

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

EXPLORING AND VALIDATING DATA MINING ALGORITHMS FOR USE IN DATA ASCRIPTION

by

Daniel P. Huynh

June 2008

Thesis Advisor: Second Reader: Simson L. Garfinkel Craig Martell

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE					Form Approved OMB No. 0704–0188
The public reporting burden for this collection of ini maintaining the data needed, and completing and suggestions for reducing this burden to Departmer Suite 1204, Arlington, VA 22202–4302. Responde information if it does not display a currently valid O	formation is est reviewing the co at of Defense, V nts should be a MB control nun	mated to average 1 hour per res ollection of information. Send cor lashington Headquarters Service ware that notwithstanding any other. PLEASE DO NOT RETUR	ponse, including the t nments regarding this es, Directorate for Info her provision of law, n IN YOUR FORM TO T	ime for review s burden estim rmation Opera o person shall THE ABOVE A	ing instructions, searching existing data sources, gathering and ate or any other aspect of this collection of information, including ations and Reports (0704–0188), 1215 Jefferson Davis Highway, be subject to any penalty for failing to comply with a collection of DDRESS.
1. REPORT DATE (DD-MM-YYYY)	2. REPO	RT TYPE			3. DATES COVERED (From — To)
20-06-2008	Master ³	s Thesis			01-07-2006-20-06-2008
4. TITLE AND SUBTITLE	intustor	5 1110515		5a. CON	
5b. GRANT NUMBER Exploring and Validating Data Mining Algorithms for use in Data Ascription				NT NUMBER	
5c. PROGRAM ELEMENT N					GRAM ELEMENT NUMBER
6. AUTHOR(S)				5d. PRO	JECT NUMBER
Daniel P. Huynh	5e. TASK NUMBER			KNUMBER	
Damer F. Huyim				54 WOD	
				51. WOH	K UNII NUMBER
7. PERFORMING ORGANIZATION N	AME(S) AN	D ADDRESS(ES)			8. PERFORMING ORGANIZATION REPORT NUMBER
Naval Postgraduate School					
9. SPONSORING / MONITORING AG	ENCY NAN	IE(S) AND ADDRESS(E	S)		10. SPONSOR/MONITOR'S ACRONYM(S)
Department of the Army					11. SPONSOR/MONITOR'S REPORT NUMBER(S)
12. DISTRIBUTION / AVAILABILITY S	STATEMEN	Т			
Approved for public release; distr	ibution is	unlimited			
13. SUPPLEMENTARY NOTES					
Digital forensics is a growing and today. As more information is sto evidence has grown in complexity evidence obtained from the field, visualization and not data reduction possible to use file metadata accur explore and validate existing algor measure the accuracy of these alg framework for media exploitation	l importan red in digi y. Today a means tha on or corre rately to a rithms tha orithms. T in compu	t field of research for ital form, the need and nalysis is reliant upon t analysis frequently t elation. This thesis fill scribe ownership of fi t may support and aid This work facilitates fu ter forensics.	current intellig l ability to ana trained exper- takes too long. ls an importan les based upor l data ascriptio urther research	gence, law lyze and ts. This, c Current t t void. Th t a hard d n. The la i into deve	w enforcement, and military organizations process this information for relevant compounded with the sheer volume of forensic tools focus on decoding and he first goal is to determine whether it is rive with multiple users. The second is to st goal of this work is to compare and eloping an automated analysis and reporting
15. SUBJECT TERMS					
Data mining algorithms, metadata	a, file ascri	ption, data carving, m	ulti-user hard	drives	
16. SECURITY CLASSIFICATION OF a. REPORT b. ABSTRACT c. T	: HIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF	19a. NA	ME OF RESPONSIBLE PERSON
Unclassified Unclassified Un	classified	UU	PAGES 77	19b. TEI	LEPHONE NUMBER (include area code)
		1	1	1	Standard Form 209 (Boy 9, 09)

Approved for public release; distribution is unlimited

EXPLORING AND VALIDATING DATA MINING ALGORITHMS FOR USE IN DATA ASCRIPTION

Daniel P. Huynh Captain, United States Army B.S., United States Military Academy, 1999

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL June 2008

Author:

Daniel P. Huynh

Approved by:

Simson L. Garfinkel Thesis Advisor

Craig Martell Second Reader

Peter Denning Chair, Department of Computer Science

ABSTRACT

Digital forensics is a growing and important field of research for current intelligence, law enforcement, and military organizations today. As more information is stored in digital form, the need and ability to analyze and process this information for relevant evidence has grown in complexity. Today analysis is reliant upon trained experts. This, compounded with the sheer volume of evidence obtained from the field, means that analysis frequently takes too long. Current forensic tools focus on decoding and visualization and not data reduction or correlation. This thesis fills an important void. The first goal is to determine whether it is possible to use file metadata accurately to ascribe ownership of files based upon a hard drive with multiple users. The second is to explore and validate existing algorithms that may support and aid data ascription. The last goal of this work is to compare and measure the accuracy of these algorithms. This work facilitates further research into developing an automated analysis and reporting framework for media exploitation in computer forensics.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Digital Forensics Science	2
1.3	Data Mining and Data Ascription	4
1.4	Outline of this Work	5
2	Related Work	7
2.1	Cross Drive Analysis.	7
2.2	Text Mining	8
2.3	Data Mining in Related Areas of Work.	9
3	A New Approach For Carved Data Ascription	11
3.1	Algorithmic Background	11
3.2	Tools and Data Formats.	17
3.3	Components of Actual Test Data	20
3.4	Assumptions and Controls.	22
3.5	Methodology	23
4	Experimental Results	27
4.1	Built Drive Data Results	28
4.2	Real Corpus Drive Data Results	28
5	Conclusion and Future Work	31
A	Data Preparation	33
B	The Weka Explorer	35
С	Test Run Results	37
Ini	itial Distribution List	61

List of Figures

1.1	The nucleus of digital forensics. Finding its niche.	3
1.2	Data Mining within the 7 step process.	4
3.1	File Ascription Start to Finish : Abstraction.	12
3.2	Application of Instance Based Learning with Knearest Neighbor	14
3.3	Application of J48 Decision Tree Learning.	15
3.4	Example of K-fold Cross Validation, K=4	16
3.5	FIWALK creates ARFF data	19
3.6	ARFF Breakdown of Components.	21
3.7	Documents and Settings File System Structure.	23
A.1	This screen capture shows the selection of the id and filename attributes which are removed from the analysis.	33
A.2	After the id and filename attributes are deleted, all of the string attributes need to be converted to nominal attributes in order to allow the extracted metadata to be analyzed by Weka	34
B .1	This screen shot shows the various classifiers built into the Weka toolkit	35
B.2	This screen shot shows the selection of the J48 algorithm	36
B.3	This screen shot shows the selection of the 1Rule algorithm	36

List of Tables

3.1	Test Hard Drive Statistics	22
4.1	Accuracy Results on Locally Built Data	28
4.2	Accuracy Results on Live Corpus Drives	29

Acknowledgements

Thanks to Simson L. Garfinkel and Jim Migletz for their mentorship and combined work. Thanks to Craig Martell who graciously volunteered to be my second reader.

CHAPTER 1: Introduction

1.1 Motivation

"We are overwhelmed with data. The amount of data in the world, in our lives, seems to go on and on increasing – and there's no end in sight [19]." Currently the intelligence and law enforcement communities rely on trained experts to conduct media exploitation of devices captured either on the battlefield or through investigative work. The analysis can require weeks and in some cases months to complete. For individuals relying on current information to plan and execute their next move, the wait can be a severe impediment to their efforts. Additionally, current tools focus on data decoding and visualization tasks and not data reduction or correlation. Such data reduction or correlation could be invaluable for these agencies wishing to act upon real time information or findings.

Providing intelligence and law enforcement personnel with an automated means of exploiting captured media within hours instead of weeks will allow for an increase in operating tempo and improved actionable intelligence. These benefits can be achieved through an automated analysis and reporting framework for media exploitation in computer forensics.

In 2006 Richard and Roussev discussed the urgent need for new tools and strategies for the rapid turnaround of large forensics targets. They focused on the acquisition and analysis of forensics evidence and argued that current forensics tools are inadequate given the increased complexity of cases, increased size of targets, better awareness of the capabilities of digital forensics, and multi-computing scenarios. "Tools to automate analysis of digital evidence are needed now – current tools are not prepared for the huge targets of tomorrow (or today) [17]."

This thesis addresses the challenging but relevant problem of automatically analyzing raw data from acquired storage media to rapidly infer information about the ownership of the individual files. Work has previously been completed in associating hard drives to owners by Garfinkel [15]. This thesis seeks to further identify ownership of individual files given one hard drive and multiple users from file metadata, a problem which has not previously been addressed. The focus of this paper will be to explore methodologies and algorithms that may be used in data ascription of files to users and to evaluate their accuracy.

1.2 Digital Forensics Science

In August of 2001, over fifty researchers, computer forensic examiners, and analysts met in Utica, New York to attend the first ever research workshop on digital forensics. Its purpose was to ignite discussion, to explore future areas of importance, and to lay a basis which would attempt to define the field more concretely [10]. With increased funding, interest, and attendees, the Digital Forensics Research Workshop has become an annual event. As technology and computing have exponentially grown over the past years, it is valuable to look back at the original stated purpose of the workshop and to think about how this field was in its infancy: "Build a taxonomy to guide and direct research. Identify the areas or categories that define the "universe" of Digital Forensic Science [10]." Much progress has been made since the first DFRWS meeting, and yet much work still lies ahead. Similar to traditional forensic science, digital forensics science finds its differences in the sources of what is to be considered evidence. Compiled from group discussions at the first DFRWS is a formal definition of digital forensic science.

Digital Forensic Science:

The use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations [10].

Digital forensics is relevant and important because of the vast increase in dependency on digital sources of information, computerized systems, networks, data storage, processing, and transmission in all aspects of our lives. From law enforcement to courts, homeland security, military operations, business and industry, and critical infrastructure digital forensics has played and will continue to play increasing rolls of importance. As technology continues to advance we will reach unprecedented volumes of data and be faced with the challenges of managing it all [10].

Figure 1.1 further illustrates where digital forensic research has found its niche, and where it lies amongst the broad battlespace given its enormous potential for application.



[10]

The founders of the first DFRWS characterized the discipline of digital forensic science with the following associated entities:

- Theory: a body of statements and principles that attempts to explain how things work
- Abstractions and models: considerations beyond the obvious, factual, or observed
- Elements of practice: related technologies, tools, and methods
- Corpus of literature and professional practice
- Confidence and trust in results: usefulness, purpose [10]

The framework first developed at the DRWS broke digital forensics down into a seven step process. They are identification, preservation, collection, examination, analysis, presentation, and lastly decision. It was this time also that data mining was classified as a key area of research under the analysis step. This thesis specifically deals with the elements of practice and the tools of the science within step five as seen in Figure 1.2.



Figure 1.2: Data Mining within the 7 step process. [10]

1.3 Data Mining and Data Ascription

"Data mining can be defined as the analysis of often large observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the owner [8]." First and foremost, for data mining to be relevant, the data sets of interest must be large; if they were not then it might be feasible to manually explore the data and make a decision. There are varying scales of data sets that may be considered to be large, but this thesis focuses on data and files on a single hard drive which may easily number to millions of files at a time. Once validated on a single hard drive, the goal is scale the data mining effort to much larger data sets across multiple hard drives and possibly networks. Secondly as defined, a goal of data mining is to find unsuspected or unknown relationships within data. Obviously there is no reason to report or to repackage already known relationships through data mining. This thesis seeks to find the relationships among files or data, specifically the ascription or ownership of the data. From a forensics perspective such correlations or relationships discovered may be used to tie criminals, terrorists, or people of interest together. Lastly, the results of data mining must be understandable and useful. Within our context of research, the end goal is to develop a readable and reportable format which may applied and incorporated into existing forensics tools for use by examiners or operatives in the field.

There exists a formal methodology for data mining which includes these basic steps:

- Determine the nature and structure of the representation of the data sets
- Decide how to quantify the data; compare how well different representations fit the data
- Choose an algorithmic process to optimize the scoring function
- Decide what principles of data management are required to implement the algorithms efficiently [8]

Data mining is often placed into the greater context of knowledge discovery in databases (or KDD) which was originated from the artificial intelligence research field [8]. The proceedings of the 9th KDD International Conference on Data Mining and Knowledge Discovery in 2003, give an invaluable recounting of the stated past accomplishments and future goals of data mining research. This was a special session to explore and discuss the future of data mining. From a historical point of view, the most notable developments are related to the creation of the World Wide Web. The web paved the road for new forms of communication, information processing, and storage which led to web mining, content mining, usage mining, search engines, and intelligent agents [11]. Additionally, the members of the conference speculated on the future "hot applications" for data mining and link analysis, E-commerce, bioinformatics and in silico drug design, and lastly multi-media data mining [11]. This thesis focuses in on relational mining and link analysis.

1.4 Outline of this Work

This thesis has three main objectives. The first is to determine whether it is possible to use file metadata accurately to ascribe ownership of files based upon a hard drive with multiple users. The second is to explore and validate existing algorithms that may support and aid data ascription. The third goal of this work is to compare and measure the accuracy of these algorithms when applied to our data. Chapter two discusses other work related to this thesis; this includes a discussion on text mining, cross drive analysis, meta data extraction, data mining relevance. Chapter three details the algorithmic background and methodology used in this thesis. Chapter four discusses the experimental results. Lastly Chapter five discusses the conclusion and introduces future work.

CHAPTER 2: Related Work

In order to gain a greater understanding of this thesis and its challenges, it is useful to discuss some of the related work and research that has been done in the field already. First discussed will be that of cross drive analysis (CDA). Second, text mining and authorship ascription will be mentioned. Subsequently, numerous applications and work with data mining will be explained and how they have been applied across different disciplines with respect to this thesis.

2.1 Cross Drive Analysis

Garfinkel's Forensic Feature Extraction (FFE) and Cross Drive Analysis (CDA) allow an investigator to analyze information across a span of data sources, rapidly identify drives of interest, and to correlate information between disk drives [15]. Some key concepts that Garfinkel's work brings about are automated feature extractors and pseudo-unique identifiers. A feature extractor is a program that can scan a disk image for identifiers and store the results in an intermediate file. Examples of some feature extractors Garfinkel built include: email address extractor, email id extractor, date extractor, cookie extractor, social security extractor, and a credit card number extractor. A pseudo-unique identifier is an identifier that has enough entropy which will ensure it will not be repeated by chance. An example of one is that of an email message id. Garfinkel additonally introduced the notion of cross drive analysis for forensic use. Garfinkel was able to successfully conduct drive attribution and ownership from within his corpus of hard drives by using email addresses, credit card numbers, and social security numbers [15].

Garfinkel's work identified several uses for cross drive analysis:

- automatic identification of hot drives
- ascription of drive ownership
- improving single drive forensics systems
- identification of social network membership [15]

This thesis builds upon Garfinkel's work by striving to ascribe at the resolution of a single file rather than an entire drive.

2.2 Text Mining

In this section, text mining will be discussed to the depth of comparing and contrasting it with data mining. There are varying opinions on how data mining and text mining are similar and different. Text mining is in itself a large and emerging field with much research that has been done and continues to grow. Text mining can be defined as the discipline of extracting meaningful information from natural language text [28]. "Compared with the kind of data stored in databases, text is unstructured, amorphous, and difficult to deal with algorithmically [28]." As data mining can be roughly described as looking for patterns in data, text mining is about looking for patterns in text. One perspective on data mining argues that it deals with the extraction of implicit, previously unknown, and potentially useful information from data. This novel information is implicit within the test data, hidden, and unknown to the user. On the other hand, text mining deals with information that is to be extracted which is already clearly and explicitly stated in the text. All in all, most scholars agree that text mining is a branch or a sibling of data mining [21].

Some current uses of text mining include text summarization, document retrieval, text categorization, document clustering, language identification, and ascribing authorship of documents [19]. In 2001, O. de Vel, Anderson, Corney and Mohay investigated email content mining for author identification and authorship attribution for forensic applications. Their test email corpus consisted of 156 documents sourced from three English authors with each contributing emails on specific subject topics. Their work focused on the ability to distinguish between authors on the use of aggregated emails topics as well as different topics. Through the use of support vector machine learning algorithms, their experiments gave positive results across a data set of email documents which ranged between 70% to 100% accuracy [9]. In 2005, Stolfo and Hershkop developed an email mining tool kit (EMT), that visualizes a wide range of detailed analyses of email and email flows from an archive. EMT's purpose was to support law enforcement agencies in analyzing and handling email evidence under investigation [27].

It is important to note that text mining looks at the actual content of text and is a successful approach towards authorship ascription. Let us bring in the concept of ownership and differentiate it with authorship. Given a file on a hard drive, a user may be both an author and owner of a

file. What happens if the user is not the author of the file, but only the owner? For example the owner of a computer system could download a file of contraband information from a criminal website. In the subsequent investigation it would be necessary to identify the owner of the file; that is the person who had downloaded the file. Whereas the author of a file is responsible for its content, the file's filesystem metadata is determined by the activities of the file's owner. To that end, this thesis only considers metadata to ascribe ownership.

2.3 Data Mining in Related Areas of Work

In 2003, the Artificial Intelligence Lab at the University of Arizona, presented an overview of case studies done with relation to their COPLINK project. The project's specific interests was how information overload hindered the effective analysis of criminal and terrorist activities by law enforcement and national security personnel. Their work proposed the use of data mining to aid in solving these issues. In their report they define data mining in the context of crime and intelligence analysis to include entity extraction, clustering techniques, deviation detection, classification, and lastly string comparators. Four case studies in the report showed how data mining was useful in extracting entity information from police narrative reports, detecting criminal identity deceptions, authorship analysis in cyber crime, and lastly criminal network analysis. Today, COPLINK is software that has been successfully deployed in the field, and works by consolidating, sharing, and identifying the information from online databases and criminal records [6].

Work done by Hewlett Packard in 2005 applied data mining to solve their problem of finding similar files in large document repositories. HP has many millions of technical support documents in a variety of collections. Their researchers sought a way to routinely locate similar documents and to delete obsolete versions. Their approach for finding similar files that scales up to a large document repository is based upon chunking byte streams to identify unique signatures. The end analysis yielded clusters of related files and was further enhanced by applying a graph bipartite partitioning algorithm [12].

The automation of organizing digital photographs has grown in popularity in recent years. Research done by Cooper, Foote, and Girgensohn in 2005 looked at ways to cluster digital photographs based on temporal information and content. Using similarity based methods, they successfully applied supervised and unsupervised clustering algorithms to their data sets after extracting time stamps from EXIF data sections embedded in JPEGS by digital cameras [7]. In 2006, Galloway and Simoff experimented with a case study redefining an approach to network data mining. In their work, they defined network data mining as identifying emergent networks between large sets of individual data items. By improving upon traditional data mining techniques, they utilized special algorithms to aid in the visualization of emergent patterns and trends in the linkage. They emphasized a human centered data mining methodology which was successful in discovering implicit relationships between data attributes. Specifically in their work they analyzed over 5000 department of motor vehicle claims and were able to detect fraudulent claims amongst the legitimate ones [13].

Shatz, Mohay, and Clark in 2006 explored a correlation method for establishing provenance of timestamped data for use as digital evidence. This work has a deep and relevant impact on digital forensics research as it reiterated the complexity issues of dealing with timestamps because of clockskew, drift, offsets, and possible human tampering. Furthermore, their work presented a new approach for discovering temporal behavior of a particular computer over a range of time by correlating local machine timestamps and an analysis of web browser records [4].

In 2006 as well, research done by Abraham explored event data mining to develop proles for computer forensic investigation purposes. Abraham analyzed computer data in search of discovering owner or usage profiles based upon sequences of events which may ocur on a system. Abraham categorized an owner profile with four different attributes: subject, object, action, and time stamp [1].

In 2007, Beebe and Clark in their work proposed pre-retrieval and post-retrieval clustering of digital forensics text string search results. Though their work is focused on text mining, the data clustering algorithms used have shown success in efficiency and improving information retrieval efforts [26].

CHAPTER 3: A New Approach For Carved Data Ascription

The motivation of this thesis is a problem that confronts law enforcement: information is found on a hard drive that has been used by multiple individuals. Some technique is needed to determine which individual is responsible for the data. Not only must the user be identified from one of many choices, but the likely accuracy of the identification must be provided in line with Daubert standards.

A more specific example of this would be if police impound a computer from a lab that was used by six different graduate students. On this computer's hard drive are files belonging to each student. But when a file carver is applied to the hard drive additional information is discovered such as terrorist plans and schematics. How do we go about identifying from our list of authorized users the most likely person to be responsible for the contraband information on that computer system? Faced with this problem, a human investigator might look at time stamps in the files or the disk sectors where the information was found. The investigator would then try to correlate this information in the files of known authorship. This thesis seeks to test a methodology that would be useful in this situation.

3.1 Algorithmic Background

We developed an architecture in which the hard disk is imaged, and then processed with an automated metadata extraction tool which builds a list of all the files and any relevant metadata. In our investigative scenario this metadata from the carved files would be compared with meta data of files from known ascription. Our experiment tested several different datamining algorithms with K-fold cross validation to determine which performed the best. Figure 3.1 on the following page illustrates an abstraction of how this methodology is done starting with a given hard disk and the end results of carved files ascribed to users.

3.1.1 The Classification Problem

"A solution for classification generated from a set of training examples will almost always be highly accurate on the same data, but far less accurate on new data [3]." In the context of this thesis, classification is defined as a way to place files into groups based upon quantitative infor-



Figure 3.1: File Ascription Start to Finish : Abstraction.

mation or attributes, where what is learned as a result, is later applied to new files for grouping. An important distinction to be made is that of training examples and test data. As suggested by Apte and Weiss, classification learning done on a set of training examples will usually be accurate when applied to the same data because it has already seen all of the exemplars. The true test is to see how the classification is applied to new test data after having been trained. In our experiment, we used a multi-user hard drive to develop our data mining approach. In testing of these different classification algorithms, we use known files as exemplars. Ultimately, we seek to find the most accurate classification model from our data set, and strive to obtain the model's true accuracy on unseen instances. This thesis explores Instance Based Learning (IBL) with K-nearest neighbors (knn) [19], J48 [19], and the 1-Rule [19] algorithms within our framework.

In theory, each algorithm chosen would work on our test data. Instance based learning was chosen because given our data set, the number of attributes was fairly small (less than 20), and

also because there was a lot training data to use. Instance based learning, requires no training. Past work by Aha and Kibler showed the effectiveness of Instanced Based Learning with ways to improve performance [2]. Based upon work by Apte and Weiss, J48 and 1-Rule were chosen because they are based upon decision tree logic which have also proven to be very powerful and effective [3]. The J48 and 1-Rule algorithms have the advantage of generating understandable rules, not requiring expansive computations, and are compatible with the kinds of metadata being analyzed in our work.

It is important to note that each of these algorithms also potentially suffer from common data mining issues such as over-fitting and dimensionality. Over-fitting occurs when our model characterises too much detail, or noise in our training data which results in poor learning performance [25]. The curse of dimensionality is a concept that applies to instance based learning as the number of attributes increases, if the number of relevant attributes remains low, learning performance decreases because of the size of the data space [25]. A solution to guard against the curse of dimensionality is weighting attributes. For simplicity, we chose to use a common standard of Euclidean distance and did not weight attributes in our model.

3.1.2 Classification Algorithms

Instance based learning with k-nearest neighbor

Instance based learning is known as a lazy classifier because it defers learning as long it possibly can. Each new instance of data is compared with an existing one using a distance metric, and the closest existing instance is used to assign the class to the new one. The closest existing instance is often referred to as the nearest neighbor classification method; when more than one nearest neighbor is used, it is called k-nearest-neighbor [19]. Instance based learning is one of the simplest forms of learning and often very accurate. With instance based learning, there are no new rules created but the knowledge representation comes from within the existing data itself. Computing the distances between instances is fairly basic when dealing with all numeric data types. The standard distance function used in most nearest neighbor classifications is that of Euclidean distance.

It is important to note that with this base function and how it is implemented within our framework the assumptions are that data is of equal importance. Weighting is not applied; weighting of attributes is possible and can be done as attributes are found to be more important than another. It is also worthy to note that this algorithm may be much slower with respect to others because it must scan the entire data set for each item to be classified.

How is instance based learning applied to our test data? Given our disk image, the instances of data are the files and their attributes. Since we are concerned with ascribing files with users, we care about the userguess attribute defined for each instance of data. Figure 3.2 below illustrates a conceptual diagram of how K-nearest neighbor is applied to our data.



Figure 3.2: Application of Instance Based Learning with Knearest Neighbor.

As File X is determined to be the closest to File 4 in distance, File X is classified as having the same userguess as File 4.

J48

Decision trees represent a supervised approach to classification which is done recursively. A decision tree is a simple structure where non-terminal nodes represent tests on one or more attributes and terminal nodes reflect outcomes. By first selecting an attribute to be the root node, a branch is made for each possible value. As a result of these splits, one value is associated with each attribute. Recursion then occurs on each branch until all instances on a node have the same classification [3].

A current and popular implementation of this approach is known as Quinlan's C4.5 model [28]; C4.5 is well known as an "industrial strength algorithm" which has been modified and updated over the years [19]. Our framework's version of C4.5 is called J48. J48 works by first choosing an attribute that best differentiates the output attribute values. Next, a separate tree branch is created for each value of the chosen attribute. The instances are then divided up into subgroups to reflect the attribute values of the chosen node. A key part of this algorithm is based upon which attribute to choose as the root node. This procedure is based upon information gain and entropy [3]. Figure 3.3 is a conceptual diagram which shows how the J48 algorithm is applied to our data.



Figure 3.3: Application of J48 Decision Tree Learning.

After a root node is chosen from our list of attributes, recursion is done on each branch to classify users for each file encountered. It is important to note the attributes selected by the algorithm for each node, and also the number of files assigned to each user. Based upon the branching comparisons, a new node such as File X would get aligned with a corresponding userguess.

1-Rule

Work done by Holte illustrated the power and relevancy of using the 1-Rule classification algorithm for many reasons, but most importantly for its simplicity and accuracy [18]. 1-Rule is a very inexpensive algorithm. It generates a one level decision tree expressed in the form of a set of rules that all test one particular attribute. It is fairly accurate and inexpensive in terms of computation. By making rules that test a single attribute and branching, each branch is associated with a different value of the attribute. Each attribute generates a different set of rules, one rule for every value of the attribute. As 1-Rule is also based upon a tree decision algorithm, it shares many similarities to J48 [18].

3.1.3 K-fold Validation As a Testing Methodology



K-Fold Cross Validation Example (K = 4)

Figure 3.4: Example of K-fold Cross Validation, K=4

K-fold cross validation is the most common approach to estimating the true accuracy of a given model and to address the problem of over-fitting [22]. K-fold cross validation is based on randomly splitting the available sample between a training set and a testing set. The approach is known as rotation estimation, the "K-fold" represents the number of partitions the data set is broken up into for testing and for training. For example, given K = 4, a data set will be

broken up into 4 equal partitions. Training is done on $\frac{3}{4}$ of the data, and testing is done on the remaining $\frac{1}{4}$ of data. The procedure is repeated K times, until every $\frac{1}{4}$ of the data has been used for testing. Without separating the data into test and training portions it is common to have a biased or lower error estimate than the true error rate.

Figure 3.4 illustrates conceptually how data is partitioned based upon the K value. For a value of K = 4, the experiment is run 4 times; in the end all of the data has been used for both training and testing.

Cross validation is a valuable method because it ensures that all examples in the data set are eventually used for both training and testing, while eliminating the bias and improving the true error estimate. Research and studies have shown that 10 fold cross validation has been the most accurate in estimating error [22] and for this reason we use 10 for our K value in this thesis. Using instance based learning, J48, and 1-Rule algorithms we analyzed their application under cross validation to all of our data.

This thesis will next review and discuss some of the relevant tools, file formats, and assumptions used for testing.

3.2 Tools and Data Formats

This section gives a brief overview of the relevant tools and file formats used in this thesis. It is important to note that there are other available tools to facilitate the work done here, but these in particular are open source.

3.2.1 Aff

The disk images for this work were stored using the Advanced Forensics File (AFF) format, an open and extensible file format designed to store disk images and associated metadata. There are numerous advantages of the AFF format, including the ability to store an entire disk image and associated information into a single file, being able to store an arbitrary amount of metadata and user defined metadata, and lastly the ability to store disk images in an encrypted form [16].

3.2.2 Afflib

AFFLIB and AFF Tools are open source library developments created by Simson Garfinkel and Basis Technology. These tools allowed the manipulation and processing of the AFF files used in this thesis. Additionally, AFFLIB has been incorporated into Brian Carrier's Sleuthkit (TSK) and Autopsy. TSK will be explained in more detail later on in this section. The version of AFFLIB used in this thesis was v3.1.3.

3.2.3 Aimage

AIMAGE is an advanced disk imager which is part of the AFFLIB suite used to image drives or data. Aimage is the program that was used to image the hard drives in this thesis. The version of Aimage used was v3.0.0a1.

3.2.4 Fiwalk

FIWALK (file and inode walk) is an open source software tool developed by Simson Garfinkel that retrieves information from disk partitions found on disk images.

It relies on the TSK programmer's interface to find all of the files in a given disk image. In addition, FIWALK is also a metadata extraction system whereby file metadata is pulled through TSK and user defined plug-ins. FIWALK can create Attribute Relation Formatted File (ARFF) files from given AFF images.

FIWALK was vital to this thesis because it was used to carve and create the ARFF files used in thesis. FIWALK version .3 was used in this thesis.

FIWALK works by:

- Finding all of the partitions on the disk.
- For each partition, walk the files.
- For each file, print the requested information.
- For each partition, walk the inodes
- For each inode, print the requested information. [14]

Figure 3.5 is an extract of actual ARFF data created from FIWALK which shows the corresponding parts of an ARFF. There are three sections created which detail most importantly the name, the attributes to be factored into, and lastly the instances of each attribute as extracted from the hard drive. Values in the data section are separated by commas.



Figure 3.5: FIWALK creates ARFF data.

3.2.5 Weka

The WEKA workbench is a collection of machine learning algorithms and data preprocessing tools. Developed at the University of Waikato in New Zealand, WEKA stands for Waikato Environment for Knowledge Analysis [19]. The algorithms within Weka can either be applied directly to a dataset or called from Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes. Weka is open source software issued under the GNU General Public License. We used WEKA's implementation of IBL, J48, and 1-Rule algorithms for this thesis. The version of WEKA used was v3.5.7.

3.2.6 Arff

An Attribute-Relation File Format (ARFF) file is an ASCII text file that describes a list of instances sharing a set of attributes. ARFF files were developed by the Machine Learning

Project at the Department of Computer Science of The University of Waikato for use with the Weka machine learning software [19]. The ARFF file format is used for moving complex data from FIWALK into WEKA. There are several components which comprise an ARFF file; they are the relation, attribute, and data sections. The relation field is the relation name, normally comprised of the header section. The attribute component represents the columns of data and their data types to be used. Lastly, the data section component lists by line the actual instances of data to be analyzed.

3.2.7 The Sleuth Kit

The Sleuth Kit (TSK) is a very well known collection of UNIX-based command line file and volume system forensic analysis tools [5]. TSK allows one to extract files from a file system disk image without using an operating system. TSK supports DOS partitions, BSD partitions (disk labels), Mac partitions, Sun slices (Volume Table of Contents), and GPT disks. Further, TSK can identify where partitions are located and extract them so that they can be analyzed with file system analysis tools. FIWALK is dependent upon libtsk and various interfaces for its operation [5].

3.3 Components of Actual Test Data

So let us take a step back and describe our what our actual data is at this point. Given a single hard drive from within a computer and multiple users, Aimage was used to capture the test data off the drive which is all the files on the hard disk. Upon creation of the AFF image, Fiwalk was used to carve out the files and metadata and to produce an ARFF file. Figure 3.6 details the breakdown of components and their mappings from an actual excerpt of actual test data.

In Figure 3.6, the name of relation for the ARFF file is fiwalk. The attributes or the metadata extracted from each file off the hard drive is also listed. For our experiment, the attribute id is purely a sequential generated number starting with the number one, associated with each instance of data. The partition attribute details which partition of the hard drive the file belongs to; typically there are 4 primary partitions and its data type is numeric. The file size details the size of the file as a numeric data type. The mtime, ctime, and atime attributes represent the modified time, created time, and access time of the file in year, month, day, hour, minutes, and seconds format. The fragments attribute details how many fragments the file is broken into in numeric format. The frag1 and frag2 start sectors detail which


Figure 3.6: ARFF Breakdown of Components.

sector the fragment begins in numeric format. The filename attribute lists the file name of the file on the hard drive in string format. The userguess attribute is a parsed string which contains a guess on who the user of the file is based upon a windows file and system structure convention. The order of attributes dictate the order in which they appear in the data section of the ARFF file. For example, in Figure 3.6 the first attribute is id of 157, and the last attribute is the userguess of Administrator.

3.3.1 Built Drive Data vs Real Corpus Data

Our experiment consisted of two different sets of test data. The first set of actual test data was comprised from a locally populated hard drive with the user profiles we setup. The second set of data came from Garfinkel's live hard drive corpus; a body of over 1000 live drives from around the world. We selected 5 drives from the live corpus that appeared to have more than one user profile. This was done through a basic grep command. It is important to distinguish between the two different types of hard drives used because the locally built drive was used to test our hypothesis. It was our ground truth to show that our technique of ascribing files to owners could work. Objectively, the hard drives selected from the real data corpus were used to further validate our findings. The four drives selected were from China, and the fifth drive was from Israel. Below is a table that breaks down a comparison of all the drives used in terms of size, and the number of user profiles identified.

Drive Type	Number of User Profiles	Drive Size
Built test drive	9	29.4 GB
Real corpus drive (cn3-03)	11	31.80 GB
Real corpus drive (cn1-1-6)	5	6.74 GB
Real corpus drive (cn3-06)	5	14.32 GB
Real corpus drive (cn4-02)	10	6.01 GB
Real corpus drive (il03)	5	15.13 GB

Table 3.1: Test Hard Drive Statistics

3.4 Assumptions and Controls

There are key assumptions and controls which must be addressed about how tests were done. The data mining algorithms we used required exemplars – data elements for which the ground truth is known. We used the path name of files residing in the Documents and Settings directory for this purpose.

For example, files contained by the path \Documents and Settings\Administrator, we assumed to belong to the user "Administrator". It is important to note that on a different operating system, different results may be found based upon file naming conventions, and the file feature extractors used; for example inode owners, unique identifiers in metadata, or sector numbers. For our local test drive, the profiles we created were: dph2007, hunter, jjm2007, simsong. By using only files within these specified directories our model was intentionally simplified.

By using a hard disk with Windows as the installed operating system we make the assumption that our tools and methodology can be validated on one type of operating system. If applied to another operating system, in order to achieve the same level of accuracy estimation, different feature extractors would need to be used.



Figure 3.7: Documents and Settings File System Structure.

3.5 Methodology

The experiment consisted of six steps in accomplishing our stated goal:

- 1. Selection and setup of a test computer
- 2. Population of test data
- 3. Imaging of the test data
- 4. Carving and creation of the ARFF file
- 5. Data preparation
- 6. Data Classification and Validation

3.5.1 Setup of the test computer

The experiment required a test computer to be set up. Our test machine was a Dell with 1GB of Ram, a 40GB hard drive, Pentium 2.8 processor, running Windows XP. As part of the setup of this machine, all previous user data and unnecessary programs were deleted and removed. Ccleaner [24] and Eraser [23] were used to ensure that all previous user data was deleted and overwritten. Additionally, the program Shred [20], as part of the Knoppix suite of forensics tools, was used to write over the empty space on the disk. Lastly, four unique user accounts were setup with unique passwords to ensure proper separation of data amongst profiles.

3.5.2 Populating the test machine with live data

Next, after the test machine was setup the hard drive needed to be populated with live test data. Over a period of three months, four individual users logged on to the system via their account profile and did various tasks on the machine. These tasks included surfing the internet, creation of documents, saving pictures, weblinks, and files off the internet, and also the importation of files and media from external sources onto their user profile. Usage of the test machine and tasks done on each profile was completely random and restricted to each individual profile. Each user was instructed to save their work and files to within their profiles and inside their "My Documents" folder.

3.5.3 Imaging of data

After the hard drive was sufficiently populated with live user test data, the next task was to actually image the hard drive with AIMAGE. The hard drive was physically pulled out of machine and plugged into an imaging workstation. This step put our live data into AFF format.

3.5.4 Carving and Creation of the ARFF File

Given the working AFF image of the test data, Fiwalk was run on the image to carve and create our ARFF file.

3.5.5 Data Preparation

With the creation of our ARFF file, we now have the file metadata that has been extracted from our test drive. As is common in data mining, before running tests on data instances, it was necessary to clean and prepare our data for use into the WEKA workbench. An important piece here was the need to convert string data into nominal data from the ARFF file. This was done based upon the requirements constraints of the algorithms used, as they do not accept string data for processing. In addition, it was important to look at relevance of the attributes to remove redundant, noisy, or irrelevant features. We determined relevance based upon how much value the attribute would add to or disrupt our learning algorithms. In our data, we only removed two attributes which were file id and filename. The file id is an artificially generated numeric attribute from FIWALK, and does not add any value to the our classification. In addition, the filenames as attributes were removed because we did not want the algorithm to learn based upon text strings associated with the file. Lastly we chose to replace all missing values for attributes. Replacing missing values places the distribution towards the mean value of the most frequent values for an attribute, and prevents the loss of information which might potentially be useful for learning [19]. See Appendix A (Data Preparation).

3.5.6 Data Classification

Weka has four different application modes which can be worked within; they are the Explorer, Experimenter, KnowledgeFlow, and lastly SimpleCLI (command line interface). The Explorer mode provides an intuitive user interface for loading, filtering, clustering, classifying, and visualization of test data. The Experimenter, KnowledgeFlow, and SimpleCLI were not used in this thesis. After the data preparation was done, WEKA now could be used to run its suite of algorithms on the test data. See Appendix B (Running the Explorer and View of Classification Algorithms)

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4: Experimental Results

The metrics for measuring and comparing our algorithms were in the form of a percentage of correctly classified instances versus incorrectly classified ones. Our classifiers attempted to predict the owner of a file and if correct, a success was counted; if not an error was counted. In addition, we attempted to look at the time it took to run the algorithm in seconds. After completing all of the experimental runs, we determined that given our sets of data and the algorithms used, time was a negligible factor to observe. Each algorithm ran within 1 to 2 seconds, with the subsequent time given to conducting 10 fold cross validation.

As we introduced two different sets of data, one from our locally built drive, and the other from the live corpus body, the overall results were promising. As we expected, the classification completed from all three of our chosen algorithms was strong when run on our locally built test drive. We achieved 97% to 98% accuracy, with 1-Rule showing the highest accuracy. The attributes for which our tree decision algorithms branched were based upon fragstartsector, fragments, and lastly mtime.

When we applied the three algorithms to our second set of data across 5 hard drives, the results were comparably accurate to our local built drive, for J48 and 1-Rule. We achieved 92% to 99% accuracy. For all 5 drives, J48 proved to be the most accurate classification. The accuracy for IBL was much lower, with accuracy ranging from 72% to 98%. Applying a different number of nearest neighbors for our instance based learning algorithm gave us an appreciation for how accuracy is affected; for example as K increased, our accuracy slightly declined for all drives tested. In our experiment we ran 3 separate runs for each with 10, 5, and 1 nearest neighbors. Even though the IBL results were not as strong as the other two decision tree based algorithms, we believe that greater accuracy might be attained if attributes were weighted according to a level of importance.

Surprisingly, there were no apparent trends among accuracy for a particular algorithm used with respect to the size of the hard drive, the number of instances used, or the number of user profiles found.

Below are the experimental results for the locally built drive and the real corpus drives. They are broken down by the classification algorithm used, the number of instances on each drive, and the percentage of correctly and incorrectly classified. See Appendix C (WEKA output Results for all Experimental Runs).

4.1 Built Drive Data Results

Table 4.1 shows the results of testing the algorithms with the drive that was built in the laboratory. The 1-Rule algorithm performed the best, with a 98.77% accuracy, although improvement in accuracy compared with J48 (6 parts in 10,000) does not appear to be statistically significant.

Locally Built Test Drive									
	Instance Based Learning								
Classification Algorithm	k = 10	k = 5	k = 1	J48	1-Rule				
Number of Instances	4987	4987	4987						
Correctly Classified	98.43%	98.71%	98. 77 <i>%</i>						
Incorrectly Classified	2.06%	2.06% 1.74% 1.56% 1.28							

Table 4.1: Accuracy Results on Locally Built Data

4.2 Real Corpus Drive Data Results

Table 4.2 on the following page shows the results of the three classification algorithms applied to five drives from the Garfinkel Real Data Corpus. In each case the classification rate for the best performing algorithm has been printed in bold. Overall, J48 is the best performing algorithm.

Drive : cn3-03									
Instance Based Learning									
Classification Algorithm	k = 10	k = 5	k = 1	J48	1-Rule				
Number of Instances	3555	3555	3555	3555	3555				
Correctly Classified	97.60%	97.91%	98.28%	99.21%	98.42%				
Incorrectly Classified	2.31%	2.08%	1.71%	.78%	1.57%				
Drive : cn3-06									
	Instanc	e Based L	earning						
Classification Algorithm	k = 10	k = 5	k = 1	J48	1-Rule				
Number of Instances	744	744	744	744	744				
Correctly Classified	72.71%	75.80%	76.20%	95.56%	94.62%				
Incorrectly Classified	27.28%	24.19%	23.79%	4.43%	5.37%				
Drive : cn1-1-6									
	Instanc	e Based L	earning						
Classification Algorithm	k = 10	k = 5	k = 1	J48	1-Rule				
Number of Instances	16581	16581	16581	16581	16581				
Correctly Classified	87.32%	87.71%	87.3%	97.92%	97.17%				
Incorrectly Classified	12.67%	12.28%	12.67%	2.07%	2.82%				
	Drive	e : cn4-02							
	Instanc	e Based L	earning						
Classification Algorithm	k = 10	k = 5	k = 1	J48	1-Rule				
Number of Instances	1463	1463	1463	1463	1463				
Correctly Classified	77.23%	79.28%	83.25%	95.89 %	92.27%				
Incorrectly Classified	22.76%	20.71%	16.74%	4.10%	7.72%				
Drive : il03									
Instance Based Learning									
Classification Algorithm	k = 10	k = 5	k = 1	J48	1-Rule				
Number of Instances	17428	17428	17428	17428	17428				
Correctly Classified	92.67%	93.27%	93.06%	98.23%	97.75%				
Incorrectly Classified	7.32%	6.72%	6.9%	1.76%	2.24%				

Table 4.2: Accuracy Results on Live Corpus Drives

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5: Conclusion and Future Work

In conclusion, based upon our experimental results, we have demonstrated with reportable accuracy that it is possible to ascribe ownership of files of a multiuser hard drive, through the use of file metadata and several data mining classification algorithms. Though this thesis has only focused in on 3 different data mining algorithms, it would be beneficial to explore other algorithms which are packaged in WEKA.

Given more time, instead of using the basic Euclidean distance for the algorithm within WEKA, one could try altering the distance function to give more weight to different attributes. We believe in this case that greater accuracy would be achieved. In addition it would be beneficial to try different heuristics rules for determining ownership. Within our experiment, ownership was defined by parsing out different user profiles from the windows file system directory structure. Given a different heuristic, it would be interesting to apply this technique to other file systems other than that of Windows and to compare results.

This thesis utilized a basic set of metadata from the files found on each hard drive; for example, filesize, partition, and fragstartsector. It would be of great value to apply more feature extractors to our hard drive data which would give more attributes to use within our selection of algorithms. For example, one could run FIWALK with a new feature extractor that pulls out metadata from the new Microsoft Word file format DOCX.

Furthermore, based upon our results it would also be interesting to create an arff for each fragment on a disk as opposed to an entire drive; this potentially would give us a higher level of granularity of inspection. Lastly, it would be useful to potentially modify TSK to get user ownership information as it is very robust and much of FIWALK depends on TSK to run. THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A: Data Preparation

This appendix shows screen captures of the data preparation and cleaning done prior to execution of the classification algorithms. Removal of id and filename as attributes are done in these steps.

• • • • • Exp	lorer
Preprocess Classify Cluster As	sociate Select attributes Visualize
Open file Open URL Open DB Gene	erate) Undo Edit Save
Choose None	Apply
Current relation Relation: fiwalk Instances: 5057 Attributes: 11	Selected attribute Name: id Type: Numeric Missing: 0 (0%) Distinct: 5057 Unique: 5057 (100%)
Attributes All None Invert Pattern No. Name I V Id D partition D filesize C time D tim	Statistic Value Minimum 1 Maximum 5057 Mean 2529 StdDev 1459.974
6 atime 7 fragments 8 frag1startsector 9 frag2startsector 10 Wilename 11 userguess	Class: userguess (Str) Image: Class: Userguess (Str) Image: Visualize All 298 297
(Remove)	
Status OK	1 2529 5057

Figure A.1: This screen capture shows the selection of the id and filename attributes which are removed from the analysis.



Figure A.2: After the id and filename attributes are deleted, all of the string attributes need to be converted to nominal attributes in order to allow the extracted metadata to be analyzed by Weka.

APPENDIX B: The Weka Explorer

This Appendix shows screen captures of the Weka Explorer and various algorithm implementations.

Preprocess Classify Cluster Associate Select attributes Visualize assifier	⊖ ⊕		_	Explorer		
assifier	Preprocess	Classify	Cluster	Associate	Select attributes	Visualize
weka : Zerołk classifiers classifier output intuctions interaction interaction for output interaction interaction interaction i	Classifier					
classifiers b aves b meta b millication folds classifier output	wekase ZeroR					
start	classifiers					
Image: State	les 📦 🛅 bayes	Classifi	er output			
Filter Remove filter	functions					
Filter Remove filter Close	O Superiorest set					
Finisc For aptions Start Stop sult list (right-click for options) Filter Remove filter Close	meta					
Filter Remove filter	▶					
Filter Remove filter	Trees split					
Nom) userguess	rules options					
Nom) userguess B Start Stop ssult list (right-click for options) Filter Remove filter Close						
Start Stop sult list (right-click for options) Filter Remove filter Close						
Filter Remove filter Close						
Filter) Remove filter Close	Start Stop					
Filter Remove filter Close	Result list (right-click for options)					
Filter Remove filter Close						
Filter Remove filter Close						
Filter Remove filter Close						
Filter Remove filter Close						
us	Filter Remove filter	lose				
us						
us						
us						
us						
us						
	atus					
	K					

Figure B.1: This screen shot shows the various classifiers built into the Weka toolkit.

Wekase IBk - K I - W U - A "weka.co	
Classifiers	
bayes	
Image: Second	
Supplie lazy	
Cross Bk tion Folds 10	
O Perceneren Barbert % 66	
LWLe options meta	
N misc	
trees rules Stop	
Filter Remove filter Cl	ose

Figure B.2: This screen shot shows the selection of the J48 algorithm

	-
functions _C 0.25 -M 2	
▶ 🔄 lazy	- 11
Test 🕨 💼 meta	-Clas
○ ▶. □ mi ing set	
▶ 📄 misc	
🔿 🗣 📷 trees st set 🛛 🦲 Set	
Cross ADTree Folds 10 BFTree	
O Percen DecisionStump	
J48 options	
📄 J48graft	
(Nom) user (LMT s	
MSP	
Stall NBTree Stop	
tesult list 🗋 RandomForest	
REPTree	
SimpleCart	
UserClassifier	-
▶	-
P Tures	
Filter Remove filter Clos	e

Figure B.3: This screen shot shows the selection of the 1Rule algorithm

APPENDIX C: Test Run Results

This appendix shows several test run results.

Enclosed are all test runs for our locally built drive. In addition a test run for J48 for each hard drive.

Instance Based Learning with 10 nearest neighbors built drive

=== Run information === weka.classifiers.lazy.IBk -K 10 -W 0 -A Scheme: weka.core.neighboursearch.LinearNNSearch -A
\"weka.core.EuclideanDistance -R first-last\""
Relation: fiwalk-weka.filters.unsupervised. attribute.Remove-R1,10-weka.filters.unsupervised. attribute.StringToNominal-Clast-weka.filters.unsupervised. attribute.ReplaceMissingValues Instances: 5057 Attributes: 9 partition filesize mtime ctime atime fragments frag1startsector frag2startsector userquess 10-fold cross-validation Test mode: === Classifier model (full training set) === TB1 instance-based classifier using 10 nearest neighbour(s) for classification Time taken to build model: 0 seconds === Stratified cross-validation === === Summary === Correctly Classified Instances 97.9346 % 4884 Incorrectly Classified Instances Kappa statistic 103 2.0654 % 0.9755 0.0044 Mean absolute error Root mean squared error Relative absolute error 2.3656 % Root relative squared error Total Number of Instances 15.789 % 4987 Ignored Class Unknown Instances 70 === Detailed Accuracy By Class === TP Rate FP Rate Precision Recall F-Measure ROC Area Class 0.987 0.002 0.994 0.987 0.99 0.997 Administrator 0.988 0.96 0.003 0.979 0.991 0.988 0.96 0.983 0.975 All Users Default User 0.997 0.97 0.002 0.978 0.97 0.974 0.999 dph2007 0.97 0.983 0.003 0.972 0.941 0.97 0.983 0.971 0.961 0.998 0.999 hunter jjm2007 0.898 0 0.957 0.898 0.926 LocalService 1 0.001 0.727 0.842 1 NetworkService 0.986 0.986 0 0.993 1 simsong === Confusion Matrix === h 0 <-- classified as b С d 1 e 2 f 7 g 0 i 0 0 | 982 3 a = Administrator

6	557	0	0	0	1	0	0	0	b = All Users
С) 9	431	1	0	8	0	0	0	c = Default User
С) 0	4	481	1	10	0	0	0	d = dph2007
C) 0	0	6	523	10	0	0	0	e = hunter
С) 0	0	2	11	745	0	0	0	f = jjm2007
C) 0	0	0	0	1	44	4	0	g = LocalService
С) 0	0	0	0	0	0	16	0	h = NetworkService
С	0 0	0	1	1	10	2	2	1105	i = simsong

Instance Based Learning with 5 nearest neighbors built drive

```
=== Run information ===
Scheme:
                 weka.classifiers.lazy.IBk -K 5 -W 0 -A
 "weka.core.neighboursearch.LinearNNSearch -A
\"weka.core.EuclideanDistance -R first-last\""
                fiwalk-weka.filters.unsupervised.
Relation:
attribute.Remove-R1,10-weka.filters.unsupervised.
attribute.StringToNominal-Clast-weka.filters.unsupervised.
attribute.ReplaceMissingValues
Instances:
                 5057
Attributes:
                 9
                 partition
                 filesize
                 mtime
                 ctime
                 atime
                 fragments
                 frag1startsector
                 frag2startsector
                 userguess
                 10-fold cross-validation
Test mode:
=== Classifier model (full training set) ===
IB1 instance-based classifier
using 5 nearest neighbour(s) for classification
Time taken to build model: 0 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances
                                               4900
                                                                       98.2555 %
Incorrectly Classified Instances
Kappa statistic
                                                  87
                                                                        1.7445 %
                                                   0.9793
Mean absolute error
                                                   0.0041
Root mean squared error Relative absolute error
                                                   0.0461
                                                   2.1677
                                                            2
Root relative squared error
Total Number of Instances
Ignored Class Unknown Instances
                                                  15.0556 %
                                               4987
                                                            70
=== Detailed Accuracy By Class ===
           FP Rate
TP Rate
                        Precision
                                                                 ROC Area
                                       Recall
                                                F-Measure
                                                                             Class
  0.987
                                         0.987
                                                     0.993
0.994
                                                                   0.995
              0
                                                                               Administrator
                            1
               0.001
                            0.989
                                                                    0.997
                                                                               All Users
  0.973
0.97
              0.001
                            0.991
                                         0.973
                                                      0.982
                                                                               Default User
                                                                    1
                            0.991
0.98
0.981
               0.002
                                         0.97
                                                      0.975
                                                                    0.999
                                                                               dph2007
  0.972
               0.002
                                         0.972
                                                      0.977
                                                                    0.997
                                                                               hunter
                                                                               jjm2007
LocalService
  0.989
0.959
              0.011
                            0.941
0.904
                                         0.989
0.959
                                                     0.965
0.931
                                                                   0.999
               0.001
                                                                   0.999
  0.688
              0.001
                            0.786
                                         0.688
                                                      0.733
                                                                               NetworkService
                            0.998
                                         0.986
                                                      0.992
  0.986
              0.001
                                                                   1
                                                                               simsong
=== Confusion Matrix ===
     a
           b
                 С
                        d
                              e
                                    f
                                                h
                                                       i
                                                            <-- classified as
                                          0
0
0
  982
0
                                                               a = Administrator
b = All Users
           3
                 0
                       1
                              2
0
                                    7
                                                0
0
                                                       0
0
                                                          T
         563
                 0
                       0
                                    1
    0
           3
               437
                              0
                                    8
                                          0
0
                                                0
                                                       0
0
                                                                c = Default User
                        1
    0
           0
                 4
                     481
                              1
                                   10
                                                0
                                                               d = dph2007
    0
           0
                 0
                       5
                           524
                                   10
                                          0
                                                0
                                                       0
                                                               e = hunter
                                                               f = jjm2007
g = LocalService
    0
0
           0
                 0
                       2
0
                              6
                                  750
                                          0
                                                0
                                                       0
0
           0
                             0
                                         47
                 0
                                    1
                                                1
    0
                                    0
                                                       2
                                                               h = NetworkService
           0
                 0
                       0
                              0
                                          3
2
                                               11
                                                               i = simsong
    0
           0
                 0
                       1
                              1
                                   10
                                                2
                                                  1105
```

Instance Based Learning with 1 nearest neighbors built drive

```
=== Run information ===
                weka.classifiers.lazy.IBk -K 1 -W 0 -A
Scheme:
 "weka.core.neighboursearch.LinearNNSearch -A
  \"weka.core.EuclideanDistance -R first-last\""
               fiwalk-weka.filters.unsupervised.
Relation:
attribute.Remove-R1,10-weka.filters.unsupervised.
attribute.StringToNominal-Clast-weka.filters.unsupervised.
attribute.ReplaceMissingValues
                5057
9
Instances:
Attributes:
                partition
                 filesize
                mtime
                 ctime
                 atime
                 fragments
                 frag1startsector
                 frag2startsector
                 userquess
Test mode:
                10-fold cross-validation
=== Classifier model (full training set) ===
IB1 instance-based classifier
using 1 nearest neighbour(s) for classification
Time taken to build model: 0 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances
                                             4909
                                                                    98.4359 %
Incorrectly Classified Instances
                                                78
                                                                     1.5641 %
                                                 0.9815
Kappa statistic
Mean absolute error
                                                 0.0037
Root mean squared error
                                                 0.0454
                                                 1.9664 %
Relative absolute error
Root relative squared error
                                               14.81
                                                         8
Total Number of Instances
Ignored Class Unknown Instances
                                             4987
                                                         70
=== Detailed Accuracy By Class ===
                                                             ROC Area Class
0.995 Admin
0.997 All 0
TP Rate
           FP Rate Precision
                                      Recall F-Measure
                                                                           Administrator
All Users
  0.988
0.998
                                       0.988
                                                   0.994
0.996
              0
                           1
              0.001
                           0.993
  0.971
0.974
0.974
                                       0.971
0.974
0.974
              0
                                                   0.985
                                                                 0.993
                                                                           Default User
dph2007
              0.002
                           0.98
                                                   0.977
0.975
                                                                0.998
0.996
              0.003
                           0.976
                                                                           hunter
                           0.941
0.98
0.875
  0.988
              0.011
                                       0.988
                                                   0.964
                                                                0.998
                                                                           jjm2007
                                                   0.99
0.875
              0
                                       1
                                                                           LocalService
  1
                                                                 1
                                       0.875
  0.875
                                                                 0.957
                                                                           NetworkService
              0
  0.988
              0
                           0.999
                                       0.988
                                                   0.993
                                                                0.998
                                                                           simsong
=== Confusion Matrix ===
                                                         <-- classified as
    a
          b
                      d
                            е
                                  f
7
                                              h
                                                    i
                                        g
  983
                Õ
                            2
                                              0
                                                    0
                                                       I
                                                          a = Administrator
b = All Users
                                                    0
0
    0
        563
                0
                      0
                            0
                                  1
                                        0
                                              0
    0
          2
                       3
                            Õ
                                        Ő
                                              Ő
                                                            c = Default User
              436
                                  8
                                                            d = dph2007
e = hunter
f = jjm2007
    0
          0
                0
                    483
                             3
                                 10
                                        0
0
                                              0
                                                    0
0
                          525
7
    0
                                              0
          0
                      3
                                 11
    0
          Õ
                 0
                       2
                                749
                                        0
                                              0
                                                    0
    0
0
          0
                 0
                      0
                            0
                                  0
0
                                       49
                                             0
14
                                                    0
                                                            g = LocalService
h = NetworkService
          0
                0
                      0
                            0
                                        1
                                                    1
          0
                 0
                                 10
                                        0
                                              2
                                                1107
                                                             i = simsong
                       1
                            1
```

J48 built drive

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2 Relation: fiwalk-weka.filters.unsupervised.attribute. Remove-R1,10-weka.filters.unsupervised.attribute. StringToNominal-Clast-weka.filters.unsupervised.attribute. ReplaceMissingValues Instances: 5057 Attributes: 9 partition filesize

mtime ctime atime fragments frag1startsector frag2startsector userguess 10-fold cross-validation Test mode: === Classifier model (full training set) === J48 pruned tree frag1startsector <= 2381464 fraglstartsector <= 119192: Administrator (985.0)
fraglstartsector > 119192 frag1startsector <= 259656: All Users (563.0)
frag1startsector > 259656
frag1startsector <= 309752: Default User (440.0)</pre> frag1startsector > 309752 fragments <= 0
| mtime <= 1194329808000</pre> | mtime <= 1193969482000: dph2007 (55.0/44.0) | mtime > 1193969482000: simsong (4.0) mtime > 1194329808000 mtime <= 1197572476000: jjm2007 (5.0)
mtime > 1197572476000: hunter (3.0) fragments > 0: dph2007 (485.0) frag1startsector > 2381464 frag1startsector <= 2785248 | fraglstartsector <= 2459368: hunter (528.0) | fraglstartsector > 2459368: jjm2007 (745.0) fraglstartsector > 2785248 frag1startsector <= 2791872 fraglstartsector <= 2789736: LocalService (49.0)
fraglstartsector > 2789736: NetworkService (16.0) frag1startsector > 2791872: simsong (1109.0) Number of Leaves : 13 Size of the tree : 25 Time taken to build model: 0.18 seconds === Stratified cross-validation === === Summary === Correctly Classified Instances Incorrectly Classified Instances 4923 98.7167 % 64 1.2833 % 0.9848 Kappa statistic Mean absolute error 0.0026 Root mean squared error 0.0395 Relative absolute error 1.3679 % Root relative squared error Total Number of Instances 12.8945 % 4987 70 Ignored Class Unknown Instances === Detailed Accuracy By Class === TP Rate FP Rate Recall F-Measure Precision ROC Area Class 0.99 0.996 0.978 0.978 0.998 0.982 0.984 0.997 0.98 0.006 0.99 0.996 0.996 0.997 Administrator All Users 0.002 0.978 0.998 Default User 0.976 0.983 0.988 0.98 0.98 0.976 0.983 0.988 0.96 0.987 0.996 0.968 0.985 0.992 0.004 0.998 dph2007 0.995 hunter jjm2007 LocalService 0.001 0.999 0.98 0 0.98 0.98 0.99 0.969 0.938 0 0.938 NetworkService 0.991 0 0.999 0.991 0.995 0.999 simsong === Confusion Matrix === d i <-- classified as а h C е f g 0 h a = Administrator b = All Users 985 2 3 0 | 0 0 1 4 1 562 0 0 0 0 0 0 c = Default User d = dph2007 439 4 0 5 1 0 0 0 0 1 9 Õ 1 484 2 Ő Ő Ő 0 e = hunter f = jjm2007 g = LocalService 4 0 0 4 530 1 0 0 0 0 749 2 3 0 0 3 0 1 0 0 0 0 0 0 48 1 0 - İ h = NetworkService
i = simsong 15 1 | 0 1111 | 0 2 0 0 0 0 0 2 0 0 Ő 4 1 1

1-Rule built drive

=== Run inform	nation ===			
Scheme: weka. Relation: fiw attribute.Rem attribute.Str attribute.Rep Instances: Attributes:	classifiers.rules. alk-weka.filters.u ove-R1,10-weka.fil ingToNominal-Clast laceMissingValues 5057 9	OneR -B 6 nsupervised. ters.unsupervised. -weka.filters.unsup	ervised.	
	partition filesize mtime ctime atime			
	frag1startsector frag2startsector userguess			
Test mode:	10-fold cross-val	idation		
fraglatartage	r model (lull tral	ning set) ===		
Iragistartsec < 119252.0 -> < 259688.0 -> < 309788.0 -> < 1464810.7122 < 1490690.7122 < 2381500.0 -: < 2789760.0 -: < 2789760.0 -: < 2791900.0 -: > 2791900.0 -: < 2791900.0 -: < 14933/4987 in:	<pre>cor: Administrator All Users Default User 3840258 -> dph2007 3840258 -> jjm2007 > dph2007 > hunter > jjm2007 > LocalService > NetworkService -> simsong stances correct)</pre>			
Time taken to	build model: 0.04	seconds		
=== Stratified === Summary ==	d cross-validation ==	_ ===		
Correctly Clas Incorrectly C. Kappa statist Mean absolute Root mean squa Relative abso. Root relative Total Number Ignored Class	ssified Instances lassified Instance error ared error lute error squared error of Instances Unknown Instances	4926 s 61 0.9855 0.0027 0.0521 1.4491 % 17.0252 % 4987 70	98.7768 % 1.2232 %	
=== Detailed A	Accuracy By Class	===		
TP Rate FP 1 0.99 0 0.998 0 0.98 0 0.98 0 0.98 0 0.98 0 0.988 0 1 0 1 0 0.99 0	Rate Precision 1 1 .003 0.97 1 .009 0.952 1 .002 0.993	Recall F-Measure 0.99 0.995 0.998 0.999 0.98 0.99 0.98 0.975 0.98 0.99 0.98 0.99 1 1 0.99 0.992	ROC Area Class 0.992 Admini 0.998 All Us 0.99 Defaul 0.987 dph200 0.997 hunter 0.987 jjm200 1 Locals 1 Networ 0.994 simsor	strator sers t User 7 Service ckService
=== Confusion	Matrix ===			
a b 985 0 0 563 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	$\begin{array}{cccc} & d & e & f \\ 0 & 2 & 0 & 6 \\ 0 & 1 & 0 & 0 \\ 40 & 2 & 0 & 7 \\ 0 & 486 & 0 & 9 \\ 0 & 2 & 528 & 9 \\ 0 & 4 & 0 & 749 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0$	g h i <- 0 0 2 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 49 0 0 0 16 0 0 0 1110	<pre>- classified as a = Administrator b = All Users c = Default User d = dph2007 e = hunter f = jjm2007 g = LocalService h = NetworkService i = simsong</pre>	2

J48 / cn3-06

```
=== Run information ===
Scheme:
                 weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:
                  fiwalk-weka.filters.unsupervised.attribute.Remove-R1,10-weka
.filters.unsupervised.attribute.StringToNominal-Clast-weka.filters.unsupervised.
attribute.ReplaceMissingValues
                16828
9
Instances:
Attributes:
                  partition
filesize
                  mtime
                  ctime
                  atime
                  fragments
                  frag1startsector
                  frag2startsector
                  userquess
Test mode:
                  10-fold cross-validation
=== Classifier model (full training set) ===
J48 pruned tree
ctime <= 1042520394000
     atime <= 1042519839000
          mtime <= 1107691738000
          mtime <= 110/691/38000
| filesize <= 397824: Default User (94.0/3.0)
| filesize > 397824: All Users (2.0)
mtime > 1107691738000
| ctime <= 1042519499000: Default User (12.0)
| ctime > 1042519499000: All Users (22.0)
     atime > 1042519839000
        ctime <= 1042519845000: All Users (126.0/2.0)
ctime > 1042519845000: Default User (15.0/1.0)
ctime > 1042520394000
     atime <= 1042520496000
          atime <= 1042520475000: NetworkService (9.0)
atime > 1042520475000: LocalService (12.0)
     atime > 1042520496000
          ctime <= 1042520586000
| atime <= 1042528338000: Administrator (424.0/1.0)
                atime > 1042528338000
                    atime <= 1042528399000: Administrator (5.0)
atime > 1042528399000
          | | fraglstartsector <= 478401: NetworkService (2.0)
| | fraglstartsector > 478401: LocalService (2.0)
ctime > 1042520586000
               mtime <= 1042521286000: All Users (3.0/1.0)
mtime > 1042521286000: Administrator (16.0/6.0)
Number of Leaves : 14
Size of the tree : 27
Time taken to build model: 0.05 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances
                                                   711
                                                                           95.5645 %
Incorrectly Classified Instances
Kappa statistic
                                                    33
                                                                            4.4355 %
                                                      0.9238
Mean absolute error
                                                      0.0212
Root mean squared error
                                                      0.1254
Relative absolute error
                                                      9.0814
Root relative squared error
                                                     36.7288 %
                                                   744
Total Number of Instances
Ignored Class Unknown Instances
                                                           16084
=== Detailed Accuracy By Class ===
                                                                                  Class
TP Rate
            FP Rate
                         Precision Recall F-Measure
                                                                    ROC Area
                              0.982
0.973
0.927
                                                        0.984
0.954
0.943
                                                                    0.928
  0.986
0.935
               0.026
                                           0.986
0.935
                                                                                  Administrator
               0.007
                                                                       0.797
                                                                                   All Users
  0.958
               0.014
                                           0.958
                                                                       0.84
                                                                                   Default User
                              0.688
                                                                       0.932
  0.647
0.571
               0.007
                                           0.647
                                                         0.667
                                                                                   LocalService
               0.01
                                           0.571
                                                                       0.899
                                                                                 NetworkService
=== Confusion Matrix ===
   a b c d e <-- classified as
32 0 0 3 3 | a = Administrator
2 145 8 0 0 | b = All Users
1 4 115 0 0 | c = Default User
 a
432
       4 115
0 1
                              c = Default User
d = LocalService
   1
1
```

1 11

4 |

```
42
```

J48 / cn4-02

=== Run information === weka.classifiers.trees.J48 -C 0.25 -M 2 Scheme: Relation: fiwalk-weka.filters.unsupervised.attribute.Remove-R1,10-weka .filters.unsupervised.attribute.StringToNominal-Clast-weka.filters.unsupervised. attribute.ReplaceMissingValues Instances: 48668 9 Attributes: partition filesize mtime ctime atime fragments frag1startsector frag2startsector userguess Test mode: 10-fold cross-validation === Classifier model (full training set) === J48 pruned tree partition <= 1 mtime <= 912179706000: vg2000 (205.0)
mtime > 912179706000 ctime <= 1169436556000 frag1startsector > 1367631 fraglstartsector <= 1368519: All Users (35.0/1.0)
fraglstartsector > 1368519 fraglstartsector <= 3685511: abc (211.0)
fraglstartsector > 3685511 ctime <= 915229332000 ctime <= 915208320000 | ctime <= 915206666000: All Users (3.0/1.0) | ctime > 915206666000: abc (98.0/2.0) ctime > 915208320000 | ctime <= 915210662000 ctime <= 915210432000 | ctime <= 915210084000 atime <= 915247580000 | atime <= 915247580000
| | fragments <= 0
| | ctime <= 915208786000: abc (5.0)
| | ctime > 915208786000
| | | mtime <= 915209756000
| | | | mtime <= 91520942000: All Users (15.0/1.0)
| | | | ctime > 91520942000: abc (3.0)
| | | | mtime > 915209756000: All Users (13.0)
| | atime > 91521084000: All Users (25.0/1.0)
| ctime > 915210084000: All Users (42.0)
me > 915210432000: All Users (42.0) ctime > 915210662000 ctime <= 915214218000: abc (100.0) ctime > 915214218000 ctime <= 915214490000: All Users (11.0/1.0) ctime > 915214490000 | ctime <= 915229124000 | fraglstartsector <= 4207415: All Users (2.0) | fraglstartsector > 4207415: abc (24.0/1.0) ctime > 915229124000: All Users (6.0) ctime > 915229332000 ctime <= 983731024000: abc (98.0/1.0) ctime > 983731024000 ctime <= 1054499964000 | atime <= 947520000000: All Users (7.0) | atime > 947520000000: Default User (3.0) ctime > 1054499964000: abc (9.0) ctime > 1169436556000 ctime <= 1169436926000: Administrator (71.0) ctime > 1169436926000: abc (5.0/1.0)

```
partition > 1
| ctime <= 916059349000
        ctime <= 916059148000: ttt (25.0)
ctime > 916059148000
             filesize <= 74
             filesize <= 10896
                       filesize <= 498
| mtime <= 915159756000
| | atime <= 915315678000: Default User.WINDOWS (3.0)
                                atime > 915315678000
| atime <= 915552000000: All Users.WINDOWS (8.0/1.0)
| atime > 915552000000: Default User.WINDOWS (3.0/1.0)
                       | mtime > 915159756000: All Users.WINDOWS (36.0)
filesize > 498: All Users.WINDOWS (123.0)
                  filesize > 10896
                      ctime < 916059348000: All Users.WINDOWS (3.0)
ctime > 916059348000
                           filesize <= 389120: Default User.WINDOWS (5.0)
filesize > 389120: All Users.WINDOWS (4.0/1.0)
    ctime > 916059349000
         ctime <= 916059358000
             mtime <= 1000572602000
                 mtime <= 915150499000
                      mtime <= 915149576000
                        atime <= 915315678000: LocalService (11.0/4.0)
atime > 915315678000: Default User.WINDOWS (2.0/1.0)
                       mtime > 915149576000: Default User.WINDOWS (18.0)
                  mtime > 915150499000
                       filesize <= 55464
                          atime <= 915151260000: NetworkService (12.0/2.0)
                           atime > 915151260000
             ctime > 916059358000: Default User.WINDOWS (55.0)
Number of Leaves : 46
Size of the tree : 91
Time taken to build model: 0.31 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances
                                           1403
                                                                 95.8988 %
Incorrectly Classified Instances
                                              60
                                                                   4.1012 %
                                               0.9458
Kappa statistic
Mean absolute error
                                               0.0106
Root mean squared error
                                               0.0864
Relative absolute error
                                               6.9944
Root relative squared error
                                              31.3723 %
Total Number of Instances
                                           1463
Ignored Class Unknown Instances
                                                   47205
=== Detailed Accuracy By Class ===
TP Rate
           FP Rate
                      Precision Recall F-Measure
                                                            ROC Area Class
                                                            0.988
                                                                        Default User
  0.956
0.894
            0.001
                          0.977
0.911
                                                 0.966
0.903
                                     0.956
             0.011
                                      0.894
                                                              0.96
                                                                        All Users
  0.979
             0.022
                          0.972
                                      0.979
                                                 0.975
                                                              0.948
                                                                        abc
vg2000
  0.995
             0
                                                 0.998
                                                              0.993
              0
                                                                        Administrator
  0.989
                                      0.989
                                                 0.965
             0.009
                          0.941
                                                              0.986
                                                                        All Users.WINDOWS
Default User.WINDOWS
                          0.941
0.99
0.5
0.75
                                                 0.937
0.581
                                                              0.746
             0.001
                                      0.89
  0.89
                                                                        LocalService
  0.692
             0.006
                                      0.692
                                                              0.881
                                                 0.692
  0.643
             0.002
                                      0.643
                                                              0.891
                                                                        NetworkService
                                      0.886
                          0.912
                                                 0.899
                                                              0.786
  0.886
             0.002
                                                                        ttt
=== Confusion Matrix ===
                                                 <-- classified as
   а
       b
            С
                 d
                              g
                                   h
                                        i
                                            j
O I
                                        0
  43
        1
            1
                0
                     0
                          0
                                   0
                                                  a = Default User
b = All Users
   1 144 16
                0
                      Ő
                          Õ
                              Ő
                                   Õ
                                        Ő
                                             Ő
                                               i
   0
     13 617
                 Ω
                      0
                          0
                              0
                                   0
                                        0
                                             0 |
                                                   c = abc
d = vq2000
            1 205
   0
                          0
                              0
                                   0
                                        0
                                             0
       0
                      0
                                               1
   0
        0
            0
                 0
                    71
                          0
                               0
                                   0
                                        0
                                             0
                                                   e = Administrator
                                               Ì
                             1
97
                                                   f = All Users.WINDOWS
g = Default User.WINDOWS
   0
       Ω
            Ω
                 0
                      0 177
                                   0
                                        0
                                             1
   0
                      Ő
                                        0
                                             οi
                 Ō
        0
            0
                         11
                                   1
   0
        0
            0
                 0
                      0
                          0
                              0
                                   9
                                        2
                                             2 |
                                                   h = LocalService
   0
        0
            0
                 0
                      0
                          0
                              0
                                   5
3
                                        9
                                             0 |
                                                   i = NetworkService
   0
                 0
                      õ
                          Ő
                               Ő
        Õ
            Õ
                                            31 |
                                                    j = ttt
                                        1
```

J48 / cn1-1-6

=== Run information === weka.classifiers.trees.J48 -C 0.25 -M 2 Scheme: Relation: fiwalk-weka.filters.unsupervised.attribute.Remove-R1,10-weka. $\verb+ filters.unsupervised.attribute.StringToNominal-Clast-weka.filters.unsupervised.\\$ attribute.ReplaceMissingValues Instances: 45510 Attributes: 9 partition filesize mtime ctime atime fragments frag1startsector frag2startsector userguess 10-fold cross-validation Test mode: === Classifier model (full training set) === J48 pruned tree atime <= 1172246400000 mtime <= 1160963874000 fraglstartsector <= 3065191
fraglstartsector <= 1996815: All Users (32.0/1.0)
fraglstartsector > 1996815: Administrator (25.0/1.0) frag1startsector > 3065191 fraglstartsector <= 4623263 | fragments <= 0 | atime <= 1162134000000 me <= 1152134000000
ctime <= 1160963826000
| ctime <= 1150910388000: Default User (47.0)
| ctime > 1150910388000: Administrator (13.0/2.0)
ctime > 1160963826000: Default User (83.0) atime > 1162134000000
| filesize <= 105643: All Users (3.0/1.0)
| filesize > 105643: Administrator (4.0) mtime > 1160963874000
| mtime <= 1172117622000: Administrator (9163.0/27.0)
| mtime > 1172117622000 fragments <= 0 ctime <= 1172118500000 mtime <= 1172117882000: LocalService (38.0)
mtime > 1172117882000 mtime <= 1172118374000 | filesize <= 81 | Illesize <- 01
| filesize <= 21: Administrator (2.0)
| filesize > 21: LocalService (4.0)
| filesize > 81: Administrator (33.0/1.0)
ntime > 1172118374000: Administrator (227.0) ctime > 1172118500000 ctime <= 1172132404000 | filesize <= 0 mtime <= 1172125866000: LocalService (9.0/1.0) | mtime > 1172125866000: Administrator (8.0/2.0)
filesize > 0: LocalService (267.0/6.0) ctime > 1172132404000 ctime <= 1172136282000 | filesize <= 1681 mtime <= 1172136056000
| mtime <= 1172132932000: Administrator (4.0)
| mtime > 1172132932000: LocalService (14.0) | mtime > 1172132052000. hotalservice (1
| mtime > 1172136056000: Administrator (8.0)
filesize > 1681
| ctime <= 1172133592000</pre> ctime <= 1172132946000: Administrator (46.0) ctime > 1172132946000

ctime <= 1172133050000: LocalService (3.0) ctime > 1172133050000 ctime > 1172133338000: Administrator (19.0/2.0) ctime > 1172133338000: LocalService (3.0) ctime > 1172133592000: Administrator (98.0) ctime > 1172136282000 ctime <= 1172227448000 ctime <= 1172223658000: LocalService (193.0/16.0) ctime > 1172223658000 mtime <= 117223690000
| filesize <= 985: NetworkService (15.0)
| filesize > 985: LocalService (2.0) mtime > 1172223690000 filesize <= 6: Administrator (3.0/1.0)
filesize > 6: LocalService (57.0/1.0) ctime > 1172227448000 | mtime <= 1172323236000 filesize <= 0 ctime <= 1172228028000: LocalService (3.0)
ctime > 1172228028000: Administrator (124.0/18.0) filesize > 0 | atime <= 1172160000000 filesize <= 86 mtime <= 1172230136000: Administrator (5.0/1.0)
mtime > 1172230136000: LocalService (15.0/2.0) filesize > 86 mtime <= 1172228032000 ctime <= 117222052000 ctime <= 1172227576000: Administrator (11.0) ctime > 1172227576000: LocalService (14.0) mtime > 1172228032000 filesize <= 662 initial = 002
i filesize <= 397: Administrator (9.0)
i filesize > 397: LocalService (7.0/1.0)
filesize > 662: Administrator (46.0/1.0) atime > 1172160000000 mtime <= 1172316560000: LocalService (159.0/2.0)
mtime > 1172316560000 mtime <= 1172319208000 filesize <= 46: LocalService (3.0)
filesize > 46: Administrator (18.0/5.0) http://www.states.com/instrator (10.0/5. | | | mtime > 1172319208000: LocalService (7.0) mtime > 1172323236000 filesize <= 66: Administrator (89.0/4.0) filesize > 66 | ctime <= 1172326606000: Administrator (27.0/2.0) ctime > 1172326606000 mtime <= 1172329832000: LocalService (13.0/1.0)
mtime > 1172329832000: Administrator (9.0/1.0) ragments > 0 frag1startsector <= 52751 mtime <= 1172164580000: LocalService (21.0/1.0) mtime > 1172164580000 frag1startsector <= 29079 fragments <= 2 fragments <= 1 frag1startsector <= 27263 mtime > 1172325250000: LocalService (7.0)
mtime > 1172325250000: Administrator (2.0) frag1startsector > 27263 | TragIstartsector > 2/263
| fragIstartsector <= 28831: Administrator (13.0)
| fragIstartsector >= 28831
| | filesize <= 1739: LocalService (4.0)
| filesize <= 1739: Administrator (2.0)
fragments > 1: Administrator (2.0) fragments > 2: LocalService (4.0) fraglstartsector > 29079
fraglstartsector <= 51519: Administrator (85.0/1.0)
fraglstartsector > 51519
fraglstartsector > 51519 mtlme <= 11/2502500000 {
 atime <= 11/25000000000: Administrator (3.0/1.0)
 atime > 1172160000000: LocalService (3.0)
 mtime > 1172302588000: Administrator (4.0) fraglstartsector > 52751
filesize <= 392
f ctime <= 117228028000</pre> fraglstartsector <= 7381711: Administrator (58.0/2.0) fraglstartsector > 7381711 ctime <= 1172224368000 mtime <= 1172132730000 filesize > 392 mtime <= 1172331844000: Administrator (1768.0/4.0)
mtime > 1172331844000

```
ctime <= 1150910398000: LocalService (2.0)
                                           ctime > 1150910398000
                                                 fraglstartsector <= 2843447: All Users (2.0/1.0)
fraglstartsector > 2843447: Administrator (21.0)
a \pm ime > 1172246400000
     ctime <= 1178936684000
            mtime <= 1172107720000
                 ctime <= 1150910346000: All Users (60.0/1.0)
                  ctime > 1150910346000
                        mtime > 1150911424000: Administrator (34.0)
mtime > 1150911424000
                               filesize <= 1246
                                     filesize <= 155
                                          mtime <= 1160963952000: LocalService (4.0/1.0)
mtime > 1160963952000: Administrator (11.0)
                                     filesize > 155
| mtime <= 1169510400000
                                                 fraglstartsector <= 2005911
| mtime <= 1157054084000: All Users (16.0)
| mtime > 1157054084000: Administrator (5.0)
                                                  frag1startsector > 2005911
| mtime <= 1161181770000
                                                       | fraglstartsector <= 10409599: Administrator (35.0)
| fraglstartsector > 10409599: All Users (3.0)
mtime > 1161181770000: All Users (12.0/1.0)
                                                    > 1169510400000: Administrator (28.0/1.0)
                                           mtime
                               filesize > 1246
                                     filesize <= 165888: All Users (28.0/4.0)
                                     filesize > 165888: LocalService (2.0/1.0)
            mtime > 1172107720000
                  ctime <= 1171783424000
                         ctime <= 1150910398000
                  | ctime <= 1150910346000: NetworkService (4.0)
| ctime > 1150910346000: LocalService (5.0)
| ctime > 1150910398000: Administrator (42.0/3.0)
ctime > 1171783424000
| filesize <= 45592</pre>
                               mtime <= 1178934240000
                                    ctime <= 1178930304000
| ctime <= 1178927990000
                                                 mtime <= 1178882602000
| mtime <= 1174219294000</pre>
                                                              frag1startsector <= 9995607
                                                                    frag1startsector <= 3711295
| fragments <= 1
                                                                                 atime <= 1174140000000
                                                                                       fraglstartsector <= 3700479: LocalService (11.0/1.0)
fraglstartsector > 3700479: Administrator (2.0)
                                                                                 atime >
                                                                                             1174140000000
                                                                                       mtime > 11/4140000000: Administrator (4.0/1.0)
mtime > 1172325748000: LocalService (3.0)
                                                              | fragments > 1: Administrator (2.0)
| fraglstartsector > 3711295: LocalService (116.0/5.0)
fraglstartsector > 9995607
| mtime <= 11721228840000: LocalService (2.0)</pre>
                                                                    mtime > 1172122884000
                                                                    | mtime > 11/212/834000: All Users (4.0/1.0)
| mtime > 11/2127816000: Administrator (3.0)
11/4219294000: All Users (7.0)
                                                        mtime >
                                                  mtime > 1178882602000
                                                        ctime <= 1178925910000: LocalService (290.0)
ctime > 1178925910000
                                                              ctime <= 1178925938000
| ctime <= 1178925930000
                                                                          mtime <= 1178925932000
                                                                             mtime <= 1178925928000: NetworkService (17.0/3.0)
mtime > 1178925928000: LocalService (6.0)
                                                                           mtime > 1178925932000
                                                                             mtime <= 1178928768000: All Users (3.0)
mtime > 1178928768000: NetworkService (2.0)
                                                                    ctime > 1178925930000: Administrator (6.0)
                                                              ctime > 1178925938000: LocalService (227.0)
                                           ctime > 1178927990000
                                                  filesize <= 116: LocalService (5.0)
filesize > 116
    filesize <= 311: Administrator (9.0)
    filesize > 311: LocalService (7.0/1.0)

                                                  frag1startsector > 311095
| filesize <= 109
                                                             esize <= 109
ctime <= 1178928906000: Administrator (3.0)
ctime > 1178928906000: LocalService (21.0/2.0)
                                                        filesize > 109
```

filesize > 299 filesize <= 520: LocalService (6.0) filesize > 520 | | | | | | | | | mtime <= 1178929736000: Administrator (23.0/1.0)
| | | | | | | mtime > 1178929736000: LocalService (5.0/2.0)
| | | | frag1startsector > 8429119: LocalService (4.0)
ctime > 1178930304000: LocalService (939.0/1.0) mtime > 1178934240000 mtime <= 1178935390000 ctime <= 1178934122000: All Users (6.0/1.0)
ctime > 1178934122000: Administrator (34.0) mtime > 1178935390000 mtime > 1178937430000: LocalService (45.0/2.0)
mtime > 1178937430000: Administrator (4.0/1.0) filesize > 45592 fraglstartsector <= 253191 | mtime <= 1178929976000: LocalService (10.0) | mtime > 1178929976000 mtime <= 1178931084000: Administrator (3.0)
mtime > 1178931084000: LocalService (12.0) frag1startsector > 253191 ctime <= 1178927990000 clime <- 11/052/050000
| frag2startsector <= 8974703: LocalService (11.0/2.0)
| frag2startsector > 8974703: All Users (2.0)
ctime > 1178927990000 mtime <= 1178929698000: Administrator (11.0)
mtime > 1178929698000 filesize <= 78975 frag2startsector <= 6663743
| filesize <= 46043: Administrator (3.0)
| filesize > 46043 filesize <= 47916: All Users (9.0) filesize > 47916 fragments <= 1
| mtime <= 1178929734000: Administrator (2.0)</pre> mtime > 1178929734000
| filesize <= 52953</pre> | mtime <= 1178930270000: All Users (2.0)
| mtime > 1178930270000: Administrator (2.0)
filesize > 52953: All Users (5.0) | | fragments > 1: Administrator (4.0)
frag2startsector > 6663743: Administrator (3.0/1.0) filesize > 78975: LocalService (3.0) ctime > 1178936684000 ctime <= 1178937968000 mtime <= 1178937026000
| mtime <= 1178937020000: Administrator (54.0)
| mtime > 1178937020000 | ctime <= 1178937020000: LocalService (5.0)
| ctime > 1178937020000: Administrator (3.0/1.0)
mtime > 1178937026000: Administrator (303.0) ctime > 1178937968000 ctime <= 1178939192000 ctime <= 1178939158000 ctime <= 1178938160000: LocalService (22.0)
ctime > 1178938160000 fragments <= 1: All Users (32.0) fragments > 1 ctime <= 1178938574000: Administrator (3.0) ctime > 1178938574000: All Users (4.0) ctime > 1178939158000: LocalService (28.0) ctime > 1178939192000 filesize <= 846 | ctime <= 1178974642000 filesize <= 2: Administrator (2.0) filesize > 2 filesize <= 3492 filesize <= 1045: All Users (121.0) filesize > 1045 | | filesize <= 1163: Administrator (6.0)
| | filesize > 1163: All Users (77.0/1.0)
filesize > 3492: Administrator (17.0/1.0) Number of Leaves : 152 Size of the tree : 303 Time taken to build model: 1.54 seconds === Stratified cross-validation === === Summary ===

Correctly Incorrect Kappa sta Mean abso Root mean Relative Root rela Total Num Ignored C	Class ly Cla tistic lute e asquar absolu tive s ber of lass U	ified ssifi ed en te en quare Inst nknov	l Insta ed Ins ror fror ed erro ances an Inst	ances stance or cances	s	16	237 344 0.9 0.0 0.0 7.8 32.5 581	41 111 865 753 % 597 % 28929		97.92 2.0	253 % 747 %
=== Detai	led Ac	curad	су Ву (Class	===						
TP Rate 0.981 0.872 0.7 0.956 0.989	FP Ra 0 0.0 0 0.0 0.0	te 03 08 49	Preci 0. 0. 0. 0.	sion 981 906 375 957 987	Reca 0.9 0.8 0.7 0.9 0.9	11 81 72 56 89	F-Me 0 0 0 0	asure .981 .888 .778 .957 .988	ROC A 0.9 0.8 0.9 0.9 0.7	rea 52 77 01 77 32	Class Default User All Users NetworkService LocalService Administrator
=== Confu	sion M	atri>	x ===								
a 260 1 0 4	b 1 442 3 5 37	C 0 28 4 0	d 9 6 2606 101	e 4 55 3 111 12901	< 	cla a = b = c = d = e =	assif = Def = All = Net = Loc = Adm	ied as ault Use Users workSer alServi inistra	er vice ce tor		

J48 / cn3-03

```
=== Run information ===
                          weka.classifiers.trees.J48 -C 0.25 -M 2
fiwalk-weka.filters.unsupervised.attribute.Remove-R1,10-weka.filters.
Scheme:
Relation:
unsupervised.attribute.String {\tt ToNominal-Clast-weka.filters.unsupervised.attribute.}
ReplaceMissingValues
                          26775
Instances:
                         9
Attributes:
                          partition
filesize
                          mtime
                          ctime
                           atime
                          fragments
frag1startsector
                           frag2startsector
                          userquess
Test mode:
                          10-fold cross-validation
=== Classifier model (full training set) ===
J48 pruned tree
ctime <= 1085981614000
      frag1startsector <= 3447143
| frag1startsector <= 3406679: Default User (136.0)
| frag1startsector > 3406679
| frag1startsector > 3413127: All Users (34.0)
| frag1startsector > 3413127
| | mtime <= 1045227608000: LocalService (4.0)
| mtime > 1045227608000: Administrator (9.0)
frag1startsector > 3447143
| frag1startsector <= 3476887
| frag1startsector <= 3450967: Default User.WINDOWS (53.0)
| frag1startsector > 3476887
| ctime <= 1045229760000
| | ctime <= 1045229760000</pre>
       frag1startsector <= 3447143
                             ctime <= 1045229760000
| fragments <= 0
                                      | ctime <= 1045228224000: Default User (2.0)
| ctime > 1045228224000: Default User.WINDOWS (3.0)
fragments > 0
                                             atime <= 1060264800000: Administrator.IMAGE (2.0)
atime > 1060264800000: All Users.WINDOWS (13.0)
                              ctime > 1045229760000
                                     ctime <= 1045229906000: NetworkService.NT AUTHORITY (10.0)
ctime > 1045229906000: LocalService.NT AUTHORITY (21.0/4.0)
```

```
atime <= 1060264800000
           frag1startsector <= 3417543
           inagistattsector <= 341/543
i fraglstartsector <= 3405639: Default User (42.0)
i fraglstartsector > 3405639: All Users (126.0)
fraglstartsector > 3417543
c time <= 1162921266000
i fraglstartsector > 3417543
                   me <= 1162921260000
fragments <= 0: Default User (2.0)
fragments > 0
| ctime <= 1162921250000: NetworkService (5.0)
| ctime > 1162921250000: LocalService (5.0)
                ctime > 1162921266000: Administrator (54.0)
     atime > 1060264800000
          atime <= 1119448800000: zsnd (2493.0/2.0)
          atime > 1119448800000
| atime <= 1173880800000
                   frag1startsector <= 3420727
                     | fraglstartsector <= 3418119: All Users (59.0)
| fraglstartsector > 3418119: LocalService (6.0/1.0)
                | frag1startsector > 3420727: Administrator (128.0/1.0)
atime > 1173880800000
                   frag1startsector <= 3483159
                     | mtime <= 1149667146000: zsnd (4.0)
| mtime > 1149667146000: All Users.WINDOWS (3.0/1.0)
frag1startsector > 3483159: zsnd (164.0/1.0)
Number of Leaves : 26
Size of the tree : 51
Time taken to build model: 0.22 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances
                                                  3527
                                                                           99.2124 %
Incorrectly Classified Instances
Kappa statistic
                                                    28
                                                                            0.7876 %
                                                      0.9816
Mean absolute error
                                                      0.0017
                                                      0.0354
Root mean squared error
Relative absolute error
                                                      2.2223 %
Root relative squared error
Total Number of Instances
                                                     17.8948 %
                                                  3555
Ignored Class Unknown Instances
                                                           23220
=== Detailed Accuracy By Class ===
TP Rate
            FP Rate
                         Precision Recall F-Measure
                                                                    ROC Area
                                                                                  Class
                                                                     0.842
              0
                             1
0.995
                                                       0.986
  0.973
                                           0.973
                                                                                   Default User
  0.995
                                           0.995
                                                         0.995
                                                                       0.997
                                                                                   All Users
  0.833
0.929
               0.001
                              0.625
0.867
                                                         0.714
0.897
                                                                       0.902
0.964
                                                                                   NetworkService
                                           0.833
               0.001
                                            0.929
                                                                                   LocalService
                              0.99
0.948
0.962
  0.995
               0.001
                                           0.995
                                                         0.992
                                                                       0.978
                                                                                   Administrator
                                           0.982
                                                         0.965
0.957
                                                                       0.983
  0.982 0.953
                                                                                   Default User.WINDOWS
All Users.WINDOWS
               0.001
                0.001
                              0.9
0.762
0.989
                                           0.643
0.941
0.967
                                                                                   NetworkService.NT AUTHORITY
  0.643
               0
                                                         0.75
                                                                       0.999
  0.941
0.967
                                                         0.842
                                                                       0.999
               0.001
                                                                                   LocalService.NT AUTHORITY
                Õ
                                                                                   Administrator.IMAGE
               0.007
  0.998
                              0.998
                                           0.998
                                                         0.998
                                                                       0.834
                                                                                   zsnd
=== Confusion Matrix ===
                                                          i
                                                                       k
                                                                            <-- classified as
           b
                         d
                                      f
                                                   h
     a
                                е
                                             g
0
0
                  С
                                                                 i
                                                                       0 | a = Default User
0 | b = All Users
  177
                  2
                         0
                               0
                                      2
0
                                                          0
0
                                                                 õ
                                                   0
     0
         218
                  0
                         0
                                1
                                                   0
                                                                 0
     0
                                                          Ō
                                                                       0 |
                                                                                 c = NetworkService
           0
                                0
                                      0
                                             0
                                                    0
                                                                 0
     0
           0
                  1
                       13
                                0
                                      0
                                             0
                                                   0
                                                          0
                                                                 0
                                                                       0 i
                                                                                 d = LocalService
                                                          0
                                                                                 e = Administrator
     0
                         0
                             189
                                             0
                                                   0
                                                                       0 1
           0
                  0
                                                                 0
                                      1
     0
            0
                  0
                         0
                                0
                                     55
                                                    0
                                                          0
                                                                 0
                                                                        0
                                                                                 f = Default User.WINDOWS
                                          101
                                                          0
                                                                                g = All Users.WINDOWS
h = NetworkService.NT AUTHORITY
     0
           0
                  0
                         0
                               0
                                      0
                                                   0
                                                                 0
                                                                       5 I
                                                   9
     0
                         0
                                0
                                      0
                                                          5
                                                                       0 |
           0
                                                                 0
                                                                                 i = LocalService.NT AUTHORITY
j = Administrator.IMAGE
     0
            0
                  0
                         0
                                0
                                      0
                                             0
                                                    1
                                                         16
                                                                 0
                                                                       0 j
                                                   0
                                                               89
     0
           0
                  0
                         1
                                0
                                      0
                                             1
                                                          0
                                                                        1 1
                                                                                 k = zsnd
                         0
                                             2
                                                   0
                                                          0
                                                                 1 2655 |
     0
           0
                                      0
                                1
```

J48 / il03

=== Run information ===

```
Scheme:
                    weka.classifiers.trees.J48 -C 0.25 -M 2
                    fiwalk-weka.filters.unsupervised.attribute.Remove-R1,10-weka.filters.unsupervised.
Relation:
attribute.StringToNominal-Clast-weka.filters.unsupervised.attribute.ReplaceMissingValues
                    92024
Instances:
Attributes:
                    a
                    partition
filesize
                    mtime
                    ctime
                    atime
                    fragments
                    frag1startsector
                    frag2startsector
                    userquess
Test mode:
                    10-fold cross-validation
=== Classifier model (full training set) ===
J48 pruned tree
atime <= 1070985600000
     ctime <= 1070959060000
           ctime <= 1070958922000
                ctime <= 1070958790000
           | ctime <= 1070958790000
| | ctime <= 1070958604000: All Users (3.0)
| ctime > 1070958604000: Default User (12.0)
| ctime > 1070958790000: All Users (39.0)
ctime > 1070958922000: Default User (227.0)
     ctime > 1070959060000
           ctime <= 1070960026000
                 ctime <= 1070959746000
                    ctime <= 1070959286000: All Users (2.0)
ctime > 1070959286000: LocalService (3.0)
                 ctime > 1070959746000: User (32.0)
           ctime > 1070960026000
                 ctime <= 1071026706000
                 ctime <= 1071026706000
| ctime <= 1070986968000
| ctime <= 1070963562000: All Users (28.0/1.0)
| ctime > 1070963562000: User (5.0)
| ctime > 1070986968000: All Users (20.0)
ctime > 1071026706000: User (2.0)
atime > 1070985600000
     ctime <= 1071120822000
| ctime <= 1070959746000
                 ctime <= 1070959286000
                       atime <= 1114005600000
| frag1startsector <= 2749447
                                  filesize <= 6300
| mtime <= 997159590000: All Users (2.0)
| mtime > 997159590000: Default User (44.0/2.0)
filesize > 6300
                             | | filesize > 6300
| | filesize <= 11776: All Users (25.0)
| | filesize > 11776: Default User (6.0/1.0)
frag1startsector > 2749447
| mtime <= 07007374444</pre>
                                  mtime <= 970811744000: User (3.0)
mtime > 970811744000: All Users (17.0/1.0)
                       atime > 1114005600000: All Users (59.0/1.0)
                 ctime > 1070959286000
                       fragments <= 0: LocalService (48.0)
fragments > 0
                             frag1startsector <= 396999: LocalService (12.0)</pre>
                             fraglstartsector > 396999
| ctime <= 1070959486000: NetworkService (13.0/3.0)</pre>
                                   ctime > 1070959486000: LocalService (3.0)
            ctime > 1070959746000
                 ctime <= 1070960220000
| filesize <= 6421: User (102.0)
| filesize > 6421
                            ctime <= 1070960016000: User (7.0)
                 | ctime > 107096016000: 0ser (7.0)
| ctime > 107096016000: All Users (2.0)
ctime > 1070960220000
                       mtime <= 1074767900000
                       | filesize <= 11549: All Users (189.0/3.0)
| filesize > 11549: User (5.0/1.0)
mtime > 1074767900000
                             ctime <= 1070963470000: All Users (4.0)
ctime > 1070963470000
                                  frag2startsector <= 6305167: All Users (2.0)</pre>
                                   frag2startsector > 6305167: User (37.0/2.0)
     ctime > 1071120822000
            frag1startsector <= 9993231
                 atime <= 1114092000000
                       atime <= 1108656000000
```

frag1startsector > 22479 atime <= 1108137600000 | filesize <= 759 mtime <= 1078501390000 mtime <= 1078212236000 atime <= 1113055200000 atime <= 1112882400000 ime <= 11128240000
ctime <= 1075457498000
ctime <= 1074949812000
f filesize <= 434176: User (13.0)
f filesize > 434176: All Users (2.0)
ctime > 1074949812000: LocalService (9.0/1.0)
ctime > 1074949812000: User (46.0(1.0)) mtime <= 1114080056000 filesize <= 4352 filesize <= 524 | mtime <= 1099651762000 ctime <= 1098477766000: User (116.0) ctime > 1098477766000 | | frag1startsector <= 1875047: User (18.0/1.0)
| | frag1startsector > 1875047: NetworkService (4.0)
mtime > 1099651762000: User (334.0/17.0) filesize > 524 mtime <= 1113498150000 ctime <= 1099531106000 mtime <= 1087226496000 filesize <= 2240: User (2.0) filesize > 2240: All Users (3.0) | mtime > 1087226496000: User (43.0/2.0)
ctime > 1099531106000
| mtime <= 1097779960000: All Users (27.0)
| mtime > 1097779960000 frag1startsector <= 9590903 mtime <= 1106294796000: User (13.0)
mtime > 1106294796000
| atime <= 1113919200000</pre> ctime <= 1109432996000 atime <= 1113660000000 | ctime <= 1105470320000: User (2.0) ctime > 1105470320000 fraglstartsector <= 9032175: All Users (18.0)
fraglstartsector > 9032175: User (4.0/1.0) | atime > 1113660000000: User (3.0) ctime > 1109432996000 filesize <= 564 | filesize <= 549: User (4.0) | filesize > 549: All Users (6.0) filesize > 564: User (24.0) atime > 1113919200000: All Users (6.0) | | atime > 1113919200000: All Users (6.0)
| | frag1startsector > 9590903
| | mtime <= 1110587976000: All Users (8.0)
| | mtime > 1110587976000: NetworkService (2.0)
mtime > 1113498150000: User (90.0/2.0) | filesize > 4352: User (468.0/15.0)
mtime > 1114080056000 frag1startsector <= 5541911 | ctime <= 1100212392000 | ctime <= 108232882000: User (2.0) | ctime > 1088232882000: NetworkService (4.0) | ctime > 1100212392000: User (42.0/1.0) fraglstartsector > 5541911 ctime <= 1091971738000: User (2.0) ctime > 1091971738000: All Users (19.0) atime > 1114092000000 ctime <= 1071125176000: User (11.0) ctime > 1071125176000 filesize <= 389 | mtime <= 1091635424000: All Users (4.0/1.0) | mtime > 1091635424000: User (13.0/1.0) filesize > 389 filesize <= 2105: All Users (132.0/17.0) filesize > 2105: User (15.0/1.0) frag1startsector > 9993231 atime <= 1084024800000 filesize <= 232313 | atime <= 1082124000000 atime <= 1077638400000: User (2133.0/10.0) atime > 1077638400000

atime <= 1081864800000 | mtime <= 1072611488000: All Users (4.0)
| mtime > 1072611488000: User (93.0/3.0)
atime > 1081864800000: All Users (4.0) atime > 1082124000000: User (3624.0) filesize > 232313 atime > 1084024800000 atime <= 1086271200000 mtime <= 1084198626000 httme <= 108419820000
| mtime <= 1083856242000
| | mtime <= 1083856242000: User (2.0)
| mtime > 1080983272000: All Users (5.0/1.0)
| mtime > 1083856242000: All Users (52.0)
mtime > 1084198626000: User (10.0) atime > 1086271200000 ctime <= 1112949930000 ctime <= 1106830664000 ctime <= 1089022372000 | ctime <= 1073319082000 | mtime <= 1071489358000: All Users (2.0)
| mtime > 1071489358000: User (92.0)
ctime > 1073319082000 filesize <= 9576 | ctime <= 1087226494000 atime <= 1094738400000 filesize <= 597: User (17.0/5.0) filesize > 597 filesize > 2014: All Users (9.0) filesize > 2014: User (3.0/1.0) atime > 1094738400000 | atime <= 1113400800000: User (21.0) atime > 1113400800000: All Users (4.0/1.0) ctime > 1087226494000 ctime <= 1088151294000: All Users (13.0) | | ctime > 1088151294000: User (4.0/1.0)
filesize > 9576: User (24.0/1.0) ctime > 1089022372000 atime <= 1098108000000: User (2915.0/3.0) atime > 1098108000000 mtime <= 1098002698000 ctime <= 1098616316000 | mtime <= 1096875572000: User (61.0) mtime > 1096875572000
| mtime <= 1097228874000: All Users (4.0)
| mtime > 1097228874000: User (4.0/1.0) ctime > 1098616316000 ctime <= 1099531276000: All Users (57.0) ctime > 1099531276000: User (9.0) mtime > 1098002698000 ctime <= 1100133458000 ctime <= 1099995288000 filesize <= 699 | filesize <= 546: User (90.0/6.0) | filesize > 546: All Users (3.0) filesize > 699: User (217.0) ctime > 1099995288000 | ctime <= 1100039384000: NetworkService (10.0)
| ctime > 1100039384000: All Users (4.0)
ctime > 1100133458000: User (1774.0/16.0) ctime > 1106830664000 | atime <= 1107705600000 filesize <= 8421 filesize <= 905 | atime <= 1107532800000 atime <= 1106928000000: All Users (3.0) atime > 1106928000000 atime <= 1107187200000: User (10.0) atime > 1107187200000 ctime <= 1107528890000: All Users (7.0/1.0) ctime > 1107528890000: User (3.0) atime > 1107532800000: All Users (66.0) filesize > 905 ctime <= 1106910794000: All Users (2.0) ctime > 1106910794000: User (83.0) | filesize > 8421
| filesize < 21678: All Users (107.0)
| filesize > 21678: User (8.0)
atime > 1107705600000 ctime <= 1112842506000 ctime <= 1109304624000 | ctime <= 1108040994000

ctime <= 1106978972000: All Users (5.0) ctime > 1106978972000: User (138.0/5.0) ctime > 1108040994000 atime <= 1108310400000: All Users (15.0/1.0) atime > 1108310400000 mtime <= 1109264522000 fragments <= 0: User (17.0)
fragments > 0 | | filesize <= 2960: All Users (5.0/1.0)
| | filesize > 2960: User (8.0)
mtime > 1109264522000: All Users (9.0) ctime > 1109304624000: User (241.0/2.0) ctime > 1112842506000 | ctime <= 1112843324000: All Users (30.0/1.0) | ctime > 1112843324000 ctime > 1112949930000 mtime <= 1114130254000 atime <= 1113832800000: User (1728.0) atime > 1113832800000 fragments > 0: User (643.0/1.0) mtime > 1114130254000 ctime <= 1229683712000 CLIME <= 1229683712000
| filesize <= 11: All Users (5.0)
| filesize > 11
| | mtime <= 1229573390000: User (3.0)
| mtime > 1229573390000: All Users (2.0/1.0)
ctime > 1229683712000: User (35.0) Number of Leaves : 140 Size of the tree : 279 Time taken to build model: 2.25 seconds === Stratified cross-validation === === Summary === Correctly Classified Instances Incorrectly Classified Instances 17120 98.2327 % 308 1.7673 % Kappa statistic 0.8932 Mean absolute error 0.0098 Root mean squared error 0.0809 Relative absolute error 14.576 % Root relative squared error Total Number of Instances 44.1412 % 17428 74596 Ignored Class Unknown Instances === Detailed Accuracy By Class === TP Rate FP Rate Precision Recall F-Measure ROC Area Class 0.973 0.9 0.676 0.986 0 0.007 0.979 0.878 0.97 0.79 0.986 Default User 0.858 All Users 0.658 0.001 0.658 0.667 0.888 NetworkService 0.901 0 0.913 0.901 0.907 0.97 0.753 LocalService 0.107 0.993 0.989 0.991 User === Confusion Matrix === b с 0 d e 1 I <-- classified as 283 a = Default User b = All Users 3 0 3 157 | 7 1005 0 C - ALL USERS C = NetworkService d = LocalService 2 2 9 | 73 2 | 0

25 3 9 2 15734 | e = User

0

1

104

- Tamas Abraham. Event sequence mining to develop profiles for computer forensic investigation purposes. In ACSW Frontiers '06: Proceedings of the 2006 Australasian workshops on Grid computing and e-research, pages 145–153. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 2006. ISBN 1-920-68236-8.
- [2] David W. Aha, Dennis Kibler, and Marc K. Albert. Instance-based learning algorithms. *Mach. Learn.*, 6(1):37–66, 1991. ISSN 0885-6125.
- [3] Chidanand Apté and Sholom Weiss. Data mining with decision trees and decision rules. *Future Gener. Comput. Syst.*, 13(2-3):197–210, 1997. ISSN 0167-739X.
- [4] Andrew Clark Bradley Schatz, George Mohay. A correlation method for establishing provenance of timestamps in digital evidence. In *Digital Investigation*, volume 3, supplement 1, pages 98–107. 6th Annual Digital Forensic Research Workshop, 2006.
- [5] Brian Carrier. The sleuth kit (tsk). Retrieved 2008-03-10 11:20:22 -0700. http:// www.sleuthkit.org/sleuthkit/desc.php.
- [6] Hsinchun Chen, Wingyan Chung, Yi Qin, Michael Chau, Jennifer Jie Xu, Gang Wang, Rong Zheng, and Homa Atabakhsh. Crime data mining: an overview and case studies. In *dg.o '03: Proceedings of the 2003 annual national conference on Digital government research*, pages 1–5. Digital Government Research Center, 2003.
- [7] Matthew Cooper, Jonathan Foote, Andreas Girgensohn, and Lynn Wilcox. Temporal event clustering for digital photo collections. *ACM Trans. Multimedia Comput. Commun. Appl.*, 1(3):269–288, 2005. ISSN 1551-6857.
- [8] Padhraic Smyth David Hand, Heikki Mannila. *Principles of Data Mining*. The MIT Press, 2001.
- [9] O. de Vel, A. Anderson, M. Corney, and G. Mohay. Mining e-mail content for author identification forensics. SIGMOD Rec., 30(4):55–64, 2001. ISSN 0163-5808.
- [10] DFRWS. A road map for digital forensic research. DTR T001-01 FINAL DFRWS Technical Report, 1(1), August 2001. http://dfrws.org/2001/dfrws-rm-final. pdf.
- [11] Usama M. Fayyad, Gregory Piatetsky-Shapiro, and Ramasamy Uthurusamy. Summary from the kdd-03 panel: data mining: the next 10 years. *SIGKDD Explor. Newsl.*, 5(2): 191–196, 2003. ISSN 1931-0145.

- [12] George Forman, Kave Eshghi, and Stephane Chiocchetti. Finding similar files in large document repositories. In *KDD '05: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 394–400. ACM, New York, NY, USA, 2005. ISBN 1-59593-135-X.
- [13] John Galloway and Simeon J. Simoff. Network data mining: methods and techniques for discovering deep linkage between attributes. In APCCM '06: Proceedings of the 3rd Asia-Pacific conference on Conceptual modelling, pages 21–32. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 2006. ISBN 1-920-68235-X.
- [14] Simson L. Garfinkel. Fiwalk program.
- [15] Simson L. Garfinkel. Forensic feature extraction and cross-drive analysis. *Digital Investigation*, 3(Supplement-1):71-81, 2006. http://dx.doi.org/10.1016/j.diin. 2006.06.007.
- [16] Simson L. Garfinkel and Basis Technology. Aff: The advanced forensic format. Retrieved 2008-03-03 10:50:49 -0700. http://www.afflib.org/.
- [17] III Golden G. Richard and Vassil Roussev. Next-generation digital forensics. Commun. ACM, 49(2):76–80, 2006. ISSN 0001-0782.
- [18] Robert C. Holte. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–90, 1993.
- [19] Eibe Frank Ian H. Witten. *Data Mining : Practical Machine Learning Tools and Techniques.* Morgan Kaufman Publishers, 2005.
- [20] Knoppix. Shred tool. Retrieved 2007-10-09 10:50:49 -0700. http://www.knopper. net/knoppix-mirrors/.
- [21] Jan H. Kroeze, Machdel C. Matthee, and Theo J. D. Bothma. Differentiating data- and text-mining terminology. In SAICSIT '03: Proceedings of the 2003 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology, pages 93–101. South African Institute for Computer Scientists and Information Technologists, , Republic of South Africa, 2003. ISBN 1-58113-774-5.
- [22] M. Last. The uncertainty principle of cross-validation. *Granular Computing*, 2006 IEEE International Conference on, pages 275–280, 10-12 May 2006.
- [23] Heidi Computers Ltd. Eraser tool. Retrieved 2007-09-10 10:50:49 -0700. http:// www.heidi.ie/node/6.
- [24] Piriform Ltd. Ccleaner tool, 2005-2008. Retrieved 2007-09-10 10:50:49 -0700. http: //www.ccleaner.com.
- [25] Tom M. Mitchell. Instance Based Learning. McGraw Hill, 1997.
- [26] Jan Guynes Clark Nicole Lang Beebe. Digital forensics text string seraching: Improving information retrieval effectiveness by thematically clustering search results. In *Digital Investigation*, volume 4, pages 49–54. 6th Annual Digital Forensic Research Workshop, 2007.
- [27] Salvatore J. Stolfo and Shlomo Hershkop. Email mining toolkit supporting law enforcement forensic analyses. In dg.o2005: Proceedings of the 2005 national conference on Digital government research, pages 221–222. Digital Government Research Center, 2005.
- [28] Ian H. Witten. Text mining. Computer Science, University of Waikato, Hamilton, New Zealand, 2004.

THIS PAGE INTENTIONALLY LEFT BLANK

Referenced Authors

Abraham, Tamas 10 Aha, David W. 13 Albert, Marc K. 13 Anderson, A. 8 Apté, Chidanand 11, 13–15 Atabakhsh, Homa 9

Bothma, Theo J. D. 8 Bradley Schatz, Andrew Clark, George Mohay 10

Carrier, Brian 20 Chau, Michael 9 Chen, Hsinchun 9 Chiocchetti, Stephane 9 Chung, Wingyan 9 Cooper, Matthew 9 Corney, M. 8

David Hand, Padhraic Smyth, Heikki Mannila 4, 5 de Vel, O. 8 DFRWS 2–4

Eshghi, Kave 9

Fayyad, Usama M. 5 Foote, Jonathan 9 Forman, George 9

Galloway, John 10 Garfinkel, Simson L. 1, 7, 17, 18 Girgensohn, Andreas 9 Golden G. Richard, III 1

Hershkop, Shlomo 8 Holte, Robert C. 15, 16

Ian H. Witten, Eibe Frank 1, 8, 12, 13, 15, 19, 20, 25

Kibler, Dennis 13 Knoppix 23 Kroeze, Jan H. 8

Last, M. 16, 17 Ltd, Heidi Computers 23 Ltd, Piriform 23

Matthee, Machdel C. 8 Mitchell, Tom M. 13 Mohay, G. 8

Nicole Lang Beebe, Jan Guynes Clark 10

Piatetsky-Shapiro, Gregory 5

Qin, Yi 9

Roussev, Vassil 1

Simoff, Simeon J. 10 Stolfo, Salvatore J. 8

Technology, Basis 17

Uthurusamy, Ramasamy 5

Wang, Gang 9 Weiss, Sholom 11, 13–15 Wilcox, Lynn 9 Witten, Ian H. 8, 14

Xu, Jennifer Jie 9

Zheng, Rong 9

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

- 1. Defense Technical Information Center Ft. Belvoir, Virginia
- 2. Dudly Knox Library Naval Postgraduate School Monterey, California