Correlation of Affiliate Performance against Web Evaluation Metrics

MATHEW J MIEHLING

Commercial-in-confidence

May 2014

Submitted in partial fulfilment of the requirements of Edinburgh Napier University for the degree of Doctor of Philosophy in the Facility of Engineering, Computing and Creative Industries

Abstract

Affiliate advertising is changing the way that people do business online. Retailers are now offering incentives to third-party publishers for advertising goods and services on their behalf in order to capture more of the market. Online advertising spending has already over taken that of traditional advertising in all other channels in the UK and is slated to do so worldwide as well [1]. In this highly competitive industry, the livelihood of a publisher is intrinsically linked to their web site performance.

Understanding the strengths and weaknesses of a web site is fundamental to improving its quality and performance. However, the definition of performance may vary between different business sectors or even different sites in the same sector. In the affiliate advertising industry, the measure of performance is generally linked to the fulfilment of advertising campaign goals, which often equates to the ability to generate revenue or brand awareness for the retailer.

This thesis aims to explore the correlation of web site evaluation metrics to the business performance of a company within an affiliate advertising programme. In order to explore this correlation, an automated evaluation framework was built to examine a set of web sites from an active online advertising campaign. A purpose-built web crawler examined over 4,000 sites from the advertising campaign in approximately 260 hours gathering data to be used in the examination of URL similarity, URL relevance, search engine visibility, broken links, broken images and presence on a blacklist. The gathered data was used to calculate a score for each of the features which were then combined to create an overall HealthScore for each publishers. The evaluated metrics focus on the categories of domain and content analysis. From the performance data available, it was possible to calculate the business performance for the 234 active publishers using the number of sales and click-throughs they achieved.

When the HealthScores and performance data were compared, the HealthScore was able to predict the publisher's performance with 59% accuracy.

Contents

List of Figures

List of Tables

List of Equations

Acknowledgements

I would like to thank my supervisory team, Bill Buchanan and Alistair Lawson for providing me with the skills and training needed to take on this research. I would also like to thank the Financial Services Authority and the School of Computing at Napier University for providing me with the opportunity and funding to pursue this degree. Finally, I would like to thank my parents, Bob and Terri Miehling as well as my girlfriend, Nicola Dowling, for their love, support and most importantly, motivation. Without any of these people, this accomplishment would not have been possible.

1 Introduction

The Internet has changed the way people work, play and stay connected with each other and the world around them. From news sites, to e-commerce and gaming, the Internet has become an intricate part of daily life. One area in which the Internet has instigated major changes is the process with which businesses advertise their goods and services. One popular method, affiliate advertising, is a performance-based sub-set of online advertising in which individuals or companies, referred to as publishers, create a web site specifically to advertise the goods and services offered by another company, called the advertiser. The research presented in this Thesis explores the correlation between the features of these publisher sites and the actual business performance of the publisher on an affiliate advertising campaign.

1.1 Motivations

This research is the product of a Financial Services Authority (FSA) sponsored studentship to explore a topic related to a weakness in e-commerce that has the potential to affect the financial services sector. The exploration into the affiliate advertising industry began with an initial evaluation of the websites and affiliate network profiles of the individuals believed to be involved in a massive affiliate fraud case in the UK in 2008 [2]. The Northumbria police were able to disclose the URLs and related entries from the publisher database of a large affiliate network that was one of the victims of this incidence of affiliate fraud. The associated losses totalled more than £200,000 with a further £215,000 of pending transactions that were cancelled when the deceit was discovered. The fraud was exposed during a manual spot-check when an employee of the affiliate advertising team at one of the affected retailers luckily noticed clues that the transactions may be fraudulent and flagged them with the affiliate network.

After completing the initial evaluation, it was clear that a system capable of automating the review of publishers on these advertising campaigns might be beneficial in preventing losses of this nature in the future. Through working closely with several digital marketing agencies, it was identified that fraud was not the only case of missed revenue in the affiliate advertising industry. A typical advertising campaign can contain thousands or even tens of

thousands of publishers that must be managed by a programme manager in order to ensure that the publisher's sites are good enough to represent the brand and products of an advertiser. Many of these sites fall into what is referred to as the long tail of affiliate advertising. The long tail typically refers to the large number of affiliates on an advertising campaign that receive relatively few customers per month. In an e-consultancy article on the topic, Hewitson notes that from September 2010 to September 2011, the top 10 publishers for one of the top affiliate networks, Affiliate Window, were responsible for 76% of the total sales for the network in that period. Hewitson suggests that the long tail can be cultivated in order to grow those publishers and earn more money for advertisers, but acknowledges that it is hard work, and more of a long-term investment [3]. Several of the digital marketing agencies I was in contact with while conducting this research confirmed that they had also identified the same trend on their campaigns. From these observations, the idea for exploring the link between the features of a publisher site and the real-world performance of that publisher in an affiliate advertising campaign was born. The ability to determine how well a publisher is likely to perform based upon information that is readily available could prove to be useful in the identification of publishers with the potential to perform well along with providing a warning about potentially problematic publishers so that actions can be taken to curtail any issues before they arise. These problematic publishers could be those with a site that is simply under-performing, incompatible with the advertising campaign, poorly designed, breaking terms and conditions, or possibly even hosting malicious content.

Currently, most affiliate networks provide programme managers with access to data that tells them ***how*** their publishers are performing but the tools and information available do not provide an insight as to ***why*** a certain publisher is a "Good" or "Poor" performer. Unless they are able to devote the time and resources necessary to look at each individual publisher site, affiliate managers also do not have a complete picture of how their product or brand is being promoted or whether their publishers are adhering to online best practice and affiliate network terms of service. For these reasons, affiliate advertising is an ideal area in which to explore the correlation between web site features and business performance.

1.2 Aims and Objectives

Current web site evaluation methods are able to determine how easy a site is to use and can even gauge the potential level of user acceptance and satisfaction a site promotes. However, this research is focused on the affiliate advertising domain where performance is measured by how well a campaign goal is fulfilled. More often than not, the campaign goal involves the generation of revenue or brand awareness for the retailer. Therefore, this Thesis aims

> **To explore the correlation between web site evaluation metrics and the real-world performance of a publisher on an affiliate advertising campaign.**

In order to achieve this aim, three questions have been defined to further focus the research.

1.2.1 Measuring Real-World Publisher Performance

The ability to derive a single measurement of the actual level of publisher performance on a campaign based upon individual features of the publisher's site could prove be a powerful tool for affiliate programme managers. In order to meet the aim of this Thesis, the first research question must first be considered:

> **Q1. How can a site's real-world performance be measured and reported in such a way that a comparison between the site's health and business performance can be drawn? HealthScore Calculation**

In order to explore the correlation between web site evaluation metrics and publisher performance, a set of significant web site features must first be identified. Any data related to these features must then be extracted from the publisher sites in order to calculate a score for each feature. These scores can then be used to test each of the features to determine their suitability in measuring overall publisher performance. Finally, a method must be devised that allows for the combination of the individual scores of the features that have been deemed suitable indicators of publisher performance into a single overall measurement for the publisher. This overall measurement will be referred to as a publisher's HealthScore for the remainder of this thesis. Completing this process will assist with answering the second research question:

> **Q2. Can scores derived from the various features of a publisher web site be combined in order to create a useful overall measurement of the site's health?**

1.2.2 HealthScore Validation

Once the HealthScore of a publisher site health has been calculated, the correlation between

the site's health and the business performance of that publisher can then be tested. The exploration of that correlation is directly related to the aim of this Thesis and will also help to answer the main research question:

> **Q3. How well can the HealthScore construct defined by this research be used as an indication of publisher performance on an advertising campaign?**

1.3 Contribution

The work presented in this thesis contributes to the field of knowledge by presenting evidence of the existence of a correlation between the feature scores used to evaluate a publisher's web site and that publisher's business performance on a related affiliate advertising campaign. In order to show this correlation, this Thesis presents six web site evaluation metrics created using ideas and concepts from previous related literature. The metrics defined are URL Similarity, Visibility, URL Relevance, Broken Links, Broken Images and Blacklist Check. A custom-built web crawler examined over 4,000 web sites and extracted the data required to test the suitability of these features in measuring potential publisher performance.

Using this data, the system calculated a score for each of the six features which were then used to compute a single HealthScore. This HealthScore is a new method of evaluating potential real-world publisher performance on an affiliate advertising campaign.

In order to determine how each of the features and the HealthScore related to the real-world performance of the publishers, a measurement of performance had to be defined. Using the metrics of number of sales and number of click-throughs, the system was able to calculate a performance score for the 234 publisher sites for which performance data was available.

With a measurement of performance now available, a comparison could be made between HealthScore and performance of each of the sites. Out of the 234 publisher sites that had associated performance data, the methods defined in this Thesis were used to correctly predict the performance of 137 (59%) of the sites.

The system used to collect and analyse the data presented in this thesis is also a contribution to knowledge. This system is an automated and repeatable evaluation process for affiliate advertising publisher sites.

1.4 Thesis Structure

This section lays out the structure for the remainder of this thesis. Chapter 2 describes the search of the literature surrounding interactive system evaluation and affiliate advertising in order to work towards the answering the three research questions. Chapter 3 presents the methodology used in the initial selection of web site features to be examined as well as the scoring method used for each of the web site features and concludes with an explanation of how the HealthScore and campaign performance scores are calculated for each of the publisher sites. Chapter 4 discusses the systems and processes involved in the gathering of feature data from the publisher sites as well as the initial observations regarding the sites themselves, the network data and the data collected from the publisher sites. Chapter 5 presents the results of the experiments involving the web site features and the comparison of the HealthScore against real-world advertising campaign performance data. Finally, Chapter 6 offers conclusions, a summary and critical appraisal of the research and then suggests areas for future research.

2 Literature Review

2.1 Introduction

This chapter presents a brief overview of the affiliate advertising value chain. The overview introduces the main players and discusses the issues faced by each. Following this introduction, comes a discussion surrounding three types of affiliate advertising programmes commonly employed on the Internet today. These include: pay-per-mille (PPM), pay-per-click (PPC) and pay-per-action (PPA). After describing affiliate advertising, this chapter reviews the relevant current criteria commonly used to evaluate the quality of interactive systems including web sites and web applications. The review also includes a discussion of a selection of the various techniques used to evaluate these criteria. Following that review is a discussion surrounding web analytics and how they relate to the different metrics and techniques used in this research to measure publisher performance. Finally, the chapter concludes with a brief overview of two related forms of affiliate fraud: Impression Spam and Click Spam along with several mitigation techniques employed to combat these types of fraud.

2.2 Online Advertising

There is currently a multitude of techniques being used to advertise and sell products online; however, this thesis is generally concerned with a sub-set of the online advertising industry known as affiliate advertising. Affiliate advertising is a performance-driven industry that offers incentives for individuals and companies to earn an income from advertising goods and services on behalf of retailers. Affiliate advertising is one of the fastest growing online marketing tactics with a 16.7% annual growth rate, which is predicted to hold up through 2016. This growth even outpaces the current top performer in online advertising, paid search [1].

As a consequence of the recent growth of this market, the global number of participants grows rapidly day-by-day. While speaking with a large unnamed affiliate network with a strong presence in the UK, the team learned that the network receives 60-70 new publisher applications per day and currently boasts a portfolio of over 60,000 active publishers. As

the number of publishers enrolled in these programmes continues to grow, it becomes extremely difficult for programme managers to keep up with the evaluation of the publisher sites on the campaigns they manage.

2.2.1 Key Players

In nearly every type of affiliate advertising programme, four key players exist: the user or customer, the publisher or affiliate, the affiliate network and the advertiser. A fifth player has also begun to break onto the scene: the digital marketing agency. There are also some cases in which the advertisers will create an in-house affiliate department to look after their own affiliate advertising programme without the help of an affiliate network or digital marketing agency, but these cases are few and are often reserved for very large advertisers such as Google, Amazon and eBay.

Industry professionals tend to use different terms for the various key players depending on where in the industry they work. These names below may be used interchangeably throughout the rest of this thesis.

2.2.1.1 The User or Customer

The user or customer is generally a normal Internet user looking to complete a purchase online or to sign up for an online service. According to a recent Forrester survey, 55% of respondents indicated that they look for deals or voucher codes related to an online transaction they wish to complete, while 32% said that they often begin their online shopping session by visiting a publisher site. The survey also found that these users visit an average of three different publisher sites before purchasing to ensure they have found the best deal available [1]. The results of the survey indicate that the primary goal of users is to save money on their online purchases. Several authors in the field have found that users prefer and will more often user publisher sites that are highly usable [4] [5] [6].

Although the research presented in this Thesis does not directly address the primary goal of users to save money, a publisher that has access to the HealthScore calculate by this system would be able to make changes that positively affect the usability of their site and thus improving the user's experience.

2.2.1.2 The Publisher or Affiliate

The publisher, or affiliate, is an individual or company that earns commission on purchases

made by customers that they have directed to the advertiser's site. Commission rates vary between advertising campaigns, but the campaigns examined in this Thesis generally had a majority of the publishers earning 3-5% commission on a relatively new campaign. Some programmes varied the amount of commission paid depending on previous experience with a publisher, and occasionally rewarded high-performing publishers with a higher commission rate. Forrester reported that commission from affiliate networks continued to be the biggest spend in affiliate advertising through 2012 [1].

Publishers participate in affiliate advertising programmes to earn revenue for influencing users to complete various actions on their site. These actions benefit an advertiser in some way, and that advertiser then pays commission to the publisher. Hasan, Morris and Probets found that users satisfied with their browsing session on a site were more likely to return [7]. In order to increase their revenues, publishers are continually looking to improve their sites as sites with low usability have been linked to revenue loss [8] [9]. The system presented in this Thesis directly relates to helping publisher solve this problem by highlighting the strengths and weaknesses of a publisher's site in regards to overall health. With this knowledge, a publisher should be well equipped to improve their site's usability and increase their uptake and sales volume [6].

2.2.1.3 The Affiliate Network

Affiliate networks such as Webgains, OMG, TradeDoubler, Affiliate Window, Buy.at and Commission Junction act as intermediaries between the advertisers and the publishers. The affiliate networks usually provide advertisers with a management dashboard and other tools that allow them to quickly view the statistics for their advertising campaigns as well as edit the details of their campaign.

The affiliate networks help publishers by making it easier to locate and sign up for advertising campaigns that are relevant to their web sites. The networks also usually provide the publisher with statistics related to how well their site is performing, similar to the dashboard offered to the advertiser.

Aside from offering tools and information to help both advertisers and publishers to manage their campaigns more effectively, the affiliate network also handles billing the advertisers and paying the commission due to the publishers. For their services, the affiliate networks take a small percentage from both the publishers and advertisers with some

networks also charging a recurring access fee to advertisers in order to use the management dashboard and other tools. The only direct benefit to affiliate networks from this research is related to the long tail of an affiliate advertising campaign, which often includes a large percentage of the publishers on that campaign [3]. By reviewing the various feature scores of the publishers on a campaign, it may be possible for an affiliate network to identify the common factor amongst long tail publishers in order to help boost the effectiveness of those sites.

2.2.1.4 The Advertiser or Merchant

The advertiser or merchant is the retailer that actually sells the goods or provides the services being advertised by the publishers. Advertisers can range from large companies like Tesco or ASDA down to small businesses run from someone's home. Some advertisers will hire a member of staff specifically to act as their programme manager in charge of running their affiliate programme. Their role generally involves recruiting publishers and ensuring that their site content is appropriate for the advertising campaigns. However, with the increase in size and complexity of advertising campaigns, advertisers have begun to outsource these responsibilities to third-party companies such as digital marketing agencies, and the role of the affiliate manager in these advertisers has shifted to acting as a liaison between the advertiser and the agency. Some of the larger advertisers such as Google, Amazon and eBay have gone as far as to skip the use of an advertising network and their in-house affiliate advertising departments deal directly with their publishers.

Pricewaterhouse Cooper (PwC), along with the UK branch of the Internet Advertising Bureau (IAB), have recently published a report related to online performance marketing in the UK. The report found that in the UK alone there are 3,000-4,000 advertisers spending a total of £814 million on advertising in this market with the finance sector doing 45% of the spending and online retailers being the next highest spenders at 20% [10].

2.2.1.5 The Digital Marketing Agency or Outsourced Programme Management (OPM) Company

Digital marketing agencies offer their clients a range of services dealing with brand management, online marketing and some also offer affiliate programme management. It is this last role that is referred to when digital marketing agencies are mentioned throughout the remainder of this thesis.

When an advertiser hires a digital marketing agency or OPM to manage their affiliate programme, the agency generally takes on the role of managing the publishers and dealing with the affiliate networks on behalf of the advertiser. The role played by the agency will generally include services such as providing advertising content to the publishers, recruiting new publishers to the campaign, auditing publisher performance throughout the campaign, managing the billing process and tracking sales/leads using the tools provided by the affiliate networks as well as various third-party programmes.

The digital marketing agencies involved in the studies presented in this Thesis conduct periodic reviews of the affiliate sites in the campaigns they are managing. These reviews are a method of auditing publisher performance and also allow the agency to ensure that the site conforms to their terms and conditions as well as those of the various affiliate networks and advertisers involved. The agencies indicated that publisher site reviews are often manual and time consuming, leading to the agency conducting spot checks on a subset of the publisher sites rather than reviewing each page on all of the publisher sites.

This research presents a methodology that would allow the agencies to conduct their periodic reviews in an automated, repeatable and unbiased manner. The automation of the review process could greatly improve the speed and frequency of the periodic reviews and also allow the agencies to get a more complete overview of the health of a publisher site due to the crawler analysing every page rather than the random selection of pages that the human evaluator would choose.

2.2.2 Types of Online Advertising

Vakratsas & Ambler define advertising as a means of influencing a customer to purchase the advertised product over a competitor's product or products [11]. The traditional method of achieving this goal has been through bombarding print, television and radio with advertisements in order to maximize the effective reach of the advertising message. Effective reach refers to the amount of people who are exposed to the advertisement and the goal of maximising this reach is to greatly increase the potential that people from the target demographic will see or hear the advertisement and will be influenced to purchase the product [12].

As the world becomes increasingly reliant on the Internet, advertisers selling products and services are moving from more conventional advertising mediums and turning toward the

Internet as a means of reaching a much wider and diverse audience. These advertisers are also becoming increasingly open to the idea of third party publishers advertising those products and services on their behalf [13].

In the past, most large advertisers had a few agents, distributors and resellers with whom they maintained personal relationships that were managed at a one-to-one level. However, the Internet enables this activity to be undertaken on a much larger scale and the predominant example of this is reflected in online advertising.

There are two major types of online advertising discussed in this thesis: display advertising and paid search. The main focus of this research is on display advertising, but a brief overview of paid search is also presented for completeness.

2.2.2.1 Display advertising

Display advertising generally includes a banner or other form of clickable advertisement that contains more than simply text, although a simple text link can also be used. There are several different types of display advertising programmes available to publishers including:

a) Pay-per-mille (PPM)

Pay-per-mille programmes use a unit of measurement called an impression which is registered when a unique user views an advertisement on the publisher's web site. In these programmes, the advertiser pays the publisher a set fee for every 1,000 impressions [14]. The PPM model of affiliate advertising is the most closely related to traditional advertising in other mediums such as television, print and radio. In a PPM programme, the advertisement may be shown to uninterested parties and therefore a portion of the advertising spend is wasted.

b) Pay-per-click (PPC)

In a pay-per-click programme, publishers display an advertisement for the products or services of the advertiser and the advertiser pays a small amount to the publisher for every click on the advertisement. These programmes can often generate clicks by users outside of the target audience which could result in a low conversion rate and wasted advertising spend [15]. Another problem that has become common in PPC campaigns is that the publishers will often use advertising content that is geared toward enticing users to click on the advertisement rather than properly explaining the goods or services for sale [16]. This is

likely because the publisher earns money based on the number of click-throughs to the advertiser's site and does not need to be concerned with whether the user will buy a product or service.

c) Pay-per-action (PPA)

In a pay-per-action programme, publishers are paid based upon attracting users to the publisher's site to complete a conversion. A conversion is an action from a set of pre-agreed actions that the advertiser considers to be favourable. Conversions can be actions such as completing a purchase at the advertiser's site, signing up for a newsletter, or may even include applying for a credit card or similar financial product [17].

The benefit for the advertiser in using the PPA model of affiliate advertising is that, in theory, the advertiser is only paying for the advertising that directly results in a favourable outcome [17]. Although the PPA model can result in fewer conversions for a publisher, the programmes examined during this study started at a 3-5% commission for new publishers which could net an effective publisher a substantial amount of commission on a PPA programme.

Throughout the remainder of this thesis, when affiliate advertising is mentioned, the focus will be primarily on the PPA model of affiliate advertising unless specifically stated.

2.2.2.2 Paid Search

Paid search, while not technically grouped in with affiliate advertising, often works similar to a PPC campaign. Search engines are in the unique position of knowing exactly what product or service the user is looking for, and that information allows them to flourish in this model of online advertising. Both advertisers and publishers use paid search to drive traffic to their sites.

In paid search, site owners pay for a spot in the sponsored search results for their choice of keywords and each click-through sent from the search engine costs the site owner a small amount of money. The amount of money per click may be a flat-rate agreed upon between the site owner and search provider, or it may involve real-time bidding on keywords based upon parameters set up by the site owner beforehand. The method of bidding varies between search providers, and some providers offer several options on how to bid for keywords [18]. However, the mechanics of how keyword bidding in paid search works are

beyond the scope of this thesis.

The Internet Advertising Bureau (IAB) have reported that paid search programmes grew by 6.8% from the first half of 2008 into 2009 to £1.06 Billion, which accounts for 60% of the total online advertising expenditure [19]. As such, it would be interesting to see if the results of this study extend to the realm of paid search.

2.2.3 How it works

Despite their subtle differences, all of the models of affiliate advertising that have been discussed throughout this section can be described using Figure 2.1. In general, the publisher is responsible for attracting users to their web site. Once a user has arrived on the publisher's site, the publisher will have a banner advertisement or some other method of promoting the advertiser's products and services. In a PPM type programme, the publisher will have earned a small amount of money simply from showing the advertisement to the user.

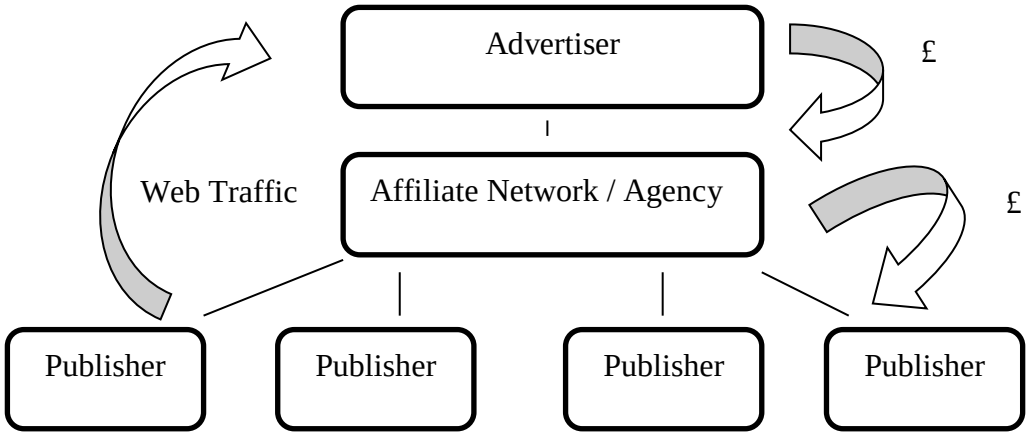

Figure 2.1 Affiliate Advertising Overview

In any other type of programme, the publisher will need to provide a reason for the user to click on the link to the advertiser's site. This may be compelling content or it may even be something useful to the customer such as a discount code. Once the user has navigated to the advertiser's site, most current implementations will place a tracking cookie onto the user's system. This cookie will be used by the advertiser and the affiliate network to determine which publisher sent the user to the advertiser's site so that the correct publisher is credited with any commission earned. The cookie will remain on the user's system for a pre-determined amount of time that can vary from campaign to campaign in order to credit the publisher even if the use does not make a purchase immediately. In a PPC campaign,

the publisher would earn a small amount of money for sending the user through to the advertiser's site.

Although the user is now on the advertiser's site, the publisher will not have earned any commission yet on a PPA programme. Once the user has completed a specific action such as making a purchase or signing up for a service or newsletter, the publisher referred to by the user's tracking cooking will then be credited with the conversion and finally earn commission [20].

As one can imagine, it can be difficult to drive traffic to a publisher website, and getting the traffic is only half of the battle. In order to get that elusive conversion, the publisher must keep their site updated with fresh, interesting information [21] as well as performing routine maintenance to check for things such as broken links that may occur when external content is taken down or moved to a new location. Because of the effort required to stay on top of the affiliate advertising game, digital marketing agencies and networks have been battling with the issue of affiliate advertising campaigns having a large percentage of their publishers that have very few conversions each month. These publishers are said to be a part of the long tail of affiliate advertising. While these publishers may not perform very well individually, the sheer number of them on a typical programme can account for 20-30% of the revenue per month in some cases [22].

2.2.4 Common Industry Performance Metrics

One of the major issues with traditional advertising is that it is very difficult to determine if advertising efforts are being wasted or are actually bringing new customers to a business [23]. The major benefit of online advertising versus advertising on traditional media like print, radio and television is that it is much easier to collect accurate statistics as to how the users are viewing and interacting with advertisements. This interaction is measured using a variety of metrics, the most important of which are introduced below.

2.2.4.1 Impressions

There are two main ways in which advertisements are seen by users: pulled advertisements are those advertisements that have been requested by the user's browser, and pushed advertisements are those that have been sent to the user via correspondence such as a newsletter or e-mail list [14].

As a metric, the number of impressions is the measurement of how many times a particular

advertisement has been seen by a unique user. An impression is counted regardless of whether a user interacts with, or even pays attention to an advertisement [15].

2.2.4.2 Click-through Rate (CTR)

When a publisher presents an advertisement or a link to an advertiser's site to the user, the user can either choose to ignore the advertisement and continue browsing or the user can choose to click-through to the advertiser's site. In the case of a user clicking on the link, a click-through is registered for the publisher.

One of the various metrics used by networks and agencies to determine how well a publisher is performing is the ratio of clicks to impressions. This measurement is the publisher's click-through rate (CTR) [17].

While CTR is commonly used to evaluate how well a particular campaign is performing, Dalessandro et al. found in a recent study [16] that this measurement alone is a poor method of determining publisher success. The authors concluded that a user clicking on an advertisement is not highly correlated with that user making a purchase at the advertiser's site. The authors have also suggested that the industry move away from using creative designed simply to entice users to click on advertisements and move toward creative that explains the products or services offered by the advertiser, and thus genuinely interests the user in visiting the advertiser's site [16].

2.2.4.3 Conversion Rate (CR)

A conversion is recorded when a user completes one or more specified actions on the advertiser's sites. These actions are generally constrained to those which provide the advertiser either with direct revenue as is the case with the user making a purchase, or information that can lead to revenue generation as can be the case when a user provides personal details when signing up to an advertiser's service such as newsletter or e-mail list.

Conversion rate is one of the most common performance metrics used to determine publisher performance on an advertising campaign. The conversion rate is the ratio of clicks generated by the publisher to the number of conversions generated by the publisher [15].

2.2.4.4 Cost-per-revenue (CPR)

Cost-per-revenue is one of the performance metrics in the world of affiliate advertising that

is closely related to a metric from traditional advertising. Like return on investment (ROI), this metric considers not only the number of conversions but also how much revenue those conversion have generated. Using this metric, advertisers are able see exactly how much it costs for a particular publisher site or advertising channel to generate one dollar [24]. Due to not having access to the revenue data necessary to compute the CPR for the sites examined, this study was unfortunately unable to use CPR as a performance measure.

2.2.4.5 Bounce Rate

Unlike the other metrics described in this section which all consider a higher score to be better, a lower bounce rate is considered to be better. A bounce is defined as a site visit that ends immediately after a user visits the landing page and does not interact with anything else on the site. The Bounce Rate is measured as the percentage of all visits that are bounces [25]. A user will generally bounce if the landing page does not display the information that the user expected to find on the site, but may also produce a bounce when accidentally clicking on an incorrect link in a search engine or other web portal and then clicking back when the mistake has been realised. Performance data related to bounce rate was not available during this study, and so it could not be used as a measurement of performance.

2.2.5 Analytics

While measuring the performance of a site is useful to publishers, collecting information about the site's users and being able to predict their behaviour are crucial to maximising a site's advertising potential [26].

As the affiliate advertising industry grows, it is becoming a more competitive environment with publishers and advertisers bidding against each other for keywords in paid search campaigns in their struggle to attract the same demographic of customers. Analytics have long been used in traditional advertising to both understand the consumer as well as to help formulate a coherent marketing strategy, and the competitive nature of affiliate advertising has caused the use of complex analytics to infiltrate the online advertising environment as well [27].

There are two general methods of retrieving analytics data: log file mining and page-tagging [28]. Both of these approaches can be adapted to the affiliate advertising industry and, in that context, these approaches share the same goal: understanding how and why a

user's experience ends in either a conversion or non-conversion. The original methods of collecting web analytics relied on log file analysis. In this method, parsing the log files can reveal information about web site visitors [28]. This information can include metrics such as:

- The last page a visitor viewed on the site or the exit page [29] [30]
- The search terms that lead the user to the site [29] [30]
- The referring site [29] [30]
- The first page the visitor saw on the site or the entry page [29] [30]
- Any errors that occurred during the site visit [29]
- The path that the user took through the site [29]
- Information about the operating system and browser of the user [29]
- The time the user spent on the site [30]

While log file analysis was able to provide a wealth of information, there were inaccuracies created by this method [28]. For example, counting unique visitors is difficult using log file analysis due to the widespread use of NAT technology on the Internet.

The issues with log file analysis helped to usher in the next generation of web analytics: page-tagging [28]. In a page-tagging system, code is inserted into the pages of a web site in order to gather statistics about visitors and traffic patterns [31]. Page-tagging is considered to be far more accurate than log file analysis because cookies allow the system to more accurately determine unique visitors and web crawlers do not activate the scripts on the page and therefore are not logged [15].

Advanced tools and services based on the page-tagging techniques such as Google Analytics allow users to view detailed statistics about how visitors interact with their site and to compare the benefits of traffic from different sources against one another [32]. The information provided about these visitors and the way they interact with the site can help a publisher to recognise which traffic sources bring high-quality traffic and which sources bring traffic that rarely converts despite the publisher's best efforts [33]. Another service that uses page-tagging is Amazon's Alexa. The main difference between Google Analytics and Alexa is that the data analysis one by Alexa is searchable on the web and available for all to see [34]. Both Google Analytics and Alex have also evolved to user techniques beyond simple page tagging. Google Analytics relies on anonymous usage data sent to the service by users of the Chrome internet browsers and Alexa boasts that data is collected

from a variety of browser extensions [34]. This usage information allows both services to estimate traffic data even for sites that have not implemented their page tagging techniques.

The use of web analytics is not limited simply to examining visitor behaviour. Interest in using analytics to predict future customer behaviour and segmenting the market based upon the results has been increasing amongst retailers [35]. Segmenting a market involves separating the existing and potential customers into a variety of groups based on several criteria. Segmenting the market allows a publisher to tailor his or her efforts to appeal to each individual segment in a unique way [36]. This extra effort can lead to an improved click-through rate for a site, meaning increased revenue for the publisher [37]. This increased revenue can be attributed to the use of techniques such as personalised landing pages based upon the search query that brought the user to the site [38].

Strategies to segment the market in affiliate advertising tend to revolve around an analysis of the referral keywords that brought the user to the site. Early research in the field used search engine logs in an attempt to categorise intent through analysing which links were clicked for each query [39, 40]. However, several studies on contextual advertising have found that personalised landing pages based upon the user's short-term search and browsing behaviour rather than simply the final search result that directed the user to the site can lead to an increase in click-through-rate [41, 37, 38].

Another aspect of understanding the consumer is to understand the user experience. Engaging the users of a web site can lead to a more enjoyable experience which increases the chance that the user is more likely to return to the site in the future [42, 43]. Returning customers are less likely to bounce, and also tend to spend more time on the site [7].

Web analytics can provide a vast amount of information that is essential to improving a site through gaining an understanding of the site's users. The downside to analytics is that completing this type of analysis requires a heightened level of access to the web servers hosting the site content. In order to set up a system like this, one would need the ability to access the log files of the web servers, modify the code of the pages on a site, or to find willing participants to install a third party program that will conduct the monitoring.

2.2.6 Impression Spam

As outlined in section 2.2.2.1, a publisher in a PPM programme earns commission for every 1,000 impressions of an advertisement. One method of defrauding this type of

programme is known as impression spam. Impression spam involves sending requests for pages that contain advertisements that users will never see, or that are simply being sent to inflate the number of impressions. As these advertisements are not reaching actual users, advertisers should not be charged for these impressions [44].

The most basic method of sending these requests is users manually refreshing pages, but that type of impression spam is low-tech and is relatively easy to detect compared to using impressions generated through malware. In 2010, Click Forensics discovered a piece of malware that opened a new browser window called a pop-under because it is opened under the currently active window. In this pop-under window advertising banners were rotated every 10-15 minutes and aside from simply displaying ads that the user was unlikely to see due to the pop-under window being covered by the active browser window, the malware would also click on these ads occasionally. This malware was earning revenue for malicious publishers through both impression spam and click spam [45].

Impression spam not only drains advertiser's budgets, but can also drastically reduce the CTR of the publisher by inflating the number of impressions without changing the number of clicks unless used in conjunction with click spam techniques. This can be detrimental to a publisher on any type of campaign, other than PPM, as some campaigns measure performance based upon CTR. An artificially inflated amount of impressions can cause the publisher to appear to be performing worse than they would without the impression spam, and so impression spam may be tempting to use against competing publishers.

2.2.7 Click Spam

In order to earn a commission in a PPC programme, publishers must rely on users clicking on their advertisements. In some cases, these clicks may originate from sources other than legitimate users. Clicks with an illegitimate source are often referred to as invalid clicks, and affiliate networks have proprietary detection techniques in order to filter out these invalid clicks so that advertisers are not charged for them. In the cases that an invalid click is detected after an advertiser has been billed, the network will issue a refund.

When one of these invalid clicks is issued with malicious intent, it is considered click fraud or click spam and can be used by malicious publishers in order to artificially inflate the number of clicks registered on their ads without providing any additional benefit to the advertiser [46]. While it is impossible to know the intent of the user or agent that has

clicked an advertisement with 100% certainty, methods have been developed to identify invalid clicks versus accidental double clicks, or other inadvertent sources of invalid clicks. These methods generally look at several aspects of a user's visit such as how many pages on the site were viewed, how many paid clicks are from the same IP address, whether the user has a tracking cookie, how quickly the user is able to traverse the pages of a site and several other aspects of the visit [47].

In the event that an invalid click is detected, the user is still directed to the requested site even though the click is not charged to the advertiser. This is done because it may be the case that a legitimate user's click was mistakenly marked as invalid and preventing a redirection to the advertiser's site would create a poor user experience for the potential customer, and could also cause a loss of revenue for the advertiser.

There are several non-malicious scenarios in which a click may be marked as invalid [44]:

- Some web crawlers may inadvertently request content that would result in a click, and these clicks are often caught and marked invalid.
- Some users may accidentally click more than once on an advertisement by double-clicking as they would when opening an application. Most tracking algorithms in place will recognise this behaviour and mark the excess clicks as invalid.

Click fraud is thought to be a common occurrence, although according to Click Forensics the click fraud rate across the 300 advertising networks they monitored in Q4 2010 was down to 19.1% from 22.3% the previous quarter [48]. It has also been reported that programmes in which the advertiser's cost per click is a flat rate that is known to the publisher had more fraud than programmes that involved keyword bidding in which the publisher did not have direct prior knowledge of the price per click [49].

Publishers are not the only members of the affiliate advertising chain that can have malicious intent. Malicious advertisers may purposefully use click spam in order to drain a competitor's paid search advertising budget for the day to gain control over high value search keywords [44]. The reasoning behind this type of click fraud is two-fold:

1. Deplete the competitor's advertisement budget without giving them any conversions, and therefore causing them to lose out on potential revenue for the day.

2. Increase the likelihood that conversions that may have gone to the competitor will instead go to the malicious advertiser as they are generally bidding on the same keywords.

No matter which malicious party is committing the fraud, the methods are generally the same. The most basic form of click spam involves users purposely clicking on an advertisement link with no intention of completing a purchase. This can be achieved with just the individual publisher clicking on advertisement links, but this method of click spam is easily detected by advertising networks and will classify all clicks from that user as invalid if detected. In order to avoid detection, the user would either need to wait a set period of time between clicks to defeat the detection algorithms as Li, Zeng and Wang observed [49] or use an anonymising proxy service such as Tor to repeatedly change their IP address, making the fraudster appear to be multiple users [50].

While it is possible for a single person to generate a significant amount of traffic in one day, the major players in click-spam are actual businesses with professional looking web sites. These companies derive profit from paying other people to visit publisher sites and complete conversion actions as well as earning revenue from advertisements on their company site. The programmes run by these companies are generally referred to as pay-to-click (PTC) or pay-to-read (PTR) and the terms are often used interchangeably. The distributed nature of these programmes makes it very difficult for affiliate networks to distinguish the actions of users on these programmes from legitimately interested users [49].

The users that take part in these programmes are paid a small amount per action, and the PTR company pockets the remainder. These actions range from simply visiting websites that display PPM banners, to clicking on PPC ads and filling in surveys on PPA sites. Many of these sites even offer affiliate programmes of their own in which users can refer others to the PTR site. The referring user will then earn a very small amount of revenue each time the referred user completes a task [51, 52].

Another popular method of producing click spam is through the use of botnets specifically designed to click on the author's ads, also known as clickbots. With this method, the fraudster must first amass a collection of infected computers, or bots, on different networks that are preferably unrelated and distributed geographically to help disguise the clicks with

legitimate click traffic [53]. Once the botnet has enough infected computers, the bot master simply instructs each computer to visit the publisher page and complete any action or actions that will earn revenue. This method presents the same problem as the PTR/PTC sites in that the infected computers may be distributed throughout the world, and the clicks are often difficult to filter out from real clicks.

There are also a number of ways to coerce users into clicking on advertisement links without their knowledge. This is known as forced click or cookie stuffing, and can be accomplished in multiple ways. Often in a PPA campaign, the publisher's site will place a cookie on a user's machine which is then read by the advertiser's site to determine which publisher is entitled to the commission is the user converts. Malicious publishers have found various methods of putting these cookies on a user's machine without user interaction and these methods can often overwrite legitimate cookies, thus denying other publishers commission that they have rightfully earned [20].

Gandhi, Jakonsson and Ratkiewicz were able to create a snippet of JavaScript code that would automatically register as a click when a page with the script on it was loaded [54]. This attack could be present in a malicious publisher's page, a legitimate page to which the malicious publisher has gained access, an e-mail message or even a forum post. Once the content is viewed, the exploit will have taken place completely invisible to the user. In order to avoid detection, the exploit was designed to only register clicks for a random percentage of visits [54].

The JavaScript code used in the attack exploited a popular method of cookie stuffing by creating an invisible iFrame which the malicious publisher site used to load the advertiser's page without the user requesting it. In fact, the user could not even see the contents of the advertiser's site as they were loaded in the invisible iFrame [55, 56].

Another method of forcing clicks out of users is through the use of malware. The malware used in this type of fraud can come from a variety of sources, but two common sources are drive-by-downloads and fake anti-virus software. Drive-by-downloads are pieces of malware that are downloaded and installed without requesting the user's permission or requiring any user interaction [57]. This type of malware is dangerous because users often do not know they are infected. On the other hand, the user must install fake anti-virus software willingly in most cases. The fake anti-virus infection sites play upon the user's

fear of downloading a virus or other malware by popping up a message that the computer is already infected and that the software can clean the computer if installed immediately. Li, et al. discovered a fake anti-virus campaign in 2011 that included 24 advertising networks and 84 fake anti-virus sites which were rotated to avoid being detected. The authors noted that when the website was accessed, there was an advertisement displayed as well as an invisible iFrame which started the redirect chain used in this attack which included Google as well as DoubleClick, although these entities were very likely unaware of their inadvertent involvement in the scam [56].

In 2012, the FBI uncovered and took down a large-scale affiliate fraud scheme in *Operation Ghost Click.* The operation is regarded as the largest cyber takedown in history as the fraudsters had earned $14 million over the course of four years [58, 59, 60]. Alrwais, et al. conducted testing on the attack infrastructure before it was taken down by the FBI and have created an in-depth report that offers a unique insight into how the attacks were conducted [61]. According to the authors, the attack changed the DNS settings on a user's machine to point them toward attacker-controlled DNS servers in Eastern Europe. When an infected user visited a legitimate publisher page and requested one of their advertisements, the attackers DNS server would instead send back one of the attacker's advertisements. Not only did this earn revenue for the attackers, but it also took revenue away from the publisher site that user was actually visiting. This technique also made it look as though the invalid clicks were coming from legitimate publishers [61].

2.2.8 Mitigation Techniques

The most common mitigation technique employed against click fraud is a direct traffic measurement technique known as clickstream analysis [62]. Clickstream analysis takes an in-depth look at the incoming clicks and uses a set of heuristics to look for conditions known to be associated with click spam [63]. One popular form of detection is searching the clickstream for duplicate clicks in a process the industry knows as de-duplication or simply de-duping [64]. Although not all duplicate clicks originate from fraudulent means, this research is only concerned with those duplicate clicks that are fraudulent.

Several techniques exist for de-duping, the majority of which rely on Bloom Filters. A bloom filter is a data structure that can be used to test whether or not an object is in a set. Bloom filters work well for detecting duplicate click data because they cannot generate

false negatives. In the context of detecting duplicate clicks, this means that a classic bloom filter-based implementation should mark every duplicate present [64]. One downfall of classic bloom filters is that there is a possibility of false positives which means that due to hash collisions, a unique click may be marked as a duplicate.

More recent work has begun to use modified versions of bloom filters in order to make them more efficient in this field. Zhang and Guan have introduced an algorithm they call Group Bloom Filters (GBF) in order to significantly reduce the memory requirement when compared to classic Bloom Filters in certain cases [65]. In the event that the GBF algorithm would still use many memory operations, the authors have also introduced an algorithm called timing Bloom Filters (TBF). The TBF algorithm introduces timing information that allows for stale or expired data to be removed in order to conserve memory [65].

Wei et al. proposed another version of bloom filters designed to reduce RAM requirements known as Detached Counting Bloom filter Array (DCBA) [66]. Each DCBA is an array of Detached Counting Bloom Filters (DCBF), which are essentially Bloom Filters that are associated with a counter that can be offloaded to a disk when full. One of the benefits of this algorithm is that the DCBA only needs to keep 10% of the elements in RAM, and yet retains 95% of its searching performance.

Hybrid detection systems use a combination of rule and anomaly-based methods, and often include both real-time "online" detection that attempt to classify a click based solely on a small amount of clickstream data as well as "offline" detection that is able to take advantage of the vast amount of data that networks store across multiple advertising campaigns [44]. Google has revealed that both rule and anomaly-based systems play a major part of their infrastructure despite being secretive about how their detection systems actually work [67].

Another method of combating the rising number of malicious publishers is to prevent them from joining a programme in the first place. Edelman outlines that it may be possible to prevent up to 71% of affiliate fraud by preventing malicious publishers from joining an affiliate programme altogether. He found that if an advertiser pays their publishers in arrears with compensation to offset the extra time before payment is received, there exists a certain point at which it is no longer profitable for fraudsters, or bad-type agents as he calls them, to participate in the programme [68]. Unfortunately, according to a more recent

survey of over 450 affiliates, 57% of good-type affiliates decide whether to join an affiliate programme based upon how often the programme pays out [13]. With the majority of affiliates basing their preference of programme on how soon they can start earning, it may be difficult for the first few networks that start extending that waiting period as it may decrease the number of good affiliates an advertiser or affiliate network can attract unless Edelman's solution is adopted en masse amongst the top affiliate networks.

Another interesting prevention mechanism involves displaying fake or bluff advertisements in programmes that automatically choose which advertisements to display [69]. In general, the method for choosing which advertisement to display on a page is to examine several factors that make up a user "profile". These factors vary from network to network, but generally take into account things such as keywords for the web page, information gleaned about the user from cookies on their machine, referring page, etc.

The bluff advertisements make use of this same user profile information, and are inserted on pages where they are highly unlikely to be clicked because the displayed text is totally unrelated to the user profile. This type of advertisement should only receive a high number of clicks from automated click-bots or poorly trained humans on PTR/PTC programmes. If a user clicks on a high number of these advertisements, the user is marked as suspicious `and their click activity can then be manually reviewed offline. A second type of bluff advertisement designed to target botnets with built-up user profiles contains specialized text, but is randomly displayed rather than targeted at the user profile. The authors feel that only a malicious user would click a large number of this type of bluff advertisement as a genuine user is unlikely to click a large number of unrelated advertisements [69].

Another method of click fraud prevention to be proposed is the idea of clickable CAPTCHAs (Completely Automated Public Turing test to tell Computers and Humans Apart) [70]. In this system, a user must complete a simple Turing test [71] before being redirected to the publisher's site. The example described by the authors showed pictures of cats and dogs and asks the user to click on a specific animal. This system works because the pictures are already manually classified for use by Petfinder.com, which provides access to their database in return for a link to adopt a pet being placed under the CAPTCHA. It was shown that 99.6% of the time, a human user was able to solve the CAPTCHA in less than 30 seconds while computers attempting to solve the same type of CAPTCHA had a

significantly lower success rate of around 1/54,000. The authors also suggest that while users may become frustrated by text-based CAPTCHAs, that clicking on pictures of cats and dogs is a more enjoyable experience and as such should not put users off using sites that employ this clickable CAPTCHA system [72].

2.3 Interactive System Evaluation

Companies and individuals often have a goal in mind when creating interactive systems such as web sites and web applications. One of the goals often involves the generation of revenue either directly through online sales, indirectly through raising brand awareness or in the case of online advertising, through earning commission by directing users to a retailer's site [16]. Unfortunately, as Dingli and Mifsud point out, although the importance of usability in these systems has been widely noted by academics and usability professionals, businesses often overlook the evaluation of the factors associated with good usability during the development process [73].

2.3.1 Measuring Website Success

Determining the best method of measuring the success of information systems has been a long-standing question, but in 1992 DeLone and McLean published their Information Systems (IS) success model (which will be referred to as the D&M success model for the remainder of this thesis) which has been widely referenced since [74]. The D&M success model introduced a multi-dimensional concept of IS success [75] which has since been updated to account for the explosion of e-commerce. The new model depicts IS success as a combination of system quality, information quality, service quality, system use (or intent to use), user satisfaction and net benefits [74] although the authors do note that the definition of success may vary depending on the business sector and purpose for the system [75].

An example of varying measurements of success lies with Google. Contrary to almost every other web site, the goal of Google’s main search portal is to ensure that visitors to the site leave as quickly as possible [76]. To that end, many of the features of the D&M success model do not apply to the Google page, and so a range of evaluation metrics with weightings tailored to the individual website being evaluated is needed to accommodate different web strategies and the features associated with determining success.

Phippen, Sheppard and Furnell maintain that some success measures relating specifically to

e-commerce have generally included measuring criteria such as return on investment (ROI), profitability, effectiveness, reliability, utility or competitive advantage [27].

In a study similar to the research presented in this Thesis, Lee and Kozar attempted to match user rankings of business website to the performance of the businesses [77]. The authors distributed surveys asking a group of online customers and a group of managers/designers to rank websites based upon information quality, system quality, service quality, and vendor specific quality. These four factors were then divided further into 14 sub-factors. In order to measure the business performance of sites, Lee and Kozar used the COMPUSTAT financial database to look up the Return on Assets (ROA) and Return on Equity (ROE) for each of the companies. The authors found that site preference rankings determined by the users closely matched the business rankings of the site with only a few exceptions [77].

2.3.1.1 Usability and Accessibility

There are several dimensions common throughout the different approaches to interactive system evaluation, but usability and accessibility are often near the top of the list as two of the most important measurements in this area [4] [5]. One definition of site or application usability is the ease with which a new or unfamiliar user can efficiently use the product without prior specific instruction [78].

As Dingli and Mifsud point out, the academic and professional community have long recognised the need for these evaluations [79, 80, 73, 8, 81]. Nielsen and Norman [4] found that users are much more likely to leave an e-commerce site in favour of a competing site than they would in a brick-and-mortar store because the "cost" of switching e-commerce sites is relatively low when compared to the same cost in a physical store. This low cost of switching sites can lead to a loss of revenue, which both Ruiz-Rodriguez and Montero et al. attribute, at least partially, to poor site usability [8, 9]. Unfortunately, it seems as though the commercial sector has not fully realised the vast importance of usability evaluation yet [82].

Websites that rate highly for usability have been shown to have a better chance of increasing their uptake and volume of sales [6], which means more revenue for the publisher, advertising network and any third party agencies involved in managing the advertising campaign.

Awareness of the inaccessibility of sites currently on the web is on the rise [83], and legislators have begun stepping up to help work toward a more accessible Internet [84]. A site can be considered accessible if it can be used by people with disabilities to the same extent that it can by those without disabilities [85].

2.3.1.2 Content, Design and Information Quality

Aside from usability, it is no surprise that website content has been ranked as being of utmost importance when evaluating the quality of a website [86, 87]. In affiliate advertising, the majority of publisher sites are driven mainly by their content. Users visit these sites expecting a variety of content that can be accessed through text, graphics and multimedia [88]. Loiacono, Goodhue and Chen argue that the information must also be useful and entertaining as they found these to be the most important predictors of a user's intent to return to a site. They also found that without the addition of new content, the information on a publisher site can become stale and offers no incentive for users to return [21]. This is a major issue in affiliate advertising as publishers stand the best chance at earning revenue by increasing the frequency with which users visit and use the site as well as improving overall user satisfaction [7].

Additionally, the presence of high-quality content on a publisher's site may entice other site owners to provide a link to the site or for a search engine spider to index the site based on the content found when crawling. A link from another site enables the users of that site to easily find and visit the publisher, and more visits create more opportunities for conversions.

The content of a site is not the only important factor as Lavie and Tractinsky noted; the aesthetics of a site could also be a major contributor to user satisfaction [89]. Zhang and von Dran also found that users often considered attractiveness to be an important factor of a site's quality [90].

2.3.1.3 Usage and Navigation

When measuring the success of an e-commerce system, many researchers agree that system use is a necessary part of a good measurement due to customer use being primarily on a voluntary basis [75, 91, 92, 93, 88, 74]. The D&M success model states that customer satisfaction will have a positive effect on customer usage [75] and it has been shown that ease of use is an important aspect to customer satisfaction and site quality [94, 87].

One way to improve the ease of use of a site and retain customers is to ensure that the site layout is well organised with a consistent look and feel to the navigation systems of each page [88]. This consistency of navigation as well as a good search facility can also contribute to a user's ability to easily find pertinent sections of a site, which users tend to consider highly when ranking a site's quality [88]. Sites lacking easy to use navigation systems that require a lot of scrolling or give the user no indication of where they are in relation to the home page are likely to increase the time required to complete tasks and user frustration [95], both of which can to lead to dissatisfaction and lower a user's perception of site quality.

2.3.2 Categorising Evaluation Approaches

Depending on what aspects of a web site are being evaluated, there are various approaches employed that examine different web site features, although some of the approaches do include features that overlap. In this thesis, the approaches discussed will be categorised based upon the way with which usability evaluation methods (UEMs) have been categorised in the literature. Nielsen originally described four categories of UEMs: automatic, empirical, formal and informal. Nielsen dismissed the automatic and formal methods based upon the state of the art in usability evaluation at the time. The technology available then was a major limiting factor for the automatic method and the formulas used in formal evaluation were difficult to apply and did not scale well [96]. In Nielsen's definition, empirical tests involved the use of actual end-users to test the system being evaluated and informal methods were based upon the general experience and knowledge of an expert evaluator rather than the complex formulas used in the formal methods [96].

In a paper discussing the methods with which UEMs had previously been compared against one another, Gray and Salzman propose that a more general two-category classification that defines a UEM as either analytical or empirical would suffice. The author's definition of an empirical UEM coincided with Nielsen's previous definition in that empirical UEMs consisted of a range of techniques often simply referred to as user testing. The authors defined an analytical UEM as a method that utilises one of the many techniques developed to assist an expert evaluator in examining a site such as heuristic evaluation, or cognitive walkthrough, among others that will be discussed in more depth later in this section [97].

In this thesis, however, the three categories of evaluation methods described by Hasan,

Morris and Probets have been adopted because the categories defined by the authors encompass all of the previously described classifications and they are also applicable across the various other types of evaluations described in this chapter as well as usability: user-based, evaluator-based and tool-based [7]. These categories very closely resembled those described by Nielsen [96], but focus more on the subject that is conducting the evaluation rather than the method used in the evaluation.

2.3.3 User-Based Evaluations

User-based evaluation methods rely on the reactions and feelings of users that share the same qualities and traits as those that will eventually become actual users of the finished system [78]. Two of the most common user-based methods in practice include surveys and laboratory testing.

2.3.3.1 Surveys

Surveys can be used in order to judge user perception of a product as well as to evaluate a specific aspect of usability [98]. A popular example of a survey used for usability evaluation is the System Usability Score (SUS) developed by John Brooke. The SUS is an instrument containing 10 items on a five point Likert scale anchored on "Strongly Disagree" (1) and "Strongly Agree" (5) that was designed to provide a high-level overview of site usability from a subjective viewpoint. The oddly numbered items are worded in a positive manner while the even numbered items are negatively worded in order to ensure that participants are kept alert whilst filling out the survey [99]. The full description of the SUS as well as how to calculate a score using the instrument can be seen in Appendix A. Although the SUS was designed to provide a high-level subjective view of system usability, Borsci, Federici and Lauriola as well as Lewis and Sauro independently discovered through factor analysis that the SUS actually has two items that are helpful in determining Learnability as well as usability [100] [101]. According to the British Standard BS ISO/IEC 25010:2011, learnability is a subset of usability that indicates how well the system helps a user learn how to use it properly [102].

Another popular survey, Loiacono, Watson & Goodhue's WebQual instrument, measures web site quality based upon a user's intent to return to the site. This instrument was developed from a combination of the Theory of Reasoned Action and the Technology Acceptance Model [103]. The WebQual instrument measures 36 items in 12 categories

using a 7-point Likert scale anchored on “Strongly Disagree” (1) and “Strongly Agree” (7). Like the SUS instrument, WebQual also includes negatively worded items in order to keep participants alert. The items of WebQual have been shown to positively relate to a customer’s intent to re-use a company’s web site, and the authors noted that usefulness, entertainment and response time were the primary indicators of site quality [21]. This is important in affiliate advertising because returning users tend to spend more time on a site [7] making conversions more likely.

2.3.3.2 Laboratory Testing (User Testing)

The second, but most common user-based evaluation method is laboratory testing. This method, often referred to as simply user testing is generally accepted as the method with the most bearing on product improvement [104, 105]. Traditionally, laboratory testing involves recruiting suitable participants to attend a testing session at a lab set up with video and audio recording equipment as well as a computer with the ability to capture the screen, mouse movements and key presses [73]. Some labs have even been outfitted with equipment to track eye movement to record user attention and area of focus on the screen during the test [106]. The users are asked to complete a set of tasks using the system being tested, usually while using the think-aloud method to verbalise their thoughts and actions as they complete the tasks [107]. When utilising this method, researchers have found that the level of instructions given during the explanation of the think-aloud process may cause the user to become more reactive. Giving explicit instructions about what to vocalise (i.e. expectations and reactions to events) rather than simply instructing the users to vocalise their internal thoughts may have an effect on the reliability of the information gathered by causing reactivity in the user. That is, explicit instructions caused a change in the cognitive process of the user, which can affect task performance [108]. Chi et al. found that this change in thinking could cause an increase task performance regarding learning tasks [109] while Chin and Schooler found that verbal overshadowing, or decreased task performance, may occur [110]. Either of these effects can be considered negative in the context of web site testing. Zhao, McDonald and Edwards explain that due to the change in the thinking process from that of the typical user, an increase in task performance may lead to missing usability issues or to the assigning of severity levels that are not useful. The authors also explain that verbal overshadowing may cause an increase in false positives, which are designated as issues of low-severity that do not have any bearing on actual site usability

[111].

While conducting multiple rounds of testing have shown to be the most effective approach to laboratory testing [105, 112], the equipment required to construct a laboratory and the travel costs for both testers and evaluators has made the cost of these tests prohibitive. In order to mitigate this high cost, Nielsen has suggested several cost saving methods that could be employed in order to allow for more frequent testing throughout the development cycle. The largest cost savings come from the recommendation to not use video recording equipment [113]. Rowley proposed that a mobile evaluation lab that could be taken to the testers might also be used to reduce costs [114] and Krug suggested that the lab and expert evaluator are not even necessary if the testing budget does not allow for these luxuries [115]. In fact, Khanum et al. noticed that children were more vocal in a field setting rather than a laboratory setting when using the think-aloud process [116] which suggests that doing away with the lab may have more benefits than simply cost savings. The study only focused on children, and so the authors were unable to determine whether this observation extends to other age groups. However, Balikrishnan et al. used the think-aloud method with a group of elderly users (65+ years old) with one of the three participants preferring to complete the activities in his home while the other two travelled to the researchers. The authors noted that all three of the participants had difficulty vocalising their thoughts while concentrating on the testing activities, but did not specify whether setting seemed to have any effect on the level of the participant's difficulties with vocalisation [117].

When designing a user testing session, two main issues must be considered: participant selection and task selection. When selecting participants for a laboratory test, it is best to select a group that is representative of the actual end users of the system. Dumas suggests doing this by creating a user profile that categorises the system users as closely as possible [105]. Krug, on the other hand, suggests that almost anybody with an understanding of the web can be used for testing a site if it means that the time and budget saved looking for participants that fit very specific user profile will allow for more rounds of testing [115]. Aside from the selection of appropriate testers, the number required has been a hotly contested issue in the field since the inception of user testing [118]. Virzi originally suggested that four to five participants were able to find 80% of the usability problems of a system and that adding more was unlikely to uncover significantly more problems. He also noted that the major problems were likely to be found with the first few testers anyway

[119]. Nielsen agrees that three to five participants were sufficient to identify 70% - 80% of the usability issues [120]. Hwang and Salvendy argue, however, that 10 (plus or minus two) participants were needed to detect 80% of the usability issues [121]. It has also been suggested that the number of testers required actually depends on the probability of detecting a usability problem in the system under test. Schmettow [122] explains that systems with issues that occur less frequently or involve complex steps to reproduce will require more testers to achieve the detection rates given by Nielsen and Virzil. In order to determine the number of testers needed, Lewis suggested that the discovery rate of usability problems should be calculated in order to estimate the sample size needed to discover most of the issues with a particular system [123].

The second issue to consider in regards to user testing is the selection of the tasks that the users will be asked to carry out. Dumas laid out a set of guidelines regarding the selection of these tasks [98]:

- Tasks must reflect an action that a normal user would want to carry out on the system.
- Tasks selected should consist of basic tasks that would be repeated frequently during normal use in order to test the core functionality of the system.
- Tasks should probe potential problem areas of the system.
- Tasks selected should be designed to explore the interface as fully as possible.
- Tasks that may be new to users or that may disrupt normal use patterns should make up some of the chosen tasks.

These tasks generally consist of asking users to find specific products on the site, complete the majority of a purchase, find information, use the search functionality and to initiate the process of sending feedback to the company [124, 125].

In the work done by Hasan, Morris and Probets, the authors found that user-based testing did not identify as many issues as evaluator-based testing, but all of the errors identified by user-based testing were considered to majorly affect the site usability and overall user experience in a negative way [7]. When compared to evaluator-based methods, user-based methods are considered to be better at detecting a lack of clear feedback, poor help facilities [126, 127], functionality issues, learnability issues [126, 127, 81], navigation issues and the use of technical jargon [81].

2.3.4 Evaluator-Based Evaluations

Evaluator-based methods rely on human evaluators following a set of guidelines in order to manually assess the quality of a site. There are several common evaluator-based methods including heuristic evaluation [128], cognitive walkthrough [107, 129], and techniques such as MiLE+ that use a set of heuristics combined with user-testing methods [130]. Evaluator-based methods came about because laboratory testing was too expensive, which led website designers to simply guessing as to the usability of their systems. These guesses were not always accurate and so experts in the field developed a method of testing that could be implemented quicker and for less cost than the traditional empirical user testing that had been the norm [131, 126, 81, 132].

2.3.4.1 Heuristic Evaluation

The first and most popular evaluator-based method discussed is heuristic evaluation [133]. In a heuristic evaluation, an evaluator inspects the interface of the system being tested and judges how well it conforms to a set of usability guidelines, or heuristics [128]. The various sets of guidelines followed in heuristic evaluations have been designed and refined by experts throughout the years [128, 131, 134, 135, 136]. Although many sets of heuristics exist, the majority of popular guidelines generally include categories such as ease of use, information quality, navigation and organisation, functionality and security [87, 78, 7, 103]. Once the set of guidelines to follow has been selected, the tests are then often performed by usability experts [6], although there has been success in having non-expert users follow the heuristics in order to evaluate a site as well [87]. Like user testing, heuristic evaluation is designed to be an iterative process that should be completed throughout various stages of the development lifecycle of a site in order to maximise the efficiency [137]. In fact, Allen et al. have successfully used heuristic evaluation on a set of screen shots of web pages in order to test the usability of a site [138] which could be done using screen mock-ups before development has even begun.

Hasan, Morris and Probets found that heuristic testing was able to find more issues than user testing, however the majority of these issues were considered minor in their role in disrupting the site's usability and overall user experience [7]. The issues more often detected by heuristic evaluation generally consisted of technical issues such as the load delay when accessing the site with a browser [88, 126, 127, 139], appearance or layout issues, inconsistency problems [126, 127, 139] and issues with security and privacy as well

as compatibility [133]. The large number of minor issues found by heuristic evaluation has been noted as one of the major criticisms of the approach. Critics of heuristic evaluation argue that these minor issues may actually be false alarms, or false positives, representing usability problems that do not affect usability or perception of site quality for normal users of the system [140, 132].

2.3.4.2 Cognitive Walkthrough

The next evaluation method to discuss is cognitive walkthrough [141]. The first version of the cognitive walkthrough method is based upon CE+, the cognitive learning theory developed by Polson & Lewis [141]. The CE+ model has three main components: problem solving, learning, and execution. The model chooses an action based upon how well that action fits with the current goal, then analyses the results of the action. If the result is not considered to have been successful in getting closer to the goal, the system marks it to be undone. All completed actions are recorded no matter whether their outcome is successful or not. This is done so that the outcome can be re-used by the execution component if the model encounters a similar situation in the future [141].

The second iteration is based upon the same general principles as the first, but in testing the first version, Lewis et al. found that only 50% of the observed errors had actually been uncovered [142]. After some refinement, the authors published the second version of cognitive walkthrough which Polson et al. describe as "a precisely specified procedure for simulating a user's cognitive processes as the user interacts with an interface in an effort to accomplish a specific task" [107]. Cognitive walkthrough is based upon the design walkthrough technique that is commonly employed in software development as a cheaper alternative to user testing. Cognitive walkthrough is designed to be used earlier in the development cycle than heuristic evaluation and can be used properly by either the system developers or usability specialists [129].

Like user testing, cognitive walkthrough is task-specific in that a group of tasks designed to test the core functionality of the system are selected for the evaluator to inspect. The cognitive walkthrough process, which is made up of a preparation and evaluation phase, is guided by a printed form containing questions designed to walk the evaluator through the inspection without requiring the user to have an advanced understanding of the cognitive psychology on which the method is based [107].

In the preparation phase, the tasks that will be tested are chosen. Polson et al. recommend choosing tasks that are composed of basic sub-tasks that users of a well-constructed system would be able to complete. For each task to be tested, the initial state of the interface, the actions that will be needed to complete the task and the goals of the user are noted by the evaluator on a sheet similar to the one shown in Figure 2.2.

Cognitive Walkthrough Start-up Sheet

Interface _____

Task Evaluator(s) _____

Task Description: Describe the task from the point of view of the first-time user. Include any special assumptions about the state of the system assumed when the user begins work.

Action Sequence: Make a numbered list of the atomic actions that the user should perform to accomplish the task.

Anticipated Users: Briefly describe the class of users who will use this system. Note what experience they are expected to have with systems similar to this one, or with earlier versions of this system.

User's Initial Goals: List the goals the user is *likely to form* when starting the task. If there are other likely goal structures list them, and estimate for each what percentage of users are likely to have them.

Figure 2.2 Abbreviated Cognitive Walkthrough Instruction Sheet (Source: [107])

The user's initial goals section of the form is meant to capture the goals that the user is actually likely to have based upon the initial interface states and background knowledge of a typical user of the system rather than the goals that the developer thinks the user should have.

Once the sheet has been filled in with the appropriate information, the evaluation phase of the walkthrough can begin. This phase involves a three part form that can be found in Appendix B. Essentially, the evaluator completes each step of the task and predicts how the changes in the state of the interface will affect the user's goals and what percentage of users will be unable to successfully complete that step based upon the information available on the interface and the typical user's background knowledge. The evaluator also records the differences in what the user would expect to happen and the feedback actually given by the

system in each step of the tasks [143].

Cognitive walkthrough has been critiqued for being an awkward and cumbersome process, which was confirmed by several reviews of the system [114, 129, 144, 145]. In response to the criticisms, the authors released a simplified third version of the method. In this version, the evaluator answers four relatively simple questions at each step of a task:

1. Will the user try to achieve the right effect?
2. Will the user notice that the correct action is available?
3. Will the user associate the correct action with the effect that user is trying to achieve?
4. If the correct action is performed, will the user see that progress is being made toward solution of the task?

Mahatody et al. note that several reviewers have critiqued the third version of cognitive walkthrough by saying that although the technique was easy to learn and apply, it was still tedious in practice [146, 147, 148, 149, 150].

2.3.4.3 Streamlined Cognitive Walkthrough (SCW)

Spencer noted that when used on software projects with large teams, that even the simplified third version of cognitive walkthrough was difficult to apply and did not consistently provide good results for three main reasons [151]:

1. The amount of time required to answer the four questions for each step of a task and to process the large amount of data produced was simply too much as development teams are generally on very tight schedules.
2. The evaluation session often led to long design discussions in an attempt to fix the problems discovered.
3. Developers would sometimes become defensive about design decisions rather than work toward fixing the usability problems.

Spencer suggested the Streamlined Cognitive Walkthrough (SCW) in order to work toward solutions to these issues. The first phase of SCW is conducted similarly to the preparation phase of cognitive walkthrough in that an appropriate task for the user to complete must be chosen. Once the tasks have been defined, the second phase involves briefing the evaluation team on each member's specific role as well as defining what should and should not happen in the evaluation phase in order to avoid wasting time with re-designs or

defensive developers. Phase three is the inspection phase and is similar to the second phase of cognitive walkthrough, but replaces the four questions with only two [151]:

1. "Will the users know what to do at this step?"
2. "If the users do the right thing, will they know that they did the right thing and are making progress towards their goal?"

After the inspection has finished, phase four involves noting down any issues discovered and the final phase is fixing those issues.

Although the SCW method adds additional phases onto the already cumbersome cognitive walkthrough method, Spencer reports completing an evaluation on a new Integrated Development Environment (IDE) with an eight member team in two and a half hours spread over two sessions that were separated by a week. The first session took just 90 minutes with the first 20 minutes devoted to the preparation phase. In this session, the team managed to cover 32 actions and uncovered 24 usability issues without defending previous design choices. In phase four, it was decided that 14 of the problems discovered were down to users not possessing the level of required knowledge, while the remaining 10 were due to a lack of system feedback to completed user actions. The author notes that 11 design ideas were also recorded (but not discussed) during the walkthrough and that six of the 11 design ideas were possible solutions to issues discovered during the walkthrough. Only the first session was covered in the report by the author, and so data is not available about the findings of the second session [151].

2.3.5 Tool-Based Evaluations

Tool-Based evaluation methods generally involve models, predictive tools and tools to conduct remote user testing.

2.3.5.1 Models

The models used in tool-based evaluations are similar to the formal usability methods described by Nielsen [96] and generally attempt to provide measurements of user performance without actually involving the users [7]. This can be accomplished through the use of processes like GOMS (Goals, Operators, Methods and Selection rules) [152]. GOMS is a system of breaking down user interactions into the most basic actions, or Operators as they are called in GOMS, in order to work toward user Goals. In GOMS and the variants developed over the years [153], high-level user Goals are divided into sub-

goals that are then in turn divided into further sub-goals until a user could reasonably complete the resulting action at which point it is considered an Operator. John and Kieras have created examples related to using a word processor to explain the difference between Goals and Operators [153]. In their example, the overarching goal is *EDIT-MANUSCRIPT* which is divided into sub-goals of *MOVE-TEXT*, *DELETE-PHRASE*, and *INSERT-WORD*. These sub-goals can be further broken down into Operators such as *MOVE-CURSOR*, *CLICK-MOUSE-BUTTON*, and *HIT-DELETE-KEY*. The authors also note that not all systems will require such a fine level of detail, and some may have stopped with the original group of sub-goals as the Operators depending on the expected level of knowledge of the typical system user [153].

2.3.5.2 Predictive Tools

Ivory's Web TANGO [154] is an automated usability evaluation platform that assists inexperienced and non-professional web developers with creating sites that do not suffer from poor design. The tool, which the authors refer to as a "quality checker" that they liken to the spell check functionality in a word processor, uses metrics such as word count, link count and graphics percentage. Human evaluators have previously manually ranked the websites used and the system attempts to predict the ranks that the humans have given to the sites. The first version of Web TANGO was a proof-of-concept implementation capable of achieving a predictive accuracy of 63% [155]. For the second version of the Web TANGO platform, the authors added more metrics and support for site classifications and the system was able to predict the rankings of sites with 94% accuracy when the sites were divided into their respective categories (community, education, finance, health, living and services) [156].

Similar to the Web TANGO system, Li and Yamada developed an automated platform to predict the rank that users were likely to give to a web site. The authors collected data from over 700 web sites across multiple online site ranking services. These sites use reviews sent by users in order to rank web sites. Using this system, Li and Yamada were reliably able to predict the rough site rank for sites in five out of seven categories [157].

Rank prediction is not the only goal of tool-based evaluation methods. There are several tools that focus on the prediction of the path that a user is likely to take through a web site. Web Criteria's SiteProfile utilises a model called Max that is based on GOMS. Max

browses a site in an attempt to perform a given task in order to simulate user testing [158]. Max does not exhibit typical user behaviour in all aspects, which has led to some criticism [159], but the agent attempts to browse as much of the site as necessary in order to complete the given task [160]. Lynch, Palmiter and Tilt reported that the accessibility values computed by Max matched the data from user tests in 8 out of 10 cases. While not conclusive, the authors were hopeful that future testing would show the same trends when using more sites [158].

The InfoScent evaluator, like Max, attempts to automatically predict the path a user is likely to take through a site. Unlike Max, the InfoScent Evaluator uses Latent Semantic Analysis (LSA) to match links on a website with a user's goal. LSA is a technique from the field of natural language processing that can be used to analyse how the terms in a corpus relate to that corpus [161]. The InfoScent Evaluator also incorporates the Information Foraging Theory, which says that users will choose their path through a web site based upon clues presented regarding the content on the other side of a link [162]. In order to rank the links, the evaluator looks at clues such as URL, alt text for graphical links and link text and then uses LSA to determine which link is most closely related to the user's goal. Links that match the user's goal closely are said to have a high Information Scent, and the evaluator chooses the link with the highest information scent and then starts the process over on the new page until the user's goal is reached. A study on the effect of information scent used eye-tracking to determine the influence of scent on a user's information seeking behaviour. The study found that when users were looking at pages with high information scent they felt more confident about which link would help them reach their goal [163].

Using the concept of Information Scent, the Bloodhound simulator [164] creates a matrix of the probability that a user with a specific search goal will click each of the links on a site. The simulator then employs the Information Scent Absorption Rate (ISAR) algorithm in order to simulate a user's path through a website when looking for that specified search goal. After the simulation, the system then produces a usability report for the site detailing how easily the requested information was able to be found. In testing, the Bloodhound system strongly correlated with user trial results in a third of the cases and moderately correlated in roughly another two thirds, with only a small portion of cases having a weak correlation [164].

The Cognitive Walkthrough for the Web (CWW) evaluation method is an automation of a user testing method but it also fits into the category of a prediction tools due to its use of LSA in order to compute the similarity between a user's search goal and the text of each link and heading on a web page [165]. While the principle of goal driven exploration from cognitive walkthrough is the basis for CWW, the model has been changed from the CE+ model [141] to Comprehension-based Linked model of Deliberate Search (CoLiDeS) [166]. As mentioned, CWW uses latent semantic analysis to estimate the similarity between the headings and links of a site and a pre-written user goal. This is accomplished with a two-step process: first, a page is divided into sub-regions and the algorithm attempts to select the most appropriate sub-region required to complete the task. The second step involves choosing the correct widget in that sub-region to complete the action. In order to pick the correct region and widget, CWW has modified the second question from cognitive walkthrough and split it into two parts as follows [165]:

a. "Will the user connect the correct sub region of the page with the goal using heading information and her understanding of the sites page layout conventions?"
b. "Will the user connect the goal with the correct widget in the attended to sub region of the page using link labels and other kinds of descriptive information?"

2.3.5.3 Remote User Testing

As previously mentioned, the main problem with user testing in a lab setting is that it is expensive. The high cost coupled with rapid development times, frequent changes throughout the development cycle and tight deadlines that do not allow for frequent re-evaluations can cause development teams to skip testing all together [167]. These problems are exacerbated when developing a website due to geographically separated target audiences, a wide range of cultural and language differences as well as rapidly developing technology that continuously changes the way sites behave. Christos suggests that under these conditions, the high cost of laboratory testing becomes even more prohibitive and the results of those tests become less relevant [168].

Remote user testing is an attempt to work toward a solution for these problems [169, 170, 171, 172, 173]. The remote nature of this evaluation method offers quite a few benefits

[168]:

- It is easier to reach a culturally and geographically diverse group of users, opening the evaluation to a worldwide audience
- It is useful in testing systems developed for hard-to-reach or de-centralised groups that would otherwise be very difficult to meet with in person.
- No travelling means a potentially large savings on travel and lodging costs.
- Users are able to take part in evaluations in a more realistic environment. Khanum and Trivedi saw increased verbalisation in child participants in a field setting when compared to the lab [116].
- The lower cost overhead creates the potential for including more participants in the study, making it more likely that a wider range of user types are involved in testing.

Of course, there are also negative aspects of having the evaluators physically separated from participants. Christos mentions thee of the major drawbacks as being [168]:

- It may be more difficult for the evaluator and participant to build a mutual understanding and trust due to the relatively limited ability to communicate with one another during the evaluation.
- While software exists that makes it relatively easy to clearly capture the participant's verbalisations, screen contents, mouse clicks and other technical aspects of the evaluation, being physically separated makes it more difficult for the evaluator to assess the user's facial expressions and other non-verbal cues.
- Having access to such a diverse pool of participants may bias the results due to the social and cultural context of an international audience.

In a paper detailing a study of think-aloud usability testing in Denmark, China and India, Clemmensen et al. elaborate on the final point regarding cultural bias. The authors found that when the evaluator and participants were from different demographics (culture, geographic region, age and even gender in some cases), the results of usability tests could be negatively affected [174].

Despite the drawbacks, West and Lehman noted that in their comparative study, there was no difference in the levels of task success and task satisfaction between remote and laboratory based testing. The authors did notice a minor difference in the time taken on tasks and the likelihood of participants giving up on a frustrating task, but noted that these

differences were deemed to be insignificant [173]. The authors postulate that these differences may be the result of the users being in a more relaxed testing environment. West and Lehman also explained that although written comments from the users allowed the evaluator to identify usability errors that ended in task failure, the results were not as comprehensive as when the evaluator and participant were in the same location [173].

2.4 Conclusions

This chapter has introduced some of the key website features that distinguish a low-quality web site from a high-quality web site. Due to the relative ease of switching from one e-commerce site to another [4], usability is often touted as being of the utmost importance [4] [5] as sites that are considered more usable tend to see an increase in sales [6]. In recent years, researchers and web designers have begun to realise that usability of a site is important for all users, including those with diverse abilities [175]. This realisation has lead designers and academics to examine the accessiblity of websites more closely [176] [177]. Another important aspect of running a successful e-commerc site is the retention of customers. User satisfaction and perception of site quality have been shown to increase on sites with pleasing aesthetics [89] and clean, consistent navigation [88] systems, and customers that feel satisfied by their use of the publisher's site are more likely to return [7].

Knowing what qualities make a site appealing to users is only half of the problem. Evaluating the quality of these metrics has been approached in differing ways throughout the years. In this research, the various approaches to website evaluation have been sorted into user-based, evaluator-based or tool-based categories depending on what entity makes the final judgement of website quality. These categories were adopted from research on usability evaluation done by Hasan, Morris and Probets [7] because although they were designed for a usability study, they can extended to the other key website features discussed in this thesis.

User-based testing methods generally gather the opinions and feelings of users about a specific site. The collective opinions of these users can then be analysed in order to evaluate the quality of that site. Although several methods exist, most generally measure features in categories such as ease of use, information quality, navigation & organisation, functionality, security and several other related web site features [87, 78, 7, 103].

Approach	Advantages	Disadvantages	Examples
User-based	• SUS [99] shown to determine usability and learnability [100] [101] • WebQual [103] items positively relate to customer re-use • Think-aloud [107] gives the evaluator first-hand glimpse of user experience • Errors identified by user-based methods generally major [7]	• Requires input from multiple users • Generally does not identify as many errors as evaluator-based methods [7] • Laboratory equipment is expensive [113] [114] [115] • Overt instructions in think-aloud can cause false-positives [111]	• Surveys (SUS, WebQual) • Laboratory Testing (Think-aloud method)
Evaluator-based	• Cheaper than user-based testing [81] [126] [131] [132] • Generally identifies more errors than user-based methods [7] • Cognitive Walkthrough is usable by developers or specialists [129] • Streamlined cognitive walkthrough is a less cumbersome version of cognitive walkthrough [151]	• Errors identified by heuristic evaluation methods generally minor [7] and sheer number could be overwhelming [132] [140] • Cognitive Walkthrough is a cumbersome process [114] [144] [145]	• Heuristic Evaluation • Cognitive Walkthrough • Streamlined Cognitive Walkthrough
Tool-based	• Does not require user involvement [7] • Can be automated [154] [157] [158] [161] [164] • Web TANGO predicted site ranking with 94% accuracy [156] • Li and Yamada reliably predicted rough site rank for five out of 7 categories [157] • Max computed accessibility values that matched user data in eight out of 10 cases [158] • Remote user testing allows for user testing in a more realistic environment [116]	• Max criticised for not accurately mimicking typical user behaviour [159] • Remote user testing may make it more difficult for the participant and evaluator to understand each other [168] • Demographic differences between the evaluator and participants can negatively affect usability test results [174]	• Models (GOMS, Max) • Predictive (WebTango, Li and Yamada) • Remote user testing

Table 2.1 Advantages and disadvantages of the three approaches

Despite the comparatively high cost of user-based testing methods, they remain popular in practice as they have a tendency to be more effective than evaluator-based methods. That

is, some researchers have concluded that user-based testing generally does not produce as many false positives [140, 132].

Although some have been critical of the possibility of false positives with evaluator-based methods, they generally excel at discovering more technical issues with a site than user-testing [88, 126, 127, 139]. These evaluation methods have also been noted to be especially suited to discovering issues with system appearance and layout, inconsistency [126, 127, 139] security and privacy as well as compatibility [133].

Several tool-based evaluators were also introduced in this chapter. These ranged from formal models such as GOMS [152] to several automated solutions designed to predict user behaviour [165, 158, 161], predict site performance [154] or to check a site's conformance to a set of guidelines [73].

In the Web TANGO system, sites were crawled and several features were examined in an attempt to predict the ranks that were given to the sites by human evaluators [154]. The first version evaluated only 11 features and was able to accurately predict the site ranks between 67-80% of the time [155]. The second version, however, examined 157 features and was able to achieve an accuracy of 94% when sites were classified into their respective page types [156].

The concept of remote user testing combines the effectiveness of user testing with the automation and ease of use of tool-based methods. Remote user testing was created to work towards a solution to many of the issues with laboratory based testing [168].

The correlation of the evaluation metrics examined in this research will be in the context of affiliate advertising. The affiliate advertising value chain typically consists of the customer, the publisher, the affiliate network, the advertiser and an outsourced programme manager (OPM). The affiliate networks help to organise these key players as they work together in an effort to drive customer traffic and sales to advertisers through a publisher's site. For their effort, publishers can be paid either based upon the number of times an advertisement is shown (PPM), the number of clicks the advertisement receives (PPC) or the number of customers that perform one of several actions that have been pre-agreed between the advertiser and publisher (PPA). The OPM generally acts as a liaison between the advertiser and the publisher as well as helping to add value to the campaign through the creation of advertisement creative and other content for the publisher's sites.

The OPM is also responsible for finding and managing publishers for the advertising campaign. In order to determine which publishers are performing well, the OPM will typically look at the statistics collected by the affiliate network along with those from third-party tools. These statistics can include site-level metrics such as the number of impressions, click-through-rate, conversion rate and cost-per-revenue. Some OPMs and publishers may also use services such as Google Analytics to collect page-level metrics such as bounce rate in order to better understand why a page is performing either poorly or well.

In the next chapter, the website features to be used in the evaluation of affiliate advertising sites along with the methodology developed for crawling publisher sites in order to extract data related to these features will be introduced.

3 Research Methodology

3.1 Introduction

This chapter will begin by introducing the feature selection process. This is followed by the definitions of the currently implemented features along with the methodology involved in calculating their scores and a hypothesis about how each feature relates to campaign performance. In order to judge each site, these scores are combined to calculate a site's overall HealthScore, which is then compared to the real-world performance of the site. The chapter concludes by detailing the process behind the calculation of the HealthScore and performance score.

3.2 Research Design

Many affiliate networks provide programme managers with access to a suite of tools designed to collect and display performance information for each of the publisher sites on their advertising campaigns. Generally, this information is collected through direct traffic measurement techniques such as clickstream analysis [62] and page-tagging techniques like web beacons [178]. The networks use the information gathered from these traffic measurement techniques to calculate performance metrics such as click through rate and conversion rate and will sometimes also include revenue-based metrics such as cost per revenue. Due to a lack of access to revenue data for the campaigns analysed, the examination of any metrics based on revenue are unfortunately beyond the scope of this work.

While page-tagging techniques, such as web beacons, are generally one of the ways that networks use to track the commission earned by publishers [179], this research focuses on the examination of information readily available to the surfer agent which does not currently have access to traffic measurement data. Readily-available information is used so that a HealthScore can be calculated for a site for which there may be no historical data available due to the site being a recent or even potential future addition to the advertising campaign. The site's HealthScore can later be compared against the business performance measurement data supplied by the affiliate networks to gauge success.

The methodology followed during this research has been informed by previous work related to the evaluation of interactive systems, especially of web sites. Traditionally the majority of web site evaluations have been conducted manually using various techniques such as having users fill out a survey [99, 180]. Another popular evaluation technique consists of evaluators that are either experts [92, 90] or experienced web users [78, 87] completing a pre-determined task using the site while verbalising their actions and thought process [141]. An alternative to these methods is to have an expert follow a set of rules or guidelines in order to discover faults with the site [128]. Lastly, the evaluation may even consist of a combination of these methods [165]. The data collected from the observations and surveys can then be analysed to determine the quality of the site being tested.

The study presented in this thesis utilised automated data collection and analysis techniques. However, unlike previous automated methods [154, 157] the main concern was not with predicting how well a site would be ranked by human evaluators. This research was concerned with how the features of a site correlate to the real-world performance of the publisher through an overall HealthScore calculated from the scores of the various features on the publisher site.

None of the instruments used in the previous literature has been adopted in its entirety because, as Li and Yamada point out, an automated solution is better suited to measure objective features rather than the subjective features found in more traditional manual methods [157]. Although not taken directly from previous instruments, the features designed for use in this research were inspired by the those described in previous literature. A summary of the origins of the currently implemented features used in this study can be seen in Table 3.1.

In order to fulfil the aim of this thesis, the real-world performance of a publisher's site must also be calculated. In their study, Lee and Kozar used information from the COMPUSTAT database to compute the ROA and ROE of the companies running the sites which they were examining. While the financial success of a company is certainly a good measurement of success, it would be very difficult, if not impossible, to obtain the same information for the majority of publishers in affiliate advertising. Instead, I have calculated a performance score for each publisher based on performance metrics supplied by the digital marketing agencies running the campaigns being analysed. Section 3.3.4 further expands upon the

creations and use of the performance score.

Feature	Rationale	From the paper	Origin
URL Similarity	Typo-squatting hurts advertiser brand and earns unwarranted commission for a publisher.	Represents typo-squatting	[9]
		Often a characteristic of a malicious page	[181] [182]
URL Relevance	Users will click more often on a search result that they consider better or that they prefer. More clicks give more chances for conversions, which increases site performance.	Search captions built using the URL relevant to the query are better. Better captions are clicked more.	[183]
		Users prefer descriptive, static URLs over non-descript, dynamic URLs	[184]
		Static URLs have an advantage in click-through rates because users can easily read the URLs	[185]
Broken Links	Broken links make sites incomplete, unprofessional, and possibly malicious. Users are likely not to trust these.	Incomplete sites rank lower than complete. Incompleteness defined in part by broken links.	[186]
		Broken links decrease site trustworthiness.	[187]
		Broken links promote a poor user experience	[35]
Broken Images	Broken images make sites incomplete, unprofessional, and possibly malicious. Users are likely not to trust these sites.	Incomplete sites rank lower than complete. Incompleteness defined in part by broken images.	[186]
		Broken images decrease site trustworthiness.	[187]
		Broken images promote a poor user experience	[35]
Blacklisted	Chrome, Firefox and Safari issue a warning when visiting. User trust is a major factor in purchasing.	Security and trust promote user acceptance.	[26]
		Successful e-commerce sites are those that users trust.	[92]
		A large portion of Russian sites in a counterfeit affiliate programme appeared in blacklists, indicating that they may have been previously used in SPAM activities	[188]
Visibility	Top 10 results are the most important, but even at top 100, only 10 sites had a non-zero score. Originally planned to do paid search, but research indicated organic was clicked MUCH more often.	Out of 8m clicks, 94% on top 10 results	[189]
		Out of 1.4bn searches, 94% clicked organic	[190]
		Out of 1.5bn searches, 95% clicked organic	[191]

Table 3.1 Origin of Framework Features

3.2.1 Initial Features

The process of selecting the initial list of features was conducted in two phases. Phase one of the feature selection process involved compiling a list of features that were hypothesised to be possible indicators of a publisher site's performance. After examining the features in previous studies, 41 features separated into four distinct groups were initially proposed. The high-level groups were Content, Security, Technical and Design and the full listing of the features is included in Appendix D.

3.2.2 Refining the Feature Set

The initial list of features was shared with a mix of computing researchers as well as the employees of a digital marketing agency located in Scotland. Feedback from these individuals indicated that several of the features were very closely related or worded in such a way that they were difficult to understand. In order to correct these issues, the feature categories were re-defined, several of the features were combined and others were renamed in order to more clearly convey what the feature was designed to measure.

Once the features were more clearly defined, the list of features to be implemented was narrowed down to six features representing a balance between content, security and usability type features. A list of 11 features to be investigated further in future work was also created.

3.3 Dimensions and Features

The six features present on the final feature list were originally inspired from previous instruments or ideas presented throughout the literature. This section presents the justification for including each of the final features below. Following that is a description of how each of the feature scores including the HealthScore and the performance score are calculated.

In order to present the final metric scores in a more human readable format and to promote the ability to easily compare feature values, I have designed a set of requirements that each of the features to be included in this instrument must meet:

r1. Any information needed to calculate a score for the feature must be readily available to the various gathering agents either on the publisher site or from an open-source repository of some kind.

r2. The final score of a feature must be an integer in the range of 0-100 inclusive with zero being the worst possible score and 100 being the best. Any features measuring values outside of this range must undergo processing to bring the values into the acceptable range.

r3. It must be possible to automate the data collection and analysis processes required to calculate the final score of a feature. This requirement does not include the automation of any set up processes conducted before the analysis is undertaken for a campaign.

In the proof-of-concept implementation of the data collection system used throughout this research, several of the features required manual intervention in either the data collection or analysis processes which does not strictly comply with the above requirements. In choosing between following the above rules or implementing more features in the timeframe allotted to development, I felt that the inclusion of more features was an acceptable reason to deviate from the requirements. Each of the features included in the current version of the implementation is also capable of being automated in future versions of the data collection and analysis systems as described further in Section 6.5.

Each of the features described in this thesis, including those features that have been planned but have not yet been implemented, belong to a high-level category designed to separate the features into logical groups. These categories exist in order to better classify the features in an effort to aid the user in understanding what the features measure and how they relate to each other. Along with this high-level category, each of the features is also assigned a modified version of the Implementation Level defined by Dingli and Mifsud to represent how easily the feature could be translated into a guideline that could be interpreted by their framework (Dingli & Mifsud, 2011). In this research, the Implementation Level score refers to how easily a feature could be implemented in a manner that fulfils the feature requirements laid out in this section. The meanings of the three possible values for implementation level are presented in Table 3.2.

Implementation Level	Meaning
Green	This feature can be implemented, and will meet all requirements. The parameters of this feature are easily measureable.
Amber	This feature will be more difficult to implement. One of the requirements may not be currently met, but with some additional work, all requirements can be met.
Red	This feature is not currently implemented as the technology required to meet all requirements is advanced well beyond the scope of this work. This feature is listed solely for informational purposes so that users can manually check the feature if desired.

Table 3.2 Implementation Level Meanings

3.3.1 Domain Analysis

This category is focused around analysing data related specifically to the domain. This could include things such as the URL for the site, information about the software being used to host a site, physical location of the server, or how well the site ranks on search engines.

3.3.1.1 URL Similarity (Amber)

Malicious users will often register a domain name that is similar to, or is a misspelling of the URL of a popular site. The owner of the newly registered domain will then create a web page designed to display a large amount of related ads, serve up malware or sometimes show a duplicate of the genuine page in order to trick users into entering login credentials [181]. These sites are usually visited when a user mistypes the proper URL for the advertiser, meaning the user already intended to visit the advertiser and the publisher has not genuinely contributed to sending the user there and should not receive commission on any purchases made. This practice, known as typo-squatting [181, 9], is generally against the terms and conditions of affiliate networks.

In the current implementation of the framework used in this study, each publisher URL is

manually examined to determine whether it is similar to that of the advertiser's URL. If it is, a score of zero is awarded while a score of 100 indicates a unique URL. In the future, techniques similar to those used in the detection of phishing URLs may prove useful in implementing the measurement of this feature in an automated fashion in order to comply with requirement r3 [192, 182].

H1. The use of typo-squatting techniques will negatively affect campaign performance.

In looking at two case studies, the percentage of direct visits was below 25% in each case (18.09% and 24.9% respectively) meaning that less than 25% of visitors arrived at the site without clicking a link on another site such as a search engine or web portal [32, 193]. Assuming this holds true for most sites, the majority of customers do not type in a website address when visiting a site, which would limit the exposure customers have to sites using typo-squatting techniques to attract traffic. For these reasons, it is hypothesised that publisher sites that use typo-squatting to attract traffic will not perform as well as publisher sites that follow the terms and conditions.

3.3.1.2 Visibility (Green)

The visibility feature was designed to capture how well a publisher site performs in a search for the various keywords associated with an advertising campaign. In the current implementation, this feature has its own agent assigned to collect the organic search results for the given keywords. The organic search results were initially targeted over sponsored results based on findings by Jerath, Ma and Park that showed 95% of the 1.5 million clicks they examined in February 2011 were on organic results [191]. In June 2011, Nielsen examined 1.4 billion searches and found that only 6% of those users chose a paid result, meaning the other 94% of potential customers chose an organic search result [190].

The visibility agent is given a list of keywords associated with the campaign, which the agent then uses to search Google, Bing and Yahoo! for each of these campaign keywords. The agent saves the top 100 results and then checks the URL of each publisher page to see if that URL is within those search results. The Chitika affiliate network reported that out of 8 million clicks sent to their network by Google in May of 2010, 95% of the traffic was sent from the first ten search results [194]. In an updated study covering the week of May 21st 2013 through May 27th 2013, Chitika reported 92% of the traffic generated to their

network from Google was from the first ten search results [195]. Limiting the calculation to the first ten results would severely lower the number of publisher sites that received a non-zero score. In order to allow for more sites to earn points in this metric, the visibility agent search range was expanded to include the top 100 results rather than simply the top ten. I hypothesised that a site that has a good visibility in organic search results for keywords directly related to the campaign is not only easier for users to find, but will also be seen by users to be well-aligned with their search goals and therefore will perform better than those sites with lower visibility.

H2. Having a good visibility score will positively affect campaign performance.

For an example of how the visibility score works, two fictitious companies have been created. These companies are the advertiser, RoadGrip Tires (RGT) and their top publisher "Mudder's Heaven", which is a blog site run by an off-road enthusiast. RGT has the following five keywords associated with the campaign as determined by the programme manager:

1. Tires
2. Heavy-duty tires
3. Snow tires
4. Off-road
5. Four Wheeling

$$S = \begin{cases} 101 - r, & r > 0 \\ 0, & r \leq 0 \end{cases}$$

Equation 3.1 Visibility Score Calculation

The visibility agent searches Google, Bing and Yahoo! for the organic search results of the keywords associated with the RGT campaign and saves the top 100 results for each. In order to calculate the score for a keyword, the agent checks where the publisher's site ranks (r) in the search results for that keyword and converts the ranking into a score (S) ranging from zero (not present) to 100 (first search result) using Equation 3.1. This conversion from r to S is done to reverse the order of the search engine rankings to remain compliant with r2 and to give the 100^{th} search result a score of 1 which is the lowest score that a site appearing in the results can receive. For this example, Table 3.3 shows the rank for each keyword in the search results for the publisher Mudder's Heaven while Table 3.4 shows the

visibility score calculated from these results.

Mudder's Heaven			
Keyword	**Google**	**Bing**	**Yahoo!**
Tires	10	15	7
Heavy-duty tires	5	3	0
Snow tires	0	0	0
Off-road	5	35	10
Four Wheeling	1	8	15

Table 3.3 Mudder's Heaven Keyword Rank (r) Example

Mudder's Heaven			
Keyword	**Google(G)**	**Bing(B)**	**Yahoo!(Y)**
Tires	91	86	94
Heavy-duty tires	96	98	0
Snow tires	0	0	0
Off-road	96	66	91
Four Wheeling	100	93	86
AVERAGE (A_G, A_B, A_Y)	76.6	68.6	54.2
SCORE	67		

Table 3.4 Mudder's Heaven Visibility Score (S) Example

Once S has been calculated for each of the keywords, the average of the scores is then calculated for each search engine (A_G, A_B, A_Y) as shown in Table 3.4. These search engine scores can then be used to calculate the overall Visibility score for each publisher's site, which is the weighted average of the search engine scores as shown in Equation 3.2. Because the weighted average is used, it is possible to assign weights (w_G, w_B, w_Y) to each of the individual search engines if the user wishes. The ability to weight individual search

engines may be important for cases in which the results of one search engine may be preferred for a particular campaign, but this study did not prefer one search engine over another and so the weight of each has been set to one, resulting in the formula behaving as a simple average.

$$\frac{(\boldsymbol{A_G} * w_G) + (\boldsymbol{A_B} * w_B) + (\boldsymbol{A_Y} * w_Y)}{(w_G + w_B + w_Y)}$$

Equation 3.2 Calculating the Overall Visibility Score

3.3.1.3 URL Relevance (Green)

The thinking behind the URL relevance feature is that a publisher site that is easy to find and identify should result in better performance than one that is difficult to recognise. This thinking is partially based on findings from a study to determine the influence that search captions have on user search behaviour. Clarke et al. found that in order to create the most useful search captions, URLs containing search terms should be chosen in order to highlight the relationship to the search query [183].

In a second study that also helped in the formulation of the URL Relevance feature, Katsanos, Tselios and Avouris examined the effect that links with strong information scent, those that are closely related to the search goal, had on user information-seeking behaviour. The authors found that links with a high information scent made users feel more confident about their link choices. As part of the information scent calculation, the URL address of a link was examined to determine relevance to the user's search goal [163]. While the URL was not the only contributing factor in determining strength of a link's information scent, it is a reasonable assumption that a URL that is seen to be closely related to a user's goal is likely to create that same confidence. Yang and Gerasoulis also found that users preferred a URL to be static and descriptive over a confusing and dynamic URL [184]. Because the keywords used to score the URL Relevance feature are derived from the campaign goals, it is hypothesised that sites with a high URL Relevance score will be seen to be well-aligned with the customer's goal, and will lead to more clicks which creates more opportunities for conversions and is likely to lead to a higher performing publisher.

H3. The presence of campaign keywords in a publisher URL will positively affect campaign performance.

Before being able to calculate a score for this feature, the campaign keywords defined by

the affiliate programme manager were examined and broken down into single words. I then assigned a value to each word based upon how well the it fit with the overall goal of the specific campaign examined. Keywords highly related to the campaign goal as originally defined by the programme manager were given high values between 40 and 50, with more generic keywords describing the business sector of the advertiser given lower scores between 20 and 30. In order to account for the large number of users that specifically search for deals on publisher sites before making online purchases [1], terms describing popular types of super affiliates (i.e. voucher, cash back, and so on) were given low values of five. Terms related to super affiliates that were also found in the keywords (discount, etc.) were given a slightly higher value of 10.

The process of calculating a score for this feature is done in two steps:

1. Points are assigned to each site based upon the highest scoring keyword present in the URL.
2. Half points are then awarded to a site for each unique keyword thereafter for up to two additional keywords.

By only awarding points for a maximum of three unique keywords, even a URL containing three keywords worth the maximum amount of points would score $50 + (50 * 0.5) + (50 * 0.5) = 100$ and would still be in the range set out in requirement r2.

In practice, an excel formula was set up to examine the URLs for each keyword and to assign points to each site based only upon the highest scoring keyword. I then manually examined the URLs to assign the half points. While the semi-manual nature of this feature does not currently fulfil the automation requirement (r3), it is possible to fully automate the score calculation of this feature, leaving only the setup as a manual process, which is not a violation of r3.

3.3.2 Content Analysis

This category relates to the actual content of the pages on a site. These metrics help the user of the system to get a better understanding of what is actually present on a site including text, pictures, video and other media.

3.3.2.1 Broken Link Analysis (Green)

A broken link is a hyperlink on a publisher's site that cannot be properly resolved and results in an error, the most common of which is known as a 404 error. These broken links

can cause customer frustration and loss of trust [186, 187] or a poor user experience if the user attempts to navigate the site using any of the broken links [35]. All of these scenarios may lead to a loss of conversions as frustrated or untrusting users are more likely to switch to using another publisher's site [4]. Therefore it is hypothesised that the presence of broken links on a publisher site will lower the site's performance on an advertising campaign.

H4. The presence of broken links on a site will negatively affect site performance.

In order to calculate the broken link score, the surfer agent keeps track of any link that cannot be properly resolved (l_b) as well as the total number of links found while crawling each publisher's site (l_t). The broken link score is the proportion of working links to the total number of links on the publisher's site. In order to comply with requirement r2, the resulting percentage is multiplied by 100 to bring the score into the range of 0-100 inclusive as shown in Equation 3.3.

$$\left(\frac{l_t - l_b}{l_t}\right) * 100$$

Equation 3.3 Broken link score calculation

3.3.2.2 Broken Image Analysis (Green)

A broken image is an image on a publisher's site that cannot be displayed properly. This is usually because the image file is not present in the location referenced in the web page's code. While a broken image is technically a specific kind of broken link, broken images have a different effect on the user experience than broken links. Unlike broken hypertext links, broken images are obvious when viewing a page and create a poor user experience [35] due to the site appearing incomplete or unprofessional [196]. This, in turn, may lead to a loss of conversions due to user frustration and loss of trust [186, 187]. This frustration and loss of trust can ultimately lead to a loss of revenue because trust is an essential component of user satisfaction [26]. It has also been shown that unsatisfied users may switch to using another web site as the cost is much lower than in a brick-and-mortar store [4]. Therefore, it is hypothesised that the presence of broken images on a publisher site will lower its campaign performance.

H5. The presence of broken images on a site will negatively affect site performance.

In order to calculate the broken image score, the surfer agent keeps track of any image that cannot be properly resolved (i_b) as well as the total number of images found while crawling

the publisher's site (i_t). The broken image score is the proportion of working images to the total image count on the site. In order to comply with requirement r2, the resulting percentage is multiplied by 100 to bring the score into the range of 0-100 inclusive as shown in Equation 3.4.

$$\left(\frac{i_t - i_b}{i_t}\right) * 100$$

Equation 3.4 Broken Image Analysis

3.3.2.3 Blacklist Check (Green)

The blacklist check in the current implementation of the framework is a measurement conducted by the Site Info agent which is described further in Section 4.2.2. The agent checks the Google Safe Browsing and Malware Domain List blacklist services to determine if any of the publisher sites are present on the lists. The Blacklist Check is a binary score meaning that if a site has a page on either blacklist the site receives a score of zero, otherwise the site will receive a perfect score of 100 for this feature.

H6. Being on a blacklist will negatively impact a publisher site's campaign performance.

In an examination of publishers involved with the Tower of Power (TowPow), an affiliate advertising programme best known for dealing in herbal remedies and counterfeit products, Kamari, Ghaemi and Mccoy discovered that a large portion of the Russian domains from that network were listed on an e-mail SPAM blacklist feed. The authors speculate that the domains may have been previously used in SPAM activities, and were added to the feed when those activities were detected [188]. With known-malicious spammers re-using their domains for affiliate advertising, a blacklist check is essential in determining the HealthScore of a publisher. As of May 2013, Chrome, Firefox and Safari combined hold 66.3% of the browser market share [197], and all three of these browsers issue a very blatant warning to any user attempting to browse to a site on the Google Safe Browsing blacklist. A warning regarding malicious content appearing on three of the major web browsers should cause a decrease in traffic to an affected publisher's site. Less traffic means fewer chances for conversions to happen and so I hypothesise that appearing on a blacklist will cause a drop in a publisher's performance.

3.3.3 Publisher HealthScore Calculation

Once the individual feature scores for a publisher's site have been computed, they can then be used to calculate the HealthScore for that site. In order to be consistent with the rest of the system described in this thesis, I have also created a set of requirements for the publisher's HealthScore to follow:

r1. The final HealthScore must take individual feature weights into account when being calculated.

r2. If a user wishes to ignore a particular feature, the calculation of the final HealthScore should allow for this without requiring a change in the formula or underlying system code.

r3. The final HealthScore must be in the range of 0-100 inclusive.

The publisher site's final HealthScore is the weighted average of the individual feature scores and is calculated as shown in Equation 3.5. The weighted average is used because it fulfils all three of these requirements for any number of n features.

The requirement (r1) to allow for feature weights is based on advice given by Agarwal and Venkatesh who noted that not all features will be equally important when evaluating sites belonging to different contexts [87]. The weighted average allows for each feature to have a different weight based on the requirements for a specific campaign and allows for a feature to be disabled by setting the weight to zero and cancelling it out of the equation.

In order to properly calculate the HealthScore for a site, some assumptions must first be met. Each individual feature score, denoted as m_i, is assumed to be in the range of 0-100 inclusive in order to comply with the feature requirements. The weight of a feature, denoted as w_i, is assumed to be in the range of 0-10 inclusive to avoid giving any one feature a disproportionate effect on the HealthScore. Each feature score is multiplied by the corresponding weight for that feature and then the products are all added together. The HealthScore is kept within the range specified in r3 by dividing the sum of the weighted scores by the sum of the weights.

$$\frac{\sum_{i=1}^{n}(\boldsymbol{m_i} * \boldsymbol{w_i})}{\sum_{i=1}^{n}(\boldsymbol{w_i})}$$

Equation 3.5 Publisher HealthScore calculation for *n* features

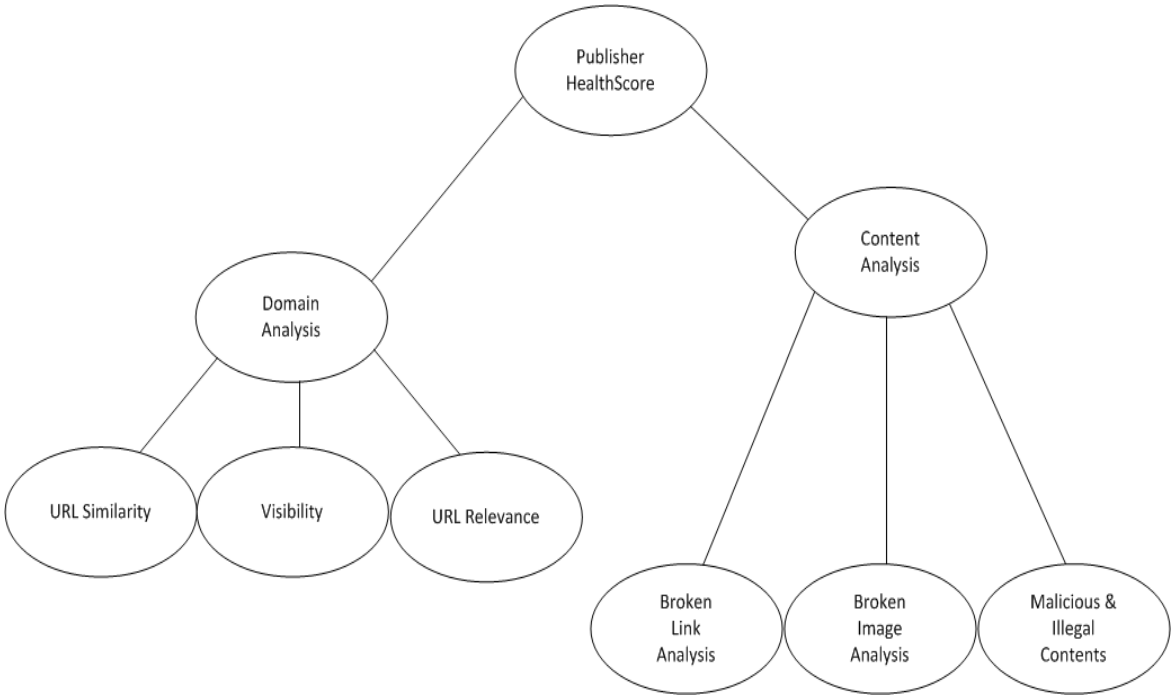

Figure 3.1 Structure of the Publisher Score Instrument

H7. A site with a good HealthScore is also likely to be classified as a good performer.

H8. A site with a poor HealthScore is also likely to be classified as a poor performer.

As shown in Figure 3.1, the HealthScore is a combination of the individual features of each of the high-level dimensions, some of which are designed to indicate "Good" performance and some to indicate "Poor" performance. Therefore, it is hypothesised that the HealthScore will be capable of indicating either level of publisher performance.

3.3.4 Publisher Performance Calculation

The initial plan to determine how well a publisher site was performing on the campaign was to use the conversion rate measurement discussed in section 2.2.4.3. The conversion rate is the proportion of sales to clicks for a publisher site, and it was originally felt that this metric would fairly and accurately capture the real-world publisher performance on the advertising campaign. However, after working with the data for a short time, it became apparent that using the conversion rate would be problematic. There were several issues with the conversion rate measurement that led to the creation of the performance score:

1. The conversion rate could sometimes lead to high volume publishers being unfairly penalised when compared to very low volume publishers (Example #1).
2. Occasionally more sales than clicks were recorded in the affiliate network data

(section 4.3.1) which caused a conversion rate that was above 100% (Example #2).

3. The conversion rate does not allow for sales and clicks to be weighted differently depending on the goal of the advertising campaign (Example #3).

In order to combat the issues inherent in the conversion rate measurement, the performance score was created. The performance score calculation uses the sales and clicks just like conversion rate calculation, but uses a weighted average of the two metrics, similar to the HealthScore (Equation 3.5) in order to address the issues encountered with the conversion rate.

For the advertising campaign used in this study, it was decided that the number of sales was far more important than the number of clicks, and that the weight of the number of clicks should be limited to prevent publishers with sheer click volume from rising above sales leaders in the performance rankings. These decisions led to the sales being weighted at five while the clicks were weighted at one.

Because the performance score calculation uses the same equation as the HealthScore, it is bound to follow the same requirements (section 3.3.3). In order to comply with those requirements, there is some pre-processing that needs to be done in order to scale the number of sales and clicks to be within the range of 0-100 inclusive. This is done by calculating the percent rank of each click and sale meaning that the scaled values for these measurements are dependent on how many clicks or sales the other publishers on the campaign have received. This method of scaling brings the values into the acceptable range laid out in r3, and also allows for the publishers to be quickly compared to one another. For an example of how this scaling works, see Table 3.5.

In order to fully explain the advantages of using the performance score over the simple conversion rate, a number of example scenarios have been created. All of the example scenarios refer to Table 3.5.

Item	Raw clicks	Raw sales	Scaled clicks	Scaled sales	CR	Performance	Swap Weights
1	1	1	9	27.00	100	24.00	12.00
2	10000	1000	63.00	72.00	10	70.50	64.50
3	2	1	27.00	27.00	50	27.00	27.00
4	100	1	45.00	27.00	1	30.00	42.00
5	1000000	1000	90.00	72.00	0.1	75.00	87.00
6	1	1000	9	72.00	100000	61.50	19.50
7	1000	0	54.00	9	0	16.50	46.50
8	10000	0	63.00	9	0	18.00	54.00
9	2	2	27.00	54.00	100	49.50	31.50
10	10000	458	63.00	63.00	4.58	63.00	63.00

Table 3.5 Performance Score Example

3.3.4.1 Example 1

A publisher from the long tail with only a single click and a single sale (item one in the example table) would have a conversion rate of 100% while a publisher with 10,000 clicks and 1,000 sales (item two in the example table) has undoubtedly earned more revenue and brand recognition for an advertiser, yet is listed as only having a 10% conversion rate. Using the conversion rate as a measure of performance in this situation would indicate that the first publisher is outperforming the second by a large amount, although further inspection reveals that this is not true. Using the performance score, the site with 1,000 sales will always have a higher score than the site with one sale.

3.3.4.2 Example 2

Affiliate network data that has not been recorded properly (section 4.3.1) may cause conversion rates to be reported at above 100%, whereas the performance score uses scaled clicks and sales, and so the score will still be a sane value for performance even on sites with more sales than clicks recorded as shown with the sixth item of Table 3.5.

3.3.4.3 Example 3

In working closely with several digital marketing agencies in Scotland, I found that it is sometimes the case that advertising campaigns are primarily interested in raising brand awareness with revenue generation as secondary focus. Simply changing the weights for sales and clicks allows for the performance score to measure this shift in priority as shown in the last column of the example table. In this case, the weights were swapped so that the number of clicks is weighted at five and the number of sales is weighted at one although

either of the weights can be set anywhere in the range of 0-10 inclusive.

3.4 Conclusion

This chapter introduced the methodology behind the feature selection process including the justification for the selection of the six implemented features. Each of the features belongs to one of two high-level categories: domain analysis or content analysis.

In order to rate each site, a score for each of these feature must be calculated and that process was outlined along with the presentation of a hypothesis as to how each feature will relate to the site's real-world performance.

The domain analysis feature group includes URL similarity, visibility and URL relevance. The URL similarity feature was designed in order to identify publishers using typo-squatting techniques [181]. The visibility feature score is based upon how well a site ranks for the organic search results for the campaign keywords. Organic search results were chosen as it has been shown that most users choose organic results when searching [191]. It has been suggested that when a site has a URL that includes terms from the search query, that URL should appear in the search caption in order to highlight the result's relationship to the user's search [183]. This thinking has been extended to test whether the presence of campaign keywords in the URL will affect publisher performance with the URL relevance feature.

The content analysis group contains broken link analysis, broken image analysis and blacklist check. Sites that have broken images and broken links on them can appear to be unfinished or unprofessional to users [196]. This feeling can lead to user dissatisfaction, which can then lead to a loss of revenue for the publisher [186, 187]. The broken link and broken image features take this into account and report the proportion of working links and images on a site. When attempting to visit a site on the Google Safe Browsing blacklist, Chrome, Firefox and Safari warn the user that the site they are attempting to visit is potentially dangerous. This warning is likely to deter potential customers from trusting the site, which is sure to lead to a loss of revenue as trust is a key component to user satisfaction [26].

An overall HealthScore was calculated using the weighted average of the scores of the six website features tested for each of the publishers in order to create a value that could be

compared against the publisher's campaign performance. This calculation was described along with how the real-world performance for each publisher was calculated in this study.

In the next chapter, the systems and processes used to gather the data needed to calculate the website feature scores will be presented. The chapter will also cover some initial observations regarding the quality and availability of the data as well as the experimental set up needed to test the hypotheses presented in this chapter.

4 Data Gathering and Trial Analysis

4.1 Introduction

This chapter presents a cloud-based platform used in the data gathering and analysis phases of this research. The platform housed several virtual machines, each responsible for a different sub-system of the overall website evaluation framework, and was created in order to provide an automated method of gathering the data required in order to compute the feature scores and HealthScore for a publisher's site. The chapter will also explain the processes employed by the various sub-systems in order to capture that information. From there, some initial observations related to the data gathered will be discussed. This discussion is followed by the experimental setup that is required in order to test the hypotheses presented in the previous chapter.

4.2 Framework Components

The data gathering and analysis platform is logically segmented into the four independent subsystems shown in Figure 4.1 on page 67. The hardware used to host the Virtual Machine (VM) instances that would make up the system included five DELL PowerEdge servers, four of which had a 3.1 GHz CPU and 16 GB of RAM with the fifth being used as a controller and having a 3.1 GHz CPU and 8GB of RAM. The hardware was hosted off-site at a local cloud provisioning company.

The data gathering portion of the platform deployed for use in this study consisted of one VM allocated 8 CPUs and 8192 MB of RAM (of which an average 7.60 GB was in use at any time) to double as the controller and database subsystems. There were also 60 VMs, each allocated a single CPU and 512 MB of RAM (of which an average of 420 MB was in use at any given time with each agent taking up around 80 MB), for the agents. It is uncertain what the unit "CPU" represents in terms of GHz as the VM instances were provisioned using the hosting company's cloud management software, however the instances of the Agent subsystem had an average CPU utilisation of 40% and the combined

Controller and Database subsystem instance had an average of 20% CPU utilisation.

url	HS	performance	broken_links	broken_images	visibility	url_relevance	url_sim	blacklist
http://www.site1.co.uk	60.00	99.50	100.00	100.00	0.00	30.00	100.00	100.00
http://www.site2.info	57.30	99.10	99.00	100.00	0.00	20.00	100.00	100.00
http://www.site3.co.uk	65.13	98.27	99.00	100.00	0.00	50.00	100.00	100.00
http://www.site4.mil	57.43	98.13	98.00	100.00	1.00	20.00	100.00	100.00
http://www.site5.gov.uk	58.43	97.87	99.00	99.00	0.00	25.00	100.00	100.00
http://www.site6.net	51.57	97.62	93.00	100.00	0.00	0.00	100.00	100.00
http://www.site7.au	57.30	96.92	99.00	100.00	0.00	20.00	100.00	100.00
http://www.site8.gov	51.74	96.32	95.00	100.00	0.00	0.00	100.00	100.00

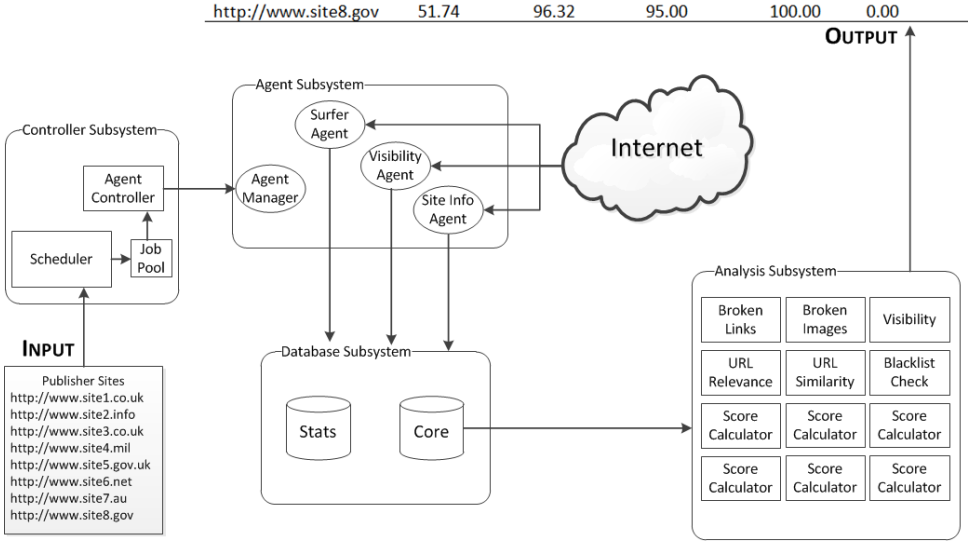

Figure 4.1 Cloud-based Data Gathering and Analysis Platform Overview

All of the VM instances were running Microsoft Windows Server 2008, and each of the smaller instances comprising the Agent subsystem had a copy of the manager, surfer, visibility and site info agents installed. This was done to simplify the process of creating new scanner instances as required based upon the current workload.

4.2.1 Controller Subsystem

The controller subsystem deals with managing/balancing the workload for individual agents and consists of three elements:

1. **Job Pool**: This details the jobs that are already allocated to individual agents and jobs that are awaiting allocation.
2. **Scheduler**: The scheduler calculates which website is next in the queue for evaluation. The scheduler is configurable and any modification to the schedule will not disrupt the operations of the overall system.

3. **Agent Controller**: This utility starts, pauses, and changes the type of the active agent. This controller also allows the core engine administrator to manually stop the agents if necessary.

4.2.2 Agent Subsystem

This subsystem consists of software programs written in C# that are tasked with gathering the required information for each of the features on individual websites. The number and types of agents that can be implemented at this layer is both flexible and extensible. Inclusion of new agents or modifications to existing agents will have minimal impact on currently deployed agents or other subsystems in the core engine. Therefore, any future upgrades to this subsystem would not require extensive downtime of the overall system. The four agents that were implemented for this study were:

1. **Agent Manager**: This agent checks the next job in the job pool and chooses which agent should be assigned that job.
2. **Surfer Agent**: This agent is the main workhorse of the entire system. The surfer agent crawls individual websites, gathering data present on them. The surfer agent uses the Fiddler2 proxy server in order to capture browsing sessions while browsing the site it has been assigned. This agent mimics a (human) user in discovering how a website will represent itself to potential customers. In order to mimic the behaviour of a human, the crawler spawns an instance of the Firefox browser and directs the browser to render the pages of the site to be scanned.

   The crawling scheme implemented for the Surfer Agents is breadth first with a limit of 5000 pages per website per scan. Although each Surfer Agent is limited in the number of pages on a site it will scan per pass, all of the pages of each site are present in the job pool and will be picked up and scanned by an agent in time. I imposed this limit in order to speed up the process of calculating initial scores for each of the sites. Future versions of the surfer agent may be designed to crawl in a more intelligent manner similar to browsing agents like Max [158]. Multiple agents with various crawling techniques in order to simulate different user types would also be an interesting addition to future versions. Techniques that take advantage of LSA, similar to those used by the InfoScent Evaluator [161] and the Bloodhound project [164], in order to more closely emulate a user seeking information are particularly interesting.

3. **Visibility Agent**: The visibility of an affiliate corresponding to a campaign is an important feature for the prediction of publisher performance. As the functionality of this agent is fundamentally different to that of the surfer agent, it has been implemented as a separate agent. The operation of this agent is outlined in more depth in section 3.3.1.
4. **Site Info Agent**: This agent is tasked with searching open source repositories and discovering the supplementary information related to a website required to calculate a score for the Blacklist Check feature. Features not yet implemented such as Content Relevance, Digital Certificate Evaluation and Readability Analysis will also have the data they require collected by this agent.

4.2.3 Database Subsystem

The database subsystem stores all the data gathered from online open source repositories and publisher websites by the various agents from the Agent subsystem. The subsystem uses a MS SQL database designed to store individual elements related to a website in a manner that allows searching without time-consuming computations and a logical relation to the respective websites so that tracking any changes/modifications to them is easily detectable. Indexing and the free text search feature allow for fast searches to be performed on the data stored in the Core database. In addition to gathering data regarding individual websites, statistics related to the platform utilisation are stored in a separate Stats database. This includes the performance, data generation, network usage, and memory usage of the core engine in order to allow for the effective tuning of the system.

4.2.4 Analysis Subsystem

The main aim of this research is to determine what features make good indicators as to whether a publisher will be a “Good” or “Poor” performer. In order to meet that aim, a significant amount of data must be processed in order to calculate the scores for each feature. The VM instance used to house the Analysis subsystem consisted of a single CPU and 2048 MB of RAM. Unfortunately, usage statistics were not recorded for this subsystem, and so the average RAM and CPU utilisation cannot be reported.

The score calculator for the features consisted of several purpose-built C# programmes each designed to calculate the scores of a single feature and to record the score in the Core database. For an in-depth explanation on how each of the feature scores was calculated, see

section 3.3. It is envisioned that future iterations of the score calculator will be a single programme capable of reading rules from a set defined outside of the code to allow for more extensibility. This method is similar to KWARESMI, described by Beirekdar, Vanderdonckt and Noirhomme-Fraiture [198]. KWARESMI is a method for expressing usability guidelines in a high-level language for use in automated usability evaluation systems.

4.3 Initial Observations

In order to calculate a score for each of the six web site features, the Surfer Agent, Visibility Agent and Site Info Agent gather data from each of the publisher sites. The Surfer Agent is responsible for gathering data found on the site through crawling while the Site Info and Visibility agents gather data about the site from external sources such as blacklist providers and search engines. Whilst crawling a site, the Surfer Agent gathers a significant amount of data beyond what is currently being used to calculate the various feature scores and the HealthScore. The feature calculations that have yet to be implemented will be able to make use of that extra data in order to evaluate more features and increase the accuracy of the system without needing to re-crawl sites as the information has already been gathered.

4.3.1 Affiliate Network Data

The data used to calculate real-world performance for the publishers was supplied by various digital marketing agencies. These agencies retrieved the data from the affiliate network systems, and based upon the non-uniformity of the data from each network, it does not appear to have undergone any additional processing by the agency and is likely straight from the network systems.

The most pronounced of the limitations encountered throughout the study were related to the performance data. Although there were over 4,000 sites scanned, the availability and completeness of the performance data limited the number of sites that were usable in the study to 234 sites.

In looking at the performance data, it seemed as though some of the networks had the capability to report a plethora of performance metrics while other networks reported only the bare minimum needed to evaluate performance. It is unclear if the data sets from

networks C and D were incomplete or if the agency only reported the minimum amount of information required to compute a performance score. It is possible that the data was purposely limited to the bare minimum necessary in order to save time as some agencies reported having to be manually copy and paste the data from the network systems due to the lack of an export feature. The full list of measurement capabilities for each network based upon the data received is summarised in Table 4.1.

Measure	**Network A**	**Network B**	**Network C**	**Network D**
Average Order Value		x		
Number of Clicks	x	x	x	x
Commission Rate		x		
Commission Earned	x	x		
Commission Level		x		
Cookie Length		x		
Cost Per Mille (CPM)	x			
Conversion Rate (CR)	x	x		
Click-through-rate (CTR)	x			
Declined Average Order Value		x		
Declined Commission		x		
Declined Number of Sales		x		
Declined Value		x		
Earning Per Click (EPC)	x			
Number of Impressions	x	x	x	
Number of Leads	x		x	
Sale Amount	x	x		
Number of Sales	x	x	x	x

Table 4.1 Affiliate Network Measurements

While there are a large variety of performance metrics listed for networks A and B, most of the fields for a majority of the publishers were blank or seemingly recorded incorrectly. For example, there were several cases of a publisher with zero impressions and hundreds or thousands of sales. The implication of this is that any metrics calculated using these fields (i.e. CPM or CTR) were calculated incorrectly. One publisher is listed as having 2801 clicks and only 285 Impressions resulting in a reported click-through-rate of 983%. Luckily, it appears as though the networks are more diligent with the recording of the

number of clicks and the number of sales as these are the values used in the calculation of the publisher performance scores, and they passed basic sanity checks for all networks.

There were two networks that appear to keep their customer data separate from the performance data. This was evidenced by the fact that the spread sheets with the performance data had publisher IDs on them which then had to be matched up with a separate spread sheet containing customer data in order to find the publisher's URL. While this in itself does not cause any major issues, the lists were often inconsistent and had different numbers of publishers on them, and several publishers with performance data did not appear on the customer sheet which made it impossible to find the URLs for those publishers.

It also appears that some affiliate networks store data about each site in the campaign separately while others keep data about each publisher, merging the data from all of the publisher's sites and making it impossible to determine which of the publisher's sites was responsible for the performance reported. When designing future versions of the framework, these differences should be taken into consideration.

In talking with various affiliate networks at the outset of this research, it was noted almost unanimously that each publisher was vetted before being added to the system. When looking at the data, however, this was not evident. Several of the URL fields contained either no URL or the URL of the advertiser rather than a publisher's site. There were also several occasions where the URL field contained multiple URLs, but this may have been done purposely in cases where the publisher uses multiple sites for one advertising campaign.

4.3.2 Sites

Before adding the publisher sites to the job pool and sending the agents to gather data, 24 surfer agents were deployed on local machines at the University for testing purposes. The agents were given a subset of the publisher sites to crawl and were observed in action. This was done in order to spot any potential issues with the agents that may have been difficult to detect when the scanners were moved to the virtual machine instances that had to be accessed through remote desktop.

Several of the sites crawled during the test scans were domain parking pages full of advertisements. Others instantly redirected the user to another domain or even to the

advertiser's site in a few cases. Unfortunately, the crawler does not record the address of pages like these and so the team was only able to inform the digital marketing agencies about a small number of sites that were witnessed exhibiting these behaviours.

Other times, the URL appeared to be that of the publisher's corporate site rather than the site being used to advertise the products and services of the campaign. Many digital marketing agencies and affiliate networks claim to audit publisher sites for compliance with network and advertiser terms and conditions, yet the site URL is not present in the affiliate network's system. These issues point toward the difficulties faced by affiliate networks and OPMs related to the continual auditing of publisher sites.

4.3.3 Crawler

Several sites highlighted issues with the design of the custom web crawler. For instance, when encountering calendars the crawler would continuously page through the calendar day by day until manually stopped by a researcher. This resulted in some sites being listed as having millions of pages and wasting valuable crawling time. In order to stop this, the crawler was changed to ignore objects that appeared to be calendars and an initial scan limit of 5,000 pages per session was added to the surfer agent.

Several of the publisher sites were unable to be crawled. These sites will be referred to as null sites for the remainder of this thesis. There are three main reasons that a site may be classified as a null site:

1. The site was unreachable when the web crawler attempted to visit. These sites were manually added back onto the end of the job pool so they would be tried again at a later time to rule out temporary outages or routing issues.
2. If a site immediately redirected the crawler to a different domain, the crawler did not attempt to crawl any further. This limitation is built into the crawler and is discussed in further depth below.
3. Any single page site that was could be identified as a temporary parking page was classified as null as these pages do not offer any value to the customer.

In the cases where a site was classified as a null site due to an immediate redirect to an external domain, the crawler halted and requested the next job. This behaviour was purposely built into the surfer agent as the agent only concerns itself with a single domain at a time. This was done in order to prevent it from running off and attempting to crawl the

whole of the Internet when following links to other domains. This also serves to ensure that all traffic captured in a single session belongs to only one site in order to properly calculate the feature scores for that site without including content from external domains. This practice appeared to be most commonly used to redirect a publisher's old URL to a new site being used for the same campaign. Presumably, the publisher used this tactic to avoid having to contact all of the agencies and networks to update each of their addresses.

This behaviour should be taken into consideration when future iterations of the surfer agent are being designed. The surfer agent should also be made to keep track of those sites which are parking pages, offline or simply redirects especially those that instantly redirect to the advertiser's page as those pages add no value to the network and should be manually reviewed by the programme manager. The ability to handle calendars and other similar objects should also be integrated into the surfer agent.

4.3.4 Feature Data

After the crawlers were moved to the cloud-based VMs and had finished collecting the data, I quickly checked over the values to ensure that the data seemed valid. During these checks, I made several observations regarding the various web site features of the publishers related to Company_A.

4.3.4.1 Broken Links

Based upon the ever-changing nature of the Internet and the relatively low level of knowledge needed to create a publisher site, I expected that several of the sites would struggle with broken links. Surprisingly, on 62% of the sites crawled, less than 1% of the links on the site were broken. With 39% of all sites having no broken links on them at all. The high level of performance in regards to maintaining links on the publisher sites was not something that I had expected to see.

4.3.4.2 Broken Images

Like broken links, I expected that several of the publishers would struggle with broken images on their sites, albeit to a lesser degree than broken links as a broken image is much easier to spot with an untrained eye. Most browsers display a broken image as a large red 'X' or other obvious placeholder which is difficult to miss when a publisher reviews their site. However, 89% of the sites for Company_A had no broken images on the site at all with 97% of the site having only 1% broken images.

4.3.4.3 Blacklist Check

Due to the low frequency with which publisher sites are audited and the large workload of programme managers, it was expected that a small amount of sites that had been blacklisted would make it onto the advertising campaigns. However, none of the sites related to Company_A appeared on either of the blacklists checked.

While this is certainly good for the advertiser, agency and affiliate network running Company_A's campaign, it unfortunately means that the effectiveness of this feature could not be tested in this case study.

4.3.4.4 URL Similarity

The URL Similarity check was done manually, and like the blacklist check, none of the sites related to Company_A had a problem with this feature. There was a publisher for Company_B that had seven sites on the campaign with URLs that matched the advertiser's URL almost exactly, but unfortunately these sites had incomplete performance data and could not be included in the analysis. These sites were, however, reported to the digital marketing agency for manual review.

Unfortunately the effectiveness of the Blacklist Check feature was unable to be determined in this study as no sites with complete performance data had a score below 100.

4.3.4.5 Visibility

While designing the visibility feature, it was envisioned as possibly having the most bearing on real-world performance. Unfortunately, it was not anticipated that so many of the sites would receive a zero score for visibility. Only 10 sites (4%) received a score above zero for this feature, which indicates that the algorithm may need to be tweaked in order to perform a more complete analysis.

4.3.4.6 URL Relevance

The URL Relevance feature was expected to be another strong indicator of publishers with good performance. It was expected that a small portion of the sites would have keywords in the URL due to there being a limited number of combinations of campaign keywords that would make sense in a website address. It is also likely that a portion of the publishers use the same site for multiple advertising campaigns and so those publishers will choose a more generic URL. As expected, 45% of the sites related to Company_A had a score below 10 for URL relevance.

4.3.4.7 HealthScore

The HealthScore is created by combining the scores from each of the features discussed above. I expected that the HealthScores would show a large cluster toward the bottom with a smaller cluster at the top and a relatively low number of sites spread between them. This theory was based upon the large number of low performing publishers that typically make up the long tail of an affiliate advertising campaign [3].

However, there was not a cluster of scores near the bottom end of the scale, and this can be attributed to the surprisingly high performance of the sites in the broken links and broken images categories.

4.3.4.8 Performance Scores

The original plan to measure real-world performance of the publishers was to use the conversion rate metric as that is what the digital marketing agencies I was working with generally used as a measurement of performance on their campaigns. However, it soon became clear that there were publisher sites, especially in the long tail, with a single click and a single sale. These sites would skew the numbers with 100% conversion rates and presumably low revenue for the advertisers. In order to mitigate this unfair advantage, rather than using the number of sales and number of clicks to calculate the conversion rate, the weighted average of these numbers was used as a measure of performance (section 3.3.4).

It would have been interesting to include the click-through-rate in this calculation but unfortunately, the unreliable reporting of impression data by either the networks or the digital marketing agencies made this impossible.

4.4 Experimental Setup

This research has defined and implemented six features to be included in the calculation for the publisher's HealthScore. In order to determine how the features relate to the performance score, the differentiation between a "Good" and "Poor" scores must first be made. The values chosen to be tested for these thresholds were based upon the range of scores for each feature.

Nine confusion matrices were created for each of the features using the three performance score thresholds and the three feature score thresholds. Using the confusion matrices, the

follow values were calculated for each test [199]:

- Precision
- True Positive Rate (Sensitivity)
- False Positive Rate
- True Negative Rate (Specificity)
- False Negative Rate
- Accuracy

Precision refers to the proportion of sites that were classified as having a “Good” feature score that are also “Good” performers. The precision is the most important of the calculated values for the features in this study, and the precision for both the “Good” and the “Poor” classifications has been calculated.

The sensitivity, or the true positive rate, identifies the proportion of sites that are “Good” performers that were also labelled as having a “Good” feature score and the specificity, or true negative rate, is the proportion of sites that are “Poor” performers and were also labelled as having a “Poor” feature score. The sensitivity and specificity are important measures, but none of the individual features are designed to capture all of the “Good” or “Poor” publishers. The sensitivity and specificity measurements are therefore more important in regards to the HealthScore as that construct is meant to identify as many of the “Good” or “Poor” publishers as possible. Finally, the accuracy is the proportion of the total number of correct predictions (“Good” and “Poor”).

The results with the best combination of measurements along with the corresponding confusion matrix for each feature can be seen in chapter 5, while the full list of confusion matrices and measurements can be found in Appendix E.

4.5 Conclusions

This chapter described the physical and logical cloud-based platform used to host the virtual machines that housed the various sub-systems responsible for data collection and analyses. The surfer agent is responsible for gathering data from the publisher sites while the visibility and site info agents gather related information from external sources such as search engines and blacklists. I also suggested implementing feature analysis rules in a high-level language like KWARESMI [198].

Upon receiving and examining the campaign performance data from the digital marketing agencies, it was apparent that the different networks relied upon varying types of performance metrics. As can be seen in Table 4.1, the amount of data reported from network C and network D was much sparser than that from either of the other two networks. All of networks reported the number of sales and the number of clicks, which allowed those values to be included in the calculation of the publisher's performance score created to represent the business performance of a publisher.

Although it seemed as though some of the networks were able to calculate several performance metrics, some of the data, such as the number of impressions, was missing or inaccurately recorded (Section 4.3.1). Throughout the crawling process there were issues with missing or malformed publisher URLs as well as several sites that were blatantly malicious or did not follow the terms of service for the network or advertisers. I reported my findings to the digital marketing agencies responsible for the affected campaigns.

After the surfer agent had collected the data from the publisher sites, several observations were noted during a manual sanity check of the data. Out of the six website features currently implemented, only four were able to be tested in this study as every site received a perfect score for both the URL Similarity and Blacklist Check features. With every site scoring a perfect score, I was unable to determine the relationship between these two features and the real-world performance of the publishers. Without knowing how either of these features affects publisher performance, it was not possible to determine an appropriate weight for them and so the weights on these features were set to zero when calculating the HealthScore. As described in section 3.3.3, this effectively turns these features off without needing to modify the HealthScore equation.

Once the data was collected and the feature scores had been calculated, it was necessary to determine which sites were to be labelled as "Good" or "Poor" performers for each of the features as well as for campaign performance. In order to do this, three threshold values were chosen for the performance score and three were chosen for each feature. Confusion matrices were created for each of the nine permutations of these thresholds, and the best fit was determined for each feature by choosing the matrix that had the best corresponding positive and negative precision.

The next chapter will focus on the reporting and evaluation of the results of these tests

along with the threshold chosen for each feature. A discussion of the implications of the results along with a critical analysis of the methodology and design decisions made throughout the research process will follow.

5 Evaluation

5.1 Introduction

This chapter describes the data collected from the publisher sites in more depth and then presents an overview of the findings. After that, an analysis of the results related to the performance score, each of the features and the overall HealthScores is presented. This analysis includes the confusion matrix corresponding to the chosen threshold for that feature along with the statistics derived from that confusion matrix. A full list of the confusion matrices produced in the tests for each feature can be found in Appendix E.

5.2 Affiliate Advertising Trial Overview

Although the trial conducted originally involved four advertisers, three of the advertisers had mostly incomplete performance data available. As such, it was not possible to conduct a meaningful evaluation of the performance of the publishers related these three advertisers. Only Company_A had enough performance data available to make a comparison between the publisher health and performance.

Even without the accompanying performance data for a majority of the sites, the sites for each of the advertisers were added to the job pool and a HealthScore was calculated for them using the methodology laid out in chapter 3. Overall, more than 4,000 publisher sites were scanned using the cloud-based infrastructure and custom purpose-built web-crawling agents described in section 4.2. The data gathering phase of the trial was conducted over the span of 20 days with an actual uptime of approximately 260 hours. During this time, the surfer agents gathered approximately 1.1TB of data from over 5.6 million pages. A score for each of the six features was then calculated from the analyses of more than 39.5 million content elements (such as links, images, HTML tags, and so on).

5.2.1 Data

Of all of the advertisers originally involved in the trial, Company_A had the largest set of URLs by far. The digital marketing agency that manages the advertising campaign in question at Company_A was able to provide 89 campaign keywords related to the goals of the campaign along with 2,224 URLs belonging to publisher sites that were enrolled on the

campaign at the time of the study. From these URLs, 523 were unable to be crawled by the surfer agents and were classified as null sites.

Although the system crawled and scored 1701 sites related to Company_A, adequate performance data was only available for 234 of the sites. More than one agency reported having to manually extract the performance data from the affiliate network system by copying and pasting the values into a spread sheet. Due to the time-consuming nature of extracting the data by personnel at the digital marketing agency, at least one agency only extracted performance data related to affiliates deemed to be active. In doing this, that agency effectively removed the performance data belonging to a large portion of their long tail and greatly reduced the number of sites usable from their campaign.

5.2.2 Findings/Results Overview

The data collected from the publisher sites was analysed to create scores for six features: broken links, broken images, visibility and URL relevance, URL similarity and blacklist check. Table 5.1 shows that out of 234 publisher sites, 68% (158 sites) scored well for the broken link category meaning that 32% (76 sites) had broken links present on the site. It can also be seen that 98% (229 sites) had no broken images on the sites. In terms of visibility, only 4% (10 sites) received a non-zero score and 44% (104 sites) had URLs containing campaign keywords. Fortunately for Company_A, the publishers and the digital marketing agencies involved, none of the sites had addresses that were considered to be typo-squatting URLs and none were on either of the blacklists checked. Unfortunately for this study, this meant the correlation between real-world performance of a publisher and the URL Similarity and Blacklist Check features was unable to be explored in this study.

Company_A Overview	**Count**	**Percent**
Sites Scanned	234	100%
Broken Links (Good)	158	68%
Broken Images (Good)	229	98%
Visibility (Good)	10	4%
URL Relevant (Good)	104	44%
URL Similarity (Good)	234	100%
Blacklist Check (Good)	234	100%

Table 5.1 Overview of Company_A Scan Findings

5.3 Performance Score Analysis

The measurement that was originally slated for use in determining the campaign performance of a publisher was that publisher's conversion rate. However, there were several issues associated with using the conversion rate (section 3.3.4) which prompted the creation of the performance score.

In looking at the distribution of performance scores shown in Figure 5.1, about 12% of the sites (28 sites) have a performance score between zero and five. This cluster of very low scores can most likely be attributed to those publishers forming the long tail of the campaign as explained in section 2.2.3. Other than these long tail publishers, the remainder of the performance scores are spread out relatively evenly and there does not appear to be any other major clusters of scores.

In order to use the performance score in comparisons with the feature scores and the HealthScore, it was necessary to determine where the cut-off between "Good" performers and "Poor" performers would lie along the range of scores. In order to determine the best value, three different performance thresholds were tested in the experiments conducted on each of the features. The values used corresponded to the cut-off for the top 75% of the scores (24.45), the mathematical mean (48.04) and the cut-off for the top 25% of the scores (73.39). The best balance between precision of "Good" and "Poor" feature scores corresponded with a performance threshold equal to the mean performance score for every feature tested. The other two thresholds were rejected because with the population balanced more heavily toward either "Poor" or "Good", there were generally too few sites on the opposite side to properly test the performance of the individual features. Therefore, any site with a performance score above 48.04 is considered a "Good" performer and any site with a performance score at or below 48.04 is considered to be a "Poor" performer.

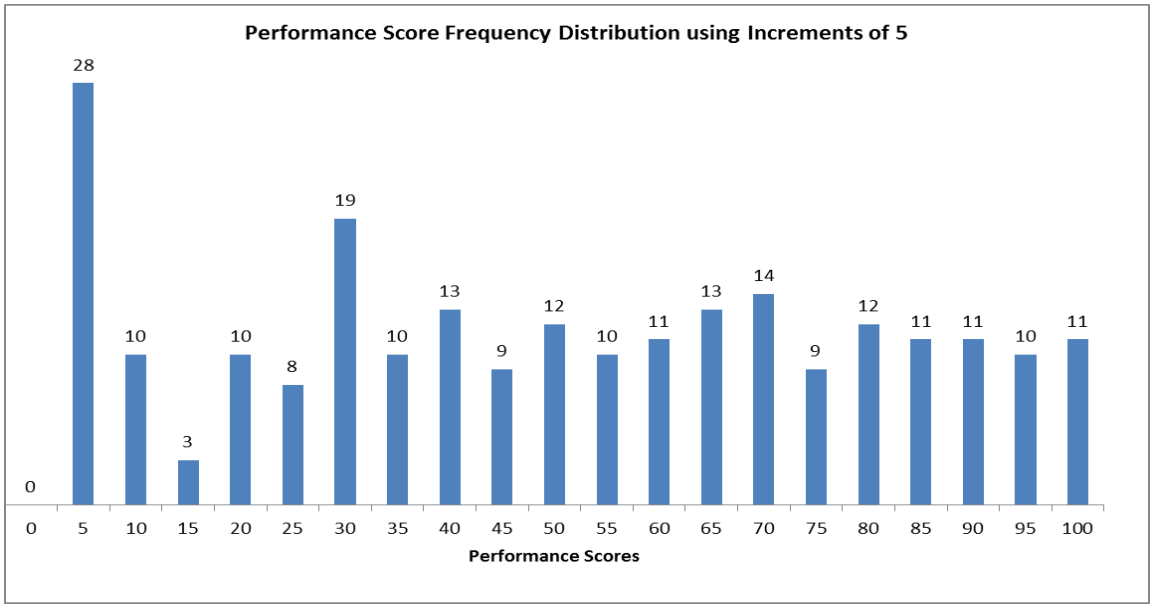

Figure 5.1 Performance Frequency Distribution

Measure	Performance
Count	234
Min	0.47
1st Quartile	24.45
2nd Quartile	48.24
3rd Quartile	73.39
Max	99.5
Mean	48.04
Median	48.24
Threshold	48.04
# above	119
# below	115

Table 5.2 Company_A Performance Statistics

After segmenting the population at the chosen threshold, it can be seen in Table 5.2 that 119 (51%) of the sites had scores that put them into the "Good" performer category and the remaining 115 (49%) sites were put into the "Poor" performer category.

5.4 Visibility Analysis

In the context of affiliate advertising, it makes sense that sites appearing high on search engine results for the keywords of an advertising campaign should be the best performing publishers as those sites should get the most exposure to visitors looking for specifically what is offered by the advertiser. Only 10 sites (4%) received a score above zero as shown in Table 5.3. I attribute this failing to the overly complex method of calculating the

visibility score and discuss methods of improving this feature in Section 6.2.1.

Measure	Visibility
Count	234
Min	0
1st Quartile	0
2nd Quartile	0
3rd Quartile	0
Max	2
Mean	0.06
Median	0
Threshold	0.00
# above	10
# below	224

Table 5.3 Company_A Visibility Statistics

The extremely low number of non-zero scores for the visibility feature forced the threshold to be set at zero meaning that any site scoring more than zero is considered to have "Good" visibility and any site scoring zero is considered to have "Poor" visibility.

While the implementation of the visibility feature is not perfect, Table 5.4 shows a Precision of 0.7 which means that out of the limited number of sites with a good visibility score, 70% (7) were also good performers. If the trend holds when there are more sites with positive visibility, then this feature could be a strong indicator of good performance.

Looking at Table 5.4, however, it can be seen that the sheer number of sites with a zero score has led to a false negative rate of 94%, meaning that out of all the sites that performed well, 94% (112) have a score of zero for the visibility feature. This means that a visibility score of zero does not mean that a site is any less likely to be a "Good" performer. In fact, the precision for the "Poor" sites was 50% meaning that the predictive power for "Poor" performers is no better than a random guess.

Measure	Value	
TP (Sensitivity)	0.06	
FP	0.03	
TN (Specificity)	0.97	
FN	0.94	
Precision (Good/Poor)	0.70	0.50
Accuracy	0.51	

Table 5.4 Visibility Calculations

Despite the small number of sites with a non-zero score, the precision of the feature was

relatively high at 70%. Giving the visibility feature a weight of 2.5 for means that the sites appearing in the search results for campaign keywords are given a boost in HealthScore and those not ranking within the results examined are only given a minor penalty.

5.5 URL Relevance Analysis

The idea behind the URL Relevance feature is that hopefully a potential customer encountering a publisher site with a URL containing campaign keywords is likely view that site as being well aligned with their goal (section 3.3.1), and is more likely to click on it.

In looking at the frequency distribution for the URL Relevance feature shown in Figure 5.2, the most noticeable trend is that 38% (88) sites have scored between zero and five for the URL Relevance Feature. This large cluster of sites on the lowest end of the score range may be due to publishers using sites that were designed for use across multiple campaigns and so the URLs are not overly specific to any single campaign. A cluster of sites at the low end of a feature's score range may also be explained by those sites being in the long tail of the advertising campaign. The second aspect of the feature scores highlighted in the frequency distribution is that no site scored above the 50 to 55 range. This is not an unexpected trend considering that in order to score above a 50 in the URL Relevance feature two campaign keywords must be present in the URL (section 3.3.1) and URLs that contain too many of the keywords may not actually make sense to a user.

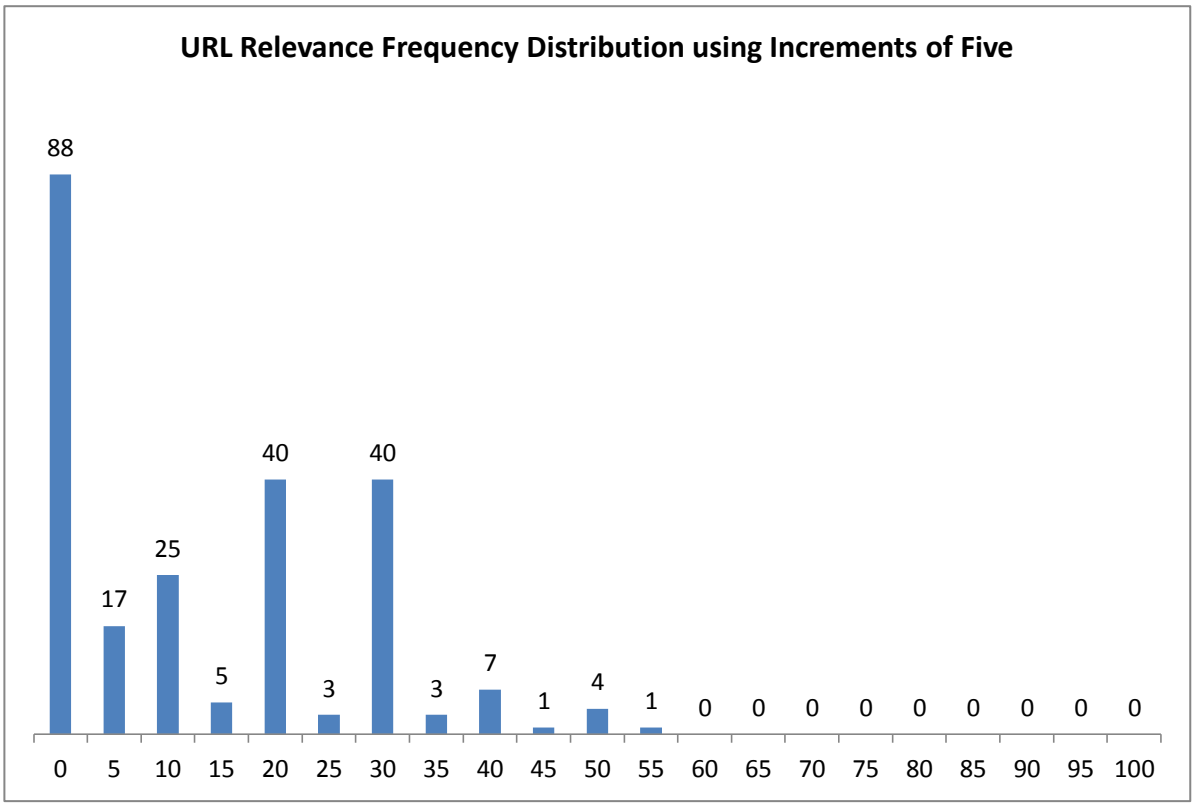

Figure 5.2 Frequency Distribution of the URL Relevance Feature

In order to determine the proper cut-off between sites with a "Good" URL Relevance score and those with a "Poor" URL Relevance score, three different threshold values were tested. These values were the median score (10), the mean score (13.52) and the cut-off for the top 25% of scores (23.13). Testing these three values against the three performance scores yielded nine confusion matrices which can be seen in Appendix F. The tests revealed that the threshold corresponding to the 3rd quartile provided the best balance of statistics. Therefore, any site with a URL Relevance score in the top 25% (greater than 23.13) is considered to have earned a "Good" score in the feature while any site with a score in the bottom 75% (less than or equal to 23.13) is considered to have a "Poor" URL Relevance score. This puts 58 sites in the "Good" category with 176 in the "Poor" category as shown in Table 5.5. The confusion matrix in Figure 5.3 corresponds to this chosen threshold.

Performance \ Relevance	Poor	Good
Poor	100	15
Good	76	43

Figure 5.3 URL Relevance Confusion Matrix (Relevance Threshold: 23.13)

Measure	URL Relevance
Count	234
Min	0
1st Quartile	0
2nd Quartile	10
3rd Quartile	23.13
Max	52.5
Mean	13.52
Median	10
Threshold	23.13
# above	58
# below	176

Table 5.5 URL Relevance Statistics

The precision of the URL Relevance feature came out at .74 for "Good" performers and .57 for "Poor" performers as seen in Table 5.6. This means that in regards to sites that were deemed to have a good URL relevance score, 74% also achieved a good performance score while 57% of the sites with poor URL relevance were also categorised as poor performers.

Measure	Value	
TP (Sensitivity)	0.36	
FP	0.13	
TN (Specificity)	0.87	
FN	0.64	
Precision (Good/Poor)	0.74	0.57
Accuracy	0.61	

Table 5.6 URL Relevance Calculations

Having achieved a precision of 74%, the URL Relevance feature is still not perfect. Like all of the individual features, it was unable to capture all of the "Good" performers by itself. In fact, of all the "Good" performers, only 37% also have a "Good" URL Relevance score. This is expected as it should not be necessary to have keywords in a site's URL to perform well on an advertising campaign, but those sites that do may have a small advantage as users may have an easier time recognising what content should be on the publisher's site [163].

Having performed as expected in the good feature score category as well as being above 50% in the poor feature score category, the weighting of the URL Relevance feature was

set to a weight of three as this puts slightly more significance behind the feature than the visibility metric. This was deemed an appropriate setting for the weight of this feature as the URL Relevance feature had the highest precision for "Good" sites while still maintaining above a 50% precision for "Poor" sites.

5.6 Broken Link Analysis

Broken links can be a source of user frustration or mistrust of a publisher's site, both of which can lead to a loss of revenue [186, 187]. The frequency distribution in Figure 5.4 shows that 74% (172) of publisher sites on the campaign for Company_A scored between 95 and 100 for broken links. With a majority of the sites being so healthy, picking a threshold was more difficult for this feature with the cut-off between "Good" and "Bad" broken link score likely to be quite high. Indeed, after testing the cut-off for the top 75% (95), the median score (99) and just above the top 75% cut-off (96), 96 came out with the best balance of statistics. The confusion matrix corresponding with this threshold is depicted in Figure 5.5.

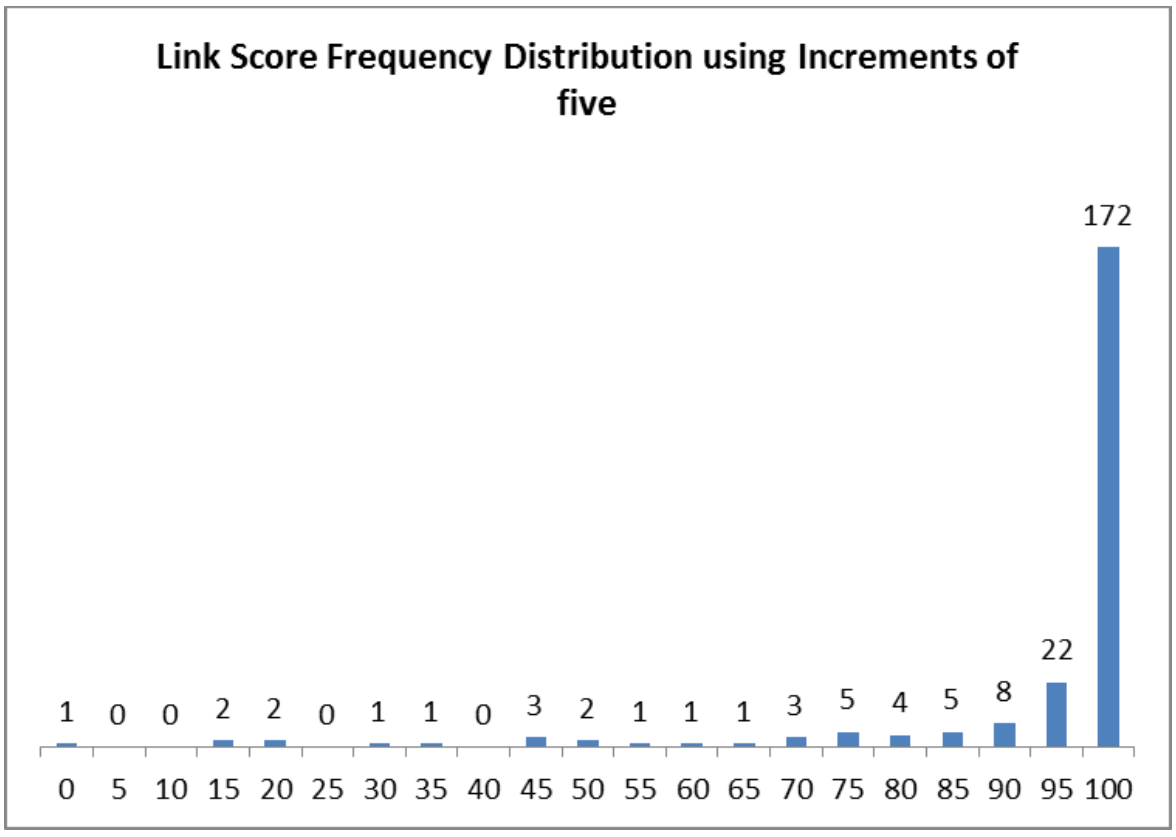

Figure 5.4 Frequency Distribution of Broken Link Scores

Performance \ Link	Poor	Good
Poor	41	74
Good	30	89

Figure 5.5 Broken Link Confusion Matrix (Link Threshold: 96)

After segmenting the sites based upon the chosen threshold, 70% (163 sites) of the sites are classified as having a "Good" score for broken links and 30% (71) of the sites are classified as having "Poor" broken links scores as shown in Table 5.7.

Measure	Links
Count	234
Min	0
1st Quartile	95
2nd Quartile	99
3rd Quartile	100
Max	100
Mean	92.51
Median	99
Threshold	96.00
# above	163
# below	71

Table 5.7 Broken Link Statistics

The cluster of sites with high scores means that there were several sites with a "Good" feature score that are actually "Poor" performers on the campaign as with the other features. This high number of false positives is confirmed in Table 5.8 which also shows that despite the high number of false positives, 55% of the sites with a "Good" feature score were also "Good" performers. Even more interesting than that, however, is the fact that out of all of the sites with a "Poor" broken link score, 58% were also "Poor" performers.

Having a "Poor" broken link score appears to be a weak indicator of being a "Poor" performer. Because of this, the weight for broken links is set to one meaning that the score is not given any extra preference when calculating the HealthScore, but that it is still considered in the calculation.

Measure	Value	
TP (Sensitivity)	0.75	
FP	0.64	
TN (Specificity)	0.36	
FN	0.25	
Precision (Good/Poor)	0.55	0.58
Accuracy	0.56	

Table 5.8 Broken Link Measurements

5.7 Broken Image Analysis

Broken images on a publisher's site can cause users to view the site as incomplete or unprofessional which may lead to a loss of revenue due to user frustration or mistrust of the site [186, 187]. For this particular campaign, that does not seem to be an issue as the broken image scores are heavily clustered with 98% (230) of the sites scoring between 95 and 100.

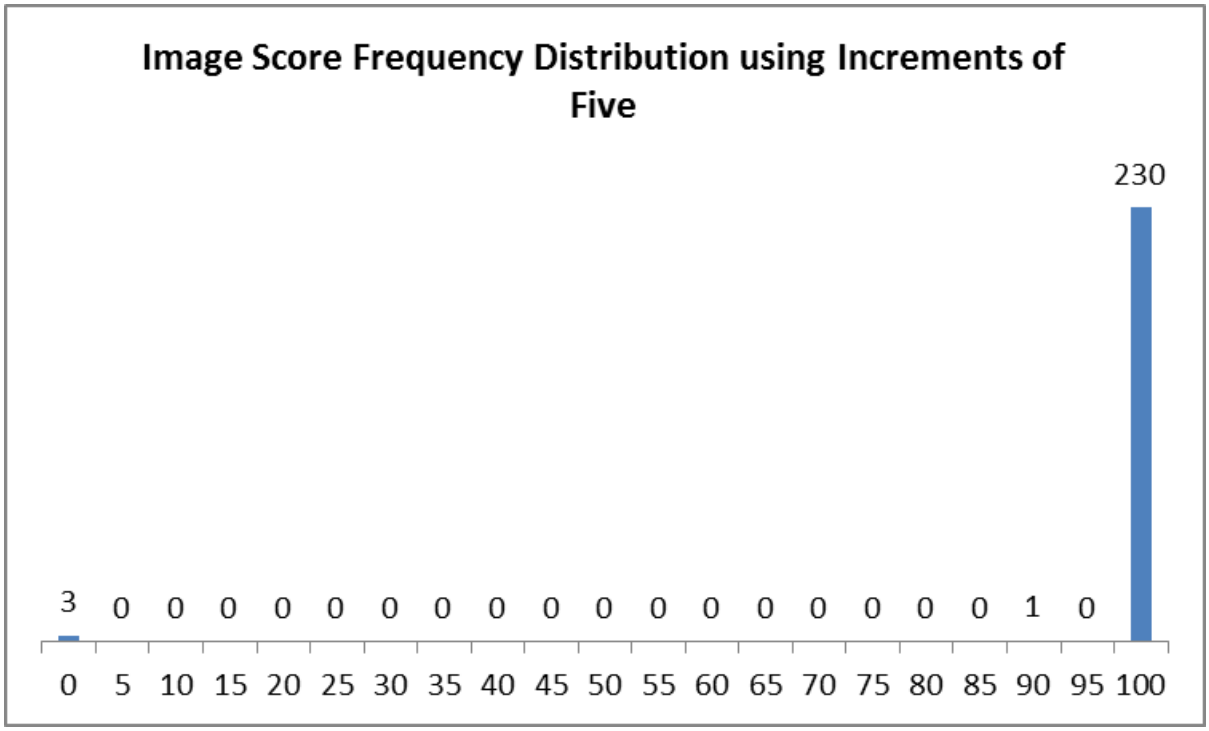

Figure 5.6 Frequency Distribution for Broken Image Scores

With the vast majority of the sites on the campaign scoring so high, the thresholds that would normally be tested were all calculated at 100 as show in Table 5.9. In order to find a threshold that worked well, more tests were run than with the other features. The thresholds

tested were the mean score (98.57) and the integers between 95 and 99. The mean down through 96 all yielded the same results, and the best were found when using 99 as the cut-off. This meant that having a single broken image on a site would cause the site to be classified as "Poor" in regards to broken links.

Measure	Images
Count	234
Min	0
1st Quartile	100
2nd Quartile	100
3rd Quartile	100
Max	100
Mean	98.57
Median	100
Threshold	99.0
# above	210
# below	24

Table 5.9 Broken Image Statistics

Even with a threshold of 99, it can be seen in the confusion matrix in Figure 5.7 that a very small portion of sites fall into the "Poor" category. In fact, only 24 sites (10%) had any broken images on them at all with the remaining 210 sites (90%) having no broken images.

Performance \ Image	Poor	Good
Poor	17	98
Good	7	112

Figure 5.7 Broken Image Confusion Matrix (Image Threshold: 99)

Despite the false positives that can be seen in Table 5.10, out of the sites assigned a "Good" image score, 53% were classified as having good campaign performance. While this does not suggest that having a "Good" score for broken images is a particularly strong indicator of a site that will perform well, out of sites with a poor image score 71% also had poor performance. This suggests that having a poor image score may in fact be a relatively strong indicator of a site that may perform poorly.

Measure	Value	
TP (Sensitivity)	0.94	
FP	0.85	
TN (Specificity)	0.15	
FN	0.06	
Precision (Good/Poor)	0.53	0.71
Accuracy	0.55	

Table 5.10 Image Score Calculations

Due to the increased performance of the broken image feature in cases of "Poor" performers, it was decided that this feature should receive more weight than the broken links feature when calculating the HealthScore. In order to minimise the effect of the relatively poor performance in regards to "Good" performers, the weight for the broken link feature was set at two.

5.8 HealthScore Analysis

The HealthScore construct is a combination of the various feature scores weighted based upon the experimental results presented so far in this chapter.

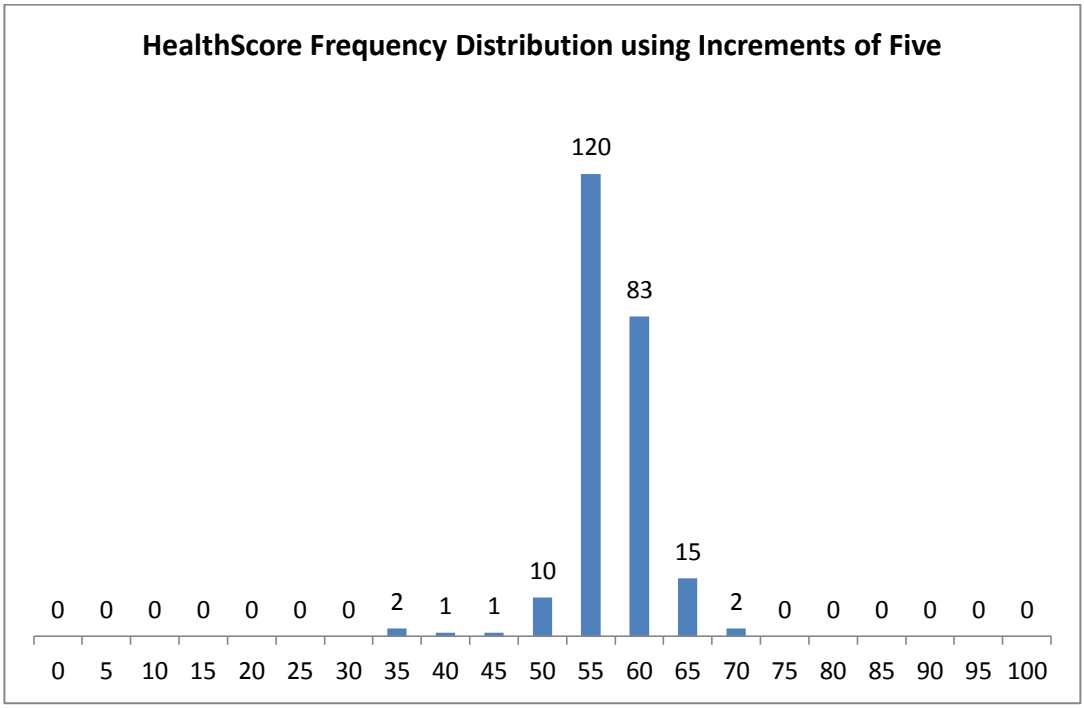

Figure 5.8 Frequency Distribution for the HealthScores

In looking at the frequency distribution shown in Figure 5.8, it can be seen that 87% (203) of the sites have a HealthScore between 55 and 65. This is most likely due to the fact that the scores for the features are very similar to each other, especially those for the broken links and broken images features. The lack if variety in feature scores has served to create a lack of variety in the HealthScore as well.

In determining the cut-off between a site with a "Good" HealthScore and one with a "Poor" HealthScore, the median (54.7), mean (54.8) and cut-off for the top 25% (57.4) were tested as possible thresholds. Of the three possible threshold values tested, the median presented the best balance between precision and the ability to capture the majority of the "Good" and "Poor" performers. As can be seen in Table 5.11, this choice put 57 publishers into the "Good" HealthScore category while 177 publishers were put into the "Poor" HealthScore category. The confusion matrix corresponding to this threshold can be seen in Figure 5.9.

Measure	**HS**
Count	234
Min	34.78
1st Quartile	52.09
2nd Quartile	54.70
3rd Quartile	57.39
Max	65.78
Mean	54.81
Median	54.70
Threshold	57.39
# above	57
# below	177

Table 5.11 HealthScore Statistics

Performance \ HealthScore	Poor	Good
Poor	69	46
Good	51	68

Figure 5.9 HealthScore Confusion Matrix (HealthScore Threshold: 54.69)

The calculations shown in Table 5.12 indicate that 60% of publishers with a good HealthScore also performed well on the advertising campaign, and 58% of publishers with a poor HealthScore were also poor performers. Unlike the individual features, the HealthScore should be capable of properly classifying the majority of "Good" and "Poor" performers and out of all the "Good" performers, 57% were also classified as having "Good" health while 60% of all the "Poor" performers were classified as having "Poor"

health.

Measure	Value	
TP (Sensitivity)	0.57	
FP	0.4	
TN (Specificity)	0.6	
FN	0.43	
Precision (Good/Poor)	0.6	0.58
Accuracy	0.59	

Table 5.12 HealthScore Calculations

These findings suggest that the HealthScore developed in this thesis may be a good indicator of how well a publisher might be able to perform when matched with an appropriate advertising campaign. For a further discussion of the results for each of the features and the HealthScore, see chapter 6.

5.9 Conclusion

This chapter presented the results of the experiments conducted on a study involving four active affiliate advertising campaigns. The various digital marketing agencies in charge of managing the campaigns were able to share over 4,000 publisher URLs in total, but only Company_A had enough associated performance data to complete a proper evaluation of the currently implemented features. The digital marketing agency for Company_A had 2,224 publisher site addresses on file, and after the agents attempted to crawl the sites, 523 of the sites were unable to be scanned. Out of the remaining 1,701 sites, performance data was only available for 234 of them and those sites correspond to the results described in this chapter.

The study consisted of evaluating the performance of six features: broken links, broken images, visibility, URL relevance, URL similarity, and blacklist check. The weights for the URL similarity, and blacklist check features were set to zero in order to disable them after discovering that all of the sites had received a perfect score in these two features, making it impossible to gauge their relationship to real-world performance.

The broken link and broken image features were better at identifying weak performers than they were at identifying good performers. In the case of sites scoring poorly in the broken image feature, 71% were also “Poor” performers on the advertising campaign in comparison to 53% of sites scoring well in the broken image feature also being “Good”

performers. In regards to the broken link feature, 58% of the sites with “Poor” broken link scores were also “Poor” performers and of those with a good broken link score, 55% were also “Good” performers.

The visibility feature managed to achieve a high precision for identifying “Good” performers. While only a very small number of sites (10) received a visibility score above zero, 70% (seven) of those sites were also “Good” performers on the campaign. However, the visibility feature is unable to detect “Poor” performers any better than a random guess. This is a strong indication that a site with a visibility score above zero is likely to perform well, but that a good visibility score is certainly not required in order to perform well. Further testing should be conducted in order to ensure that this trend also applies to data sets with a larger number of sites with non-zero visibility scores.

In regards to the URL Relevance feature, 74% of sites with a “Good” score were also “Good” performers while 57% of sites with a “Poor” URL Relevance score were also “Poor” performers.

Individually, the feature scores are not designed to provide an absolute prediction of how a site may perform, but the HealthScore combines the features into a single score in order to allow for this prediction. Out of the sites that had “Good” HealthScores, 60% were also “Good” performers on the campaign. The percentage of sites with a “Poor” HealthScore that were also “Poor” campaign performers was slightly lower at 58%.

Unlike the individual scores, the HealthScore should be able to classify a majority of the “Good” and “Poor” sites. Out of all of the publisher sites, the HealthScore was correctly able to identify 57% of all of the “Good” performers and 60% of all of the “Poor” performers and had an overall accuracy of 59%.

While this level of performance does suggest that the HealthScore is able to correctly classify a majority of publisher sites, there is still room for improvement. Unfortunately, when using data from a real campaign, it is not possible to control the distribution of performance amongst the publishers of the campaign. The experiments should be repeated with more campaigns in order to see if the results extend beyond this campaign, especially to those with a wider variety of performance amongst the publishers.

The next chapter will discuss the conclusions drawn from this research along with the limitations of the methodology used and present a further discussion on the implications of

the results presented in this chapter. It will also present several paths that this research could take in the future in order to further explore the automated evaluation of potential publisher performance.

6 Conclusions and Future Work

6.1 Introduction

In this chapter, the final conclusions that have been drawn from the work discussed throughout the rest of the thesis are presented. Section 6.2 discusses the concept of using web site features as indicators of potential site performance while section 6.3 presents the validation of the HealthScore construct. Section 6.4 offers a summary of the three research questions along with a critical appraisal of the progress of answering them. Finally, the chapter concludes with section 6.5 presenting the path for future work.

6.2 Website Features as indicators of potential performance

The first objective set in order to achieve the aim of exploring the correlation between web site features and business performance of a publisher on an advertising campaign was related to the selection of those features that may be suitable for use in predicting the performance of a publisher.

In order to determine which web site features would be examined as potential indicators of performance in an affiliate advertising context, an initial list of features that were likely to be important had to be identified. Chapter 2 details the thorough review of the current academic and professional literature surrounding the different methodologies of evaluating interactive systems, such as web sites, in various contexts. The literature covered user-based testing, evaluator-based testing and tool-based testing techniques as well as a plethora of website and affiliate advertising success measurements. Once a list of features was created, the list was refined from the original 41 features to a more manageable set of 17 web site features. Although there are 17 features defined, time constraints limited the number of features chosen for implementation to just six. The other 11 features that have yet to be implemented can be found in Appendix F. The six currently implemented website features are visibility, URL relevance, URL similarity, broken link analysis, broken image analysis and blacklist check. After crawling the sites for data related to these six features, the blacklist check and URL similarity features yielded a perfect score for all 234 sites

making it impossible to determine the effect of these features on publisher performance. As such, the weights for these two features were set to zero which effectively eliminated them from the HealthScore calculation.

It should be noted that the individual feature scores are not meant to be a complete picture of overall site performance, but rather the level of health of that individual feature. It is through the combination of the various feature scores that the predictive power is realised. DeLone echoes this sentiment and points out the need for a "comprehensive success construct" rather than a simple measurement for system success [74]. For example, a site with enough broken links to receive a "Poor" feature score will not necessarily be a "Poor" performer on the advertising campaign. The presence of broken links is likely to reduce the publisher's performance, but does not automatically make the publisher fall into the "Poor" performance category. The reporting of the feature score makes it possible for users to see that there are broken links on the site that could be fixed in order to improve the performance of the publisher.

Each of the four remaining features underwent a series of experiments in order to determine the best threshold value to differentiate between sites being classified as having a "Good" or "Poor" score for that feature. After a threshold was chosen, each of the four features was able to achieve a precision above 50% in both "Good" and "Poor" predictions, but none of the features were without faults.

6.2.1 Visibility

The visibility feature was born from the idea that the keywords associated with an advertising campaign will be well aligned with the goals of a customer likely to convert. It follows that a publisher site that ranks well in organic search for those keywords is also well aligned with the customer's goals, and should attract more conversions.

However, as can be seen in section 3.3.1, only ten (4%) publisher sites received a non-zero visibility score. Visibility was essentially transformed into a binary feature meaning that any non-zero score was considered a "Good" feature score. Out of all of the sites with any visibility score, 70% were also "Good" performers. On the other hand, the feature only identified 6% of the entire population of "Good" performers. While this is a very low percentage, remember that the individual features are designed to give a complete picture of the publisher's performance. These numbers point towards the fact that a site does not need

to rank well on organic search for the campaign keywords in order to perform well but those that do are likely to be "Good" performers. This trend fits with the hypothesis made in regards to the visibility score.

H2. Having a good visibility score will positively affect campaign performance.

In order to improve the overall detection rate of the visibility score, there is a need to re-examine the process with which the score is calculated. Rather than combining the results for all of the keywords, it would be interesting to investigate:

- Checking more search results or possibly even more search engines to see if more matches can be found.
- Pruning the keyword list to remove any words that no site ranks for in order to remove keywords that may have been poorly chosen.
- Automatically mining the campaign keywords from the advertiser's web site similar to a simplified version of the process used by the search engines to rank the publisher sites, and may improve the results. This would help to remove the subjectivity introduced when determining which keywords were most closely rated to the campaign.
- Weighting the search engines based upon either preference of the programme manager or which search engine has previously sent the most high-quality traffic to the sites on the advertising campaign.
- Calculating the final visibility score using only the keywords that rank for each individual site to avoid adding zeros to the calculation when a site does not rank for a keyword. This would ensure that the only sites to receive a score of zero would be those that do not rank for any of the campaign keywords, in which case a zero is more appropriate.

6.2.2 URL Relevance

The URL Relevance score is based around similar thinking to that behind the visibility feature: sites that are well aligned with a customer's goals are likely to attract high-quality traffic. A site with campaign keywords in the URL is clearly identifying its relevance to that campaign, and relevance is a part of the D&M success model [74].

Like visibility, URL Relevance was able to achieve a high precision. Out of all of the sites

with a “Good” URL Relevance score, 74% were also “Good” performers and out of those with “Poor” URL Relevance, 57% were “Poor” performers. Compared to visibility, URL Relevance was able to identify a slightly higher percentage of all of the “Good” performers with 36%. This is still not very high, but does point toward a publisher’s site with keywords in the URL being likely to be “Good” performer, but that it is not a necessity. This trend fits with the hypothesis formed in regards to URL Relevance.

H3. The presence of campaign keywords in a publisher URL will positively affect campaign performance.

One issue with the calculation of the URL Relevance feature score is that adding a set amount per keyword creates clusters of scores as can be seen in Figure 6.1.

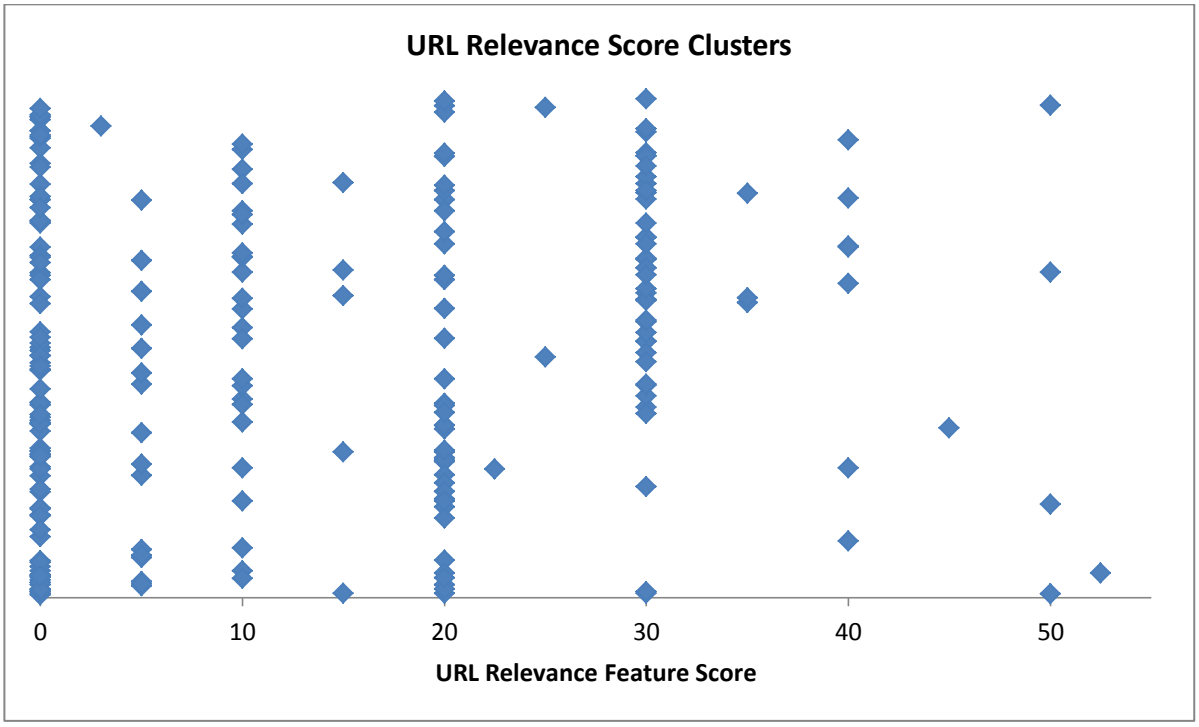

Figure 6.1 URL Relevance Feature Score Clusters

As mentioned in the previous section, it would be interesting to investigate the effects of using automatically mined phrases from the advertiser’s web site as the campaign keywords and using the frequency of appearance of each to rank them. Not only would this help to remove some of the subjectivity from the process, but it would also allow the URL relevance feature to be more automated as the only manual step left would be the validation of the mined keywords to be conducted in co-operation with the programme manager.

6.2.3 Broken Link Analysis

A site with broken links can leave users with an impression that the site is unfinished or that the publisher is unprofessional, both of which are likely to negatively influence user satisfaction, trust and purchasing behaviour.

Of all the sites with broken links, 57% are "Poor" performers. However, of all "Poor" performers, only 36% had a "Poor" broken link score. Thus, having broken links is not a requirement of a "Poor" performer, but that having broken links may be a weak indicator that a site is a "Poor" performer. This trend fits with the hypothesis formed in regards to broken links.

H4. The presence of broken links on a site will negatively affect site performance.

The broken link feature seems as though it performs as expected, but the relatively small percentage of sites with a "Poor" broken link score (32%) may introduce bias. Evaluating the performance of sites with a more varied degree of broken links may produce different results. There is not much that can be done to change the broken link calculations in order to help improve the accuracy, but even as a weak indicator it is still an important feature even if it is only used to help publishers maintain sites by pointing out broken links on their site.

6.2.4 Broken Image Analysis

Like broken links, broken images can make a site feel incomplete and unprofessional and cause a user to be dissatisfied with the publisher's site.

Of all the sites with broken images, 70% are "Poor" performers. Like broken links, however, of all "Poor" performers, only 15% had a "Poor" broken image score. Thus, this feature seems to indicate that a site with broken images is likely to also be a "Poor" performer, but even "Poor" performers may have no problem with broken images. This trend follows the hypothesis in regards to broken images.

H5. The presence of broken images on a site will negatively affect site performance.

The large number of publishers with very high scores in this feature may be due to the fact that broken images are usually displayed as a square with a large red 'X' or other easy to recognise indicator designed to make it easy to spot when reviewing. It would follow then that these errors mostly go unnoticed by those publishers that do not have either the time or

the inclination to review the publisher site regularly. This lack of time or motivation may also be indicative of the level of effort put into the creation of the site in the first place or the promotion of the site and may lead to poor campaign performance.

While the precision of this feature is quite high, there were only 24 (7%) sites with any broken images. This means that, like the broken links feature, there may be some bias introduced due to the extremely low number of "Poor" scores.

There is not much that can be done to change the broken image feature in order to improve the accuracy, but even as a weak indicator it is still an important feature even if it is only used to help publishers maintain sites by pointing out broken images on their site.

6.3 HealthScore Validation

The second main objective of this research was related to the possibility of measuring the potential performance of a publisher site in a way that could be validated. In order to achieve this objective, such a measurement was created.

The ability to calculate a HealthScore for a publisher in order to predict the business performance of that publisher has the potential to drastically change the way the affiliate advertising industry audits existing publishers and screens potential additions to a campaign. The main difficulty lies in the validation of the HealthScores assigned to the publisher sites as these must be compared against some measurement of actual site performance. Previously, system use has been found to be a good measurement of system success [75, 91, 92, 93, 88], and in the D&M success model system use includes the number of visits and number of sales [74] which are very similar to the two measurements used in this research to create the publisher's performance score: the number of clicks and the number of sales.

While the individual feature scores are not designed to be completely indicative of publisher performance, the HealthScore was created to use those scores to derive a measurement that was capable of predicting a publisher's performance on a suitably matched campaign. Before being able to validate the HealthScore construct, the measurement of publisher success must first be defined.

6.3.1 Performance Score

The idea behind the performance score is to measure the real-world performance of the

publisher's site in a manner similar to Lee and Kovar's attempt at matching user rankings of business websites to actual business performance. This study was done using the corporate sites of major companies and comparing user web site preference to offline business performance of the companies. Even though the business performance being measured was that of the entire company and not necessarily constrained to e-commerce, the authors found that user's preference rankings closely matched that of business performance (with the exception of two sites being swapped) [77].

The original plan to measure a publisher's level of performance on the campaign in this study was to use the conversion rate measurement. This number is simply the ratio of unique visitors to the number of conversions as presented in Section 2.2.4.3, which are similar to two of the measurements used in the D&M success model for the system use dimension. While the conversion rate was originally thought to be an adequate measure of performance, after working with the data for a short time, it became apparent that there were several issues with using conversion rate as the sole unit of performance measurement (section 3.3.4).

In order to combat these issues, the weighted average of the number of conversions and the number of clicks was calculated in a similar manner to that of the HealthScore equation (Equation 3.5) to form the publisher's performance score. The number of conversions was given a higher weight than the number of clicks as a sale is guaranteed revenue while a click or visit only gives the possibility of revenue.

In order to determine how each of the features relate to a site's campaign performance, the performance score was used to classify each of the sites as either a "Good" or "Poor" performer in relation to the advertising campaign. During the experiments conducted on the features, three performance score thresholds were also examined to find the best fit. The best balance between "Good" and "Poor" precision for each of the features occurred when using the mean performance score (48.04) as the threshold.

In examining the distribution of "Good" and "Poor" conversion rates, there were 112 publishers listed on either side creating an exact 50/50 split. With the new performance score, there are 119 "Good" performers and 115 "Poor" performers meaning the distribution has not changed significantly, but the publishers with a single click and a single sale no longer come out as the top performers.

6.3.2 HealthScore

Individually, the previously mentioned features help to highlight potential issues of a publisher's site, but they are not meant to provide a complete picture of potential publisher performance when looking at the various feature scores by themselves. The HealthScore is the combination of all of the features, and is meant to give a quick, easy to interpret indication as to how a publisher should perform on a suitably matched advertising campaign.

Despite the small range of HealthScores, out of all the publisher sites with a "Good" HealthScore, 60% were also "Good" performers and out of those with a "Poor" HealthScore, 58% were "Poor" performers.

H7. A site with a good HealthScore is also likely to be classified as a good performer.

H8. A site with a poor HealthScore is also likely to be classified as a poor performer.

Unlike the individual features, two hypotheses were made regarding the HealthScore and the trend in the HealthScore performance appears to support both of the hypotheses.

6.4 Summary and Critical Appraisal

The main aim of this research was to explore the correlation between the web site evaluation features of a publisher site and that publisher's performance on an affiliate advertising campaign. In order to fulfil this aim, three questions were defined.

> **Q1. How can a site's real-world performance be measured and reported in such a way that a comparison between the site's health and business performance can be drawn?**

The first question relates to the ability to measure the real-world performance of a publisher's site. Section 2.2.4 explains that out of the digital marketing agencies that provided data for this research, the majority use conversion rate as a major indication of publisher performance. However, using the conversion rate presented several issues that are highlighted along with specific examples in Section 3.3.4. To alleviate those issues, the performance score incorporates the same two factors are combined to calculate the conversion rate (number of clicks and the number of sales), but allows for them to be assigned an individual weight. This weighting allowed the number of sales to be prioritised over the number of clicks. While the use of the performance score helped to solve some of

the issues encountered with the conversion rate, more work should be done surrounding the validation of this new measurement of performance. If revenue data were available for the publishers, it would be possible to compare each publisher's performance score to their revenue earned.

Q2. Can scores derived from the various features of a publisher web site be combined in order to create a useful overall measurement of the site's health?

The second question is surrounding the selection of the appropriate features to be used in achieving the aim of this thesis. The four features that yielded results represent a start in creating a measurement of publisher site health for use as a predictor of campaign performance. While these four features all achieved scores that supported their respective hypotheses and have served well as a proof-of-concept implementation, there is work yet to be done in order to verify if the trends hold true for larger data sets from other publishers and networks. Having a group of publisher sites with a more varied range of campaign performance may have provided a better test environment for the feature evaluations. It was not possible to control for this in this particularly study as the experiments were completed using real performance data rather than examples created in the lab. This means that in the future, very large data sets may be necessary in order to get a good mix of performance scores or a method of fabricating publisher sites with realistic associated performance data would need to be created. There is also room for improvement with the addition of more implemented features. Having more feature scores included in the calculation of the HealthScore would have likely contributed to the HealthScore being a more accurate measurement of publisher performance.

Q3. How well can the HealthScore construct defined by this research be used as an indication of publisher performance on an advertising campaign?

The third, and main research question was concerned with how well the HealthScore functioned as a measurement of prediction and how this would be validated. The results from the comparison of the HealthScores and performance scores are a promising start, but there is still work to be done before it can be conclusively declared as a good indicator of real-world publisher performance. The HealthScore was able to achieve a precision of 60% on predictions of "Good" performers and a precision of 58% on "Poor" performers with an overall accuracy of 59%. This means that while the HealthScore was able to predict which classification a publisher falls into a majority of the time, there is still much room for

improvement. The first version of Ivory's Web TANGO system was able to achieve an accuracy of 67-80% with 11 features [155]. However, upon improving the existing features and adding more to total 157, the system was able to achieve a 94% accuracy [156].

The research presented in this thesis serves to show that predicting a publisher's real-world performance using web site evaluation metrics from the publisher's site is a possibility. With the implementation of more features and refinement of the algorithms behind the calculation of current feature scores, I feel that the predictive power of the system presented in this thesis, like Ivory's, will increase.

6.5 Future Work

The work presented in this thesis has several areas where improvements could be made such as the re-working of the calculations for some of the web site features, the addition of new metrics and the examination of sources beyond the publisher's site.

6.5.1 Feature score improvements

While all four of the features implemented in this appear to fulfil the hypotheses set for them, the features were not without fault. None of the features achieved a sensitivity or specificity above 50%, which means that none of the individual features was able to classify more than half of all the "Good" or "Poor" performers into the correct category. While the features are not meant to be able to classify all of the publishers on their own, having features that are able to identify a larger proportion of "Good" or "Poor" performers will help to improve the accuracy of the HealthScores predictions. One method of improving this accuracy is through improving the algorithms behind the currently implemented features. The Visibility and URL Relevance features achieved the highest precisions for predicting "Good" performers, but more testing is required before being able to conclusively say that either of these features can be used in the creation of a strong predictor of "Good" performers.

In regards to the visibility feature, there are simply not enough sites with non-zero scores to determine if the trends shown in this study will extend beyond this data set. Visibility was also the only feature that was completely incapable of helping to predict a "Poor" performer. In order to improve the overall detection rate of the visibility score, there is a need to re-examine the process with which the score is calculated. Currently, the visibility

score is calculated based upon how well a site ranks on several search engines for all of the campaign keywords. In the study conducted on Company_A there were 89 campaign keywords and it is likely that many of the publisher sites only ranked for small number of them, if any at all. It would be interesting to investigate how well the visibility score would work if keywords that returned no results for any sites were pruned from the list or if the score was calculated only from the rankings of keywords that the site actually had a rank.

The current method of having the keywords selected and ranked by the programme manager runs the risk of introducing a significant amount of bias, which could possibly account for the very low visibility scores. The concept of automatically mining the campaign keywords from the advertiser's web site would be an interesting path to examine in future research regarding this feature. Incorporating a method similar to Latent Semantic Analysis (LSA) to determine which keywords are important to the advertiser may better capture the spirit of the campaigns and help to improve the result by removing some of the subjectivity.

As for the URL Relevance feature, a list of keywords that are more closely related to the campaign would likely help to boost the number of sites scoring positively in this metric. The use of a technique such as LSA would also benefit the URL Relevance feature by providing a method of computing the appropriate weights for the individual keywords based upon actual relevance to the advertiser's site. Currently, the keywords are weighted by a person, which does not provide the same level of granularity that using the actual keyword relevance would allow. The similarity in keyword weighting creates scores that are similar to one another, and causes feature score clumps as shown in Figure 6.1.

As outlined in Section 3.3, each feature must adhere to a set of three rules, the last of which required that it is possible to automated to data collection and analysis needed to calculate a score for the feature. While implementing the analysis system used in this research, a concession was made regarding the URL Similarity feature in that the automation was not implemented. In order to complete that automation of this feature, it may be helpful to look at some of the techniques used in spell checking solutions. Huang, Murphey and Ge developed a system capable of detecting and correcting typographical errors related specifically to the automotive domain [200]. The authors pointed out that domain specific typo identification is different than the normal spell checking process in word processing

software due to the large number of self-defined words that may be commonly used in a specific domain. Using the automatically mined keyword list would provide a campaign specific knowledge base that could then be used to examine the URLs of the publisher sites and find misspellings of campaign specific words (namely organisation name) in order to automate the URL similarity feature check.

Along with automating this measurement, it may also prove useful to extend the feature. The main idea behind the URL Similarity feature is to detect malicious sites that are using typosquatting to take advantage of the target organisation's brand name. Extending the feature to include anti-phishing techniques would likely be more useful than simply detecting if the target organisation's name is present in the URL. Studies have found that metrics such as a large number of sub-domains, a longer than normal URL and a shorter than normal domain name are common in phishing URLs [192] [182]. Apart from examining the URL, Banerjee, Rahman and Faloutsos also discovered that phishing sites rely heavily on http redirects in order to fool anti-phishing tools. The network-layer behaviour of these sites is significantly different from that of a non-malicious site and allows the authors to identify malicious sites with a high degree of accuracy [201]. Extending the feature to include considerations such as these would likely lead to a much more robust feature in future versions of the analysis system. The URL Relevance feature also included a manual component when it came to calculating the scores for the feature. The formula used to calculate this feature score awards a site full points for the first campaign keyword that appears in the URL and half points for up to two additional unique keywords. Rather than taking the time to write and test a complete program or excel formula to assign the half points, I chose to go with the quicker option of manually assigning the half points. This is a simple task to automate, but the feature is currently in violation of the automation rule because of this time saving compromise.

While the rule related to feature automation does not include any set up required prior to data gathering, the URL Relevance feature would likely benefit from some automation in that area as well. This feature currently relies on a list of campaign keywords supplied by the campaign manager from either the advertiser or the digital marketing agency. If the system were to use web content mining techniques in order to extract key ideas form the advertiser's site, it may be possible to automate the production of the campaign keyword list. Turney and Pantel describe a concept called the bag of words hypothesis, which comes

from the field of information retrieval, as stating that the frequency of words in a document often indicates the relevance of those words to the document [202]. Following on this principle, using the frequency of terms on the advertiser's site may produce a list of keywords that could both replace the manually produced list and provide insight to the advertiser into the messages conveyed by their site. Implementing this level of automation for the URL Relevance feature may lead to a more robust metric in future implementations of the analysis system.

6.5.2 Additional features

The performance of the HealthScore shows that the features used to calculate it do not cover all of the characteristics that are able to determine publisher site performance. Like the first version of Ivory's Web TANGO [154], the research presented in this thesis could benefit greatly from the addition of more features incorporated into the HealthScore calculation as well as more publisher sites with accompanying performance data. In order to change the way that a feature score is calculated or to add new features, the underlying code for the existing agents must be changed. More extensibility with a database to store and retrieve rules written in a high-level language like KWARESMI would make it easier to test more features in the future [198]. During feature refinement phase of this research, eleven features were identified that were unlikely to be implemented in the allotted development time. Those features, which are listed below, have been classified into the categories of domain analysis, content analysis, campaign compatibility, terms and conditions and site profile.

6.5.2.1 Domain Analysis

a) Digital Certificate Evaluation (Green)

The digital certificate evaluation feature includes several checks related to a site's digital certificate that are envisioned as being implemented into the site info agent. It has not been decided if the various checks will be combined to create a single score as was done with the visibility score, or if they will be counted individually to avoid the possibility of having a large number of poorly scoring sites like the visibility feature. An improperly issued certificate, an expired certificate, or a certificate from an unknown or non-credible authority could pose a serious technical risk to the user.

The checks will include:

- Credibility: Is the authority that issued the certificate a known-good Certificate Authority?
- Validity: Is the certificate still valid, or has it expired?
- Domain Mismatch: Was the certificate issued to the correct domain?

6.5.2.2 Content Analysis

a) Content Relevance (Amber)

The content relevance feature is designed to determine how well a publisher site fits in with an advertising campaign based upon the content being relevant to the campaign. A feature based upon relevance is trivial for a human reviewer to evaluate, as has been done in several previous evaluation instruments [87, 27, 203, 204, 205]. However, when attempting to automatically evaluate content relevance in order to comply with feature requirement r3, the problem becomes difficult.

The automation of this feature has not been fully defined as of yet, but current thinking is that the agent would use an approach similar to that employed by the visibility agent. The agent tasked with measuring this feature would take the list of campaign keywords defined by the affiliate manager and use a technique such as Latent Semantic Analysis (LSA) in order to rank the relevance of the site in a manner similar to Cognitive Walkthrough for the Web (CWW) [165] . Of course, some thought would need to be exercised in preventing abuse as once malicious publishers have learned which algorithm is being used, keyword stuffing could become a problem. This abuse is likely to be detectable using spam detection techniques similar to those used by Ntoulas et al. [206].

Because LSA and the anti-abuse techniques of this feature are beyond the scope of this thesis, and because keyword stuffing is a real problem that already happens, this feature has not been implemented yet.

b) Link Analysis (Amber)

The link analysis feature is an extension to the broken links feature. The extension envisioned is likely to use the techniques and methods from CWW [165] in order to determine how relevant the links on a page are to the content of the page. Irrelevant links will subtract from the feature score. This feature has not yet been fully defined as LSA is beyond the scope of this thesis.

6.5.2.3 Campaign Compatibility

This group of features refers to how compatible the website is with the advertising campaign to which it belongs.

a) Sector Classification (Amber)

It is envisioned that future versions of the framework will have various business sector profiles that have been built, allowing the system to attempt to categorise the site being scanned into the appropriate business sector. Li and Yamada [157] and Ivory, Sinha and Hearst [156] have had promising results when the sites being scanned have been split up into their proper categories.

b) Target Demographic (Red)

This relates to the intended audience for a website. If the site is selling maternity supplies, it is likely the target demographic would include groups such as families. If a site is focused on a specific area of academic research, then the audience is likely to be people educated about that research area. Each of these sites should contain very different advertisements in order to reach the maximum amount of interested parties. The technology to support this type of automated content analysis is beyond the scope of this work, and so this feature has not been implemented.

6.5.2.4 Terms & Conditions

This category refers to the terms and conditions set down by the affiliate network, the advertiser and/or the digital marketing agency. The features in this category generally exist to ensure that these terms and conditions are being followed.

a) Invisible Elements (Amber)

In future versions of the framework, this feature will provide insight into the elements on a page that are not visible to the user. There are several reasons that some elements may not be visible to the user ranging from images that are being used as spacers instead of using CSS, web beacons used to keep track of visitor statistics, up to invisible iFrames that can be used to invisibly load an advertiser's page and earn commission for a malicious publisher.

6.5.2.5 Site Profile

This category relates to what type of website is being scanned and who the target audience is for that site. The category also contains features that are designed to measure how stale

the information contained on the site is, based upon how frequently the site is updated.

a) Insecure Protocol Use (Amber)

When transferring sensitive user information such as real name, address, billing information and other personally identifying information, it is of the utmost importance to use secure protocols. When a web site fails to use secure transfer protocol, this sensitive information is sent in clear-text making it much more susceptible to being intercepted by a malicious third party.

Each page of a site would need to be analysed to search for semantic clues to determine if it was asking for any sensitive information. When such a page is found, the surfer agent should ensure that the page is using HTTPS. If a site contained pages that failed to protect sensitive user information, it would receive a zero no matter how many such pages are discovered, as only one unsecured page is needed to leak damaging information. If no such pages are found, the site is given a score of 100.

b) Site Size (Green)

The size of a site is calculated by the Surfer Agent during the course of the web crawl. Fiddler2 is capable of breaking down the site size by content-type of which text/HTML and text/plain are used to calculate the site size in Bytes. This feature corresponds roughly to the indexable text size feature used by Li and Yamada in their automated approach [157]. In the implementation used for this study, this information was not extracted and so was not examined. In order to make use of this information in the future, pre-processing would need to be done in order to meet the requirement of being in the range of 0-100 inclusive.

c) Load Time (Green)

The loading time for each page is recorded by the web crawler and the average is taken to represent the loading time for a publisher site. The load time feature has been used in several previous instruments (Cao2004) (Muylle2004) [27]. Long wait times when loading a page can cause customer frustration and possibly even a loss of conversions due to customers switching to another site. In 2009, Shopzilla reported that a site redesign dropped the average load times from 7 seconds to 2 seconds. This drop in latency was accompanied by a 7-12% increase in conversions [207].

In the implementation used in this study, the load time data was not extracted and therefore

could not be examined.

d) Readability Analysis (Red)

This Usability Analysis feature refers to the readability of a web site. In order to determine this, the scanner will use the same methods as those employed in Microsoft Word when ranking the readability of a document. These methods include the Flesch-Kincaid readability scale for determining the reading level of the content of a site and a spelling and grammar check.

e) Update Frequency (Amber)

The measure of the freshness of web content is important because a site that is not updated frequently could indicate an abandoned or poorly maintained site [21]. This feature is the only one currently defined for the framework that cannot be effectively measured on the first crawl of a publisher site. It is envisioned that the surfer agent will create a hash from the code used on each site and once a site has been crawled once, the surfer agent will compare the hashes to see if the content has changed on subsequent crawls. A change in content will be counted as an update.

6.5.3 Beyond the Publisher's Site

From the evaluation that was part of the original motivation for this research, it was noted that several inconsistencies were present in the customer information of the affected affiliate network [2]. An additional feature designed to seek out these anomalies may enable the automated flagging of accounts for a closer inspection by an employee. The system might be able to draw upon research related to typical anomaly detection system such as an anti-virus or an anomaly-based Intrusion Detection System (IDS). The addition of a feature such as this could be the beginning of a security dimension. Keeping track of a publisher's risk score in conjunction with the already implemented web site tests could give an indication of not only potential performance, but also whether or not the affiliate is genuine, malicious or undetermined. In the case of malicious or undetermined, the case could be moved to the fraud team of the affiliate network for further investigation.

References

[1] Forrester Research, “Affiliate Marketing - The Direct and Indirect Value That Affiliates Deliver to Advertisers,” 2012.

[2] M. Miehling, W. Buchanan, J. Old, A. Batey and A. Rahman, “Analysis of Malicious Affiliate Network Activity as a Test Case for an Investigatory Framework,” in *Proceedings of the 9th European Confeence on Information Warfare and Security*, Thessaloniki, 2010.

[3] O. Hewitson, “Making the Affiliate Long Tail Wag: Part One,” 20 January 2012. [Online]. Available: http://econsultancy.com/us/blog/8712-making-the-affiliate-long-tail-wag-part-one.

[4] J. Nielsen and D. Norman, “Web-Site Usability: Usability on the Web isn't a Luxury,” 2000.

[5] L. J. Najjar, “Designing e-commerce user interfaces,” in *Handbook of Human Factors in Web Design*, Second ed., K. L. Vu and R. W. Proctor, Eds., Mahwah, New Jersey: Lawrence Erlbaum, 2011, pp. 587-598.

[6] S. Y. Chen and R. D. Macredie, “The assessment of usability of electronic shopping: A heuristic evaluation,” *International Journal of Information Management*, vol. 25, no. 6, pp. 516-532, 2005.

[7] L. Hasan, A. Morris and S. Probets, “A comparison of usabiliy evaluation methods for evaluating e-commerce sites,” *Behaviour & Information Technology*, vol. 31, no. 7, pp. 707-737, 2012.

[8] R. Ruiz-Rodriguez, “An Auxiliary Tool for Usability and Design Guidelines Validation of Web Sites,” in *15th International Conference on Computing*, Mexico City, 2006.

[9] T. Moore and B. Edelman, “Measuring the Perpetrators and Funders of

Typosquatting," in *Financial Cryptography and Data Security*, R. Sion, Ed., Tenerife, Springer Berlin Heidelberg, 2010, pp. 175-191.

[10] Internet Advertising Bureau UK, "The Value of UK Online Performance Marketing," Price Waterhouse Cooper, 2013.

[11] D. Vakratsas and T. Ambler, "How advertising works: What do we really know?," *Journal of Marketing*, vol. 63, no. 1, pp. 26-43, 1999.

[12] E. A. Riordan and H. M. Cannon, "Effective reach and frequency: Does it really make sense?," *Journal of Advertising Research*, vol. 34, no. 2, pp. 19-28, 1994.

[13] S. Collins, "AffStat Affiliate Marketing Statistics," 6 March 2010. [Online]. Available: http://affstat.com/2010/03/06/2009-affiliate-summit-affstat-report/.

[14] IAB, "Glossary of interactive advertising terms v. 2.0," 2011. [Online]. Available: http://www.iab.net/media/file/GlossaryofInteractiveAdvertisingTerms.pdf. [Accessed 20 April 2012].

[15] A. Kaushik, Web Analytics 2.0, Sybex, 2009.

[16] B. Dalessandro, R. Hook, C. Perlich and F. Provost, "Evaluating and Optimizing Online Advertising: Forget the Click, But There are Good Proxies," 2012. [Online]. Available: http://ssrn.com/abstract=2167606.

[17] Y. J. Hu, "Performance-based pricing models in online advertising," *Available at SSRN 501082*, 2004.

[18] S. Yuan, A. Z. Abidin, M. Sloan and J. Wang, "Internet Advertising: An Interplay among Advertisers, Online Publishers, Ad Exchanges and Web Users," *Computing Research Repository (CoRR)*, vol. abs/1206.1754, 2012.

[19] K. Holton, "UK Internet ad spend overtakes TV for first time," 29 September 2009. [Online]. Available: http://www.reuters.com/article/idUSTRE58S4IL20090929.

[20] B. Edelman, "CPA Advertising Fraud: Forced Clicks and Invisible Windows," 7 October 2008. [Online]. Available: http://www.benedelman.org/news/100708-1.html.

[21] E. T. Loiacono, D. Chen and D. L. Goodhue, "WebQual TM Revisited: Predicting the Intent to Reuse a Web Site," *AMCIS 2002 Proceedings. Paper,* p. 46, 2002.

[22] R. Daniele, A. J. Frew, K. Varini and A. Magakian, "Affiliate Marketing in Travel and Tourism," in *Information and Communication Technologies in Tourism 2009*, W. Höpken, U. Gretzel and R. Law, Eds., Springer Vienna, 2009, pp. 343-354.

[23] R. T. Rust, K. N. Lemon and V. A. Zeithaml, "Return on marketing: using customer equity to focus marketing strategy," *Journal of Marketing,* pp. 109-127, 2004.

[24] Atlas Institute, "Evaluating the Impact of the New Cost Per Revenue Metric," 2011.

[25] L. Hasan, A. Morris and S. Probets, "Using Google Analytics to Evaluate the Usability of E-Commerce Sites," in *Proceedings of the 1st International Conference on Human Centered Design: Held as Part of HCI International 2009*, San Diego, 2009.

[26] H.-P. Shih, "An empirical study on predicting user acceptance of e-shopping on the Web," *Information and Management,* vol. 41, pp. 351-368, 2004.

[27] A. Phippen, L. Sheppard and S. Furnell, "A practical evaluation of Web analytics," *Internet Research,* vol. 14, no. 4, pp. 284-293, 2004.

[28] A. Kaushik, Web Analytics: An Hour A Day, John Wiley & Sons, 2007.

[29] D. Peacock, Statistics, Structures & Satisfied Customers: Using Web Log Data to Improve, 2002.

[30] S. Xue, "Web usage statistics and Web site evaluation: a case study of a government publications library Web site.," *Online Information Review,* vol. 28, no. 3, pp. 180-190, 2004.

[31] E. T. Peterson, Web Analytics Demystified: A Marketers Guide to Understanding How Your Web Site Affects Your Business, Celilo Group Media, 2004.

[32] B. Plaza, "Monitoring web traffic souces effectiveness with Google Analytics: An experiment with time series," *Aslib Proceedings,* vol. 61, no. 5, pp. 474-482, 2009.

[33] B. Plaza, “Google Analytics for measuring website performance,” *Tourism Management,* vol. 32, no. 3, pp. 477-481, 2011.

[34] Alexa, “How are Alexa’s traffic rankings determined?,” [Online]. Available: https://alexa.zendesk.com/hc/en-us/articles/200449744-How-are-Alexa-s-traffic-rankings-determined-. [Accessed February 2014].

[35] I.-A. Zara, B. C. Velicu, M.-C. Munthiu and M. Tuta, “Using analytics for understanding the consumer online,” *Annals. Economics Science Series.,* no. XVIII, pp. 788-793, May 2012.

[36] J. W. Thomas, “Market Segmentation,” 2007. [Online]. Available: http://www.decisionanalyst.com/publ_art/MarketSegmentation.dai.

[37] A. Ortiz-Cordova and B. J. Jansen, “Classifying Web search Queries to Identify High Revenue Generating Customers,” *Journ of the American Society for Information Science and Technology,* vol. 63, no. 7, pp. 1426-1441, 2012.

[38] J. Yan, N. Liu, G. Wang, W. Zhang, Y. Jiang and Z. Chen, “How much can behavioral targeting help online advertising?,” in *Proceedings of the 18th international conference on World wide web*, Madrid, 2009.

[39] A. Broder, “A taxonomy of web search,” *SIGIR Forum,* vol. 36, no. 2, pp. 3-10, September 2002.

[40] D. E. Rose and D. Levinson, “Understanding user goals in web search,” in *Proceedings of the 13th international conference on World Wide Web*, New York, 2004.

[41] A. Broder, M. Fontoura, V. Josifovski and L. Riedel, “A semantic approach to contextual advertising,” in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, Amsterdam, 2007.

[42] T. P. Novak, D. L. Hoffman and Y.-F. Yung, “Measuring the Customer Experience in Online Environments: A Structural Modeling Approach,” *Marketing Science,* vol. 19, no. 1, pp. 22-42, 2000.

[43] E. Toufaily and J. Perrien, "Customer loyalty to a commercial website," *Journal of Business Research,* 2012.

[44] N. Daswani, C. Mysen, V. Rao, S. Weis, K. Gharachorloo and S. Ghosemajumder, "Online Advertising Fraud," in *Crimeware: Understanding New Attacks and Defenses,* M. Jakobsson and Z. Ramzan, Eds., Addison-Wesley Professional, 2008.

[45] M. J. Schwartz, "Malware Driven Banner Ad Attacks Rising," 28 January 2011. [Online]. Available: http://www.informationweek.co.uk/security/cybercrime/malware-driven-banner-ad-attacks-rising/229200034?subSection=Cybercrime.

[46] V. Dave, S. Guha and Y. Zhang, "Measuring and fingerprinting click-spam in ad networks," in *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, Helsinki, 2012.

[47] D. Eroshenko and M. Bloch, "How to defend your website against click fraud," Clicklab, 2004.

[48] Click Forensics, Inc, "Click Fraud Rate Drops to 19.1 Percent in Q4 2010," 26 January 2011. [Online]. Available: http://www.adometry.com/news/release.php?id=1.

[49] X. Li, D. D. Zeng and L. Wang, "Click frauds and price determination models," in *Proceeding of 2012 IEEE International Conference on Intelligence and Security Informatics*, 2012.

[50] Tor, 2013. [Online]. Available: https://www.torproject.org.

[51] ClixSense, "Affiliates," 2013. [Online]. Available: http://www.clixsense.com/en/Affiliates. [Accessed February 2013].

[52] neobux, 2013. [Online]. Available: http://www.neobux.com/. [Accessed February 2013].

[53] B. Stone-Gross, R. Stevens, A. Zarras, C. Kruegel and G. Vigna, "Understanding fraudulent activities in online ad exchanges," in *Proceedings of the 2011 ACM*

SIGCOMM conference on Internet measurement, New York, 2011.

[54] M. Gandhi, M. Jakobsson and J. Ratkiewicz, "Badvertisements:Stealthy Click-Fraud with Unwitting Accessories," *Journal of Digital Forensic Practice,* vol. 1, no. 2, pp. 131-142, 2006.

[55] N. Provos, P. Mavrommatis, M. A. Rajab and F. Monrose, "All your iFRAMEs point to Us," in *Proceedings of the 17th conference on Security symposium*, Berkeley, 2008.

[56] Z. Li, K. Zhang, Y. Xie, F. Yu and X. Wang, "Knowing Your Enemy: Understanding and Detecting Malicious Web Advertising," in *Proceedings of the 19th ACM Conference on Computer and Communications Security*, Raleigh, 2012.

[57] J. Narvaez, B. E. Endicott-Popovsky, C. Seifert, C. Aval and D. A. Frincke, "Drive-by-downloads," in *2010 43rd Hawaii International Conference on System Sciences*, 2010.

[58] Krebs, "Biggest Cybercriminal Takedown in History," 11 November 2011. [Online]. Available: http://krebsonsecurity.com/2011/11/malware-click-fraud-kingpins-arrested-in-estonia/. [Accessed January 2012].

[59] S. Sengupta and J. Wortham, "7 Charged in Web Scam Using Ads," 9 November 2011. [Online]. Available: http://www.nytimes.com/2011/11/10/technology/us-indicts-7-in-online-ad-fraud-scheme.html?_r=0.

[60] United States District Court Southern District of New York, "Indictment," November 2011. [Online]. Available: http://www.wired.com/images_blogs/threatlevel/2011/11/Tsastsin-et-al.-Indictment.pdf.

[61] S. A. Alrwais, A. Gerber, C. W. Dunn, Oliver Spatscheck, M. Gupta and E. Osterweil, "Dissecting ghost clicks: ad fraud via misdirected human clicks," in *Proceedings of the 28th Annual Computer Security Applications Conference*, New York, 2012.

[62] P. Chatterjee, D. L. Hoffman and T. P. Novak, "Modeling the Clickstream:

Implications for Web-Based Advertising Efforts," *Marketing Sciences*, vol. 22, no. 4, pp. 520-541, 2003.

[63] N. Kshetri, "The Economics of Click Fraud," *IEEE Security and Privacy*, vol. 8, no. 3, pp. 45-53, May-June 2010.

[64] A. Metwally, D. Agrawal and A. El Abbadi, "Duplicate detection in click streams," in *Proceedings of the 14th International Conference on World Wide Web*, New York, 2005.

[65] L. Zhang and Y. Guan, "Detecting Click Fraud in Pay-Per-Click Streams of Online Advertising Networks," in *Proceedings of the 28th International Conference on Distributed Computing Systems*, Washington DC, 2008.

[66] J. Wei, H. Jiang, K. Zhou, D. Feng and H. Wang, "Detecting Duplicates over Sliding Windows with RAM-Efficient Detached Counting Bloom Filter Arrays," in *2011 6th IEEE International Conference on Networking, Architecture and Storage*, 2011.

[67] A. Tuzhilin, "The Lane's Gifts v Google Report," 2006. [Online]. Available: http://googleblog.blogspot.co.uk/pdf/Tuzhilin_Report.pdf. [Accessed 2010].

[68] B. Edelman, "Deterring Online Advertising Fraud through Optimal Payment in Arrears," in *Financial Cryptography and Data Security*, R. Dingledine and P. Golle, Eds., Springer Berlin Heidelberg, 2009, pp. 17-31.

[69] H. Haddadi, "Fighting online click-fraud using bluff ads," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 2, pp. 21-25, April 2010.

[70] R. A. Costa, R. J. G. B. De Queiroz and E. R. Cavalcanti, "A Proposal to Prevent Click-Fraud Using Clickable CAPTCHAs," in *2012 IEEE Sixth International Conference on Software Security and Reliability Companion*, 2012.

[71] A. M. Turing, "Computing Machinery and Intelligence," *Mind*, vol. 59, no. 236, pp. 433-460, October 1950.

[72] J. Elson, J. R. Douceur, J. Howell and J. Saul, "ASIRRA: a CAPTCHA that exploits interest-aligned manual image categorization," in *Proceedings of the 14th ACM*

conference on Computer and, 2007.

[73] A. Dingli and J. Mifsud, "USEFul: A Framework to Mainstream Web Site Usability," *International Journal of Human Computer Interaction,* vol. 2, no. 1, pp. 10-30, 2011.

[74] W. H. DeLone, "The DeLone and McLean model of information systems success: a ten-year update," *Journal of management information systems,* vol. 19, no. 4, pp. 9-30, 2003.

[75] W. H. Delone and E. R. KcLean, "Information systems success: The quest for the dependent variable," *Information Systems Research,* vol. 3, no. 1, pp. 60-95, 1992.

[76] Google, "Ten things we know to be true," [Online]. Available: http://www.google.com/intl/en/about/company/philosophy/.

[77] Y. Lee and K. A. Kozar, "Investigating the effect of website quality on e-business success: An analytic hierarchy process (AHP) approach," *Decision Support Systems,* vol. 42, no. 3, pp. 1383-1401, 2006.

[78] R. Benbunan-Fich, "Using protocol analysis to evaluate the usability of a commercial web site," *Information & Management,* vol. 39, no. 2, pp. 151-163, 2001.

[79] R. E. Luna, J. I. Panach, J. Grigera, G. Rossi and O. Pastor, "Incorporating usability requirements in a test/model-driven web engineering approach," *Journal of Web Engineering,* vol. 9, no. 2, pp. 132-156, 2010.

[80] J. Offutt, "Quality Attributes of Web Software Applications," *IEEE Software,* vol. 19, no. 2, pp. 25-32, 2002.

[81] L.-C. Law and E. T. Hvannberg, "Complementarity and convergence of heuristic evaluation and usability test: a case study of universal brokerage platform," in *NordiCHI '02 Proceedings of the second Nordic conference on Human-computer interaction*, Copenhagen, 2002.

[82] M. Swaak, M. de Jong and P. de Vries, "Effects of Information Usefulness, Visual Attractiveness, and Usability on Web Visitors' Trust and Behavioral Intentions," in *Professional Communication Conference, 2009. IPCC 2009. IEEE International*,

Waikiki, 2009.

[83] A. Ramasastry, “Should web-only businesses be required to be disabled accessible?,” *Find Law's Legal Commentary,* 2002.

[84] U.S. Dept. of Justice, “Section 508 of the rehabilitation act,” 2001. [Online]. Available: http://www.access-board.gov/sec508/guide/1194.22.htm.

[85] J. M. Slatin and S. Rush, Maximum accessibility: making your web site more usable for everyone, Addison-Wesley Professional, 2003.

[86] M. Cole, R. M. O'Keefe, P. Y. Chau, A. Massey, M. Montoya-Weiss and M. Perry, “From the user interface to the consumer interface: results from a global experiment,” *International Journal of Human-Computer Studies,* vol. 53, no. 4, pp. 611-628, 2000.

[87] R. Agarwal and V. Venkatesh, “Assessing a Firm's Web Presence: A Heuristic Evaluation Procedure for the Measurement of Usability,” *Information Systems Research,* vol. 13, no. 2, pp. 168-186, June 2002.

[88] J. W. Palmer, “Web Site Usability, Design, and Performance Metrics,” *Information Systems Research,* vol. 13, no. 2, pp. 151-167, June 2002.

[89] T. Lavie and N. Tractinsky, “Assessing dimensions of perceived visual aesthetics of web sites,” *International Journal of Human-Computer Studies,* vol. 60, no. 3, pp. 269-298, 2004.

[90] P. Zhang and G. M. von Dran, “Satisfiers and Dissatisfiers: A Two-Factor Model for Website Design and Evaluation,” *Journal of American Society for Information Science,* vol. 51, no. 14, pp. 1253-1268, 2000.

[91] J. D'Ambra and R. E. Rice, “Emerging factors in user evaluation of the World Wide Web,” *Information & Management,* vol. 38, no. 6, pp. 373-384, 2001.

[92] C. Liu and K. P. Arnett, “Exploring the factors associated with Web site success in the context of electronic commerce,” *Information & Management,* vol. 38, pp. 23-33, 2000.

[93] A. Molla and P. S. Licker, “E-commerce systems success: An attempt to extend and

respecify the DeLone an Maclean model of IS success," *Journal of Electronic Commerce Research* , vol. 2, pp. 131-141, 2001.

[94] S. Devaraj, M. Fan and R. Kohli, "Antecedents of B2C Channel Satisfaction and Preference: Validation e-commerc Metrics," *Information Systems Research,* vol. 13, no. 3, pp. 316-333, September 2002.

[95] A. A. Jones, "The Impact of Website Navigational Usability Characteristics On User Frustration and Performance Metrics," Master Dissertation, Ohio University, 2012.

[96] J. Nielsen, "Usability Inspection Methods," in *Conference Companion on Human Factors in Computing Systems*, New York, 1995.

[97] W. D. Gray and M. C. Salzman, "Damaged Merchandise? A Review of Experiments That Compare Usability Evaluation Methods," *Human-Computer Interaction,* vol. 13, no. 3, pp. 203-261, 1998.

[98] J. S. Dumas, "User-based Evaluations," in *The Human Computer Interaction Handbook*, J. K. Jacko and A. Sears, Eds., Manhwah, NJ: Lawrence Erlbaum Associates, Inc, 2003.

[99] J. Brooke, "SUS - A quick and dirty usability scale," in *Usability evaluation in industry*, P. W. Jordan, B. Thomas, B. A. Weerdmeester and I. L. McClelland, Eds., London, Taylor & Francis, 1996, pp. 189-194.

[100] S. Borsci, S. Federici and M. Lauriola, "On the dimensionality of the System Usability Scale: a test of alternative measurement models," *Cognitive Processing,* vol. 10, no. 3, pp. 193-197, 2009.

[101] J. R. Lewis and J. Sauro, "The Factor Structure of the System Usability Scale," in *1st International Conference on Human Centered Design: Held As Part of the HCI International 2009*, San Diego, 2009.

[102] BS ISO/IEC 25010:2011, Systems and software engineering - Systems and software Quality Requirements of Evaluation (SQuaRE) - System and software quality models, British Standards Institution, 2011.

[103] E. T. Loiacono, R. T. Watson and D. L. Goodhue, “WebQual: An Instrument for Consumer Evaluation of Web Sites,” *International Journal of Electronic Commerce,* vol. 11, no. 3, pp. 51-87, April 2007.

[104] S. Rosenbaum, J. A. Rohn and J. Humburg, “A toolkit for strategic usability: results from workshops, panels, and surveys,” in *SIGCHI conference on Human Factors in Computing Systems*, 2000.

[105] J. S. Dumas and J. E. Fox, “Usability Testing: Current Practice and Future Direction,” in *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, CRC Press, 2007, pp. 1130-1143.

[106] L. Cooke, “How do users search web home pages?,” *Technical Communication,* vol. 55, no. 2, pp. 176-194, 1 May 2008.

[107] P. G. Polson, C. Lewis, J. Rieman and C. Wharton, “Cognitive Walkthroughs: a method for theory-based evaluation of user interfaces,” *International Journal of man-machine studies,* vol. 36, no. 5, pp. 741-773, 1992.

[108] M. C. Fox, A. K. Ericsson and R. Best, “Do procedures for verbal reporting of thinking have to be reactive? A meta-analysis and recommendations for best reporting methods,” *Psychological Bulletin,* vol. 137, no. 2, pp. 316-344, March 2011.

[109] M. T. H. Chi, N. De Leeuw, M.-H. Chiu and C. Lavancher, “Eliciting Self-Explanations Improves Understanding,” *Cognitive Science,* pp. 439-477, 1994.

[110] J. M. Chin and J. W. Schooler, “Why do words hurt? Content process, and criterian shift accounts of verbal overshadowing,” *European Journal of Cognitive Psychology,* vol. 20, no. 3, pp. 396-413, 2008.

[111] T. Zhao, S. McDonald and H. M. Edwards, “The impact of two different think-aloud instructions in a usability test: a case of just following orders?,,” *Behaviour & Information Technology,* 2012.

[112] D. A. Becker and L. Yonnotta, “Modeling a library web site redesign process: developing a user-centered web site through usability testing,” *Information*

Technology and Libraries, vol. 32, no. 1, pp. 6-22, 2013.

[113] J. Nielsen, "Guerilla HCI: Using Discount Usability Engineering to Penetrate the Intimidation Barrier," 1 January 1994. [Online]. Available: http://www.nngroup.com/articles/guerrilla-hci/. [Accessed November 2012].

[114] D. E. Rowley and D. G. Rhoades, "The cognitive jogthrough: a fast-paced user interface evaluation procedure," in *SIGCHI Conference on Human Factors in Computing Systems*, 1992.

[115] S. Krug, "Usability testing on 10 cents a day," in *Don't make me think: a common sense approach to web usability*, 2000, pp. 139-153.

[116] M. A. Khanum and M. C. Trivedi, "Exploring Verbalization and Collaboration during Usability Evaluation with Children in Context," *International Journal of Computer Science Issues,* 2013.

[117] S. Balakrishnan, S. S. B. Salim and J. L. Hong, "User Centered Design Approach for Elderly People in Using Website," in *2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT)*, 2012.

[118] D. Wixon, "Evaluating usability methods: why the current literature fails the practitioner," *Interactions,* vol. 10, no. 4, pp. 28-34, July 2003.

[119] R. A. Virzi, "Refining the test phase of usability evaluation: how many subjects is enough?," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 34, no. 4, pp. 457-468, 1992.

[120] J. Nielsen and J. T. Hackos, Usability Engineering, Boston: Academic Press, 1993.

[121] W. Hwang and G. Salvendy, "Number of people required for usability evaluation: the 10±2 rule," *Communications of the ACM,* vol. 53, no. 5, pp. 130-133, May 2010.

[122] M. Schmettow, "Sample size in usability studies," *Communications of the ACM,* vol. 55, no. 4, pp. 64-70, April 2012.

[123] J. R. Lewis, "Evaluation of procedures for adjusting problem-discovery rates estimated from small samples," *International Journal of Human-Computer*

Interaction, 2001.

[124] T. Brinck and E. Hofer, "Automatically evaluating the usability of web sites," in *Conference on Human Factors in Computing Systems CHI'02*, 2002.

[125] M. Kuniavsky, Observing the User Experience: A Practitioner's Guide to User Research, Burlington, MA: Morgan Kaufmann Publishing, 2003.

[126] A. Doubleday, M. Ryan, M. Springett and A. Sutcliffe, "A comparison of usability techniques for evaluating design," in *2nd conference on designing interactive systems*, Amsterdam, 1997.

[127] L. Fu, G. Salvendy and L. Turley, "Effectiveness of user testing and heuristic evaluation as a function of performance classification," *Behaviour & Information Technology*, vol. 21, no. 2, pp. 137-143, 2002.

[128] J. Nielsen and R. Molich, "Heuristic Evaluation of User Interfaces," in *SIGCHI conference on Human factors in computing systems: Empowering people*, 1990.

[129] C. Wharton, J. Bradford, R. Jeffries and M. Franzke, "Applying cognitive walkthroughs to more complex user interfaces: experiences, issues, and recommendations," in *SIGCHI conference oog Human factors in computing systems*, 1992.

[130] L. Triacca, A. Inversini and D. Bolchini, "Evaluating Web Usability With MiLE+," in *Seventh IEEE International Symposium on Web Site Evolution*, Budapest, 2005.

[131] J. Nielsen and R. L. Mack, Eds., Usability Inspection Methods, New York, NY: John Wiley & Sons, 1994.

[132] E. J. Simeral and R. J. Branaghan, "A comparative analysis of heuristic and usability evaluation methods," in *Annual Conference Society for Technical Communication*, 1997.

[133] W. S. Tan, D. Liu and R. Bishu, "Web evaluation: heuristic evaluation vs. user testing," *International Journal of Industrial Ergonomics*, vol. 39, pp. 621-627, 2009.

[134] J. C. Bastien and D. L. Scapin, "A validation of ergonomic criteria for the evaluation

of human-computer interfaces," *International Journal of Human-Computer Interaction,* vol. 4, no. 2, pp. 183-196, 1992.

[135] D. L. Scapin and J. C. Bastien, "Ergonomic Criteria for evaluating the ergonomic quality of interactive systems," *Behaviour & Information Technology,* vol. 16, no. 4-5, pp. 220-231, 1997.

[136] S. Weinschenk and D. T. Barker, Designing effective speech interfaces, New York, NY: John Wiley & Sons, Inc, 2000.

[137] J. Nielsen, "Enhancing the explanatory power of usability heuristics," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: celebrating interdependence*, Boston, Massachusetts, USA, 1994.

[138] M. Allen, L. M. Currie, S. Bakken, V. L. Patel and J. J. Cimino, "Heuristic evaluation of paper-based Web pages: A simplified inspection usability methodology," *Journal of Biomedical Informatics,* vol. 39, no. 4, pp. 412-423, 2006.

[139] C. Bach and D. L. Scapin, "Comparing Inspections and User Testing for the Evaluation of Virtual Environments," *International Journal of Human-Computer Interaction,* vol. 26, no. 8, pp. 786-824, 2010.

[140] E. T. Hvannberg, E. L.-C. Law and M. K. Larusdottir, "Heuristic evaluation: comparing ways of finding and reporting usability problems," *Interacting with computers,* vol. 19, no. 2, pp. 225-240, 2007.

[141] P. G. Polson and C. H. Lewis, "Theory-based design for easily learned interfaces," *Human-Computer Interaction,* vol. 5, no. 2-3, pp. 191-220, 1990.

[142] C. Lewis, P. G. Polson, C. Wharton and J. Rieman, "Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces," in *SIGCHI conference on Human factors in computing systems: Empowering people*, 1990.

[143] T. Mahatody, M. Sagar and C. Kolski, "State of the Art on the Cognitive Walkthrough Method, Its Variants and Evolutions," *International Journal of Human-Computer Interaction,* vol. 26, no. 8, pp. 741-785, 2010.

[144] H. W. Desurvire, J. M. Kondziela and M. E. Atwood, “What is gained and lost when using evaluation methods other than empirical testing,” *People and Computers,* pp. 89-89, 1992.

[145] D. L. Cuomo and C. D. Bowen, “Stages of user activity model as basis for user-centered interface evaluations,” in *Human Factors and Ergonomics Society Annual Meeting*, 1992.

[146] N. E. Jacobsen and B. E. John, “Two Case Studies In Using Cognitive Walkthrough For Interface Evaluation,” Pittsburgh, 2000.

[147] B. E. John and H. Packer, “Learning and using the cognitive walkthrough method: a case study approach,” in *SIGCHI Conference on Human Factors in Computing Systems*, 1995.

[148] J. Huart, C. Kolski and M. Sagar, “Evaluation of multimedia applications using inspection methods: The Cognitive Walkthrough case,” *Interacting with Computers,* vol. 16, pp. 183-215, 2004.

[149] S. Riihiaho, “Experiencies with usability evaluation methods,” *Licentiate thesis. Helsinki University of Technology. Laboratory of Information Processing Science,* 2000.

[150] A. Sears and D. J. Hess, “Cognitive Walkthroughs: Understanding the Effect of Task-Description detail on evaluator performance,” *International Journal of Human-Computer Interaction,* pp. 185-200, 1999.

[151] R. Spencer, “The Streamlined Cognitive Walkthrough method, working around social constraints encountered in a software development company,” in *ACM CHI,* 2000.

[152] S. K. Card, T. P. Moran and A. Newell, The psychology of human-computer interaction, CRC Pressl LLC, 1983.

[153] B. E. John and D. E. Kieras, “The GOMS family of user interface analysis techniques: comparison and contrast,” *ACM Transactions on Computer-Human Interaction,* vol. 3, no. 4, pp. 320-351, December 1996.

[154] M. Y. Ivory, “Web TANGO: Towards Automated Comparison of Information-centric Web Site Designs,” in *ACM CHI 00 Conference on Human Factors in Computing Systems*, 2000.

[155] M. Y. Ivory, R. R. Sinha and M. A. Hearst, “Preliminary Findings on Using Quantitative Measures to Compare Favorably Ranked and Unranked Information-centric Web Pages,” in *6th Conference on Human Factors and the Web*, Austin, 2000.

[156] M. Y. Ivory, R. Sinha and M. A. Hearst, “Empirically Validated Web Page Design Metrics,” in *CHI 2001, ACM Conference on Human Factors in Computing Systems, CHI Letters*, 2001.

[157] P. Li and S. Yamada, “Automated Web Site Evaluation - An Approach Based on Ranking SVM,” in *Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT '09. IEEE/WIC/ACM International Joint Conferences on*, Milan, 2009.

[158] G. Lynch, S. Palmiter and C. Tilt, “The Max Model: A Standard Web Site User Model,” in *5th Conference on Human Factors & the Web*, Gaithersburg, 1999.

[159] P. Pirolli, “A Web Site User Model Should at Least Model Something About Users,” Internet Technical Group, April 2000. [Online]. Available: http://www.internettg.org/newsletter/mar00/critique_max.html. [Accessed January 2013].

[160] C. Tilt, “Response to Pirolli's Critique of MAX model,” 30 April 2000. [Online]. Available: http://www.internettg.org/newsletter/mar00/response_critique_max.html. [Accessed January 2013].

[161] C. Katsanos, N. Tselios and N. Avouris, “InfoScent evaluator: a semi-automated tool to evaluate semantic appropriateness of hyperlinks in a web site,” in *Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments*, Sydney, 2006.

[162] P. Pirolli and S. K. Card, “Information foraging,” *Psychological Review - New York*, vol. 106, pp. 643-675, 1999.

[163] C. Katsanos, N. Tselios and N. Avouris, "Evaluating website navigability: validation of a tool-based approach through two eye-tracking user studies," *New Review of Hypermedia and Multimedia,* vol. 16, no. 1-2, pp. 195-214, 12 March 2010.

[164] E. H. Chi, A. Rosien, G. Supattanasiri, A. Williams, C. Royer, C. Chow, E. Robles, B. Dalal, J. Chen and S. Cousins, "The bloodhound project: automating discovery of web usability issues using the InfoScent simulator," in *SIGCHI Conference on Human Factors in Computing Systems,* Ft Lauderdale, 2003.

[165] M. H. Blackmon, P. G. Polson, M. Kitajima and C. Lewis, "Cognitive walkthrough for the web," in *SIGCHI Conference on Human Factors in Computing Systems,* 2002.

[166] M. Kitajima, M. H. Blackmon and P. G. Polson, "A comprehension-based model of Web navigation and its application to Web usability analysis," in *People and Computers XIV—Usability or Else!,* Springer, 2000, pp. 357-373.

[167] L. Baresi, F. Garzotto and P. Paolini, "From Web Sites to Web Applications: New Issues for Conceptual Modeling," in *Conceptual Modeling for E-business and the Web,* Springer Berlin Heidelberg, 2000, pp. 89-100.

[168] F. Christos, K. Christos, P. Eleftherios, T. Nikolaos and A. Nikolaos, "Remote usability evaluation methods and tools: A survey," in *Proceedings of the 11th Panhellenic Conference in Informatics,* 2007.

[169] F. S. H. Krauss, "Methodology for remote usability activities: A case study," *IBM Systems Journal,* vol. 42, no. 4, pp. 582-593, 2003.

[170] H. R. Hartson, J. C. Castillo, J. Kelso and W. C. Neale, "Remote evaluation: the network as an extension of the usability laboratory," in *Proceedings of the SIGCHI conference on Human factors in computing systems,* 1996.

[171] M. Y. Ivory and M. A. Hearst, "The State of the Art in Automating Usability Evaluation," *ACM Computer Surveys,* vol. 33, no. 4, pp. 470-516, December 2001.

[172] T. Tullis, S. Fleischman, M. McNulty, C. Cianchette and M. Bergel, "An empirical comparison of lab and remote usability testing of web sites," in *Usability Professionals Association Conference,* 2002.

[173] R. West and K. Lehman, "Automated Summative Usability Studies: An Empirical Evaluation," in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, 2006.

[174] T. Clemmensen, Q. Shi, J. Kumar, H. Li, X. Sun and P. Yammiyavar, "Cultural Usability Tests – How Usability Tests Are Not the Same All over the World," in *Usability and Internationalization. HCI and Culture*, Springer Berlin Heidelberg, 2007, pp. 281-290.

[175] M. Y. Ivory, J. Mankoff and A. Le, "Using Automated Tools to Improve Web Site Usage by Users with Diverse Abilities," *Human-Computer Interaction Institute,* p. 117, 2003.

[176] G. Brajnik, "Using Automatic Tools in Accessibility and Usability Assurance Processes," in *User-Centered Interaction Paradigms for Universal Access in the Information Society*, vol. III, C. Stary and C. Stephanidis, Eds., Springer Berlin Heidelberg, 2004, pp. 219-234.

[177] M. Vigo and G. Brajnik, "Automatic web accessibility metrics: Where we are and where we can go," *Interacting with Computers,* vol. 23, no. 2, pp. 137-155, 2011.

[178] D. Martin, H. Wu and A. Alsaid, "Hidden surveillance by Web sites: Web bugs in contemporary use," *Communications of the ACM - Mobile computing opportunities and challenges,* vol. 46, no. 12, pp. 258-264, December 2003.

[179] N. Schmücker, "Web Tracking," 2011.

[180] E. T. Loiacono, R. T. Watson and D. L. Goodhue, "WebQual: A measure of website quality," *Marketing theory and applications,* vol. 13, no. 3, pp. 432-438, 2002.

[181] Y. M. Wang, D. Beck, J. Wang, C. Verbowski and B. Daniels, "Strider Typo-Patrol: Discovery and Analysis of Systematic Typo-Squatting," in *Proc. 2nd Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI)*, 2006.

[182] A. Aggarwal, A. Rajadesingan and P. Kumaraguru, "PhishAri: Automatic Realtime Phishing Detection on Twitter," *CoRR,* 2013.

[183] C. L. Clarke, E. Agichtein, S. Dumais and R. W. White, "The influence of caption features on clickthrough patterns in web search," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, Amsterdam, 2007.

[184] T. Yang and A. Gerasoulis, "Web Search Engines: Practice and Experience," in *Computing Handbook*, Third ed., A. Tucker, T. Gonzalez and J. Diaz-Herrera, Eds., Chapman & Hall/CRC Press, 2014.

[185] Google, "Dynamic URLs vs. Static URLs," 22 September 2008. [Online]. Available: http://googlewebmastercentral.blogspot.com/2008/09/dynamic-urls-vs-static-urls.html.

[186] A. Everard and D. F. Galletta, "How Presentation Flaws Affect perceived site quality, trust, and intention to purchase from an online store," *Journnal of Management Information Systems*, vol. 22, no. 3, pp. 56-95, 2006.

[187] R. Molich and J. Nielsen, "Improving a human-computer dialogue," *Communications of the ACM*, vol. 33, no. 3, pp. 338-348, March 1990.

[188] M. Kamari, S. Ghaemi and D. McCoy, "Folex: An analysis of an herbal and counterfeit luxury goods affiliate program.," in *eCrime Researchers Summit (eCRS), 2013*, 2013.

[189] L. A. Granka, T. Joachims and G. Gay, "Eye-Tracking Analysis of User Behavior in WWW Search," 2004.

[190] Nielsen, "The UK Search Marketing Landscape," 23 August 2011. [Online]. Available: http://econsultancy.com/us/blog/10586-ppc-accounts-for-just-6-of-total-search-clicks-infographic.

[191] K. Jerath, L. Ma and Y.-H. Park, "Consumer Click Behavior at a Search Engine: The Role of Keyword Popularity," *Johnson School Research Paper Series*, 2012.

[192] S. Garera, N. Provos, M. Chew and A. D. Rubin, "A framework for detection and measurement of phishing attacks," in *Proceedings of the 2007 ACM workshop on recurring malcode*, Alexandria, 2007.

[193] M. N. Rees, “Take Pride in America Phase III: Quarterly Progress Report, Period Covering April 25, 2012-July 24, 2012,” Public Lands Institute, Nevada, 2012.

[194] Chitika Insights, “The Value of Google Result Positioning,” 2010.

[195] Chitika Insights, “The Value of Google Result Positioning,” 2013.

[196] J. S. Downs, M. B. Holbrook and L. F. Cranor, “Decision strategies and susceptibility to phishing,” in *Proceedings of the second symposium on Usable privacy and security*, Pittsburgh, 2006.

[197] W3Counter, “Web Browser Market Share,” May 2013. [Online]. Available: http://www.w3counter.com/globalstats.php?year=2013&month=5. [Accessed June 2013].

[198] A. Beirekdar, J. Vanderdonckt and M. Noirhomme-Fraiture, “KWARESMI - Knowledge-based Web Automated Evaluation with REconfigurable guidelineS optiMIzation,” in *Proceedings of the 9th International Workshop on Design, Specification, and Verification of Interactive Systems DSV-IS*, 2002.

[199] R. Kohavi and F. Provost, “Glossary of Terms,” *Machine Learning,* vol. 30, pp. 271-274, 1998.

[200] Y. Huang, Y. L. Murphey and Y. Ge, “Automotive diagnosis typo correction using domain knowledge and machine learning,” in *Computational Intelligence and Data Mining (CIDM), 2013 IEEE Symposium on*, Singapore, 2013.

[201] A. Banerjee, M. S. Rahman and M. Faloutsos, “SUT: Quantifying and mitigating URL typosquatting,” *Computer Networks,* vol. 55, no. 13, pp. 3001-3014, 2011.

[202] P. Turney and P. Pantel, “From frequency to meaning: Vector space models of semantics,” *Journal of artificial intelligence research,* vol. 37, no. 1, pp. 141-188, 2010.

[203] S. Muylle, R. Moenaert and M. Despontin, “The conceptualization and empirical validation of web site user satisfaction,” *Information & Management,* vol. 41, no. 5, pp. 543-560, 2004.

[204] S. J. Barnes and R. Vigden, “Measuring web site quality improvements: a case study of the forum on strategic management knowledge exchange,” *Industrial Management & Data Systems,* vol. 103, no. 5, pp. 297-309, 2003.

[205] M. Cao, Q. Zhang and J. Seydel, “B2C e-commerce web site quality: an empirical examination,” *Industrial Management & Data Systems,* vol. 105, no. 5, pp. 645-661, 2005.

[206] A. Ntoulas, M. Najork, M. Manasse and D. Fetterly, “Detecting spam web pages through content analysis,” in *Proceedings of the 15th Internationl Conference on World Wide Web*, New York, 2006.

[207] P. Dixon, “Shopzilla’s site redo - you get what you measure: Velocity 2009,” *o’reilly conferences,* 22 - 24 June 2009.

[208] F. Montero, P. Gonzales, M. Lozano and J. Vanderdonckt, “Quality models for automated evaluation of web sites usability and accessibility,” in *International COST294 Workshop on User Interface Quality Model*, Rome, 2005.

[210] G. Spacagna and F. Kilander, *Ubiquitous Computing for Big Data Insight: Helpful Tool or Privacy Breaker?,* 2012.

[211] S. K. Bera, S. Dutta, A. Narang and S. Bhattercherjee, “Advanced Bloom Filter Based Algorithms for Efficient Approximate,” *CoRR,* vol. abs/1212.3964, 2012.

[212] F. D. Davis, R. P. Bagozzi and P. R. Warshaw, “User acceptance of computer technology: a comparison of two theoretical models,” *Management Science,* vol. 35, no. 8, pp. 982-1003, August 1989.

[213] Specialist Interest Group in Software Testing, “Standard for Software Component Testing,” 2001.

[214] B. Edelman, “Spyware Still Cheating Merchants and Legitimate Affiliates,” 22 May 2007. [Online]. Available: http://www.benedelman.org/news/052107-1.html.

[215] S. Grazioli and S. L. Jarvenpaa, “Deceived: Under Target Online,” *Coomunications of the ACM - Mobile Computing Opportunities and Challenges,* vol. 46, no. 12, pp.

196-205, Decenmber 2003.

[216] E. Wales, “E-Commerce Counts Cost of Online Card Fraud,” *Computer Fraud & Security*, vol. 2003, no. 1, pp. 9-11, January 2003.

[217] V. V. Akwukwuma and A. O. Egwali, “E-Commerce: Online Atacks and Protective Mechanisms,” *Asian Journal of Information Technology*, vol. 7, no. 7, pp. 394-402, 2008.

[218] M. Lek, B. Anandarajah, N. Cerpa and R. Jamieson, “Data Mining Prototype For Detecting E-Commerce Fraud,” in *Proceedings of the Ninth European Conference on Information Systems*, 2001.

[219] J. T. Quah and M. Sriganesh, “Real-Time Credit Card Fraud Detection Using Computational Intelligence,” *Expert Systems with Applications,* vol. 35, no. 4, pp. 1721-1732, 2008.

[220] M. Wu, R. C. Miller and S. L. Garfinkel, “Do Security Toolbars Actually Prevent Phishing Attacks?,” in *Proceedings of the SIGCHI Conference on Human Factors*, New York, 2006.

[221] M. Thelwall, “A History of Webometrics,” *Bulletin of the American Society for Information Science and Technology*, vol. 38, no. 6, pp. 18-23, 2012.

[222] Financial Services Authority, “Data Security in Financial Services,” 2008.

[223] J. M. Pavia, E. J. Veres-Ferrer and G. Foix-Escura, “Credit card incidents and control systems,” *Journal of Information Management*, vol. 32, no. 6, pp. 501-503, December 2012.

[224] ACFE, The 2007 Fraud Examiners Manual, Austin, Texas: Association of Certified Fraud Examiners, 2007.

[225] J. Pereira, “Skimming devices target debit-card reader; thieves are installing them to steal customers' data, then empty banka ccounts,” *Wall Street Journal*, p. B1, 8 March 2007.

[226] Ned, “HandySwipe portable magnetic card reader,” 2006. [Online]. Available:

http://www.camelspit.org/handyswipe/.

[227] Acidus, "Stripe Snoop," 2005. [Online]. Available: http://stripesnoop.sourceforge.net/.

[228] K. J. Barker, J. D'Amato and P. Sheridon, "Credit card fraud: Awareness and prevention," *Journal of Financial Crime,* vol. 15, no. 4, pp. 398-410, 1 November 2008.

[229] A. K. Sood and R. J. Enbody, "Crimeware-as-a-service -- A survey of commoditized crimeware in the underground market," *International Journal of Critical Infrastructure Protection,* 2013.

[230] E. Kraemer-Mbula, P. Tang and H. Rush, "The cybercrime ecosystem: Online innovation in the shadows?," *Technological Forecasting and Social Change,* vol. 80, no. 3, pp. 541-555, 2013.

[231] D. Adsit, "Error-Proofing Strategies for Managing Call Center Fraud," 19 February 2011. [Online]. Available: http://www.isixsigma.com/operations/call-centers/error-proofing-strategies-managing-call-center-fraud/. [Accessed May 2012].

[232] C. Arthur, "Googe, Facebook and others join forces for anti-phishing scheme," 31 January 2012. [Online]. Available: http://www.guardian.co.uk/technology/2012/jan/31/google-facebook-phishing.

[233] FFA, "Card-Not-Present Fraud," 2011. [Online]. Available: http://www.financialfraudaction.org.uk/Financial-cnp-fraud.asp. [Accessed July 2012].

[234] I. Kirschenbaum and A. Wool, How to build a low-cost, extended-range RFID skimmer, PhD Thesis, 2008.

[235] J. Gilbert and N. Archer, "Consumer identity theft prevention and identity fraud detection behaviours," *Journal of Financial Crime,* vol. 19, no. 1, pp. 20-36, 1 January 2012.

[236] C. E. Drake, J. J. Oliver and E. J. Koontz, "Anatomy of a Phishing Email," 2007.

[237] S. Hilley, “New instant phishing pop-up kits on the rampage,” *Computer Fraud & Security,* vol. 8, pp. 10-11, 2007.

[238] MillerSmiles, “The Beginner's Guide to Phishing,” 4 August 2004. [Online]. Available: http://www.millersmiles.co.uk/identitytheft/Article-Part1-Beginners-Guide-to-Phishing.php. [Accessed October 2009].

[239] FFA, “Fraud The Facts 2012,” 2012. [Online]. Available: http://www.financialfraudaction.org.uk/Publications/.

[240] B. Krishnamurthy, “Method and apparatus for automatic identification of phishing sites from low-level network traffic”. United States Patent US 8,141,150 B1, 2012.

[241] E. Ferguson, J. Weber and R. Hasan, “Cloud Based Content Fetching: Using Cloud Infrastructur to Obfuscate Phishing Scam Analysis,” in *Proceedings of the 8th World Congress on Services*, 2012.

[242] N. Arachchilage, S. Love and M. Scott, “Designing a Mobile Game to Teach Conceptual Knowledge of Avoiding Phishing Attacks,” *International Journal for e-Learning Security*, vol. 2, no. 1/2, March/June 2012.

[243] C.-C. Yang, S.-S. Tseng, T.-J. Lee, J.-F. Weng and K. Chen, “Building an Anti-phishing Game to Enhance Network Security Literacy Learning,” in *Proceedings of 2012 IEEE International Conference on Advanced Learning Technologies,* 2012.

[244] P. A. Barraclough, M. A. Hossain, M. A. Tahir, G. Sexton and N. Aslam, “Intelligent phishing detection and protection scheme for online transactions,” *Expert Systems with Applications,* 2013.

[245] T. Blizzard and N. Livic, “Click-fraud Monetizing Malware: A Survey and Case Study,” in *Proceedings of the 2012 7th International Conference on Malicious and Unwanted Software*, Redmond, 2012.

[246] L. Barnard and J. Wesson, “A Trust Model for E-commerce in South Africa,” in *Proceedings of the 2004 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries,* Stellenbosch, Western Cape, 2004.

[247] S. K. Asare and A. M. Wright, “The Effectiveness of Alternative Risk Assessment and Program Planning Tools in a Fraud Setting,” *Contemporary Accounting Research,* vol. 21, no. 2, pp. 325-352, 2004.

[248] W.-C. Chiou, C.-C. Lin and C. Perng, “vv,” *Information & Management,* vol. 47, no. 5-6, pp. 282-290, 2010.

[249] QUIS, “About the QUIS, version 7.0,” 2003. [Online]. Available: http://www.lap.umd.edu/quis. [Accessed October 2011].

[250] A. Beirekdar, J. Vanderdonckt and M. Noirhomme-fraiture, “A Framework And A Language For Usability Automatic Evaluation Of Web Sites By Static Analysis Of Html Source Code,” in *4 th International Conference on Computer-Aided Design of User Interfaces*, 2002.

[251] A. Fernandez, S. Abrahao and E. Insfran, “Empirical validation of a usability inspection method for model-driven Web development,” *Journal of Systems and Software,* pp. 161-186, 2013.

[252] T. H. Davenport, P. Barth and R. Bean, “How 'Big Data; is Different,” *MIT Sloan Management Review,* 2012.

[253] H. Chen, R. H. Chiang and V. C. Storey, “Business intelligence and analytics: from big data to big impact,” *MIS Quarterly,* vol. 36, no. 4, pp. 1165-1188, 2012.

[254] K. Groves, “The Limitations of Server Log Files for Usability Analysis,” 25 October 2007. [Online]. Available: http://boxesandarrows.com/the-limitations-of-server-log-files-for-usability-analysis/. [Accessed June 2010].

[255] G. Brajnik, “Automatic web usability evaluation: what needs to be done?,” in *6th Human Factors and the Web Conference*, Austin, 2000.

[256] K. L. Norman and E. Panizzi, “Levels of automation and user paricipation in usability testing,” *Interacting with Computers,* vol. 18, no. 2, pp. 246-264, 2006.

[257] T. Tiedtke, C. Martin and N. Gerth, “AWUSA - a tool for automated website usability analysis,” in *PreProceedings of the 9th Int. Workshop DSV-IS*, 2002.

[258] M. Y. Ivory and A. Chevalier, “A study of automated web site evaluation tools,” *University of Washington, Department of Computer Science,* 2002.

[259] D. E. Rowley, “Usability testing in the field: bringing the laboratory to the user,” in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems: Celebrating Interdependence*, 1994.

[260] w3schools, “Browser Statistics,” May 2013. [Online]. Available: http://www.w3schools.com/browsers/browsers_stats.asp. [Accessed June 2013].

[261] J. D. Velasquez, L. E. Dujovne and G. L'Huillier, “Extracting significant Website Key Objects: A Semantic Web mining approach,” *Engineering Applications of Artificial Intelligence*, vol. 24, no. 8, pp. 1532-1541, 2011.

Appendix A The System Usability Scale (SUS)

The SUS is a 10-item instrument designed as a quick and dirty usability scale. The 10 items are:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

The items are rated on a five point Likert scale anchored from "Strongly Disagree" (1) and "Strongly Agree" (5). A question with no answer is given a score of three, which is right in the middle of the scale. In order to calculate a score using the SUS, the score for each item must first be determined in the following manner:

- For the odd numbered items (1,3,5,7,9): The score is the item number minus one.
- For the even numbered items (2,4,6,8,10): The score is five minus the item number.

Once the contribution from the individual items has been determined, sum the contributions and multiple that sum by 2.5. This will yield a rough usability score between zero and 100, giving an overview of subjective usability of the system being tested.

Appendix B Cognitive Walkthrough Form

Cognitive Walkthrough For A Step

Task _____ **Action #_____**

1.Goal structure for this step.

1.1. Correct goals. What are the appropriate goals for this point in the interaction? Describe as for initial goals.

1.2. Mismatch with likely goals. What percentage of users will not have these goals, based on the analysis at the end of the previous step? Check each goal in this structure against your analysis at the end of the previous step. Based on that analysis, will all users have the goal at this point, or may some users have dropped it or failed to form it? Also check the analysis at the end of the previous step to see if there are unwanted goals, not appropriate for this step, that will be formed or retained by some users. (% 0 25 50 75 100)

2. Choosing and executing the action.

Correct action at this step: _

2.1. Availability. Is it obvious that the correct action is a possible choice here? If not, what percentage of users might miss it? (% 0 25 50 75 100)

2.2. Label. What label or description is associated with the correct action?

2.3. Link of label to action. If there is a label or description associated with the correct action, is it obvious, and is it clearly linked with this action? If not, what percentage of users might have trouble? (% 0 25 50 75 100)

2.4. Link of label to goal. If there is a label or description associated with the correct action, is it obviously connected with one of the current goals for this step? How? If not, what percentage of users might have trouble? Assume all users have the appropriate goals listed in section 1. (% 0 25 50 75 100)

2.5. No label. If there is no label associated with the correct action, how will users relate

this action to a current goal? What percentage might have trouble doing so? (% 0 25 50 75 100)

2.6. Wrong choices. Are there other actions that might seem appropriate to some current goal? If so, what are they, and what percentage of users might choose one of these? (% 0 25 50 75 100)

2.7. Time-out. If there is a time-out in the interface at this step does it allow time for the user to select the appropriate action? How many users might have trouble? (% 0 25 50 75 100)

2.8. Hard to do. Is there anything physically tricky about executing the action? If so, what percentage of users will have trouble? (% 0 25 50 75 100)

3. Modification of goal structure. Assume the correct action has been taken. What is the system's response?

3.1. Quit or backup. Will users see that they have made progress towards some current goal? What will indicate this to them? What percentage of users will not see progress and try to quit or backup? (% 0 25 50 75 100)

3.2. Accomplished goals. List all current goals that have been accomplished. Is it obvious from the system response that each has been accomplished? If not, indicate for each how many users will not realize it is complete.

3.3. Incomplete goals that look accomplished. Are there any current goals that have not been accomplished, but might appear to have been based on the system response? What might indicate this? List any such goals and the percentage of users will think that they have actually been accomplished.

3.4. "And-then" structures. Is there an "and-then" structure, and does one of its subgoals appear to be complete? If the subgoal is similar to the supergoal, estimate how many users may prematurely terminate the "and-then" structure.

3.5. New goals in response to prompts. Does the system response contain a prompt or cue that suggests any new goal or goals? If so, describe the goals. If the prompt is unclear, indicate the percentage of users who will not form these goals.

3.6. Other new goals. Are there any other new goals that users will form given their

current goals, the state of the interface, and their background knowledge? Why? If so, describe the goals, and indicate how many users will form them. NOTE that these goals mayor may not be appropriate, so forming them may be bad or good.

Table B.1 Cognitive Walkthrough Form

Appendix C Feature Origins

Table C.1 Origin of Framework Features

Appendix D Initial Features

Risk	Name	Category
MM-001	Broken Image	Content
MM-002	Broken Links	Content
MM-003	Certificate Credibility	Security
MM-004	CGI/PERL Integration	Technical
MM-005	Click jacking	Security
MM-006	Contact Details Present	Content
MM-007	Cookie Stuffing	Security
MM-008	Cookies	Security
MM-009	Cross-fertilisation of sites	Technical
MM-010	CSS Style Integration	Design
MM-011	Cyclical/Reciprocal Links	Technical
MM-012	DNS Registration Info	Technical
MM-013	Duplicate Data on Site	Content
MM-014	External Copy and Paste	Content
MM-015	Externally Hosted Content	Content
MM-016	Forced Click	Security
MM-017	Hosting Server Software	Technical
MM-018	Incomprehensible Content	Content
MM-019	Inappropriate Links	Content
MM-020	Invisible Frames	Content
MM-021	Invisible Image	Content
MM-022	Irrelevant Content	Content
MM-023	Jump/Doorway Page	Content
MM-024	Like jacking	Security
MM-025	Link Depth	Content
MM-026	Link Farm	Content
MM-027	Link Manipulation	Technical
MM-028	Link Obfuscation	Technical
MM-029	Load Time	Content
MM-030	Loaded Images	Technical
MM-031	Multiple Affiliate Links	Content
MM-032	Old Server Software	Technical
MM-033	Old Technology	Technical
MM-034	Original Layout	Design
MM-035	Payment page is not SSL	Security
MM-036	Remote Region Hosting	Technical

MM-037	Template Used	Design
MM-038	Text defined with graphics	Content
MM-039	Valid Certificate	Security
MM-040	Webpage Uptime	Technical
MM-041	Word Template Used	Design

Appendix E Full Results

Broken Links

24.45	95	
Perf \ Link	Poor	Good
Poor	20	38
Good	42	134

TP (Sensitivity)	0.76	
FP	0.66	
TN (Specificity)	0.34	
FN	0.24	
Precision (Good/Poor)	0.78	0.32
Accuracy	0.66	

24.45	96	
Perf \ Link	Poor	Good
Poor	25	33
Good	46	130

TP (Sensitivity)	0.74	
FP	0.57	
TN (Specificity)	0.43	
FN	0.26	
Precision (Good/Poor)	0.8	0.35
Accuracy	0.66	

24.45	97	
Perf \ Link	Poor	Good
Poor	26	32
Good	50	126

TP (Sensitivity)	0.72	
FP	0.55	
TN (Specificity)	0.45	
FN	0.28	
Precision (Good/Poor)	0.8	0.34
Accuracy	0.65	

48.04	95	
Perf \ Link	Poor	Good
Poor	34	81
Good	28	91

TP (Sensitivity)	0.76	
FP	0.7	
TN (Specificity)	0.3	
FN	0.24	
Precision (Good/Poor)	0.53	0.55
Accuracy	0.53	

48.04	**96**	
Perf \ Link	**Poor**	**Good**
Poor	**41**	**74**
Good	**30**	**89**

TP (Sensitivity)	**0.75**	
FP	**0.64**	
TN (Specificity)	**0.36**	
FN	**0.25**	
Precision (Good/Poor)	**0.55**	**0.58**
Accuracy	**0.56**	

48.04	97	
Perf \ Link	Poor	Good
Poor	43	72
Good	33	86

TP (Sensitivity)	0.72	
FP	0.63	
TN (Specificity)	0.37	
FN	0.28	
Precision (Good/Poor)	0.54	0.57
Accuracy	0.55	

73.39	95

Perf \ Link	Poor	Good
Poor	45	131
Good	17	41

TP (Sensitivity)	0.71	
FP	0.74	
TN (Specificity)	0.26	
FN	0.29	
Precision (Good/Poor)	0.24	0.73
Accuracy	0.37	

73.39	96

Perf \ Link	Poor	Good
Poor	53	123
Good	18	40

TP (Sensitivity)	0.69	
FP	0.7	
TN (Specificity)	0.3	
FN	0.31	
Precision (Good/Poor)	0.25	0.75
Accuracy	0.4	

73.39	97

Perf \ Link	Poor	Good
Poor	57	119
Good	19	39

TP (Sensitivity)	0.67	
FP	0.68	
TN (Specificity)	0.32	
FN	0.33	
Precision (Good/Poor)	0.25	0.75
Accuracy	0.41	

Broken Images

24.45	95

Performance \ Image	Poor	Good
Poor	1	57
Good	3	173

TP (Sensitivity)	0.98	
FP	0.98	
TN (Specificity)	0.02	
FN	0.02	
Precision (Good/Poor)	0.75	0.25
Accuracy	0.74	

24.45	98.6

Performance \ Image	Poor	Good
Poor	2	56
Good	3	173

TP (Sensitivity)	0.98	
FP	0.97	
TN (Specificity)	0.03	
FN	0.02	
Precision (Good/Poor)	0.76	0.4
Accuracy	0.75	

24.45	99

Performance \ Image	Poor	Good
Poor	7	51
Good	17	159

TP (Sensitivity)	0.9	
FP	0.88	
TN (Specificity)	0.12	
FN	0.1	
Precision (Good/Poor)	0.76	0.29
Accuracy	0.71	

48.04	95	
Performance \ Image	Poor	Good
Poor	2	113
Good	2	117

TP (Sensitivity)	0.98	
FP	0.98	
TN (Specificity)	0.02	
FN	0.02	
Precision (Good/Poor)	0.51	0.5
Accuracy	0.51	

48.04	98.6	
Performance \ Image	Poor	Good
Poor	3	112
Good	2	117

TP (Sensitivity)	0.98	
FP	0.97	
TN (Specificity)	0.03	
FN	0.02	
Precision (Good/Poor)	0.51	0.6
Accuracy	0.51	

48.04	**99**	
Performance \ Image	**Poor**	**Good**
Poor	**17**	**98**
Good	**7**	**112**

TP (Sensitivity)	**0.94**	
FP	**0.85**	
TN (Specificity)	**0.15**	
FN	**0.06**	
Precision (Good/Poor)	**0.53**	**0.71**
Accuracy	**0.55**	

73.39	95	
Performance \ Image	Poor	Good
Poor	3	173
Good	1	57

TP (Sensitivity)	0.98	
FP	0.98	
TN (Specificity)	0.02	
FN	0.02	
Precision (Good/Poor)	0.25	0.75
Accuracy	0.26	

73.39	98.6	
Performance \ Image	Poor	Good
Poor	4	172
Good	1	57

TP (Sensitivity)	0.98	
FP	0.98	
TN (Specificity)	0.02	
FN	0.02	
Precision (Good/Poor)	0.25	0.8
Accuracy	0.26	

73.39	99	
Performance \ Image	Poor	Good
Poor	20	156
Good	4	54

TP (Sensitivity)	0.93	
FP	0.89	
TN (Specificity)	0.11	
FN	0.07	
Precision (Good/Poor)	0.26	0.83
Accuracy	0.32	

Visibility

24.45	0.1	
Performance \ Visibility	Poor	Good
Poor	55	3
Good	169	7

TP (Sensitivity)	0.04
FP	0.05
TN (Specificity)	0.95
FN	0.96
Precision (Good/Poor)	0.7 0.25
Accuracy	0.26

48.04	**0.1**	
Performance \ Visibility	**Poor**	**Good**
Poor	**112**	**3**
Good	**112**	**7**

TP (Sensitivity)	**0.06**
FP	**0.03**
TN (Specificity)	**0.97**
FN	**0.94**
Precision (Good/Poor)	**0.7 0.5**
Accuracy	**0.51**

73.39	0.1	
Performance \ Visibility	Poor	Good
Poor	169	7
Good	55	3

TP (Sensitivity)	0.05
FP	0.04
TN (Specificity)	0.96
FN	0.95
Precision (Good/Poor)	0.3 0.75
Accuracy	0.74

URL Relevance

24.45	10	
Perf \ Relevance	Poor	Good
Poor	38	20
Good	92	84

TP (Sensitivity)	0.48
FP	0.34
TN (Specificity)	0.66
FN	0.52
Precision (Good/Poor)	0.81 0.29
Accuracy	0.52

24.45	13.5	
Perf \ Relevance	Poor	Good
Poor	38	20
Good	92	84

TP (Sensitivity)	0.48
FP	0.34
TN (Specificity)	0.66
FN	0.52
Precision (Good/Poor)	0.81 0.29
Accuracy	0.52

24.45	23.1	
Perf \ Relevance	Poor	Good
Poor	51	7
Good	125	51

TP (Sensitivity)	0.29
FP	0.12
TN (Specificity)	0.88
FN	0.71
Precision (Good/Poor)	0.88 0.29
Accuracy	0.44

48.04	10	
Perf \ Relevance	Poor	Good
Poor	72	43
Good	58	61

TP (Sensitivity)	0.51	
FP	0.37	
TN (Specificity)	0.63	
FN	0.49	
Precision (Good/Poor)	0.59	0.55
Accuracy	0.57	

48.04	13.5	
Perf \ Relevance	Poor	Good
Poor	72	43
Good	58	61

TP (Sensitivity)	0.51	
FP	0.37	
TN (Specificity)	0.63	
FN	0.49	
Precision (Good/Poor)	0.59	0.55
Accuracy	0.57	

48.04	**23.1**	
Perf \ Relevance	**Poor**	**Good**
Poor	**100**	**15**
Good	**76**	**43**

TP (Sensitivity)	**0.36**	
FP	**0.13**	
TN (Specificity)	**0.87**	
FN	**0.64**	
Precision (Good/Poor)	**0.74**	**0.57**
Accuracy	**0.61**	

73.39	10	
Perf \ Relevance	Poor	Good
Poor	100	76
Good	30	28

TP (Sensitivity)	0.48	
FP	0.43	
TN (Specificity)	0.57	
FN	0.52	
Precision (Good/Poor)	0.27	0.77
Accuracy	0.55	

73.39	13.5	
Perf \ Relevance	Poor	Good
Poor	100	76
Good	30	28

TP (Sensitivity)	0.48	
FP	0.43	
TN (Specificity)	0.57	
FN	0.52	
Precision (Good/Poor)	0.27	0.77
Accuracy	0.55	

73.39	23.1	
Perf \ Relevance	Poor	Good
Poor	136	40
Good	40	18

TP (Sensitivity)	0.31	
FP	0.23	
TN (Specificity)	0.77	
FN	0.69	
Precision (Good/Poor)	0.31	0.77
Accuracy	0.66	

HealthScore

24.45	54.7	
Perf \ HealthScore	Poor	Good
Poor	38	20
Good	82	94

TP (Sensitivity)	0.53	
FP	0.34	
TN (Specificity)	0.66	
FN	0.47	
Precision (Good/Poor)	0.82	0.32
Accuracy	0.56	

24.45	54.8	
Perf \ HealthScore	Poor	Good
Poor	39	19
Good	94	82

TP (Sensitivity)	0.47	
FP	0.33	
TN (Specificity)	0.67	
FN	0.53	
Precision (Good/Poor)	0.81	0.29
Accuracy	0.52	

24.45	57.4	
Perf \ HealthScore	Poor	Good
Poor	54	4
Good	123	53

TP (Sensitivity)	0.3	
FP	0.07	
TN (Specificity)	0.93	
FN	0.7	
Precision (Good/Poor)	0.93	0.30
Accuracy	0.46	

48.04	**54.7**	
Perf \ HealthScore	**Poor**	**Good**
Poor	**69**	**46**
Good	**51**	**68**

TP (Sensitivity)	**0.57**	
FP	**0.4**	
TN (Specificity)	**0.6**	
FN	**0.43**	
Precision (Good/Poor)	**0.6**	**0.58**
Accuracy	**0.59**	

48.04	54.8	
Perf \ HealthScore	Poor	Good
Poor	74	41
Good	59	60

TP (Sensitivity)	0.5	
FP	0.36	
TN (Specificity)	0.64	
FN	0.5	
Precision (Good/Poor)	0.59	0.56
Accuracy	0.57	

48.04	57.4	
Perf \ HealthScore	Poor	Good
Poor	102	13
Good	75	44

TP (Sensitivity)	0.37	
FP	0.11	
TN (Specificity)	0.89	
FN	0.63	
Precision (Good/Poor)	0.77	0.58
Accuracy	0.62	

73.39	54.7	
Perf \ HealthScore	Poor	Good
Poor	91	85
Good	29	29

TP (Sensitivity)	0.5	
FP	0.48	
TN (Specificity)	0.52	
FN	0.5	
Precision (Good/Poor)	0.25	0.76
Accuracy	0.51	

73.39	54.8	
Perf \ HealthScore	Poor	Good
Poor	102	74
Good	31	27

TP (Sensitivity)	0.47	
FP	0.42	
TN (Specificity)	0.58	
FN	0.53	
Precision (Good/Poor)	0.27	0.77
Accuracy	0.55	

73.39	57.4	
Perf \ HealthScore	Poor	Good
Poor	137	39
Good	40	18

TP (Sensitivity)	0.31	
FP	0.22	
TN (Specificity)	0.78	
FN	0.69	
Precision (Good/Poor)	0.32	0.77
Accuracy	0.66	

Appendix F Unimplemented Features