redirecting of traffic, that can be observed to find where de-heading takes place. For example if every request to a certain page or subdomain is redirected in 3 seconds to another page, we can relatively safely assume that such patterns found in web server logs can be interpreted as de-heading and re-head them safely. [5,3]

When developing an algorithm to implement re-heading one must take into consideration the peculiarities of the service in question. Also, it must be considered an iterative process that can needs assignment of sufficient processing and memory capabilities. [5,4]

Web usage mining with web server logs comes with its own challenges that are not simple to solve, even if server logs have existed for quite some time and are relatively uniform in nature. At least server generated tracking cookies should be implemented to have simple tracking capabilities. Challenges related to this field depend heavily on the system used and can have large benefits if implemented well. Taking the intended analysis methods in account when engineering the web service can have en effect on providing more accurate data. [5,4]

One of the tasks in analysing data is to separate different users from each other. For example different robots and automated web crawlers are generating a growing part of web usage data. Being able to separate these from actual humans using web sites is useful in looking for usage patterns and user flows. Looking into user agents and IP ranges are effective ways to recognize well-known and common robots. Experiments with using clickstream pattern analysis to recognize not so known robot behaviour from human behaviour have shown good results.

3 Data Mining

In this section the focus is placed on giving a general understanding of the field of data mining. The section describes the concepts of data mining and how it has been growing and developing recently.

3.1 Introduction to Data Mining

Data mining is a process of extracting new information or knowledge from existing data. This term can be misleading, since data is being mined, to gather information and knowledge, but the term is still not information mining. This term is still keeps focus on aspects that are important in data mining. [10,5]

Data is being used and needed in large quantities to harvest for new knowledge. Data mining is the process where data is gathered and processed in several phases to draw conclusions and harvest the new knowledge. [10,5]

Although data mining originates from 1980s, still in 1990s the field was not mature and in process of being defined. Some of the concepts existed, such as data models and analysis algorithms. In 1999 group of big companies from different fields joined together to standardise data mining and that led to creation of CRISP-DM, the Cross-Industry Standard Process for Data Mining. Data mining was not tied to any specific tools by then, but merely in concepts that could be used with any data sets or tools. [11,5]

CRISP-DM created a conceptual model to define the process that is data mining. This conceptual consists of 6 steps that are executed in numerical order and can be iterated. [11,5]

1. Business Understanding or organizational understanding is a step where understanding of an underlying field of interest is studied to be capable of understanding the questions that can and are useful to be answered. Data mining algorithms can process data and answer different questions related to data, but to be able to ask the questions that are relevant to the field, data miner needs to have broad understanding of the field of study. The result of this step could be a set of questions the data mining would try find answers to. Such as “which pages direct customer to contact us” or “which content

makes customers return to our service". [11,6-7]

2. Data Understanding is the step during which information and understanding is gathered on what data there is available for research, where is these data gathered from, who gathered it and what kind of problems may arise from it. This step includes verification of data's reliability and accuracy of data. [11,7-8] For characteristics of this step in case of this study see Chapter 2.1.

3. Data Preparation is the step that includes processing of data at hand, so it can be used with data mining algorithms and tools. Data often comes in various forms, and these forms can be unconventional and not easily turned into columns and paragraphs to process. Data preparation step includes joining data sets together, reducing data to only ne necessary data, working to solve possible problems like missing data and reformatting data to be consistent. [11,8-9] In Chapter 4.2 this study explains this step in the context of web server logs.

4. Modeling means building models for the data to process. In data mining a model means a computerized representation of real-world observations. These models can be designed to predict or to classify the data being processed. Classification means to find certain groups of data rows that fit same class and use that to classify data. Predictive models give predictions of certain columns of data rows. This is the step where data mining goes from preparation to processing and producing information. The nature of the model depends on the problem at hand. It can be in a form of a decision tree or a neural network etc. For the analysis of server logs, the model gets the form of association rules. Model of this study is found in Chapter 4.3.2. [11,9]

5. Evaluation step consists of evaluating the results of data mining. Questions that need to be answered are: are there false positives, were interesting patterns found, were the chosen methods correct for this data set and question and if the data did include interesting information to be found. There are mathematical and logical techniques to evaluate data mining project's success, but evaluation must also include human element that may have operational understanding that can be difficult to measure in mathematical methods. [11,10]

6. Deployment step consists of actually running and possibly automating the data mining process, meeting the end users of this information, possibly incorporating data to another systems and databases, perfecting the model in use and measuring if the in-

formation produced by data mining is functional and usable to the needs of the organization. [11,10-11]

Development of computer processing power and information technology has driven gathering, storing, accessing, analyzing, processing and managing of larger amounts of data and information in real time, which has created new possibilities in mining information and using data beyond ways that were previously possible. [12,192]

This has kept data as an important part of information technology development and research. While development IT has brought changes in many fields, in the field of data mining it has created such new opportunities that the whole field has started to grow rapidly and gather importance on much larger audience. [12,192-193]

3.2 'Datafication' and Recent Growth of Data Mining

Currently data is being among the most valuable resource within some organizations [1, 12]. For example Google has built a huge business of its search engine, which is based on gathering huge amounts of data and providing search capabilities to this data. New possibilities created by the fast progressing field of data mining affect production capacity, profitability, and competence and are changing the way many fields of business are viewed. There are different opinions about Big Data, Data Science and similar concepts, because of the fast progress on data-centric studies. [12,192-193]

A new term 'Datafication' has emerged to describe this paradigm, of data becoming more valuable source of more information and it's importance in Business Analytics. This concept can be understood as how new information is created from data related to all aspects of life. [12,193]

Currently our society can store data from many fields. Most of if not all people participate in creating more and more data daily. Many fields where data can be collected, could not be thought of before as source of useful data. This is what 'Datafication' means, ability to take anything related to human activity and turn it into data that can be stored, processed and mined. [12,193]

‘Datafication’ can be considered through for example how social media has created great amounts of data about human friendships and relations, in a way that can be mined and learned from by machine learning. This process is developing fast, as new methods of ‘Datafication’ bring new areas of human interaction to realm of data mining. Adoption of new devices, such as smart phone, has already expanded the amount of data gathered by our daily actions, but this process is going forward fast, for example by Google Glass, which brings human look in to field of data gathering and mining. [12,193]

4 Association Rule Mining

In this section the focus is placed on introducing the concept of association rule analysis and the different phases it consists of.

4.1 Introduction to Association Rule Analysis

In recent years a lot has been written about web usage mining. This has meant finding ways to get new information of users by analysing their behaviour. One of the most used methods has been association rule mining. Association rule mining is used to find patterns in data, by looking for events that associate with each other. [7,74]

Association rule mining was originally developed to analyse data of transactions, to find the knowledge which products were bought together. [7,75] This is common in retail to want to know which retail products go to the same shopping baskets. In this thesis we want to find out which use cases give hints of other behaviour, in other words show association of certain actions done by users.

Association rule generation is done, by using sets of actions, in this case server log rows. The actual association rules are relations of specific action sets. The validity or meaningfulness of association rule is measured mostly by two concepts: support and confidence. [7,75]

Below there are formulas of how to calculate support, lift and confidence.

Support is used to measure how frequently the action set is in the transaction data. It measures how many times the set of transactions occurs relatively to the size of data set.

$$\text{Support(X)} = \frac{|\{t \in D | X \subseteq t\}|}{|D|}$$

[7,75]

In the example, if support of ({X_{12}=1, X_{17}=1, X_8=1}) is 0.2 and number of transactions is 1000, that would indicate that all three items are present in 200 of those transactions.

Confidence of a rule measures how likely the rule to take place. Rule's confidence of 1 means every time the item set in the rule premise is present, also the rule conclusion is. Smaller the confidence the less likely the presence of premise and conclusion is. It is calculated by formula below.

$$\text{Confidence(r)} = \frac{\text{support}(\text{Y} - \text{X})}{support(X)}$$

[7,75]

In the example, if confidence of a rule that X_{17} leads to X_{12} is 0.5, it means that in group of 1000 transactions, if X_{17} is present in 500 transactions with support of 0.5, X_{12} is present in 250 of these transactions.

Lift is used to count how many times more likely is that items in the rule would occur together than these items occurring on separately in the data. It is used to calculate if there is statistically relevant connection in occurrence of items in a rule.

$$\text{lift(A -> B)} = \frac{\text{Confidence}(\text{A} -> \text{B})}{support(b)}$$

[12,28]

In the example if a rule that X_{17} leads to X_{12} has confidence of 0.5 and X_{12} has support of 0.35, the lift of these two items occurring together is 1.428. So it could be described that likelihood of X_{17} and X_{12} occurring together is 1.428 more likely than these items occurring separately of each other.

Other measures used to measure the relevance of association rule are leverage and conviction.

Leverage is also measure for statistical dependency of certain items of in larger data set. It is meant to show how many more occurrences item set has as a set compared to items appearing separately. Difference to lift is that it can suffer from rare item problem.[14]

$$\text{leverage}(X \rightarrow Y) = P(EX \cap EY) - (P(EX)P(EY))$$

[15]

In the example if X_{17} and X_{12} together has support of 0.25, X_{17} has support of 0.5 and X_{12} has support of 0.35, this indicates that the leverage of this rule is 0.075.

Conviction is different way to measure the probability of association rules accuracy by comparing support and confidence of items in the rule. It differs from lift by taking into account also the probability of items existing separately,

$$\text{conviction}(X \rightarrow Y) = \frac{1 - \text{Supp}(Y)}{1 - conf(X \rightarrow Y)} = P(EX)P(E\neg Y)/P(EX \square E\neg Y)$$

[16,255-260]

In the example, if example rule that X_{17} leads to X_{12} has confidence of 0.5 and X_{12} has support of 0.35, the conviction for this rule is 1.3.

One problem with the association rule mining is that it does not take into account the notion of temporal distance of actions. So when looking for patterns the miner does not mind in which order or how far between actions took place, when looking for patterns. Some believe that the actual order of actions is crucial in finding usage patterns, which can have the most valuable information. [7,76]

Before association rule mining can be applied to set of data, the data must be processed. This processing aims to make the data analysable be the mining software.

There are several higher-level tasks that also need to be considered after the low level processing of changing data format.

Higher level tasks that need consideration before association rule mining can take place are data cleaning where unusable data is removed, user identification where users that can be identified are given explicit id to ease finding patterns, session identification where page views considered single session are categorized to same session, server log rows that can be tracked to single page view are reduced to one row and the missing data patterns that caching can create is taken to consideration. [17,10]

Different weights can be defined, so that different actions taken can be the ones that are looked for. For example in this thesis, shopping in online stores is studied to observe buying of products. Because of this emphasis, rows with events leading to buying can be given heavier weight. Weight can be binary or relative amount of wanted behaviour in the users session. The wanted behaviour can be anything that is found on the log, often purchase or opening of specific document. [17,10]

There are several different algorithms used in association rule mining. The most common algorithms are Apriori, Eclat and FP-growth algorithms.

Apriori is the first and the most well known one. Agrawal et al. developed it in 1994. It soon became the most known data mining algorithm in use. It was know as basket case analysis, because of it's most common use in retail. Soon it was discovered to be useful also in the fields of medical, banking, telecommunication and website navigation analysis. [13,26]

Eclat algorithm uses a different approach to going through item sets in the data. Eclat is more effective in fast counting of support of each item set and dropping sets that have support less that minimum of support defined to be of interest. Eclat uses what is called vertical database layout, it keeps large amount if information in memory. Eclats weakness is using more memory for its operation than Apriori. [18,112]

To be able to increase processing efficiency, new algorithm FP-Growth was developed in 2000. This differentiated from Apriori by using complicated tree structure that uses link list and/or hash table structure. The biggest improvement brought by FP-Growth is

that it goes through the whole data only twice. This kind of algorithms are usually used with large sets of data, so having methods that need to go through the data less is very good feature in making its use more scalable to larger data sets with less time. FP-Growth applications are more complicated to write than similar applications written with Apriori. [13,26]

To be able to use association rule analysis, other data mining algorithms must be used in preparation. [13,26]

4.2 Preprocessing

To enable executing association rule analysis with web server logs, the data needs to be processed. This corresponds to step 3 of CRISP-DM model presented on page 10. There are several different methods to execute the analysis and the different software need data in different file formats. Later in this chapter the file formats and methods that need to be used to convert data from list of actions taken to sessions that consist of many actions are described.

Preprocessing is also very important in limiting data to only the data that is useful for the specific analysis. Preprocessing can also be useful in grouping data in purposeful groups

The first step to accomplish in preprocessing server logs for association rule analysis is to decide which format one needs to have the logs converted to. Requirements for data format were defined by the choice of software to execute association rule analysis. In the present study it was decided to use Rapidminer and Weka as the analysis software. Rapidminer needs to have data for association rule mining in CSV file format, while Weka uses ARFF file format.

Both software packages Weka and RapidMiner need to have the data grouped as rows where each row corresponds to the list of events that have happened in a specific session of using the web server, while columns of the file are the list of all events being taken in to consideration.

CSV file format consists of data rows that are plain text. In data rows different columns are separated with a delimiter such as ";" or "," symbols. CSV file can also have a header row that defines the names of data columns.

Below as Listing 2 there is an example of a CSV file row using ";" as delimiter.

```
1;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;1;0;0;0;0;0;
```

Listing 2. Example row of a CSV file created for this example purpose.

This example row has two events that did take place represented by number 1 and several columns that imply that the event did not take place represented by number 0.

ARFF file format consists of two parts. The header part where data in the file is described and data part where the actual data is.

Below as Listing 3 there is an example of very simple ARFF file. This has been simplified from the real data.

```
@RELATION ACCESS_LOG
@ATTRIBUTE   "GET   /fi/terminal-operations/new-page-375.html
HTTP/1.1" {true, false}
@ATTRIBUTE   "GET   /fi/extranet/ohjeet-319.html   HTTP/1.1"
{true, false}

@DATA
false,true
true,false
```

Listing 3. Example of an ARFF file created during this study.

In this example there are two transactions that can be measured and two sessions that show if the transaction did take place.

Header contains the name of the data set and data attributes and the attribute types. In case association rule analysis the nominal data type is used and the definition of attribute also consists of possible values. In this case the possible values are True and False.

ARFF file data part is similar to CSV where data of different columns is separated with delimiter and data is in plain text. Data has to be according to what has been defined in the header part of the file.

If this were shopping basket analysis, as was the most common use for association rule analysis before, the log rows would be products bought and they would be grouped to shopping lists of specific shopper and formed to rows with everything that was shopped in single visit to the store. Columns would represent available products and each row would represent specific shopping session committed by a single customer.

With web server logs this grouping had to be done by finding which log rows were part of the same session. In the case of this study, the choice was to define session as a group of events coming from same source with maximum gap of 30 minutes between requests. The definition of the same source here depends on the web server logs. Some logs have users IP address and information about users browser and operating system, while other logs have less information. Because of this variation, defining the source is more reliable with some logs than others, but definitely not very reliable in any case. Still information available on the source can be enough to define the source with enough reliability, to have potentially useful information.

Web servers have different formats for web server access logs. The format being used is usually also configurable. Because of this preprocessing has to take log format in use in to account when preparing for preprocessing. Also different server setups and configurations can cause the data in the log files to differ, or often some data is missing or in not coherent or usable for some reason.

Below in Listings 4-6 there are examples of different log formats and different problems or characteristics that follow the server setups.

Listing 4 is a row of web server log from Nasa web server access log. This log has most of the information that web server logs usually have in usable format, but it is lacking information about the user, such as the possible user name and information about browser and operating system of the user.

```
in24.inetnebr.com - - [01/Aug/1995:00:00:01 -0400] "GET
/shuttle/missions/sts-68/news/sts-68-mcc-05.txt HTTP/1.0"
200 1839
```

Listing 4. Example row of Nasa web server log data. [19]

In Listing 5 there is Nginx web server access log. This has more information about the user than previous log, such as referrer and user agent. Referrer gives information about which web page had the link that was followed to this page and if the link was followed to the page, the User agent gives information about users browser and operating system. It can also tell if user was web spider or robot.

```
10.1.1.237 - - [16/Dec/2013:03:24:16 +0200] "GET /?
HTTP/1.0" 200 19039
"http://www.sonera.fi/tutustu+ja+osta/tarjoukset+ja+asiakas
edut/yllatyskalenteri" "Mozilla/5.0 (Windows NT 6.1; WOW64;
rv:25.0) Gecko/20100101 Firefox/25.0" "84.250.23.175"
```

Listing 5. Nginx web server log example from Sonera Joulukampanja.

Listing 6 is a web server log file row from Apache web server. It has mostly the same information that the previous web server log had, but there is one fundamental difference for data miner. This Apache is used in setup where Varnish cache server is used between Apache server that processes PHP code and web users browser. In this log file one does not find the IP address of the actual user, but the IP address can be seen from the cache server. IP being internal network IP is a good giveaway hint of that.

```
10.1.6.44 - - [14/Dec/2014:03:39:21 +0200] "GET
/uploads/media/default/0001/01/cedbf7b60c6ba7243aaa8a01c333
d214bb6ce414.png HTTP/1.1" 200 4528
"http://www.huoletonjoulu.fi/" "Mozilla/5.0 (Linux; Android
4.1.1; HTC One S Build/JRO03C) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/39.0.2171.59 Mobile Safari/537.36"
```

Listing 6. Apache web server log row example.

In various preprocessing phases useful data rows are selected and useless rows are left out. Which rows are important and which are not, is one of the most important and difficult choices of preprocessing. The same rows can be useful for some analysis and not so for others depending on what kind of goals are defined for the analysis.

In the case of web server logs in this thesis, it was decided that the interest would be in page loads and not other kinds of requests made to the server. Every page load is usually accompanied with several other requests of images, CSS files and other kind of files needed to present the page. Ii was decided that image, CSS and JavaScript files would be considered more as metadata of the page loads and not useful in trying to understand user behaviour. Thus, events related to these elements are omitted.

In the example case of Nasa web server logs presented earlier one could track page loads by looking for .html files among the requests and included only these log rows. This depends on the web service and the best method to separate page loads from other files has to be decided case by case.

Listing 7 is en example of the data in Nasa row with a page request on top and a file request.

```
ip-pdx6-54.teleport.com - - [01/Aug/1995:00:01:17 -0400]
"GET /history/history.html HTTP/1.0" 200 1602

uplherc.upl.com - - [01/Aug/1995:00:01:18 -0400] "GET
/history/apollo/apollo-17/apollo-17-patch-small.gif
HTTP/1.0" 200 14977
```

Listing 7. Two example rows of a log file with page and file requests. [8]

To be able to have to right data to get correct and useful information, there is importance in having business understanding of the web site or service that is being analysed. While decisions need to be made on what data to include, like mentioned earlier in this chapter, there can also be exceptions to these rules. For example in many cases it is useful to leave out image files, that would create extra noise to the analysis, but in some cases downloading of some specific image or PDF file can be important use of the service and could be considered an action to follow.

As executing association rule analysis is processor and memory intensive, there needs to be decisions to be made during preprocessing, whether some data can be left out to ease technical constraints on the computer equipment used. This is a complicated issue, because the more page loads from different parts of the web service can be analysed, the more chances of learning something useful there is. Especially as the num-

ber of columns in association rule mining data file increases, the need for the computer's RAM memory grows very fast.

Many of the phases in preprocessing brings up the question of whether it is better to leave out some data that could be useful, but that could also be just extra noise in the processing or bring an extra constraint that prevents from coming to useful conclusions. This paradox can be solved to some extent by trying our different variations of the preprocessing and considering the whole data mining as a very iterative process. Preprocessing and associations rule analysis have to be repeated several times over which different parameters and different data choices to see with combination of parameters and data selection brings the most useful results.

4.3 Executing Analysis

In this study two different programs were chosen to execute the association rule analysis. The selection was done considering which programs were commonly used, free to use and had good reputation. There were other programs that could have been as good, but it was necessary to choose a limited number of software and Weka and Rapidminer 5 were both commonly used, their usage well documented and they could be had for no cost. RapidMiner also has version 6, but it is proprietary, contrary to version 5, which is the reason the older version was used in this study.

The execution phase of the analysis is quite simple, the analyst gives the program the data it needs in a valid format, decides the parameters for the analysis and executes the analysis. What makes the difference in this phase is how preprocessing was done and which parameters are used.

4.3.1 Parameter Options in Association Rule Mining

There are several parameters that have an effect on the association rule mining. Below parameters that can be given to Weka are listed and explained to give an understanding on how association rule mining is affected by choice of parameters.

Parameters that can be given in Weka for Association rule mining using FP Growth algorithm are the following:

- Maximum number of items in the item set. This parameter defines the number of items that can be taken in to consideration while processing data.
- Required number of rules. This parameter defines how many rules will be shown after processing is finished.
- Metric to rank the rules generated. Options are confidence, lift, leverage and conviction. These terms are defined earlier in chapter 4.1.
- The minimum metric score of a rule, depending on the metric of the rule that was chosen to be the metric to rank the rules.
- Upper bound for minimum support as a fraction or number of instances. This can be either support defined as fraction, or integer defining number of instances.
- The lower bound for the minimum support. The lower bound for the minimum support as a fraction or number of instances. This can be either support defined as fraction, or integer defining number of instances.
- The delta by which the minimum support is decreased in each iteration as a fraction or number of instances.
- Find all rules that meet the lower bound on minimum support and the minimum metric constraint. Turning this mode on will disable the iterative support reduction procedure to find the specified number of rules.
- Only consider transactions that contain these items
- Only print rules that contain these items.
- Use OR instead of AND for must contain list(s). Use in conjunction with -transactions and/or –rules

[20]

The following parameters have the most important effect on the execution of analysis:

- Metric to rank the rules generated. This defines which metric is used to evaluate rules and it defines how the rules effectiveness is calculated. This is probably the most important of these parameters.
- The minimum metric score of a rule. This defines which rules are included and defines together with chosen metric which possible rules are included in the rules created.

- Upper bound and lower bound for minimum support. These two metrics define the upper and lower limits of support where rules are being mined. Weka starts looking rules that have minimum support of upper bound for minimum support. If specified amount of rules are found, Weka stops after this. If enough rules are not found, Weka starts lowering the minimum support until enough rules are found or lower bound for minimum support is met.

In this thesis the data mining process is executed with tools created for this thesis, with the exception of Weka. To understand the capabilities and configuration of the association rule mining with Weka thorough understanding of Weka options is needed.

4.3.2 Results of Analysis

The association rule mining on log files of from three different sources was conducted. These logs were:

1. Nasa web site server access logs from August 1995.

2. Liandersson.fi site of an electoral candidate web server access logs from March 2015.

3. Sonera's Joulukampanja (Finnish teleoperators Christmas promotion) web server access logs from December 2013.

Even though data from all these sources were quite different in structure, they were formatted to a similar format in the preprocessing phase. An important difference was also that depending on the web site structure, the data attribute number had variety. That had an impact on the processing of the data.

Listing 8 is an example of association rule mining of web server logs of liandersson.fi from March 2015. March web server logs included 605 235 rows of data, that was parsed in preprocessing to 7150 sessions with 2081 attributes. These attributes included different page and feed loads. The algorithm that was used is FP Growth. Processing took 33 seconds on a 2011 Mac Book Pro laptop with 2,4 GHz Intel Core i5 processor. Rules are ordered by confidence.

Premises	Conclusion	Support	Confidence	Lift	Conviction
GET /category/blogi/ HTTP/1.1	GET /liandersson/ HTTP/1.1	0,06	0,49	1,47	1,30
GET /kysy-lilta/ HTTP/1.1	GET / HTTP/1.1, GET /liandersson/ HTTP/1.1	0,03	0,47	2,04	1,46
GET /kampanjan-tukijat/ HTTP/1.1	GET /liandersson/ HTTP/1.1	0,05	0,47	1,42	1,26
GET / HTTP/1.1, GET /kampanjan-tukijat/ HTTP/1.1	GET /liandersson/ HTTP/1.1	0,03	0,47	1,41	1,26
GET /tapahtumakalenteri/ HTTP/1.1	GET /liandersson/ HTTP/1.1	0,04	0,46	1,37	1,23
GET / HTTP/1.1, GET /tapahtumakalenteri/ HTTP/1.1	GET /liandersson/ HTTP/1.1	0,03	0,43	1,30	1,18
GET / HTTP/1.1	GET /liandersson/ HTTP/1.1	0,23	0,41	1,22	1,12
GET /category/blogi/ HTTP/1.1	GET / HTTP/1.1, GET /liandersson/ HTTP/1.1	0,05	0,35	1,51	1,18
GET /tapahtumakalenteri/ HTTP/1.1	GET / HTTP/1.1, GET /liandersson/ HTTP/1.1	0,03	0,32	1,38	1,13
GET /kampanjan-tukijat/ HTTP/1.1	GET / HTTP/1.1, GET /liandersson/ HTTP/1.1	0,03	0,31	1,34	1,12

Listing 8. Association rules from liandersson.fi web server logs.

Listing 9 is an example of association rule mining of web server logs of Nasa web server logs. This log example consists of 5000 rows of Nasa web server logs from Nasa web site from 1995 August. Processing took 33 seconds on a 2011 Mac Book Pro laptop with 2,4 GHz Intel Core i5 processor. Rules are ordered by confidence.

Premises	Conclusion	Support	Confi-dence	Lift	Conviction
GET /history/apollo/apollo.html HTTP/1.0	GET /facilities/lc39a.html HTTP/1.0	0,03	0,31	3,59	1,32
GET / HTTP/1.0, GET /shuttle/missions/missions.html HTTP/1.0	GET /ksc.html HTTP/1.0	0,02	0,3	1,38	1,12
GET /shuttle/missions/missions.html HTTP/1.0, GET /shuttle/missions/sts-70/mission-sts-70.html HTTP/1.0	GET /shuttle/countdown/ HTTP/1.0	0,02	0,3	1,43	1,13
GET /shuttle/missions/missions.html HTTP/1.0, GET /shuttle/missions/sts-70/mission-sts-70.html HTTP/1.0	GET /shuttle/missions/sts-71/mission-sts-71.html HTTP/1.0	0,02	0,3	5,14	1,35

Listing 9. Association rules from Nasa web server logs.

Listing 10 is the resulting rule set from association rule analysis using 100 000 rows of data from Sonera Joulu Kampanja 2013. The time the processing took was 0.128 seconds with 2011 Mac Book Pro laptop with 2.4 GHz Intel Core i5 processor.

```
1.    [GET    /?intcmp=MOB-Yllatyskalenteri-omatsivut-etusivu-lightbox
HTTP/1.0=false]:  228  ==>  [GET  /?someid=16  HTTP/1.0=false]:  228
<conf:(1)> lift:(1) lev:(0) conv:(0.9)

2.  [GET  /?someid=1  HTTP/1.0=false]:  215  ==>  [GET  /?someid=16
HTTP/1.0=false]: 215   <conf:(1)> lift:(1) lev:(0) conv:(0.85)

 3.   [GET   /?intcmp=MOB-Yllatyskalenteri-omatsivut-etusivu-container
HTTP/1.0=false]:  132  ==>  [GET  /?someid=16  HTTP/1.0=false]:  132
<conf:(1)> lift:(1) lev:(0) conv:(0.52)

 4. [GET /? HTTP/1.0=false]: 83 ==> [GET /?someid=16 HTTP/1.0=false]:
83   <conf:(1)> lift:(1) lev:(0) conv:(0.33)

 5.    [GET    /?intcmp=MOB-Yllatyskalenteri-omatsivut-etusivu-lightbox
HTTP/1.0=false,  GET  /?someid=1  HTTP/1.0=false]:  195  ==>  [GET
/?someid=16  HTTP/1.0=false]:  195      <conf:(1)>  lift:(1)  lev:(0)
conv:(0.77)

 6.    [GET    /?intcmp=MOB-Yllatyskalenteri-omatsivut-etusivu-lightbox
HTTP/1.0=false,  GET  /?intcmp=MOB-Yllatyskalenteri-omatsivut-etusivu-
container HTTP/1.0=false]: 118 ==> [GET /?someid=16 HTTP/1.0=false]:
118   <conf:(1)> lift:(1) lev:(0) conv:(0.47)

 7.    [GET    /?intcmp=MOB-Yllatyskalenteri-omatsivut-etusivu-lightbox
HTTP/1.0=false,  GET  /?  HTTP/1.0=false]:  69  ==>  [GET  /?someid=16
HTTP/1.0=false]: 69   <conf:(1)> lift:(1) lev:(0) conv:(0.27)

 8. [GET /?someid=1 HTTP/1.0=false, GET /?intcmp=MOB-Yllatyskalenteri-
omatsivut-etusivu-container HTTP/1.0=false]: 104 ==> [GET /?someid=16
HTTP/1.0=false]: 104   <conf:(1)> lift:(1) lev:(0) conv:(0.41)

 9. [GET /?someid=1 HTTP/1.0=false, GET /? HTTP/1.0=false]: 58 ==>
[GET /?someid=16 HTTP/1.0=false]: 58    <conf:(1)> lift:(1) lev:(0)
conv:(0.23)

10.   [GET   /?intcmp=MOB-Yllatyskalenteri-omatsivut-etusivu-container
HTTP/1.0=false,  GET  /?  HTTP/1.0=false]:  23  ==>  [GET  /?someid=16
HTTP/1.0=false]: 23   <conf:(1)> lift:(1) lev:(0) conv:(0.09)
```

Listing 10. Association rules from Sonera Joulukampanja web server logs.

Similarly to data mining in general, association rules are not necessarily very useful without the interpretation of results by an analyst who has insight to the data that is the basis of rules and some insight in to what could be useful in the new information created by the data mining process.

The premises column shows an event or number of events that are statistically linked to the event or events that are mentioned as conclusion. Numbers that are after these are the statistical measures that show the statistical relevance of rule in question.

As an example in Listing 11 there is a rule from linandersson.fi. It has one event as premise and two events as conclusion. So this rule means that users that visit the page that lists the campaign supporters' page are statistically likely to visit the front page and the page that describes who Li Andersson is. The numbers after premise and conclusion define how statistically likely the rule is.

GET /kampanjan-tukijat/ HTTP/1.1	GET / HTTP/1.1, GET /li-andersson/ HTTP/1.1	0,03	0,31	1,34	1,12

Listing 11. Example association rule from liandersson.fi web server logs.

As a second example in Listing 12 there is a rule from Nasa.gov from 1995. This is similar to the previous rule in structure. The rule shows that that users visiting the front page and the shuttle mission's page are likely to visit the Kennedy Space Station web page as well.

GET / HTTP/1.0, GET /shuttle/missions/missions.html HTTP/1.0	GET /ksc.html HTTP/1.0	0,02	0,30	1,38	1,12

Listing 12. Example association rule from Nasa web server logs. [8]

Listing 13 is an example rule that is mined from Sonera web server logs. This rule is in a different format than the other two example rules, but follows the same rules of the first listing premises, then conclusions and finally measures to show how reliable these rules are. This particular rule has confidence of 1, which means that the premise and conclusion follow each other every time. This makes the rule very reliable, but probably means that these events are part of the same page load and so this rule does not really

tell anything about the user behaviour, which is usually the information of interest in association rule mining.

```
[GET /?intcmp=MOB-Yllatyskalenteri-omatsivut-etusivu-
lightbox HTTP/1.0=false]: 228 ==> [GET /?someid=16
HTTP/1.0=false]: 228 <conf:(1)> lift:(1) lev:(0)
conv:(0.9)
```

Listing 13. Example association rule from Sonera Joulukampanja web server logs.

In the example cases listed above different parameters were used. As described in above chapters about data mining, it is usually an iterative process where mining needs to be executed over and over trying out different parameters and data sets or different preprocessing methods of data sets. This iterative approach is needed to find the most interesting results and try out different methods.

In both of the example rule sets of liandersson.fi and nasa.gov the same parameters of minimum confidence 0.3, and minimum support 0.1 were used. The maximum number of items was not limited and the rules were organized by ranking the rules by highest confidence.

Sonera case was ranked by using the highest number of confidence, to order the rules. Minimum confidence was defined as 0.3 and maximum number of items was not limited.

One of the main challenges there was in the executing of the analysis was that for example the Nasa data set can have quite large number of different events that can happen and when the matrix of events and sessions grows large in the sense of having thousands of columns the processing becomes very memory constraining. This was the main reason the Nasa data set that could be analysed as one batch, was considerably smaller than data sets of liandersson.fi or Sonera Joulukalenteri.

4.4 Postprocessing

Postprocessing is an important phase of a data mining process, where the acquired information is processed to be more human understandable. It often includes removing

noise and pruning rules that are not considered important. It can also consists of visualising the data or writing explanations to summarize the meaning or relevance of the information created.

Postprocessing can be divided in to four steps: knowledge filtering, interpretation and explanation, evaluation and knowledge integration. [21,1-2]

The knowledge filtering phase consists of trying to remove data or rules that are not important. This often means either to see which rules are not statistically strong or are not important for some reason. [14,3-5]

In the Sonera data there were rules that had confidence of 1, which meant that these rules were valid every time. Looking closer to what the premise and conclusion were, it could be determined that both requests were part of single page load, so this rule did not give any relevant information about user behaviour and should be ignored. Rules like this create noise that can make it more difficult to find the rules that enable acquiring useful new information. Because of this there needs to be postprocessing to clean the rule sets from non-relevant data.

The interpretation and explanation phase consists of turning the acquired information in to more usable format. If data mining has been done to gather information for an end user, this phase can consist of writing explanations, visualizing data or summarizing. This phase can also include checking if information now acquired conflicts with previously acquired information. [21,1-2]

The evaluation phase includes evaluating the rules or other kind on information acquired. The validity of the created models should be evaluated and it should be inspected whether this information was answering the questions that were the hypotheses for the whole data mining experiment. Widely accepted criteria exist for this phase, such as classification accuracy, comprehensibility and computational complexity. [21,1-2]

The last phase is knowledge integration. This phase consists of integrating the acquired and evaluated information with other information acquired with different methods or previously acquired information. [21,1-2]

In this thesis, the technical tool to execute the different phases of association rule analysis was limited to knowledge filtering. The postprocessing script takes into consideration the URLs of interest. An analyst can define which page loads are interesting and a postprocessor filters all the rules that do not have the URLs of interest in the conclusion part of the rule.

This is important in finding rules of interest among possibly very numerous rules created by the analysis.

4.5 Planning Data to Log

The purpose of the study was to research possibilities to data mine knowledge of data from web servers logs. These logs contain data of web server behaviour, but to be able to mine useful information of this data, the data has to be coherent and include as much information as possible from the users and their behaviour. Any missing pieces of information can render much of the data useless.

This thesis described earlier how different log files consist of several bits and pieces of data and how this data can be mined to create new information. There is great variance in how useful this information is or can be, depending on how the data logging is done.

As an example, the Nasa and Liandersson.fi server logs could be mined for usage patterns, while Sonera log files contained very little information about user behaviour. That example points out how different the results can be, even if the data was logged and it contained most of the useful information.

Some server logs evaluated as possible example data for this thesis contained so little information that the date could not be used as an example. For example some web servers used cache proxy in a way that the web server logged only the cache server IP address as the only information of the users. This made defining sessions based on the available data impossible and that leads to determining of any usage patterns impossible.

Logging a web page request can be used to make observations of user behaviour, but it also has its limits. More complicated the web services become and less dependant on page loads compared to JavaScript actions, the more difficult it is to have a complete view of all usage behaviour only based on access logs.

Access logs can log every JavaScript initiated request, so in some cases using JavaScript based requests instead of page loads, does not make a difference to the usability of the server logs. On the other hand, in cases such as Sonera Joulukampanja, in one page load the browser loads several different pages of content and JavaScript actions change what is shown to the user. This creates a situation where web server logs only show one page load, but very different viewing sessions of these different content items can occur in the browser.

The logging of usage behaviour has to be moved more to the level of application, because current servers cannot get information of many possible user actions that do not cause a web page request to the server. Actions such as hovering or clicking can be handled by JavaScript, showing content to user, without new page loads or any interaction with the server. This change in the way web services often work has transferred the responsibility of logging user actions from the server to the application.

This is currently often done using external services using client run JavaScript to give external service information of user behaviour to track service usage. These services and their usage is out of the scope of the present study.

The most important thing to do is to make sure user IP, possible user id, user agent, timestamp, request and the return code are logged and show the actual users information. Also the application developers have to define all actions that are of interest and that do not trigger page load by themselves otherwise this page load does not make the server log any important data related to the event by itself.

5 Technical Execution of Analysis

The previous chapter described how association rule analysis can be used to generate information about web server usage using server logs as data. Firstly, this chapter de-

scribes tools being used to execute data mining. Secondly, this chapter shows how these methods are used to data mine an example data set. This data set is used to prove that these methods can produce results using real world data and technologies described in the study. The third part of this chapter then shows how the same tools are used to data mine data from a web service currently in a production phase and what is needed to implement the mining.

5.1 Scripts Used

To preprocess data for association rule analysis a Python script that turns server access log files in to CSV file that can be imported into Weka and analysed was written. The main tasks of this script are to read a log to memory, to collect data that we need for processing, to create sessions from several queries and finally write to the new CSV file with columns representing the events found in the log and rows representing sessions.

Listing 14 is a shell script that takes the log file as parameter and uses preprocessing script to preprocess data, calls Weka to process data and after processing calls for postprocessing script to look for association rules that have conclusions defined to be of interest.

```
#! /bin/bash
rm output.arff
rm rules-output.txt
/usr/bin/python preprocessor-only-html-arff.py "$1"
java          -cp          /Applications/weka-3-6-11-oracle-
jvm.app/Contents/Java/weka.jar  weka.associations.FPGrowth  -
C 0.30 -I 3 -T 3 -N 100 -t output.arff > rules-output.txt
/usr/bin/python postprocessor.py rules-output.txt
```

Listing 14. Bash script to process logs.

Listing 15 is the preprocessing script used to preprocess logs before association rule analysis.

```
#!/usr/bin/python

import csv
import time
from sys import argv
```

```
contentFile = argv
outputFile = open('output.ARFF', "a+")
outputFile.write('@RELATION ACCESS_LOG\n')
dictionaryOfHosts = {}
URLList = set()
substring = '.html'

with open(contentFile[1]) as csvfile:
    reader = csv.DictReader(csvfile, fieldnames = ("Host",
"Identd", "Userid",  "Date Time", "Timezone", "URL", "Sta-
tus code", "Size", ), delimiter=' ')

    for row in reader:
        getParameter = row['URL'].split(" ")[1]
        if   substring   in   row['URL']   or      getParame-
ter.endswith('/'):
            subDict = {}
            subDict['Host'] = row['Host']
            subDict['URL'] = row['URL']
            subDict['Date Time'] = row['Date Time']
            if subDict['Host'] in dictionaryOfHosts.keys():
                dictionaryOf-
Hosts[subDict['Host']].append(subDict)
            else:
                dictionaryOfHosts[subDict['Host']] = []
                dictionaryOf-
Hosts[subDict['Host']].append(subDict)

            UrlStringVariable = row['URL']
            UrlStringVariable       =       UrlStringVaria-
ble.replace(",","")
            URLList.add(UrlStringVariable)


for uniqueURL in URLList:
    outputFile.write('@ATTRIBUTE "' + uniqueURL + '" {true,
false}\n')

outputFile.write('\n')
outputFile.write('@DATA\n')
sessions = {}
sessionCounter = 0
currentHost = ""

for contentList in dictionaryOfHosts.values():
    previousTime = 0

    for logRow in contentList:
        print logRow['Date Time']
        currentTime                                          =
time.mktime(time.strptime(logRow['Date Time'],
         "[%d/%b/%Y:%H:%M:%S"))
        if currentTime - previousTime < 1800:
            sessions[sessionCounter].append(logRow)
        else:
            sessionCounter += 1
            sessions[sessionCounter] = []
            sessions[sessionCounter].append(logRow)

        previousTime = currentTime
```

```
for sessionName, session in sessions.items():
    if len(session) == 1:
        continue
    print str(sessionName) + ' ' + str(len(session))
    for column in URLList:
        match = 0
        for row in session:
            if row['URL'] == column:
                match = 1
        if match == 1:
            outputFile.write('true,')
        else:
            outputFile.write('false,')
    outputFile.write('\n')

outputFile.close()
```

Listing 15. Preprocessing script.

Listing 16 is a postprocessing script to find rules that have conclusions that have been defined as interesting.

```
#!/usr/bin/python

with open('rules-of-intrest.txt') as f:
    intrestList = f.read().splitlines()

rulesOutput = open('rules-output.txt', 'rb')
for line in rulesOutput:
           lineAfterSplit = line.split('=>', 1)
           if len(lineAfterSplit) == 2:
                     for interest in intrestList:
                               if  interest  in  lineAfter-
Split[1]:
                                         print line
```

Listing 16. Postprocessing script.

The file rules-of-intrest.txt mentioned in the code above consist of HTTP requests that interest the research if they are found as rule conclusions. Listing 17 is an example content of such file with two request to search from conclusions is below:

```
GET
/file/1688/LonghaulflightsfromHEL_timetable_03052013.pdf.ht
ml HTTP/1.1=false
GET /en/cargo/guidelines.html HTTP/1.1=false
```

Listing 17. Example content of rules-of-interest.txt.

Simplified example above shows potential structure of a file that consists of rules that are interesting in the postprocessing phase.

5.2 Data Visualization

Association rule mining can be an effective way to mine information on user behaviour. Often this can create such large quantities of association rules, that it is time and resource consuming to study through all possible rules to find to possibly interesting rules. There are several ways of finding interesting rules, also discussed in Chapter XX about postprocessing association rules. Other way to help make association rules more accessible for finding useful rules is visualization of association rule data sets. [22, 1-3]

Visualization is used as a way to have abstract and concrete ideas more easily understandable in several different field. These fields consist of education, engineering and science. Visualization can be divided in to two different phases: exploration phase and presentation phase. [22,1-3]

In the exploration phase visualization is used to enable finding patterns that could be more difficult to see in data is presented in other formats. This includes a lot of manual work in filtering, zooming and rearranging data. This is used to find patterns in the data that are considered useful. [22,1-3]

In the presentation phase of visualization of the data, the data is visualized in way that is planned to help larger audience understand the data. In this phase the most important function is to emphasize for the audience the key figures of the data. [22,1-3]

Figure 1 is an example of graph-based visualization using association rules generated using liandersson.fi server logs with Rapidminer.

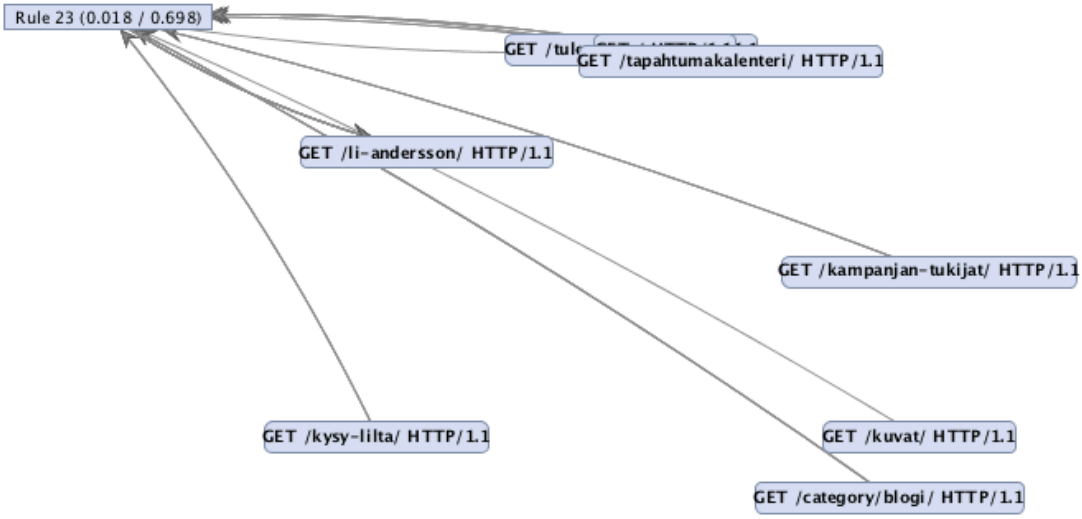

Figure 1. Visualization shows how certain requests have statistical likelihood of occurring with other requests. In this case several different request are likely to occur in the same session with a request to load page about Li Andersson.

Commonly used charts to visualize association rules are different plot charts such as scatter plot and two key plot, graph based visualizations and matrix-based visualizations. [22,3-20]

5.3 Geolocation

Geolocation means locating users geographically, i.e. to know where they are on a map. In the case of studying user behaviour on a web service this information can be useful. The more analysts have data on the users the more chances there are to find useful patterns of user behaviour. For example, the users can be grouped by their city or country and when grouping users by these attributes useful patterns can emerge.

There are different methods in finding out where web service user is based geographically. The most common ways are either asking directly from the web browser via HTML5 Geolocation API or analysing the IP address that the user uses to access the service. All browsers that conform to HTML5 specification offer this functionality of being able to give web server details of their location, if the server requests that and the user gives the browser permission for the request. [23,306]

Mobile devices have most advanced capabilities in giving their location information. This is based on possibilities of GPS, cell-tower based location and by analysing WI-FI hotspots in vicinity. GPS is the most accurate and can locate a device to a area consisting of only less than 10 meters, but needs to have access to information coming from GPS satellites and for this reason it mostly works only outside. The other two methods can work also in doors but are less accurate. [23,307]

There are several IP address based methods to locate a specific user. Most of these are based on services keeping manually maintained databases of geolocation IP address sets. There are several methods to map an IP address by 'whois' information, administrative locations and host names, but none of these are very accurate. [24,71-73]

As mentioned in the beginning of this chapter, the geolocation can give an important new attribute to classify and group data, or as is the case in this study, to group users by their locations. Due to time constraints this study does not use location as an attribute in the association rule analysis, but locations could be added as new attributes to find what kind of web usage patterns that would show.

Location based services are most commonly different web services that use location either to localize the content shown to user or add targeting. Localized content can be such as local weather or local news or similar localized content. Localization can also be used to restrict content to be available only in certain area. Localized add targeting is using geolocation as a way to find better targeting to user based on their location. [25,1220]

As an example of how geolocation has been used in scientific research is a study done by Stanford University and Microsoft Research that studied what kind of recipes where read in different areas and can that be used to study nutrition and food consumption in different areas, by data mining server logs. The study was able to learn about different dietary habits in different areas and find agreement between sodium use in the recipes and rates of admission to a hospital in certain area. [26,1399-1400]

There are also privacy concerns related to using geolocation data of users. The four most common threats that exist in web service technology used in HTML5 standard of geolocation information are the following [23,306]:

1) There is no way to give only partial information. Even if the service wants the country of the users, it always gets rights for exact location or none.
2) There is no way to give only current location, granting permission to location gives on going permissions for full tracking.
3) There is no way to give permissions for one page or purpose, giving permission to a page gives it to the whole domain.
4) There is no process of service having to confirm permission after making changes to their service. Domain holds the permission even if the underlying service is refactored or changed completely.

Concerns mentioned above need to be addressed for the geolocation technology to be safe and mature technology that is being used in everyday applications.

6 Scaling Data Mining to Larger Sets

This Chapter describes different problems and areas of concern that are related to scaling data science applications to large sets of data.

6.1 Big Data

While data mining can be used in a relatively small scale with running the data mining software in a single laptop, many real world uses need to use very large data sets or work in real time using large amounts of memory. These scalability issues can cause computing power requirements to exceed resources of a single server to be sufficient for data mining. In the field of web server logs the actual logs can be distributed on multitude of servers and any single web server might not have the processing power or memory to spare from its other functions, to do association rule analysis to predict user behaviour. These factors makes scaling to large sets or real time results an issue that needs Big Data solutions to be used in production environment.

Change brought by big data is clearly visible for example in retailing books. Retailers in could see which books were popular and which not. In some cases they might have programs to make most loyal customers get some kind of membership card that could be used to link certain transactions with certain individuals. In the age of selling books in online stores, mechanisms of tracking users and their interests changed quite a bit. Online bookstores can gather quantitative data on everything online shoppers look at, how they navigate, which campaigns, reviews or advertisements had impact on their navigation. Then all individual users could be grouped in the groups with similar interests and soon algorithms could tell with high accuracy which books would interest whom. This kind of data was never accessible in such detail before online retailing and Big Data. *[27,4]*

The story of Amazon and similar online retailers that were born on the era of digitalization have had such a big advantage from their capability to use data as a way to drive their sales, that it is almost not visible. Companies like Amazon have had capabilities that were not imaginable some years ago. This power of big data is not limited to being used only in online retail, but can be used to develop organizations from any field. Measurements of very different operations can be made faster and in much more precisely than before, which in turn will lead to better judged decisions. *[27,4]*

Spreading of tools and methods developed for managing big data is changing the way experience and leadership is perceived. It is creating a transition in how management is seen today and will lead different corporate management and leadership were data and it's mining are some of the most important tools of management. [27,4]

During 2000s big amounts of data were a technical problem that had to be solved. Greater amounts of data put serious constraints on CPU and storage capacity of IT infrastructure. Terabytes of data that were gathered were a serious scalability problem for years. Now after CPU and storage have become cheaper and have larger capacity, this data has turned from burden to large wealth. [28,4]

Not only is big data considered big in the sense of large data volume's but also by the speed data is being gathered and processed, and by the great variation of data coming from very different directions. This diversity of data together with the technical means and statistical analysis of data, is together what can be considered big data. [28,4]

Big data can be defined by three factors that all begin with V.

Volume is usually considered the defining attribute of big data, as the name implies, big data needs data storage's big in volume. In interviews about big data, many associate big data with data volumes of terabytes or even petabytes. Big data quantities can also be measured in other metrics than volumes of bytes, such as database records, time or files. [27,6]

Each day after 2012, approximately 75 Exabyte's of data were created in a month. That number is seen growing rapidly. In 2014 more data was moving in Internet every second, than had existed in Internet in 1994. This growth of data volume has created very different playing field for data mining in recent years. [28,4]

Velocity in many aspects of big data, the velocity of data can be bigger actor than the volume of data. Real time information gathering and processing can be of big advantage to companies. [285]

Velocity in big data can be thought of as how rapidly data is generated and how fast it can be delivered and processed. There are more and different kind of sensors that generate more and more real time or almost real time data that can be gathered and processed. For example there is lot of data gathered in real time from mobile phones, robotic manufacturing machines, thermometers, microphones listening for movement in secured area or video footage processed for faces that want to be found from crowds. This growing data being generated with higher frequency is important aspect of big data. Especially real time processing and analytics of such high frequency is challenge and characteristic of big data. [27,7]

Variation as a defining characteristic of big data means that while volume and velocity play important role in defining what is big data, also the large variation in data brought together for analysis is has a major role in defining big data. Data for big data analysis can be gathered from online sources, such as clickstreams, log files and social media material. Different actors have gathered this kind of data for some years. Even if this kind of data has been gathered, most of organizations have not been able to make full use of it. Big change in recent years has been that now more actors see the value of analysing all the data they have been gathering. [27,7]

Big data is not new, but understanding of how all the possibilities what can be done with big data is the bigger innovation. Recently the use of data with different levels of

structure has been a new innovation. Some data is also difficult to categorize, as it comes from a large variety of sources, such as audio and video. Combining these different varieties of data is another level of big in big data. [27,7]

6.2 Current Challenges and Future Possibilities

There is some controversy over the issue of big data, which is related to it being a new concept that is bringing change to ways business and governmental organizations operate. In the next paragraphs some of the topics of controversy are listed. [29,3]

There is discussion if big data analytics is that different from just data analytics, data has just become larger and will continue to do so. [29,3]

Big data discourse has been accused of trying to sell Hadoop based computer systems. Hadoop is a distributed file system framework used to high level scaling of distributed computing. Hadoop might not suit every need perfectly and while Hadoop is the most common base for big data system, there is competition that could suit better for some needs. From example for small to medium sized organizations. [29,3]

When analysing data in real time, the volume of the data might not have an important role, but the frequency how fast it can be analysed. [29, 3]

There are also some claims that accuracy of big data analysis could be over hyped. As there are more and more moving factors, chances of misleading correlations also grow larger. [4, 3] For example S&P 500 stock index had correlation with butter production in Bangladesh and other correlations that most likely were big coincidence. [30,3]

Even if the bigger data gets, more possibilities there are to find correlations, but also risks of having noise that misdirects or covers correlations also grows. As an example there is often an assumption that twitter users represent population globally, while this often is not true. [29,3]

There are also ethical questions related to big data. Questions such as, is it ethical to mine data about people without their knowledge about it [48,3] or publish data where

people's privacy might be violated. Often data with very much detail is considered more valuable, but also more detail gives more changes to intrude privacy. Therefore there are contradictions between effective data and privacy in the field of big data. [31,33]

One example of this contradiction is out of source data mining. It means an organization sends its data to other to be subjected to data mining. Often the customer of out of source data mining might not want the data miner to get more results then was planned by the customer or found suitable in terms of privacy. This can lead to anonymizing data to an extent that not only is privacy better guarded, but also potential information in the data lost to the extent that data mining will be futile. With less detail in the data, many patterns get lost. [31,33-34]

New digital divisions are also created by development on the field of big data. The capability to analyse big data grows gaps between organizations that can and cannot mine big data effectively. There is also big differences among organizations on level of access to big data that can define their capability to participate in race to analyse data on different fields. [29,3]

The field of web server logs is equal among organizations so that most organizations have the potential to log data from their servers and make use of it, if organisations believe it is important to them.

6.3 Technologies Commonly Used in Big Data

While big data is a concept to describe multitude of approaches, techniques and methods, there are some tools that have become almost synonymous to big data. One of the most well known is Apache Hadoop. Even though Hadoop has a special place in big data, there is an increasing number of software tools designed specifically to deal with processing big data. In this chapter a brief introduction to some of these tools is provided to give the reader a better understanding of the tools available in the field of big data.

Hadoop is more a set of tools than one single tool. The most common minimal setup consists of the following:

- **Hadoop Common**: The basic tools that Hadoop needs for its other modules to work together.
- **Hadoop Distributed File System (HDFS™)**: A distributed file system that enables to fast and reliable access for the different modules to the data that is processed.
- **Hadoop YARN**: A framework that takes care of the job scheduling and cluster resource management.
- **Hadoop MapReduce**: A YARN-based software that enables parallel processing of large data sets. [32]

The basic modules of Hadoop enable the framework to function, most of the actual work is done by different Hadoop projects such as the Ambari tool for managing Hadoop clusters, Pig data-flow language and parallel computation framework, Hive provides the data warehouse infrastructure, Mahout is data mining and machine learning library and Zookeeper provides coordination service for distributed applications. [32]

Analysis done in this thesis could have been done with a Hadoop and Mahout combination, but it was unnecessary due to the relatively small amount of data and no need for real time analysis capability.

6.4 Comparing Hadoop Based Technologies to Their Competition

In this chapter there is an evaluation on how Hadoop based tools differ from other distributed computing systems that have been developed for similar kinds of use cases. The Chapter points out differences in these computing frameworks and shows the strengths and weakness of different distributed computing environments.

6.4.1 HPC and Grid Computing Tools

HPC (High Performance Computing) and Grid Computing Tools have similarities with Hadoop in the sense that they distribute computing load to network of clusters that have shared file system. The difference is that SAN (Storage Area Network) hosts these shared file systems. They are designed for work that is processing intensive. This

differs from big data in the sense that big data often has network speeds as a bottleneck. MapReduce makes the big difference, because with if fast data access to all nodes is common in Hadoop setups, but is more limited with environments using SAN. [33]

HPC and Grid Computing Tools rely on APIs that give great control to the user. This forces the user to control the tools that take care of the data flow. On the contrary, MapReduce works on a higher level and MapReduce user does not need to interfere with the data flow. DataReduce is also more effective in handling coordination of different processing jobs, because it works on shared-nothing architecture. All tasks processed by MapReduce are completely independent and failed tasks are automatically noticed and rescheduled to healthy machines. [33]

6.4.2 Volunteer Computing Technique

In Volunteer Computing Technique the work is divided in so called work units. Work units are then sent to machines across the globe for analysing. When a distant computer has finished processing the analyses, the outcome is delivered back to the networks controller and the client is sent a new task. The server guarantees accuracy by assigning the same task to three different client's and accepts the outcome if two clients deliver the same results. In theory this concept of distributed computing looks similar to MapReduce. The main difference is that in Volunteer Computing Technique work units are processing intensive, and cannot have access to large data sets. For this to be effective it must be possible to process the work unit in a shorter period of time than is required to send the work to a processing unit. MapReduce is designed to run longer jobs with dedicated hardware with access to the file system of the data on a very fast connection. [33]

6.4.3 Relational Database Management System

Relational databases have been designed to work with data that can be measured in Gigabytes at most. MapReduce can be configured to easily process datasets with petabytes. MapReduce differs structurally on how the database is updating and having

access to data, which makes MapReduce scale to levels several magnitudes higher. [33]

7 Action Points to Develop Web Service Logging

The main target of this thesis is to provide action points on how to develop data logging in web services, to be able to extract more possibly useful information about user behaviour. Action points are based on the studying of web server logs and their analysis methods described earlier in the paper.

These action points are listed below with short explanations of why they are relevant and how they should be implemented.

1 Data to be saved. The single most important thing in web server log planning is to take into account what are the data columns that are saved. Earlier in this study different log formats and different collections of data that single log data row consists of were mentioned. There is variation in what can be interesting depending on the situation, but some conclusions can be made with the knowledge gathered in this study.

Each log row should have at least the following information:

- Data about the user, such as username, IP address and user agent information. Username is often not known to the web server, if the application layer handles all authentication information.
- Data in about the requested resource. This is usually the request that includes information about the type of request, the requested file and protocol of the request. The requested information is usually the single most valuable information.
- The timestamp of the request. It is important to know when the request was made and also to understand if it was part of specific session. User name and other user specific information and possible cookie information can help to determine which requests are part of the same session.

2 The recognition of users. Depending on the web service the users can be all logged in and the real world identity can be known to web service administration or all of the users can be anonymous to the web service. Nevertheless the understanding of which user does which request is the basis of having possibility to find behaviour patterns of users.

With anonymous users the usage patterns can be minimal and consist of IP address and user agent information together with the time of requests and their frequency, like was the case in this study. If users log in or their cookies can be logged and tracked through several sessions, more material exists to look for interesting patterns of user behaviour.

3 Understanding the infrastructure and its possible limitations. As an example of the Sonera case in this study clearly shows, not taking the infrastructure in to account in logs can render them useless. In the Sonera Joulukampanja example case the cache server blurs the number of requests from the web server and also the web server logs the IP address of the cache server as source of the requests.

In this example one can see that such web server logs are of very limited use, compared to what could have been produced by the cache server. Cache server default option was not to log every event, so default configuration provided very non optimal data to work with. In some cases there would have been a need to get user information or some other data from the web server and other data from the cache server and combine these during the preprocessing of the log files.

4 Logging data in the application level. The Sonera example also showed another important shortcoming of web server log considering usability of these logs in understanding user behaviour. Most of the data was loaded in the same page load as one html document and several JavaScript, CSS and image files. User interaction in the page then revealed different parts if this information as traditionally would have clicking links and having new page loads.

Because of this way of loading lot of information in one load and then showing it according to click that did not need page loads, the web server logs became very inaccurate measurement of user behaviour. This shortcoming can only be overcome by either avoiding this kind of paradigm of page structure or by programming the logic of auditing user behaviour in to the logic of the web application.

In this case extra requests to get the html for every click would have slowed the service and therefor been burden to the user experience, so only way to have the user behaviour logged without compromising service level is to use JavaScript in auditing the behaviour.

This is often done with using external services, as was also in this case, but the amount of data to get and analyse afterwards is limited to reports of user behaviour, not to the level of actual data rows, so to be able to use association rule analysis with services having similar structures as the Sonera Joulukampanja web service, the application level needs to programmed to audit user behaviours that would not be logged otherwise.

8 Discussion

This chapter provides an evaluation of how useful information this thesis was able to produce and if this information can be used in practice in future and how useful it can be. This thesis had two goals. The first was to create a software pipeline that can create usable association rules from web server log data. The second was to create action points that can be useful in planning web server data logging.

8.1 Usefulness of Data Produced

In the cases of linadersson.fi and Nasa web site the rules produced by association rule analysis clearly show relations between different paged loaded by users. So it can be considered successful in the sense of being able to produce information of user behaviour.

The study also shows the structure of a web service and how events are logged create limits of analysing the user behaviour. Specifically in case of Sonera Joulukampanja the structure where most user interactions are handled by JavaScript and do not necessarily produce requests to the web server causes issues to consider. This structure renders many of the aspects required for observation based on web server logs useless in trying to understand user behaviour.

Considering these two different outcomes, it can be considered that the methods selected were useful in some cases, but also clearly show that as web services have increasingly complex infrastructures the data logging needs to be developed to match the new infrastructure, to be as useful as possible in data mining.

8.2 Usefulness of Produced Methods and Action Points

The purpose of this study was to produce pre- and postprocessing tools required for creating association rules from web server logs and analyse the success of this production to create action points on developing web server log creation. This was accomplished in the sense the method to produce rules was created successfully and

using the method on different log files proved that it worked on some, but also showed weaknesses in how the data was logged in more complex environments.

These weaknesses were the basis to write the action points for data logging in web service environments, which are the product and value of this study. The number of different web service infrastructures and problems that might come up with different setups was limited, so this list of action points cannot be perfectly sufficient for every possible situation, but gives guidelines in to what to take into consideration in terms of data logging when developing a web service.

To scientifically verify the results that can be achieved with action points listen in this study, new web server logging configuration should be made for the services in question. Also the new data is would need to be produced by the web servers, that could be reviewed to verify that these actions points produce measurable results. Unfortunately this is out of scope for the present project. So to configure web servers with these instructions and to analyse the data produced would need to be another project.

9 Conclusions

This thesis was written to study what kind of logs are needed to have effective data mining performed to gather information about the user behaviour of a web service. For the data mining method to be used association rule analysis was chosen.

The beginning of the study included theory on the subjects of web server logging and data mining in general. After giving a short introduction to these topics the study got to the actual subject of data mining the server logs.

After the field of web server logs and data was introduced, the thesis could proceed to the actual topic. In the field of association rule analysis the problems that had to be solved was creating a tool set and environment to execute association rule analysis on web server logs. This pipeline was developed to work in a way that could take standard web server logs and with small modifications create association rules. Finally, interesting rules could be selected among all the rules mined in the postprocessing step.

After the pipeline was created it could be seen what kind of results and problems would come up with three different data sets, which consist of different log formats and different data in them.

After the whole pipeline was developed the study shows what kind of results this system was able to produce and how they could be interpreted. The study does not go very deep into the inner works of web services from which the example data was gathered, so there is no deep analysis of the business meaning of the rules. The study does, however, explain how rules are built and what kind of conclusions can be derived from these rules.

The most important part of the study was creating the guidelines on what kind of data is needed in the logs to be able to have possibly useful rules. The action points can help in creating logging rules, to have richer possibilities of data mining useful information from web server logs.

It was easy to find features that were lacking in the example data sets and find data that was not being correct for the purposes of association mining. Therefore it was possible to find different development needs in the data and analyse how the data logging could be developed to have better results of the analysis.

In the end it could be seen how useful results could be found, and also the fact that the study was able to create new useful information. The study found problems to solve in data logging of services in question, as well as solutions to how these issues can be solved by different approaches to logging of the data.

One clear finding of the study is that the changing paradigm of web site structure, both as the technical structure of the web server infrastructure and the internal structure of how content is being organized in the html data files. This change has rendered the traditional default ways of logging web usage to be less useful with contemporary web sites.

This proved the need to think of web service logging as a broader topic which cannot be done with the default configurations of web site analytics. Often there is need for

application level to take actions to log events that the web server would not log otherwise.

References

1 Marek K. Using Web Analytics in the Library. Library Technology Reports 2001; 47(5):5-7.

2 Burby J. A. B. Web Analytics Definitions – Version 4.0. Washington: Web Analytics Association; 2007.

3 Waisberg D, Kaushik A. WEB ANALYTICS: EMPOWERING CUSTOMER CENTRICITY. The original Search Engine Marketing Journal 2009; 2(1):5-11.

4 Reddy K S, Varma G, Babu I R. Preprocessing the web server logs: an illustrative approach for effective usage mining. *ACM SIGSOFT Software Engineering Notes* 2012; 37(3) :1-5.

5 Koch D, Brocklebank J, Grant T, Roach R, Mining Web Server Logs: Tracking Users and Building Sessions. Cary, NC: SAS Institute Inc.; 2002.

6 Baysal O, Holmes R, Godfrey M W. Mining usage data and development artifacts. In Mining Software Repositories (MSR), 2012 9th IEEE Working Conference on IEEE, 2012; 98-117.

7 Géry M, Haddad H. Evaluation of web usage mining approaches for user's next request prediction. In Proceedings of the 5th ACM international workshop on Web information and data management. ACM; 2004. p.74-81.

8 Dumoulin J. Web server logs [online]. United States. NASA Kennedy Space Center; URL: http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html. Accessed 1 March 2015.

9 Cooley, R., Mobasher, B., & Srivastava, J. Web mining: Information and pattern discovery on the World Wide Web. In Tools with Artificial Intelligence, 1997. Proceedings., Ninth IEEE International Conference. IEEE; 1997. P.558-567.

10 Han J, Kamber M, Pei, J. Data Mining, Southeast Asia Edition: Concepts and Techniques. 2006. Morgan Kaufmann

11 North M. Datamining for the masses. 2012. A Global Text Project Book.

12 Ayankoya K, Calitz A, Greyling J. Intrinsic Relations between Data Science, Big Data, Business Analytics and Datafication. Proceedings of the Southern African Institute for Computer Scientist and Information Technologists Annual Conference 2014 on SAICSIT. 2014. ACM.

13 Chen C M, Liao S H. Association Rule Algorithms for Logical Equality Relationships. In Computer and Information Technology Workshops, 2008. CIT Workshops 2008. IEEE 8th International Conference on. 2008. IEEE. P.26-30.

14 Hahsler M. A Probabilistic Comparison of Commonly Used Interest Measures for Association Rules [online]. United States. Southern Methodist University; URL: http://michael.hahsler.net/research/association_rules/measures.html#conviction Accessed 5 May 2015.

15 Piatetsky-Shapiro G. Discovery, analysis, and presentation of strong rules. Knowledge discovery in databases,. 1991. AAAI. p.229-238.

16 Brin S, Motwani R, Ullman J D, Tsur S. Dynamic itemset counting and implication rules for market basket data. In ACM SIGMOD Record 1997.26(2):255-264.

17 Srivastava T, Desikan P, Kumar V. Web mining–concepts, applications and research directions. In Foundations and Advances in Data Mining. 2005. Springer Berlin Heidelberg. p.275-307.

18 Moens S, Aksehirli E, Goethals B. (2013, October). Frequent itemset mining for big data. In Big Data, 2013 IEEE International Conference. 2013. IEEE. p.111-118.

19 Dumoulin J. Web server logs [online]. United States. NASA Kennedy Space Center; URL: http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html. Accessed 1 March 2015.

20 Hall, M. Weka Documentation [online]. New Zealand. Pentaho. URL: http://weka.sourceforge.net/doc.dev/weka/associations/FPGrowth.html. Accessed 22 April 2015.

21 Bruha I, Famili A. Postprocessing in machine learning and data mining. *ACM SIGKDD Explorations Newsletter*. ACM. 2000. *2*(2):110-114.

22 Hahsler M, Chelluboina S. Visualizing association rules: Introduction to the R-extension package arulesViz. *R project module*. 2011. R project module. P.223-238.

23 Kim H, Lee S, Kim J. Exploring and mitigating privacy threats of HTML5 geolocation API. In *Proceedings of the 30th Annual Computer Security Applications Conference*. 2014. ACM. p. 306-315.

24 Katz-Bassett E, John J P, Krishnamurthy A, Wetherall D, Anderson T, Chawathe Y. Towards IP geolocation using delay and topology measure ments. In Proceedings of the 6th ACM SIGCOMM conference on Internet measurement. ACM; 2006. p.71-81.

25 Gueye B, Ziviani A, Crovella M, Fdida S. Constraint-based geolocation of internet hosts. Networking, IEEE/ACM Transactions 2006. *14*(6):1219-1232.

26 West R, White R W, Horvitz E. From cookies to cooks: Insights on dietary patterns via analysis of web usage logs. In Proceedings of the 22nd international conference on World Wide Web . International World Wide Web Conferences Steering Committee; 2013. p.1399-1410.

27 McAfee A, Brynjolfsson E. Big data: the management revolution. *Harvard business review* 2012; (90):60-6.

28 Russom P. Big data analytics. TDWI Best Practices Report, Fourth Quarter. 2011. p.4.

29 Kim G, Trimi S, Chung J. Big-Data applications in the Government sector. March 2014. Communications of the ACM CACM Homepage archive 2014; 57(3): 78-85.

30 Zaslavsky A, Perera C, Georgakopoulos D. (2013). Sensing as a service and big data. *arXiv preprint arXiv:1301.0159*. 2013. p.3.

31 Ellison N B. Social network sites: Definition, history, and scholarship.Journal of Computer‐Mediated Communication 2007; 13(1):210-230.

32 Apache Software Foundation. Welcome to Apache™ Hadoop®![online]. USA. Apache Software Foundation. 22 September 2009. URL: http://hadoop.apache.org/. Accessed 13 March 2015.

33 Katal A, Wazid M, Goudar R H. Big data: Issues, challenges, tools and Good practices. In Contemporary Computing (IC3), 2013 Sixth International Conference. IEEE. 2013. p. 404-409.