Journal of Engineering and Applied Sciences 14 (19): 7048-7058, 2019
ISSN: 1816-949X
© Medwell Journals, 2019

A Model for Evaluating Digital Forensic Tools

Precilla M. Dimpe and Okuthe P. Kogeda
Department of Information Technology, Tshwane University of Technology, Pretoria, South Africa
precilladimpe, Kogeda@gmail.com

Abstract: Digital Forensic Investigators (DFIs) rely on tools to assess, gather and analyze digital evidence. They are used to unravel criminal acts and prove crime in a court of law. However, most of these tools are used without being evaluated because tool evaluation is expensive and time consuming. In addition, most DFIs assume that a tool would do exactly what the vendor claims it would do. If a tool is not evaluated, it remains unknown whether the results it produces are reliable or not. Unreliable results may jeopardize the whole forensic investigation process and in some cases lead to improper civil judgements resulting in criminals walking free thereby being encouraged to commit the same crime again. This may also lead to time wasting, trial and error, loss of money etc. Therefore, in this study, we designed and implemented a model for evaluating digital forensics tools to help DFIs with evaluating the tools that they would want to use. We used data from the Computer Forensic Tool Testing (CFTT) project which we aggregated and classified using Bayesian networks. We implemented our model using Java programming language and MySQL database. We tested using the data from the CFTT project in conjunction with the feedback provided by DFIs to recommend a suitable tool to use for investigations based on the task a DFI wants to perform, the category of the tool and its cost. The model attained a utility performance of 91.7%.

Key words: Digital forensics, Bayesian networks, tools, cybercrime, digital forensic investigators, MySQL, evaluation, criminals, evidence

INTRODUCTION

Over the past years, we have seen a vast increase in cybercrime activities and its impact on the society. Digital forensics has been one of the forensic sciences that has been used to investigate such activities. It uses scientifically proven and derived methods concerning the preservation, interpretation, identification, collection, analysis, presentation and documentation of digital evidence resulting from digital sources for the purpose of furthering or facilitating the rebuilding of operations that are planned (Selamat *et al*., 2008).

The problem in digital forensics is that most tools used by Digital Forensic Investigators (DFIs) have not been evaluated because it is time consuming and expensive. In addition, DFIs assumed that proprietary tools from reputable vendors can do exactly what the vendor claims it can do. Vendor evaluation has not been documented and proven publicly (Beckett and Slay, 2007). However, vendor's claim that the tools are able to perform the tasks required by a DFI and therefore it is up to a DFI to make sure that the tool can do exactly what the vendor claims it can do. A DFI risks loss of integrity if doubt can be introduced into the accuracy of the tools and actions deployed in the presented evidence (Armstrong, 2003). An inferior tool can compromise investigations because the results produced by the tool are used in a court of law to convict criminals or prove innocence (Carrier, 2002).

Tools play an important role in investigations and without them DFIs cannot conduct investigations. A reliable tool must be used in order to produce reliable results because the reliability of digital evidence is of vital significance given the forensic context of the discipline (Van Den Bos and Van Der Knijf, 2005). To ensure that the evidence presented in the court of law is reliable and accurate, tools must be evaluated. Efforts have been taken by researchers to come up with models or techniques for evaluating digital tools but most of them do not address the time consumption problem experienced by DFIs when it comes to tool evaluation as they require a DFI to manually evaluate the tools. As a result, the gap between tools and their evaluation still exists. In an attempt to close this gap, we designed and implemented a model for evaluating digital forensics tools using Java, MySQL database and Bayesian Network (BN). Our model uses test results from the Computer Forensic Tool Testing (CFTT) project in conjunction with the feedback provided by DFIs to recommend a suitable tool to use for

Corresponding Author: Precilla M. Dimpe, Department of Information Technology, Tshwane University of Technology, Pretoria, South Africa investigations based on the task they want to perform, the category of the tool and its cost. The feedback provided by DFIs influences the tool's reliability level, since, it is used to build on historical data from the CFTT project. If the feedback is negative, the tool's reliability level decreases and if it is positive, it increases. Our model is useful in assisting DFIs to make informed choices about acquiring and using tools. Furthermore, tool-testing organizations can use our model to publish their test results in one platform to make them easily accessible to DFIs.

Digital forensics tools: In our definition, a tool is a hardware or software used to achieve a goal or carry out a particular function (Dimpe and Kogeda, 2017). It is used to recover deleted files, create a disk image, collect data from a digital device, analyze data, etc., examples of hardware tools are Image MASSter Solo-3 (DHS, 2013) and LinkMASSter-2 (Cengage Learning, 2010). Examples of software tools are listed in Table 1. These tools differ in functionality, complexity and cost. Some are designed to serve a single purpose while others offer a number of functions, some of the market leading commercial tools cost a lot of money while others are free (open source). The nature of the investigation determines which tool is appropriate for the task at hand (Arthur and Venter, 2004). In this study, we focus our research on software tools.

Literature review: Tool evaluation is used for validating and verifying the quality of the tools. The importance of this procedure is to improve the confidence of software developers and DFIs that the software is fit for the purpose. In an attempt to evaluate digital forensic tools, we explored models or techniques that have been proposed by other researchers. Their strengths and limitations are discussed in this study.

The CFTT (Anonymous, 2001) project was aimed at providing a measure of guarantee for tools used by law enforcement agencies in investigations. They followed seven steps which include: establish categories of forensic requirements, identify requirements for a specific category, develop test assertions based on requirements, develop test code for assertions, identify relevant test cases, develop testing procedures and report test results. Thus, the vendor and testing organizations review the results to ensure a certain level of fairness. However, the disadvantage of it is that by the time the results are publicly available, the version of the tested tool might be deprecated (Vandeven, 2014). In spite of that the CFTT project has extensive experience in tool evaluation, hence, we took advantage of that in our model by using their test results in conjunction with the DFI's feedback to ensure that tool upgrades and patches are considered.

The Scientific Working Group on Digital Evidence (Anonymous, 2018) developed testing templates and guidelines with the aim of helping parties that embark on tool testing. The guidelines include developing a test plan and performing test scenarios. The test plan should include test purpose, scope, methodology and requirements to be tested. Their methodology has been implemented and tested but unlike the CFTT project, their results are only released to the United State (US) law enforcement agencies and not to the public. Which makes it challenging to ascertain whether their methodology is suitable for tool evaluation or not.

Pan and Batten (2009) developed a methodology for evaluating digital forensics tools by using a partition testing approach. They used Orthogonal Array (OA) to test the performance of a tool against itself or against other tools on the same constraint. Their methodology reduced the effect of incorrect observations with large values by using Taguchi's logarithmic function. Pan and Batten (2009) outlined that Taguchi's method alone is not sufficient to reduce the impact of outliers. As a result, they created a theorem to define the maximum number of suspicious samples acceptable.

The researchers claimed that their methodology allows testers to compare the performance of tools without consuming a large amount of time or using advanced equipment and can be fully automated in the

Table 1: Digital forensics tools (software)

Tool name	Description
FTK imager	FTK Imager is a free extension of FTK, developed by AccessData to generate images from other types of storage devices and hard drives (Vandeven, 2014)
EnCase	EnCase is a widely known computer forensics tool designed by Guidance Software to analyze, collect and report on evidence (Vandeven, 2014)
X-Ways Forensics	X-Ways Forensics uses diverse data recovery methods and search functions to find files that are deleted. It includes bit accurate imaging of a disk to provide a comprehensive examination of a case (Irmler *et al.*, 2013)
Device seizure	It is an analysis and acquisition tool for examining mobile devices (Anonymous, 2012). It consist of a driver pack that is designed to maintain the integrity of device acquisition (Anonymous, 2012)
Oxygen forensics	Oxygen Forensics is a mobile forensics tool designed to acquire and analyze data from mobile devices (DHS, 2015)
Adroit photo forensics	Adroit Photo Forensics recovers graphic files of several types using proprietary GuidedCarving and SmartCarving technologies (DHS., 2012)

future. Even though their methodology was tested and the results proved the validity and effectiveness of their methodology, the researchers acknowledged that it might not be feasible to use their methodology for any type of tool because it is very difficult to develop robust testing measures for every category of tools.

Wilsdon and Slay (2006) proposed an evaluation framework to validate the accuracy and reliability of tools. Their framework uses black box testing techniques by making use of reference sets and test cases. It consists of 6 phases including: acquiring software, identification of the software functionalities, development of the result acceptance spectrum, executing test and evaluate results and releasing evaluation results. The development of result acceptance spectrum uses the methodology for documenting the result acceptance spectrum from ISO 14598.1-2000 which divides potential results set into 4 groupings, exceeds requirements, target range, minimally acceptable and unacceptable. If a function does not meet the acceptance range, then that function and all dependents and co-dependents are rendered as incorrect. The functions found to be below the acceptable range are regarded as failed. Those that are in or above an acceptance rating are regarded as passed.

Wilsdon and Slay (2006) claim that the framework offers advantages such as efficient process, community input, various environment testing and a community point of contact. Their methodology divides results into the 4 above-mentioned groupings which also includes minimally acceptable. In their research, if a function is minimally acceptable, it still falls under the acceptance range. Minimally acceptable shows that a function did not meet all its requirements, therefore, it should not be regarded as passed. Given the forensic context of the discipline, a tool must produce reliable results. It must either exceed requirements or be on target range in order to be regarded as passed.

Guo *et al*. (2009) developed a methodology for validation and verification of forensics tools that was achieved by stipulating the requirements of each mapped function. Their focus was on the searching function in which the searching function was mapped and its requirements specified. A reference set was developed to validate and verify tools that have the searching function. The researchers claim that their methodology, offers benefits such as detachability, flexibility, tool neutrality and transparency. However, they stated in their conclusion, "even if the methodology is promising, it needs to be tested" this obviously shows that their method was not tested. Therefore, the reliability of their methodology is not known. In any research, testing is paramount because it provides unambiguous evidence and confidence regarding the performance and limitations of a tool (Iacob and Constantinescu, 2008). However, the research of Wilsdon and Slay (2006) and Guo *et al*. (2009) that was reviewed above was never tested. To prove beyond reasonable doubt, our model has been tested and results obtained are presented in this study.

Baggili *et al*. (2007) created a systematic database driven testing model for mobile forensics tools. Their model takes tool-testing standards and alters them, so that, the process model is programmatically driven by the database system. Based on the process model and the proprietary nature of mobile phones, a relational database schema was developed to aid in illustrating the different data requirements for mobile phones tool testing. Factors that should be recognized when forensic acquisition is performed and the data that should be stored in the database were presented in the form of an Entity Relationship Diagram (ERD). The data includes log files of all forensic examination and testing procedures for different mobile phones. They used the database driven approach to demonstrate the calculation of General Error Rate (GER) and the Feature Error Rate (FER). The researchers claimed that a database solution could help in the formation of calculated error rates based on the tool's past stability and in establishing the degree of reliability for various tool sets.

Kubi *et al*. (2011) evaluated XRY 5.0 and UFED 1.1.3.8 mobile forensics tools based on NIST smartphone tool specification and test cases using Daubert principle as a point of reference towards the admissibility of digital evidence. The evaluation was executed by using 6 phases which includes: collection, identification, preservation, examination, analysis and reporting. A graphical representation was used to compare the results which showed that most of the time XRY 5.0 exceeded UFED 1.1.3.8 in terms of performance.

The research of Baggili *et al*. (2007) and Kubi *et al*. (2011) only focuses on mobile forensics which is a specific category of digital forensics that deals with mobile tools. However, our model covers a more general area in digital forensic.

Hildebrandt *et al*. (2011) proposed a common scheme for evaluating digital forensics tools. Their model was divided into hard criteria and soft criteria. The soft-criteria includes: general acceptance within the expert community, publication of the method, standards for the usage of the application, intention of the investigation and personal familiarity with the application. The hard criterion was divided into the must-criteria, should-criteria and can-criteria. The must-criteria is concerned with the core functionality of a tool including logging, protection of the integrity of the gathered data, protection of the authenticity of the gathered data, protection of the confidentiality of the gathered data, access restriction for the gathered data and protection of the integrity of the source data.

Subsequently, the can-criteria are driven by the potential extortions for forensic software which includes system heterogeneity, minimality of expected system rights and open source. Their research does not focus only on the model for evaluating tools but it also focuses on attacker models, a framework for the development of forensic software, legal and technical requirements for digital forensics tools. Their model is broad in scope and has been tested. However, it is not concerned with tool testing; it uses results from tool testing organizations such as the National Institute of Justice (NIJ). The challenge with relying only on tool testing organization is that they cannot keep up with tool versions and patches. Similarly to their model, our model uses test results from the CFTT project. However, it does not only use their test results but it also uses feedback provided by DFIs to ensure that tool versions and patches are taken into consideration.

MATERIALS AND METHODS

Design and implementation: In this study, we discuss the system architecture and show how different components of the model work together. We also provide insight into the implementation of our model and how Bayesian network was used for recommendations. We further, discuss how the user interface researches in detail.

System architecture: Systems architecture is considered to be the conceptual model that describe the behavior and structure of a system (Jaakkola and Thalheim, 2010). The architecture of our model is made up of three layers: the presentation layer, application layer and database layer. The presentation layer contains the components that implement and display the user interface and manage user interaction (Wellhausen). The application layer is the layer that hosts the web server and recommender engine. The web server is responsible for processing request from the user. A user (DFI) interacts with the system to search for a tool that can perform a desired task. The web server then communicates with the recommender engine which uses Bayesian network to select a tool that can perform the task that the user requires using the information in the database which is hosted in the database layer. Once the tool is found, the web server returns the recommended tool to the user as shown in Fig. 1.

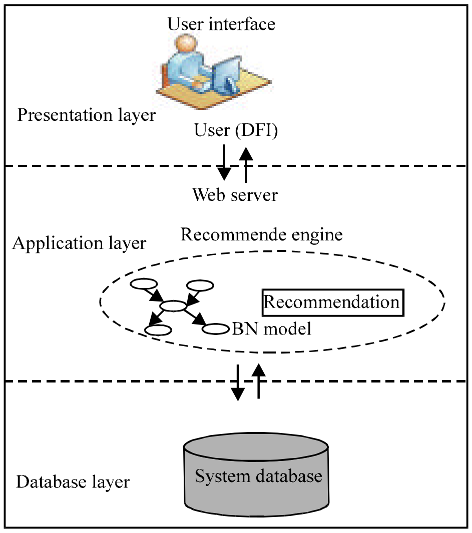


Fig. 1: System architecture

Bayesian networks: Bayesian Networks (BNs) are graphical models for reasoning under uncertainty where the nodes represent variables and arcs represent direct connections between them (Kevin and Ann, 2004). We choose to use Bayesian networks because it shows good prediction accuracy even with small sample sizes. The idea behind this research is for DFIs to provide feedback on tools after using them. The model then uses that feedback to build on its current data (historical data) in order to enhance its recommendation. Bayesian networks can be used for this purpose because it can be immediately updated when new data is presented (Kragt, 2009).

We used literature review, data from the CFTT project and a survey that was conducted by the Digital Forensic Investigation Research Laboratory to determine possible factors affecting the selection of a tool. Survey data from the Digital Forensic Investigation Research Laboratory was used to determine factors that DFIs takes into consideration when purchasing a tool (Horny, 2014). According to the survey they conducted, the factors include: feature set (task), cost and ease of use. However, their data was not used to assign states but rather to guide us on what DFIs takes into consideration when purchasing a tool. The actual data that we used to assign state variables was derived from literature review and CFTT project.

Bayesian network construction: The first step in constructing a Bayesian network is to build a directed acyclic graph, followed by an assessment of prior and conditional probability in each node (James, 2018). For the assignment of prior probabilities, literature review together with data from the CFTT project were used. On

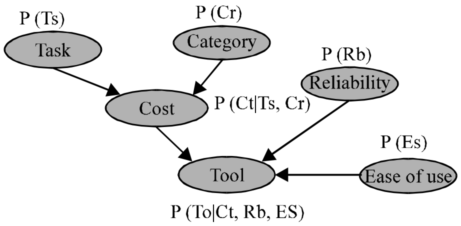


Fig. 2: Bayesian network

the other hand, the conditional probability associated with each node was calculated using Eq. 1. Whereby the probability of event of a variable state is calculated based on the fact that some important evidence has been observed.

Out of approximately 103 tools that were tested by the CFTT project, only 6 were selected for our research, namely: FTK Imager, X-Ways Forensics, EnCase Forensic, Adroit photo forensics, Oxygen Forensics and Device seizure. By making use of the 6 tools that were selected, we were able to construct a Bayesian network with 6 nodes which consist of the following variables: Task (Ts), Category (Cr), Cost (Ct), Reliability (Rb), Ease of use (Es) and Tool (To) as shown in Fig. 2:

$$P(X|Y) = \frac{P(Y|X)T(X)}{P(Y)} \qquad (1)$$

Where:

P(X) = The prior Probability of the event X without any knowledge about the event Y

P(X|Y)= The conditional Probability of X, given the event Y

P(Y|X)= The conditional Probability of Y, given the even X

P(Y) = The marginal Probability of the event Y, acting like a normalizing constant

Using a Bayesian network in Fig. 2 as an example, the probability that in the modelled system the tool is FTK (To = FTK), given that the task is data acquisition (TK = Da), category is computer forensic (Cr = CF), cost is free (Ct = Fr) was calculated using in Eq. 2:

$$P(To = FTK|\ Tk = Da, Cr = Cf, Ct = Fr) = \frac{P(To = FTK.Tk = Da, Cr = Cf, Ct = Fr)}{P(Tk = Da, Cr = Cf, Ct = Fr)} \qquad (2)$$

The above example was used to demonstrate how Bayesian network works at the backend to recommend a tool. In the next study, we demonstrate how the user interact with the system to search and view the recommended tool.

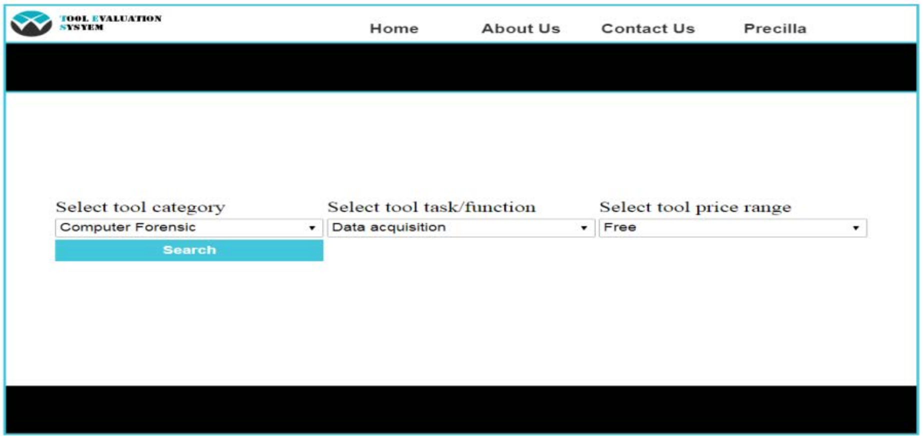
User interface: In this study, we illustrate how the user interface works including how the user interacts with the system to get the required tool. We discuss in detail how the tool evaluation, recommendation and feedback page works.

Tool evaluation page: The tool evaluation page allows the user (DFI) to search for a tool that can perform the desired task. The model looks for the tool that can perform the desired task in the selected category within the desired price range which can either be free, low or medium. By clicking the search button, the model searches for the tools that meets the user's requirements and recommends those tools to the user as shown in Fig. 3.

Recommendation page: The model recommends a tool that meets the user's requirements including a brief description about the tool, its functions and reliability level as shown in Fig. 4 and 5. The reliability level of a tool are calculated using historical data from the CFTT project in conjunction with the feedback from the users. However, at the inception of the system, recommendations are only made using the data from the CFTT project. Users are required to provide feedback on the tools after using them by clicking on the feedback link.

Feedback page: The purpose of the feedback page in Fig. 6 is to get information from DFIs about the performance of tools. The feedback provided by DFIs influences the tool's reliability level because it is used in conjunction with historical data to build on the tool's reliability level. If the feedback is negative, the tool's reliability level decreases and if it is positive, it increases. Feedback was weighed based on the user's level of expertise, e.g., an intermediate user's feedback weighs less than that of an expert as shown in Table 2. However, feedback from a beginner and novice user were ignored because they have limited knowledge in the area of digital forensic tools. This procedure was carried out using a decision matrix due to its ability to weigh multi-dimensional decisions of a decision set (Qureshi *et al*., 2013).

If a tool performed as expected on a particular test case, it is rated as reliable (Feedback (F = 1)) for that test case. If not, it is rated as partly reliable or unreliable (Feedback (F = 0)). Equation 3 and 4 were adapted from a decision matrix where F_r is the feedback result, F $_s$ (feedback score) is the feedback provided by DFIs, W_s is the weighted score assigned to DFIs based on their level of expertise and T_{ws} is the total weighted score. After the feedback is provided, we calculated the new reliability

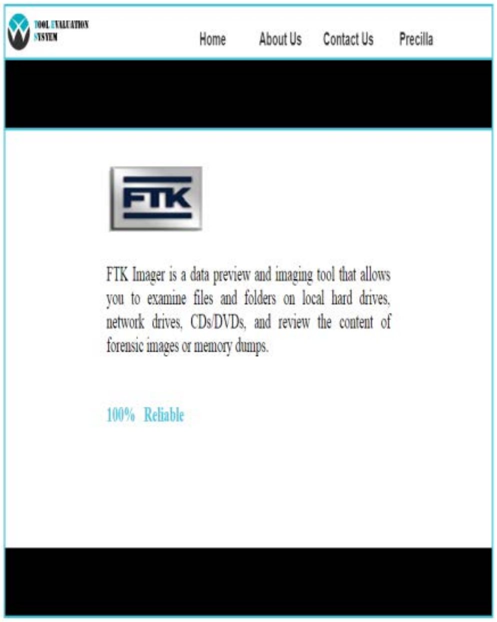


Select tool category: Computer Forensic

Select tool task/function: Data acquisition

Select tool price range: Free

Fig. 3: Tool evaluation page



FTK Imager is a data preview and imaging tool that allows you to examine files and folders on local hard drives, network drives, CDs/DVDs, and review the content of forensic images or memory dumps.

100% Reliable

Fig. 4: Recommendation page

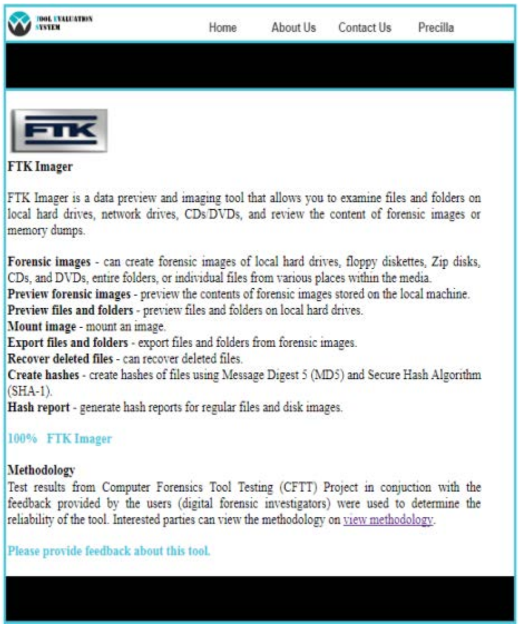
level of a tool given feedback results (F_r). Where, N_{rb} is the new reliability level of a tool and O_{rb} is the old one:

$$N_{Rb} = \frac{O_{Rb}+F_r}{2} \tag{3}$$

$$F_r = \sum \frac{F_S \times W_S}{T_{ws}} \tag{4}$$

RESULTS AND DISCUSSION

In order to test and evaluate the performance of our model, we used functional testing. In functional testing,



FTK Imager

FTK Imager is a data preview and imaging tool that allows you to examine files and folders on local hard drives, network drives, CDs/DVDs, and review the content of forensic images or memory dumps.

Forensic images - can create forensic images of local hard drives, floppy diskettes, Zip disks, CDs, and DVDs, entire folders, or individual files from various places within the media.
Preview forensic images - preview the contents of forensic images stored on the local machine.
Preview files and folders - preview files and folders on local hard drives.
Mount image - mount an image.
Export files and folders - export files and folders from forensic images.
Recover deleted files - can recover deleted files.
Create hashes - create hashes of files using Message Digest 5 (MD5) and Secure Hash Algorithm (SHA-1).
Hash report - generate hash reports for regular files and disk images.

100% FTK Imager

Methodology
Test results from Computer Forensics Tool Testing (CFTT) Project in conjuction with the feedback provided by the users (digital forensic investigators) were used to determine the reliability of the tool. Interested parties can view the methodology on view methodology.

Please provide feedback about this tool.

Fig. 5: Recommendations are made based on the user's requirements

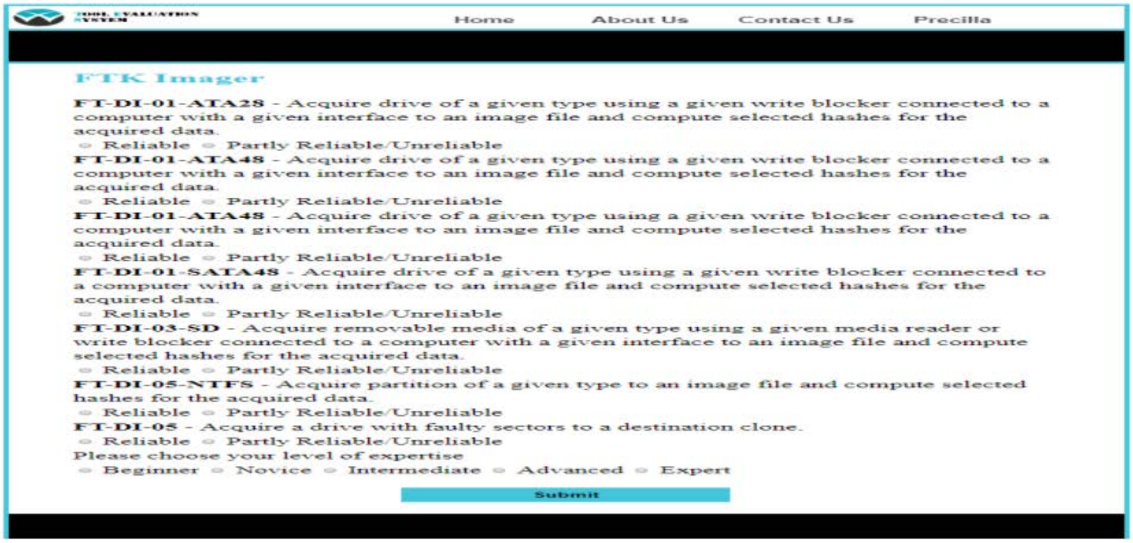
test cases are designed based on the information from the requirements to ensure that the system or software conforms with all the requirements (Nidhra and Dondeti, 2012). It guarantees that the functionality stated in the requirement specification works. Using functional testing, we were able to:

We determined the functional requirements of our model, the functional requirements were determined in order for us to know what to test and how to test it. The core functional requirements of our model are as follows:

- The model shall recommend a suitable tool based on the task, category and cost

Table 2: DFI's level of expertise (Aamodt and Plaza, 1994)

Expertise level	Description	Weighted score
Beginner	Has knowledge or an understanding of basic techniques and concepts in digital forensic	Feedback disregarded
Novice	Individuals who have a certain level of experience gained in experimental scenarios and/or classroom or as a trainee in the job	Feedback disregarded
Intermediate	Individuals who are able to complete a digital forensic task. They may occasionally require help from an expert but they can independently perform a task	60%
Advance	This individual has the skill to perform digital forensic task without assistance	80%
Expert	Professional who has extensive experience acquired through study and practice	100%



FTK Imager

FT-DI-01-ATA28 - Acquire drive of a given type using a given write blocker connected to a computer with a given interface to an image file and compute selected hashes for the acquired data.
○ Reliable ○ Partly Reliable/Unreliable

FT-DI-01-ATA48 - Acquire drive of a given type using a given write blocker connected to a computer with a given interface to an image file and compute selected hashes for the acquired data.
○ Reliable ○ Partly Reliable/Unreliable

FT-DI-01-ATA48 - Acquire drive of a given type using a given write blocker connected to a computer with a given interface to an image file and compute selected hashes for the acquired data.
○ Reliable ○ Partly Reliable/Unreliable

FT-DI-01-SATA48 - Acquire drive of a given type using a given write blocker connected to a computer with a given interface to an image file and compute selected hashes for the acquired data.
○ Reliable ○ Partly Reliable/Unreliable

FT-DI-03-SD - Acquire removable media of a given type using a given media reader or write blocker connected to a computer with a given interface to an image file and compute selected hashes for the acquired data.
○ Reliable ○ Partly Reliable/Unreliable

FT-DI-05-NTFS - Acquire partition of a given type to an image file and compute selected hashes for the acquired data.
○ Reliable ○ Partly Reliable/Unreliable

FT-DI-05 - Acquire a drive with faulty sectors to a destination clone.
○ Reliable ○ Partly Reliable/Unreliable

Please choose your level of expertise
○ Beginner ○ Novice ○ Intermediate ○ Advanced ○ Expert

Fig. 6: Feedback page

- The model shall inform the user (DFI) if the required tool is not available
- The model shall allow the user to provide feedback on the tool
- The model shall ignore feedback from the beginner and novice user and only consider feedback from intermediate, advanced and expert user
- The model shall calculate feedback based on the weighted score
- The model shall use feedback provided by users to update the data in the database

We developed test cases: Test cases were developed based on the functional requirements specified in step 1. We tested our model, the testing was executed using test cases developed in step 2, a sample of some of the test cases used in carrying out this research can be found in Appendix A. Test cases were grouped into test runs and each test run contained 6 test cases. In total, we had 10 test runs for each function which contained 60 test cases. At the completion of each test, expected results were compared to actual results to determine if the function works as it should. If the actual results are the same as the expected results, status is pass (S = 1) and if not, status is fail (S = 0). The results for each test run were calculated using Eq. 5. The results from the calculations were used to plot our graphs in Fig. 7 and 8:

$$TR_n = \sum \frac{S}{T_{tc}} \tag{5}$$

We analyzed the results: Results are analyzed and discussed in.

Model evaluation: In our first experiment, the system's ability to recommend a suitable tool to use for an investigation and to inform a DFI if the required tool was not available was tested. The results obtained are shown in Fig. 7.

Figure 7, the first few runs of the model was not satisfactory, the model failed to recommend the required tool in some cases even if the required tool was available. This was caused by incorrect variable declaration and using the model without considering the information in

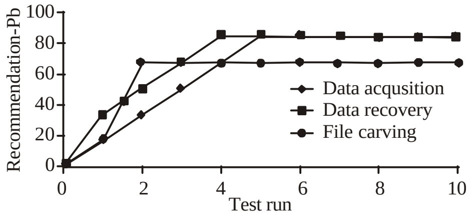


Fig. 7: Results of the recommendation

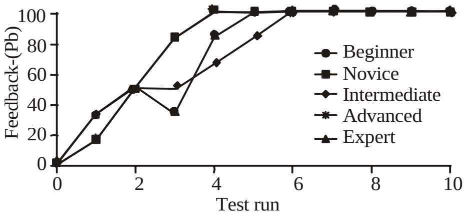


Fig. 8: Feedback

the database. After declaring the right variable and considering the information in the database, the model started to recommend the required tool. The model also failed to inform the DFI if the required tool was not available. It could not do so because the message that was supposed to inform the user that the required tool is not available was not defined on the server. That was corrected by defining the message on the server. After doing so, the model started to inform the user when the required tool was not available. In the second experiment, we tested the model's ability to calculate DFI's feedback based on their level of expertise and update the tool's information in the database accordingly as shown in Fig. 8. In the first run, we could only provide feedback on FTK. We encountered errors when we attempted to provide feedback on other tools because the data was not loaded to the session accordingly and as a result, the model was failing to submit the data stored in the session. To resolve that we recreated the session and loaded the data accordingly. Feedback from a beginner and a novice user was supposed to be ignored and should have not influenced the tool's reliability level. However, the model considered the feedback from them due to a logical error which also influenced the feedback from an intermediate, advance and expert user.

The performance measurement for our model was also evaluated using MATLAB due to its ability to predict the system's behavior. MATLAB is a simulator that can evaluate design, diagnose problems with an existing design and test a system under various conditions (Houcque, 2005). Therefore, it was employed in this study to observe the utility performance of our model as shown in Fig. 9. The utility performance of our

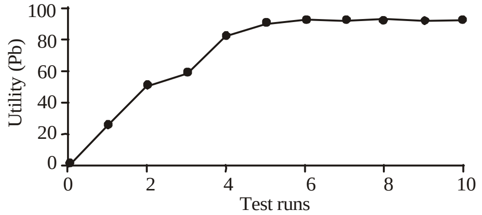


Fig. 9: Utility

model started at a low rate but kept improving until it reached a consistency of 91.7% utility which can be improved by fine-tuning the model and performing additional training.

In this study, we reviewed the models or techniques which do not address time consumption problem experienced by DFIs when it comes to tool testing as they require a DFI to manually evaluate the tool. In addition, some do not include all aspects of a tool and has not been tested. According to the survey, we conducted which was closed-ended and we managed to interview 10 digital forensic experts, 97.5% of the participants agreed that indeed it is necessary to evaluate a tool before using it but it is not practical for them to do so because it takes a significant amount of time. However, they pointed out that it would be very helpful if they could have official government entities or forensic institutes that can take care of tool evaluation. The CFTT project and SWGDE were established for that purpose and have done so excellently. However, they cannot meet the demands of DFIs because they take months to thoroughly evaluate a single tool. By the time they make their results available, the version of the tested tool might have already been upgraded.

Our proposed model seeks to address the limitations of the above-mentioned models or techniques by introducing a time-saving way of evaluating digital forensics tools and ensuring that tool patches and upgrades are taken into consideration. DFIs are not required to manually evaluate the tools, our model recommends a suitable tool for them to use for investigations based on the task they want to perform, the category of the tool and its cost. Our model uses test results from the CFTT project in conjunction with the feedback provided by DFIs as a knowledge base to recommend a tool. It ensures that recommendations are not only made based on historical data from the CFTT project but also current data from DFIs in the form of feedback. For example, the last test results from the CFTT project were released in 2016, ever since, then, tool patches and upgrades have been released. Therefore, feedback from DFIs was used to address this gap. DFIs who are using the tools are in a better position to provide us with the information about the current status of a tool.

CONCLUSION

Tool evaluation is a necessary and mandatory component of forensic science given that an unreliable tool may lead to unreliable result. In this study, we identified that most tools are used without being evaluated which is a problem that this study aimed to solve by developing a model for evaluating digital forensics tools. The model was developed using Java, Bayesian Network and MySQL server. Our model uses test results from the CFTT project in conjunction with the feedback provided by DFIs to recommend a suitable tool to use for investigations based on the task, category and cost. It saves time because it does not require a DFI to manually evaluate the tools, unlike the model and techniques reviewed in this study. In addition, it also ensures that tool versions and patches are catered for by using the feedback provided by DFIs. Furthermore, the model showed a utility of 91.7% after simulation. Therefore, our model can help DFIs with tool evaluation and tool makers to improve their tools. In addition, tool testing organization such as CFTT project and NIJ can use our web-based model to publish their test results to make them easy accessible to DFIs in one platform. However, our model is unable to validate DFIs expertise level, it assumes that the expertise level provided by DFIs is correct without validating them, more research is needed to find ways to validate DFIs expertise level.

RECOMMENDATIONS

In future, we intend to use different techniques and improve the utility of the model. A combination of techniques may be helpful in improving the performance of the model.

Appendix A: Test cases for recommendation

Function tested		Recommendation capability of the model			
Test ID		Test objective	Expected results	Actual results	Pass/Fail
1	Recommendation _ data acquisition_1.1	The model shall recommend a suitable tool if the task is data acquisition, category is computer forensic and price range is free. If the required tool is not available, the model must inform the user	The model must recommend a suitable tool based on the requirements. If the tool is not available, the model should inform the user	The model successfully recommended a suitable tool based on the requirements	Pass
1	Recommendation_ data acquisition_1.2	The model shall recommend a suitable tool if the task is data acquisition, category is computer forensic and price range is low. If the required tool is not available, the model must inform the user	The model must recommend a suitable tool based on the requirements. If the tool is not available, the model should inform the user	The required tool was available but the model did not recommend it	Fail
1	Recommendation_ data acquisition_1.3	The model shall recommend a suitable tool if the task is data acquisition, category is computer forensic and price range is high. If the required tool is not available, the model must inform the user	The model must recommend a suitable tool based on the requirements. If the tool is not available, the model should inform the user	The required tool was not available but the model did not report it	Fail
1	Recommendation_ data acquisition_1.4	The model shall recommend a suitable tool if the task is data acquisition, category is mobile forensic and price range is free. If the required tool is not available, the model must inform the user	The model must recommend a suitable tool based on the requirements. If the tool is not available, the model should inform the user	The required tool was not available but the model did not report it	Fail
1	Recommendation_ data acquisition_1.5	The model shall recommend a suitable tool if the task is data acquisition, category is mobile forensic and price range is low. If the required tool is not available, the model must inform the user	The model must recommend a suitable tool based on the requirements. If the tool is not available, the model should inform the user	The required tool was not available but the model did not report it	Fail
1	Recommendation_ data acquisition_1.6	The model shall recommend a suitable tool if the task is data acquisition, category is mobile forensic and price range is high. If the required tool is not available, the model must inform the user	The model must recommend a suitable tool based on the requirements. If the tool is not available, the model should inform the user	The required tool was available but the model recommended the least reliable instead of the most reliable	Fail

REFERENCES

Aamodt, A. and E. Plaza, 1994. Case-based reasoning: Roundational issues, methodological variations and system approaches. Artificial Intelli. Commun. IOS Press, 7: 39-59.

Anonymous, 2001. General test methodology for computer forensic tools. National Institute of Standards and Technology, Gaithersburg, Maryland, USA.

Anonymous, 2012. Paraben device seizure Version 4.3 evaluation report. National Institute of Justice, Washington, DC., USA.https://manualzz.com/doc/6952177/paraben-device-seizure-version-4.3-evaluation-report

Anonymous, 2018. SWGDE recommended guidelines for validation testing. Scientific Working Group on Digital Evidence, New York, USA. https://www.swgde.org/documents/Current%20Documents/SWGDE%20Recommended%20Guidelines%20for%20Validation%20Testing,

Armstrong, C., 2003. Developing a framework for evaluating computer forensic tools. Proceedings of the 2003 International Conference on Evaluation in Crime Trends and justice: Trends and Methods Conjunction with the Australian Bureau of Statistics, Canberra Australia, March 24-25, 2003, Canberra, Australia, pp: 1-8.

Arthur, K.K. and H.S. Venter, 2004. An investigation in to computer forensic tools. Comput. Sci., 1: 1-11.

Baggili, I.M., R. Mislan and M. Rogers, 2007. Mobile phone forensics tool testing: A database driven approach. Intl. J. Digital Evidence, 6: 168-178.

Beckett, J. and J. Slay, 2007. Digital forensics: Validation and verification in a dynamic work environment. Proceedings of the 2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07), January 3-6, 2007, IEEE, Waikoloa, Hawaii, USA., pp: 266a-266a.

Carrier, B., 2002. Open source digital forensics tools: The legal argument. Master Thesis, Stake Center Locating, Kernersville, North Carolina, USA.

Cengage Learning, 2010. Computer Forensics Investigating Data and Image Files. Cengage Learning, Boston, Massachusetts, USA.,.

DHS., 2012. Test results for graphic file carving tool: Adroit photo forensics 2013 v3.1d. National Institute of Standards and Technology, Gaithersburg, Maryland, USA.

DHS., 2013. Test results for digital data acquisition tool: Image Masster solo-4 Forensic. National Institute of Standards and Technology, Gaithersburg, Maryland, USA. https://www.ncjrs.gov/pdffiles1/nij/235710.pdf

DHS., 2015. Oxygen forensic suite 2015-analyst v.7.0.0.408: Test results for mobile device acquisition tool. Oxygen Forensics Inc., Alexandria, Virginia. https://www.dhs.gov/sites/default/files/publications/Oxygen%20Forensic%20Suite%202015%20-%20Analyst%20v7.0.0.408%20Test%20Report_Final_0.pdf

Dimpe, P.M. and O.P. Kogeda, 2017. Impact of using unreliable digital forensic tools. Proceedings of the International World Congress on Engineering and Computer Science WCECS Vol. 1, October 25-27, 2017, San Francisco, USA., pp: 118-125.

Guo, Y., J. Slay and J. Beckett, 2009. Validation and verification of computer forensic software tools-searching function. Digital Invest., 6: S12-S22.

Hildebrandt, M., S. Kiltz and J. Dittmann, 2011. A common scheme for evaluation of forensic software. Proceedings of the 2011 6th International Conference on IT Security Incident Management and IT Forensics, May 10-12, 2011, IEEE, Stuttgart, Germany, ISBN:978-1-4577-0146-7, pp: 92-106.

Horny, M., 2014. Bayesian networks. Ph.D Thesis, Department of Health Policy and Management, Boston University, Boston, Massachusetts.

Houcque, D., 2005. Introduction to Matlab for Engineering Students. Northwestern University, Evanston, Illinois,.

Iacob, I.M. and R. Constantinescu, 2008. Testing: First step towards software quality. J. Appl. Quant. Methods, 3: 241-253.

Irmler, F., K. Kroger and R. Creutzburg, 2013. Possibilities and modification of the forensic investigation process of solid-state drives. Multimedia Content Mob. Devices, 8667: 866-714.

Jaakkola, H. and B. Thalheim, 2010. Architecture-driven modelling methodologies. Inf. Modell. Knowl. Bases, 225: 97-116.

James, I.J., 2018. Survey of evidence and forensic tool usage in digital investigations. University College Dublin, Belfield, Dublin. http://dfire.ucd.ie/?p=858

Kevin, B. and E. Ann, 2004. Bayesian Artificial Intelligence. CRC Press, Boca Raton, Florida, USA.,.

Kragt, M.E., 2009. A Beginners Guide to Bayesian Network Modelling for Integrated Catchment Management. Landscape Logic, Durham, North Carolina,.

Kubi, A.K., S. Saleem and O. Popov, 2011. Evaluation of some tools for extracting E-evidence from mobile devices. Proceedings of the 2011 5th International Conference on Application of Information and Communication Technologies (AICT), October 12-14, 2011, IEEE, Baku, Azerbaijan, ISBN:978-1-61284-831-0, pp: 1-6.

Nidhra, S. and J. Dondeti, 2012. Black box and white box testing techniques-A literature review. Intl. J. Embedded Syst. Appl., 2: 29-50.

Pan, L. and L.M. Batten, 2009. Robust performance testing for digital forensic tools. Digital Invest., 6: 71-81.

Qureshi, M.A., M. Salman and R. Khalid, 2013. Development of a framework for strategic outsourcing in developing countries. Intl. J. Mater. Mech. Manuf., 1: 92-96.

Selamat, S.R., R. Yusof and S. Sahib, 2008. Mapping process of digital forensic investigation framework. Intl. J. Comput. Sci. Network Secur., 8: 163-169.

Van Den Bos, J. and R. Van Der Knijff, 2005. TULP2G-an open source forensic software framework for acquiring and decoding data stored in electronic devices. Intl. J. Digital Evidence, 4: 147-166.

Vandeven, S., 2014. Forensic images: For your viewing pleasure. SANS Institute, Bethesda, Maryland. https://pdfs.semanticscholar.org/0f17/8513b0ca856 62a92a34c9d42a09562a1f0ee.pdf?_ga=2.250350534. 1741136004.1551269898-214331 2028.1535543438

Wilsdon, T. and J. Slay, 2006. Validation of forensic computing software utilizing black box testing techniques. Proceedings of the 4th International Conference on Australian Digital Forensics, December 4, 2006, Edith Cowan University, Perth Western Australia, pp: 1-10.