ISSN: 0974-8571 Vol.13 No. 2 December, 2021

International Journal of Computational Intelligence in Control

The Web Spam Taxonomy and Algorithms for Detection and Prevention of Web Spamming - A Systematic Review

Asim Shahzad[#], Jamaluddin Mir[*], Aamer Khan[#], Muhammad Asshad[#], Muhammad Zeeshan[#], Ahsan Zubair[#], Muhammad Naeem[#]

[#] *Department of Computer Science, Abbottabad University of Science & Technology, Hevellian, Pakistan*
E-mail: asim.shaz@gmail.com
[*] *Faculty of Computer Science & Information Technology, Universiti Tun Hossein Onn Malaysia, Batu Pahat, Malaysia*

Received: 09th April 2021
Revised: 03rd June 2021
Accepted: 29th June 2021
Published: 15th July 2021

Abstract: In today's world, internet is a globally used information system through which everyone glances for online information using search engines (SEs). They usually see the results on a search engine's first page before replacing it with another search query. Search engines attempt to return outstanding results in the search engine result pages (SERP) in response to a user's search query. However, results on the first page of SERPs are not always up to par. Moreover, due to web spamming's continuous evolution, it has now become more challenging for search engines to provide excellent quality search results. Furthermore, many providers of information are sought by the user, therefore, presenting every piece of data on the first page of SERPs is impossible. Therefore, most people use search engine optimisation (SEO) techniques to bring their websites to the top of SERPs' first page. However, some people misuse search engines using illegal methods to get their website to the top of the SERP's first page. These prohibited techniques are known by various names, such as web spam, search-engine spamming, web poisoning, and spamdexing. In web spamming, spammers redirect the users to a low ranked website, which reduce the performance of a search engine and spammers improve their business profit. In addition, adversarial information retrieval got much attention from both the industry and academic researchers. This article, therefore, presents the taxonomy of current web spamming techniques using which spammers can gain special ranks for their websites. This article also presents a systematic literature review of algorithms used for the detection and prevention of web spamming. We have divided all the current web spam detection and prevention algorithms into four categories based on the information type they use: Link-based web spam detection techniques, content-based web spam detection techniques, web spam techniques based on non-traditional data, and combined approaches. We have also subcategorised the link-based technique into five categories by the principles and ideas used. We have then compared these five methods with each other based on algorithms, complexity, working, and type of information used. Ultimately, we have shortened the perceptions and unravelled the rules used for the construction of the web spam detection algorithm.

.

Keywords—Spamming, Web Spam, Spamdexing, Web Poisoning, Content Spamming, PageRank, TrustRank, HITS, Link Spamming, Cloaking, Search-Engine Spamming.

I. Introduction

A search engine (SE) is a system software used for searching web pages' data on the World Wide Web (WWW) based on the user's search statement (user query). As the information on the web is increasing with every passing day, SE is recognised as a weapon to penetrate the web. An index of websites' contents is built so that information could be extracted from the web. Every search engine returns a list of results in response to the user's query submitted to SE. Every SE has a rank algorithm to rank the results. The majority of users only check the first page of SERPs[1]. Enhancing the standard and quality of the web-page content is the proper and legitimate method to improve the website's rank in SERPs, but it is expensive and time-consuming [1]. Another approach is to use illegal and improper techniques to boost websites' reputation, which can cause massive traffic to the websites. Some SEO specialists also use fraudulent techniques [2] and are distinguished as Black Hat SEOs (search-engine optimisers). Such methods guarantee that a web page is available to a SE and increase the chance that the web page will be at the top position in SERP. The process of cheating SEs is known as web spamming[2]. Several information systems face this issue, and the spam is penetrating every system be it web, email, social media, reviews platform, or blogs. The idea of web spam initially began in 1996 and web spamming has now been recognised as a critical and standard issue [3]. Web spam is a major challenge currently faced by the SE industry [4]. The goal of web spam is to artificially raise the page rank (PR)[3] of a site in query results so that it appears higher than it should be compared to other, non-spamming sites [5]. This not only ruins the quality of search results for users, but also wastes their time. As the number of spam pages grows, so does the number of pages that search engine crawlers and indexers have to sift through [6]. When the number of spam pages increases, the number of spam pages examined by crawlers and indexers also increases. Consequently, the SE will waste its valuable resources as the searching time increases [7]. According to Gyongyi and Garcia, this is an illegal activity performed by an individual to improve the page rank of a web page [8]. The adverse impact of the abundance of spam on the internet has been recognised as a critical challenge faced by SEs [9]. Web spam undermines users' trust in a SE resulting in shifting of the user to another SE. This is a different problem because it costs a user nothing to shift from one SE to another [10]. Spam pages are the source of the distribution of adult content, malware, and phishing attacks. Eiron et al. ranked a hundred million web pages using the Page-Rank algorithm [11]. They discovered that out of the top twenty, eleven results were websites with sexually explicit content that achieved great ranking due to content-based and link-based web spamming [12]. Most importantly, web spam requires a SE provider to expend an inordinate amount of resources in storage and computation. The worldwide financial losses resultant from spam in 2017 were estimated at a staggering $450 billion. In 2009, that number was $130 billion, but in 2016 alone, the United States saw over $1.3 billion in financial losses from spam. New hurdles are appearing continually, namely the speedy growth of the worldwide web and its simplification of content-creation tools (e.g., blogs, free-access pages, and wiki) along with a decrease in the cost of website maintenance and development (e.g., WordPress, Drupal, hosting, and domain registration) [13]. Spam evolution itself is a big challenge, especially the emergence of new web spamming techniques that cannot be apprehended by previous methods. Silverstein et al. analysed a large SE query log and concluded that a user only checks the first-page of results for 85% of their queries [14] and after clicking only a few links on top of SERPs they rephrase their query [15]. Therefore, the owners of websites try to manipulate SE rankings by applying unethical methods like undeserved link formulations (e.g., link wheel,

[1] https://www.wordstream.com/serp

[2] http://www.webspam.org/seo-spam-what-is-spamdexing/

[3] https://en.wikipedia.org/wiki/PageRank

link farms, link pyramids, and two-way link exchange.), doorway pages, keywords stuffing, cloaking, URL redirections, invisible text, and meta-tag stuffing. Castillo et al. found that 6% of web pages were categorised as spam in the English language [16]. Becchetti et al. revealed that the host-level spamming is 22% [17], while Benczur et al. estimated that at 16.5% [18]. A few researchers worked on the countries and top-level spam distribution [19]. According to their report, the spam rate is 13.8% in English pages, 22% in the German pages, 25% in the French pages, and 9% in the Japanese pages. According to their studies, 20% of web pages in *.com and 70% of web pages in *.biz domains are spam.

This study has three goals. First, this study aims at explaining the web spam taxonomy and identifying all possible methods used for web spamming. Second, it explores all possible web spam detection and prevention algorithms and reveals the underlying construction rules for web spam detection and prevention algorithms. Finally, it builds the perception and provides the roadmap for further research in the area. We organised this article as follows. In Section 2, we provided a summary of search engine optimisation and its types. In Section 3, we provided a summary of the web spamming taxonomy and its types. We further categorised the on-page and off-page web spamming. In Section 4, we discussed the need for web spam detection. In Section 5, we provided a detailed overview of web spam detection algorithms. In Section 5.A, we discussed detection algorithms for content-based spamming. In Section 5.B, we provided thorough coverage of detection algorithms for link-based spamming. In Section 5.C combined approaches for spam detection is discussed. In Section 5.D, we discussed the spam detection algorithms based on untraditional features. Existing techniques are compared in Section 6. In Section 7, we discussed the challenges in spamdexing techniques. Finally, we compiled fundamental rules used for the construction of web spam detection and prevention algorithms in Section 8 followed by a conclusion in Section 9.

II. SEARCH ENGINE OPTIMIZATION

SEO is the process of optimizing a web page to increase its visibility in search engine results. SEO specialists improve web page relevance by optimizing the page content and making sure it can be indexed accurately. [20]. There are two types of SEO techniques: on-page SEO and off-page SEO. Figure 1 below shows the different types and techniques of SEO.

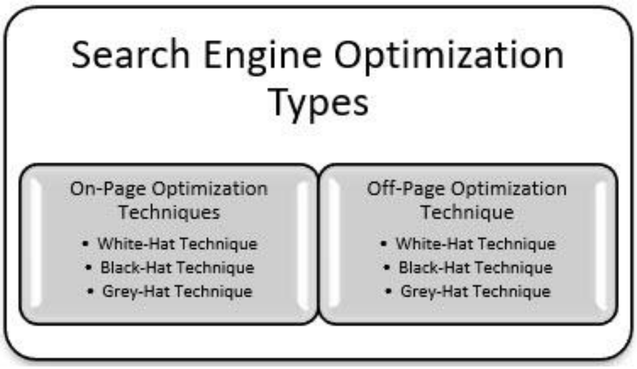

Figure 1:Types of Search Engine Optimisation

On-page Optimisation: It is also known as on-site optimisation. In this technique, web pages are optimised to get a higher rank on SERPs. On-page optimisation deals with the content of a web page that affects the SE rankings [21]. It incorporates best content, excellent keyword selection, proper insertion of keywords on relevant tags, and measurement of the keyword's frequency.

Off-page Optimisation: This method is also utilized for enhancing the rank of a website in SERPs. There are many factors that contribute to off-page optimization, such as social media marketing and social bookmarking. However, the most important focus is on developing incoming links to a web page [22]. This can be done by submitting the web page's URL to search engines, social media outlets, listing directories, and other repositories. [21].

White-Hat SEO: It is a proper SEO method, which fulfils SE guidelines[4] and is being used for a long time. It includes research on keywords, keyword analysis, and rewriting of Meta contents and Meta tags. The objective of white hat SEO is to improve the accessibility of both SE and users towards a web page [22].

Grey-Hat SEO: Although the use of grey-hat techniques to increase the page rank of a web page is technically legal, it is ethically questionable and could one day become a black-hat technique [23]. The examples of grey-hat tactics are three-way linking, building irrelevant links, duplicate content to some extent, and abnormally high keyword density.

Black-Hat SEO: It is not an appropriate method to gain SE rankings through illicit means by violating the SE guidelines. This technique is known as web poisoning or spamdexing [24]. It is volatile and aggressive and therefore discouraged by search engines. Black-hat techniques are designed to exploit search engines rather than provide a good experience for users. It includes invisible text, link wheels, link pyramids, cloaking, doorway pages, and mirror websites [21]. Spammers use Black Hat SEO techniques for raising their websites to the top in SERPs. Section 3 discussed how spammers use black hat web spamming techniques and the complete taxonomy of web spamming.

III. Web Spamming Taxonomy

Spamdexing is a portmanteau of 'spam' and 'indexing'. The purposeful distortion of search engine rankings in order to elicit higher web traffic on undeserving pages is known as spamdexing, or web spamming It manipulates the SE ranking to get more web traffic on a particular web page. This results in a higher ranking in all significant SEs. [25]. The first commercial SE Lycos was developed in 1995. Spamdexing has concurrently emerged with commercial SE. Then, SEs tried to deal with this challenge [21], [25]. Davison presented an article on web spam detection using the machine-learning method; academia took this subject for further discussion and a discussion started on web spam in universities [16]. AIRWeb workshops are recognised as a platform where researchers can exchange ideas on web spam since 2005 [26]. Web spam is the outcome of using unscrupulous techniques to manipulate search results [8], [27], [28]. Perkins described web spam as the effort to cheat page-ranking algorithms associated with SEs [28]. The researchers have discovered and recognised different spamming techniques. Web spamming activities can broadly be categorized as either being off-page or on-page. Off-page spamming techniques encompass link-based spamming, social-based spamming and click Spamming. On-page spamming activities on the other hand, include the creation of Mirror websites, HTML-based spamming, Architecture-based spamming, and Trust-based spamming. [29].

A. *On-page Spamdexing Techniques*

The on-page spamming technique is further categorised into three categories: content-based, HTML-based, and architecture-based web spamming.

[4] https://support.google.com/webmasters/answer/7451184?hl=en

1) Content-based spamming: Content-based spamming refers to manipulation of the content of a web page. Its distribution consists of hyperlinks, the number of words in the text of a page, and content repetition [30]. The content-based spamming is the first and the most comprehensive form of web spamming. Content-based spamming is a favourite among spammers. This is because information-retrieval models used by SEs, e.g., BM25 [31], Vector Space Model (VSM) [32], or Statistical Language Model (SLM) [33], rank a web page based on its content and spammers exploit this weakness. For instance, in examining the Term Frequency-Inverse Document Frequency[5] (TF-IDF) scoring, where TF-IDF is used to calculate a weight, which signifies the significance of a word in a document. It compares the frequency of a word used inside an individual document to the whole data set (collection of all documents). The significance of a term increases proportionally with the number of times a term appears in an individual document; this is called Term Frequency (TF). Nevertheless, if a term appears many times in multiple documents, it is a problem [34]. Therefore, TF-IDF also balances this value through the term's frequency in an entire document set; this value is called Inverse Document Frequency (IDF).

$$TF - IDF\,(q,p) = \sum_{t\in q \wedge t\in p} TF(t).IDF(t) \qquad (1)$$

where q represents the query, p represents the page, and t is the term. A spammer can increase the TF of terms appearing on a web page. For every term in each document, TF-IDF is computed [34]. Typically, a spammer can focus on one term individually or he could be interested in the highest TF-IDF terms in a particular document (e.g., creating tags for blog posts) [35]. There are various aspects based on which we can classify content-spam in the following categories.

Keyword Stuffing: Careful insertion of keywords into a web page for increasing the keyword's count, variation and density in a web page is called Keyword Stuffing [36]. Keyword Stuffing is beneficial since it makes a web page seem suitable and noticeable for a web-page crawler. For example, a fraudulent-scheme promoter desires to drag internet surfers to a web page to promote his scam [16]. He puts invisible text suitable for the follower page of a popular sports group on his web page, assuming that the web page will be indexed as a follower page and will get several visits from sports enthusiasts. Earlier techniques of indexing applications calculated how frequently a keyword emerged and used it for relevance level [37]. However, modern SEs can efficiently analyse a web page for Keywords Stuffing and discover whether the frequency of keywords is consistent with other websites designed mainly to drag SE traffic. Moreover, large web pages are pruned, so on a single web page, extensive dictionary-list indexing is not possible [38].

Doorway Pages: These are also called Gateway pages. These web pages are cheap-quality pages with minimal content but are alternatively stuffed with pretty similar phrases and keywords [37]. These pages are invented to get a high rank in SE results but are useless for the visitors searching for information. Generally, a Gateway page will have "Click here to Enter" on the web page. Google removed BMW in 2006 for using Gateway pages to bmw.de[6] (German website of the Company).

Body Spamming: The body of a web page is modified in this case. Body spamming is the conventional method of content-based spamming because body spamming is cheap and concurrently supports the implementation of several strategies [38]. For example, if a spammer desires to get a high page rank for only a few predefined queries, he can use a repetition tactic by overstuffing the page's body with the

[5] http://www.tfidf.com

[6] http://edition.cnn.com/2006/BUSINESS/02/07/google/

terms that appear in the queries [39]. Furthermore, if the spammer's target is to cover a broad set of queries, he will use many arbitrary terms at once. To hide such contents, that are used for boosting the rank of a website, spammers give it the background colour, so, in this way, only crawlers can read and recognise it.

Scraper Sites: These sites are developed using different applications and are designed to 'scrape' the content from several websites including SERPs to generate 'content' for a spam web page. The content's presentations on these websites are unique but are a combination of contents obtained from other places, usually without approval. Such web pages are usually full of ads (pay per click advertisement[7]) or these pages redirect users to other websites [40]. It is effortless for scraper web pages to outrank the official web pages for their data and organisation names.

Article Spinning: Instead of scraping the content from other web pages, spammers, in article spinning, rewrite the already existing articles to bypass the penalties applied by SEs for copied content. Spammers do the article spinning by contracted writers or using automated applications and tools using the neural network or dictionary database [41]. Spammers also use unrelated words to rank their website higher in SERPs. The page created by the spammer utilising this technique may appear in many query words [37]. Like article spinning, some web pages use machine translation to translate the web pages' content in many languages, without human involvement; this usually results in a meaningless text [41].

Title Spamming: Some SEs give more attention and importance to the title of a document. Spammers may use foreign terms in this tag [38]. Therefore, if SE gives higher weights to these terms of the tag, the web page will also get a higher rank. As the title tag is significant for information retrieval [42], spammers got leverage to overstuff the title tag to obtain an overall high ranking.

URL Spamming: Some SEs also recognise a tokenised URL of a web page as a zone [43]. Therefore, spammers generate a URL for the web page from the keywords that appeared in targeted queries. For instance, a spammer wishes to be ranked higher for the search term “cheap iPhone,” the spammer might generate a URL "cheap-phones.com/cheap-phones/iPhone.html."

Anchor Text Spamming: Advantages of anchor text regarding the ranking were discovered in 1994 [44] and immediately the spammers build their strategy for spamming through anchor text [10]. With the desired anchor text, spammers create the links (usually irrelevant anchor text to the linking-web-page content) to get the 'right' terms for the target web page [38]. After the link-based ranking algorithm [11], [44] was introduced in the market, the content-based spam issue was partially defeated. However, spamming is continuously evolving and soon after, spammers began constructing the link wheels, link pyramid, and link farms [9], [38], [45].

2) HTML-Based Web Spamming: This technique of spamming consists of Meta-tag stuffing, text hiding, and URL redirection. HTML-Based spamming is categorised as follows.

Meta-Tag Stuffing: Web page designers use the HTML Meta tag for a short description of the web page [22]. Usually, spammers place unrelated terms in this tag; with these additions, SE algorithms consider these web pages as essential. The web page will then get a higher rank due to these irrelevant terms. Meta tag plays a vital role in the document representation. Therefore, the addition of spam terms in the meta tag might be essential from a spammer's point of view [46], [47]. Due to massive spamming, SEs are giving less importance to this tag; some SEs even ignore it.

Invisible or Hidden Text: Spammers hide the unrelated text by giving it the colour of the web page's background. They might use some other techniques for hiding the text, e.g., by using tiny font size, using HTML code (no frame sections) for hiding it, zero-sized DIVs, alternative attributes, and 'no script' sections. SE can permanently or momentarily block a whole website for having hidden text or a

[7] https://en.wikipedia.org/wiki/Pay-per-click

few of its web pages [22]. Nevertheless, the invisible text is not always spamming; some web designers are using it to enhance the accessibility of a website.

URL Redirections: URL redirection involves the taking off a visitor to another web page without their intervention, e.g., JavaScript, Meta refresh tags, Flash, and Server-side or Java redirects. However, SEs do not consider permanent or 301 redirects as malicious behaviour [48].

3) Architecture-Based Spamming: This The architecture-based spamming focuses on the design of a website. It monitors SE crawlers to check what a search engine crawls for. Architecture-based spamming is further categorised as follows.

Cloaking: Spammers use several methods for cloaking. It delivers a web page to the SE spider entirely different from the visitors' web page (human users). Spammers might mislead the SEs about the content on a specific website. However, some website designers are using it to increase the accessibility of a website to visitors with disabilities or provide human visitors with data that SE is not able to parse or process [49]–[52]. It is also used to provide users' content based on their location; Google is also using the IP delivery technique to deliver results. IP delivery is a form of Cloaking. Another method currently in use for Cloaking is Code Swapping. It involves optimising a web page for high ranking and, after achieving a high rank, swapping another new web page in its place. Google consider these techniques as Sneaky Redirects [53].

Cybersquatting: It is a combination of two different terms: cyber (computer network) and squatting (unlawfully occupy the property). In this technique, spammers register a domain name which may be a trademark or a replica of some famous brand on the internet. It means using other's identity to get more profit [22], [54].

Typosquatting: It is also a combination of two words: Typo and Squatting. The prefix Typo, in this context, refers to "typographical mistake" [22]. In this method, a spammer registers a domain name similar to the name of the website of a famous brand, like facebook.com; if users made a typographical mistake in the domain name, e.g., "facbook.com", it redirects the visitor to facbook.com, which is not the user's choice. This method is used to deceive unaware visitors [55].

B. *Off-Page Web Spamming Techniques*

In these techniques, spammers try to increase the backlinks (incoming links to a web page) to their sites rather than practising spamming methods on their websites. These spamming methods deal with tactics to promote web pages thereby indexing the web pages to top rank in SERPs [21], [56]. The off-page spamming techniques are further categorised into Link-based spamming, Trust-based spamming, social-based spamming, Mirror websites, and Click spamming.

1) Link-Based Web Spamming: It refers to the presence of links between web pages present for purposes other than quality [57]. In Link-based spamming, the spammers manipulate the link structure and create hundreds of incoming links to get a high rank. Incoming links to a website are a critical measuring factor for a SE's ranking algorithms. SE relies on link building to update the PageRank (PR) of websites. Link spamming takes advantage of weaknesses in link-based ranking algorithms which gives web pages top ranking if other top-ranked web pages link to it [22], [36]–[38], [46]. These link-based spamming techniques aim at influencing other link-based rankings algorithms, e.g., the HITS algorithm[8]. link-based spamming can be categorised, based on many factors, as elaborated below.

Link Farms: It is a close-knit network of web pages linking each other to manipulate SE's ranking algorithms [22]. These networks are identified facetiously as shared admiration societies. After Google

[8] http://pi.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture4/lecture4.html

started the first Panda update in Feb 2011, the usage of links farms is dramatically reduced. This update introduced necessary reforms in its spam detection algorithm. Another prevalent type of link-based spamming is the use of link-building software. It is used for automating the SE optimisation process [57].

Private Blog Networks: It is a group of dedicated web pages used as the primary source of contextual links, leading to the spammer's main website and is created to obtain top SE ranking [58]. For getting backlinks from leading authority web pages, the owners of Private Blog Networks (PBN) websites use auction domains or expired domains with backlinks from top authority web pages. Since 2014, Google is targeting and penalising PBN users. Google is running deindexing campaigns against these PBN users.

Hidden Links: For improving the link popularity, spammers put hyperlinks on their web pages where visitors will not see them. Highlighted text with a secret link can help rank a web page on top for matching that phrase [57].

Sybil Attacks: A Sybil attack deals with the forging of various identities to serve a malicious purpose. It is named after the dissociative-personality-disease patient 'Sybil.' In this type of attack, spammers create multiple web pages at several interconnected domain names, e.g., spam blogs [59]. The only purpose of making these blogs is commercial publicity and passing the link authority to target websites. Usually, spammers design these "splogs" in a misleading way that produces the effect of a reliable website but upon close investigation will usually be found written by rewriting tools or poorly written and hardly readable content [58]. Furthermore, web-spammers also use Guest Blog Spam. It is adding a guest blog on the website to gain links to other websites. Unfortunately, it is usually confused with legal forms of guest blogging with goals other than putting links [22], [58].

Buying Expired Domains: Spammers also use crawling tools to identify expired domains or monitor DNS[9] records of the domains which are about to expire. Spammers then buy these expired domains for replacing the web pages with links to their web pages. Although it is possible for Google to reset the link data but this has not been verified on expired domains [57], [60]. If a buyer wants to keep the whole previous Google ranking data for a domain, the buyer must take it before it is dropped. Some of these methods might be used for building a Google bomb [60].

Cookies Stuffing: It is also known as cookies dropping and is an affiliate marketing method in which a visitor accepts third-party cookies from a web page irrelevant to that visited by the user, often without the visitor being aware of it. It will generate revenue for the spammer doing the cookies dropping [61]. Cookies stuffing not only produces dishonest affiliate sales but also has the power to overwrite cookies of other affiliates, basically robbing their legally earned commissions [37], [61].

Link Wheel: It is also known as Link Prism or Link Pyramid. It is a compelling link-building strategy because it attempts to emulate natural human-linking standards; Google and other SEs always like natural link-building. Rather than obtaining links to inappropriate websites and sources, it gets links from relevant web pages on the internet [62]. Moreover, it can multiply link-count over a given period because the link pyramid pages are linked to each other and other targeted websites or particular URLs. It is the newest and most efficient link-building technique introduced by the spammers [63], [64].

Buying Links: Some spammers may purchase the backlinks from other high page rank websites. Because purchasing backlinks is usually the fastest method to get them, particularly when the income produced by high ranking web pages is greater than what covers the cost of buying the backlinks required to achieve them [65].

Link Exchange: It is the method of exchanging a website's URL with other websites. Currently, link exchange is not useful for increasing the page rank. Because SEs have become intelligent and shady

[9] https://dyn.com/blog/dns-why-its-important-how-it-works/

linking techniques have become identifiable. SEs can penalise or even ban websites participating in link exchange programs [62], [65].

Accessible Pages: Spammers do not own Accessible Pages but they can modify these pages. Spammers can also edit all websites that can be revised and updated by the users to add links to spam websites if proper anti-spam steps are not taken [22], [37]. Automated spambots can quickly make the user-editable section of a website unusable. The example of Accessible Pages are blogs with public comments, Wikipedia pages, general discussion groups, questions and answers websites, and open user-controlled directories. Spammers utilise the opportunity to transform external pages by generating backlinks to their web pages [38], [46].

2) Trust-Based Web Spamming: This is a type of off-page spamming [66]. In this technique, spammers create a link with a page that already has top ranking in SERPs. Trust-Based Spamming includes host parasitic attacks and piracy ads [22]. Similarly, in Social-based spamming, the leading social websites like Facebook and Twitter are used for advertising spammed pages. In this technique, spammers show unwanted spam content on Social Networking platforms with user-created content like comments, chat, and links. Social spamming can be performed in various ways including blasphemy, hate speech, insults, fraudulent reviews, fake friends, malicious links, and privately identifiable information [22].

Web-spammers also use Mirror websites. In this technique, spammers host several websites with conceptually the same content using distinct URLs. Some SEs assigned higher page ranks to the results where the searched keyword resembled the URL [37], [38].

3) Click Spamming: Since SEs use clickstream data as indirect feedback to adjust the ranking function, spammers eagerly create fraudulent clicks to twist these functions towards their web pages [38], [67]. To accomplish their aim, spammers submit keywords to a SE [68] and then click on the link leading to their target web page [69]. To cover the abnormal behaviour, spammers frequently use the click scripts on several machines or in botnets [70]. The other benefit of spamdexing to create fictitious clicks comes from online advertisements [38], [71]. Spammers also click on the competitor's advertisement to reduce their funds and bring their funds to zero and display their advertising in the same spot [38].

IV. NEED FOR THE DETECTION OF WEB SPAMMING

Due to the following adverse effects, spamdexing detection has become a considerable challenge faced by all search engines [72].

- Web spam reduces the worth of search results and removes legitimate revenue from web pages.
- Web spam reduces the trust of a visitor in a SE, resulting in shifting of user to another SE. This is a different problem because it costs a user nothing to shift from one SE to another.
- Spam pages are the primary sources of Pornography, Malware, Viruses, Trojans, and Phishing attacks.
- Web spam has an economic influence since top-ranking presents sizeable free ads and improved size of the web traffic.
- A critical challenge is in tagging, i.e., distinguishing between the most suitable tags for the provided content and removing the spam tags.
- Web spam forces a SE provider to consume a substantial amount of storage and computation resources.

V. Techniques for Combating Web Spamming

The SE experts and researchers are combating spamming techniques [73] and have presented several techniques to resist it. The proposed techniques can be classified into four broad categories. The first category comprises methods that analyse content features, such as detecting content duplication, language models and word count. The second group of methods encompasses link-based information, such as link-based trust and distrust propagation, neighbour graph connectivity, link pruning, study of statistical anomalies and graph-based label smoothing. The third group uses combined approach by integrating content-based techniques with link-based techniques. Finally, the last group consists of techniques that exploit clickstream data, query-popularity information [49], [74], HTTP-sessions information [75], data-mining techniques [76], user-behaviour data [77]–[79], and some other techniques [80], [81].

A. Content-based Techniques for Combating Web Spam

Fetterly et al. carried out essential research on the identification of content-based spam and supplied the foundation for algorithms for identification and prevention of content-based spam. [19], [82]–[84]. Generally, in content-based methods, statistical analysis is used for spam detection. [84]. The primary way that spammers generate spam web pages is using software [85] and software applies woven and phrase stitching techniques to generate pages. These pages are not designed for actual users. As a result, these pages exhibit odd properties. [85]. Different researchers have found that spam page URLs have an excessive amount of dots, digits, dashes and lengths. [38]. The researchers noted that the majority (80%) of the longest identified hostnames led to adult web pages, while a smaller proportion (11%) pointed to financial credit web pages. They realized that these web pages have a duplicated structure. Many spam web pages on the same host will have little variation in word count. Researchers have also found that spam web pages change rapidly. The researchers observed that, on average, the amount of weekly changes made to a given host for all pages could be used to identify most of the very active spam hosts. All the features that are recommended can be seen in the article [84]. In other investigations carried out by them [82], [83], they concentrated on content duplication and discovered that huge clusters with identical content are spam. To detect duplicate content and said clusters, the shingling method [86] is applied which is a Rabin fingerprint-based method [18], [87]. Especially in the first step, on a page, they fingerprint each of the *n* words by using a primitive polynomial[10] (PA) and obtain a token from the first step. Secondly, all these tokens are fingerprinted by them using another primitive polynomial (PB) using extension transformations and prefix deletion. Then, *m* different fingerprinting functions are applied to every string obtained from the second step. The least of the *n* resulting values for all *m* fingerprinting functions is maintained. The document was finally represented as *m* fingerprints, and clustering was performed by using the transitive closure of the nearly identical relationship. The list of famous phrases is also mined by them by ordering the (*i, s, d*) triplets lexicographically and taking adequately long runs of triples with matching *s* and *i* values. Based on this study, it can be concluded that phrases that are derived from the machine-generated content can be monitored starting from the 36th position. So, one can use these phrases as an extra input, parallel to traditional spam terms, for "word's bag" spam classifier.

The research article [19] looked further into the matter and found a few more new content-based features. They integrated all of these components into a classification model that deployed boosting and bagging frameworks, in addition to C4.5. They reported 97.8 per cent true negative and 86.2 per cent

[10] http://mathworld.wolfram.com/PrimitivePolynomial.html

true positive rates for boosting 10 C4.5 trees. In [88], they demonstrated how machine learning models and features can significantly enhance the efficacy of web spam detection algorithms. The authors attained outstanding classification results using cutting-edge learning models, such as LogitBoost, RandomForest, and low computational-content features. They also pinpointed the worldwide and computationally intensive features. The proper and careful choice of a machine-learning model is critical. For example, PageRank (PR) yields little additional quality improvement.

In [60], researchers worked on the HTML-page structure to identify automatically-generated spam pages and they introduced HTML-page structure-based features. This work is similar to the work done by Fetterly et al. [82], [83] but here they performed non-traditional pre-processing steps. They kept only the layout of the page and removed all the content from the page. Therefore, they investigated page duplication by analysing the structure of a page instead of its content. The fingerprinting techniques [87], [89] are applied by them with subsequent clustering to identify clusters of structurally near-identical spam web pages. Mishne et al. introduced a spam identification technique in blogs [90]; they compared the language models [91] for comments and web pages. This research's fundamental idea is that due to the erratic behaviour of spam comments, these models are likely to be radically different for a spam web page and blog. They used Kullback-Leibler divergence to measure the inconsistency between language models (probability distributions) Θ_1, Θ_2:

$$KL(\Theta_1 \parallel \Theta_2) = \sum_w p(\, w|\Theta_1) \log \frac{p(w|\Theta_1)}{p(w|\Theta_2)} \qquad (2)$$

An advantageous characteristic of this technique is that training data is not required in this method.

In [92], [93], researchers analysed the linguistic features for the identification of web spam. They examined many Natural Language Processing (NLP) factors like emotiveness, lexical and content heterogeneity, lexical legitimacy, use of passive and active voices, syntactical heterogeneity and entropy, and many others. Several elements are proposed by Benczúr et al. based on the appearance of keywords on a web page, which would either indicate that the page is hugely spammed or of excellent advertising value. [74]. The authors also studied the discriminatory effect of the following features: popular keywords by Google AdWords and the number of Google AdSense ads on a web page, Online Commercial Intention value allocated to a URL in a Microsoft adCenter and Yahoo Mindset Classification of a web page as either non-commercial or commercial. [38]. Their spam detection accuracy is more than 3% higher than the work done by Castillo et al. [26] who did not consider these features. For the identification of cloaking, a similar technique was applied. Chellapilla et al. analysed the advertising click-through logs and Search Engine (SE) query logs in order to assess their potential for monetisability and popularity among users. [49]. Monetisability is defined as the revenue generated by advertisements appearing in response to users' search keywords, while query popularity refers to popular keywords used by users to search for relevant information online. [94]. The authors used the top five thousand keywords from all keyword categories and requested the top two hundred links for every keyword four times, using different agent-fields to mimic requests from a Crawler (C) and a User (U). Additionally, they used the cloaking test (Equation 3), which is the updated version of the cloaking-detection test that was originally proposed. [50], [51].

$$CloakingScore\ (p) = \frac{\min[D(c_1,u_1),D(c_2,u_2)]}{\max[D(c_1,c_2),D(u_1,u_2)]} \qquad (3)$$

where

$$D(a_1, a_2) = 1 - 2\,\frac{a_1 \cap a_2}{a_1 \cup a_2} \qquad (4)$$

The equation above reflects the normalized-term-frequency variation for two copies of a web page described as a set of terms. Researchers discovered that their proposed method is successful in cloaking detection, with 0.75 and 0.985 accuracies for monetizable and popular queries respectively. Yet, there are some faults in their work. Their proposed technique can have a relatively high false-positive rate; for example, legitimately generated dynamic web pages also include several links and terms on each access. To address this drawback, they strengthened their earlier suggested technique by utilizing the structural features of a web page. [52]. The updated process swapped out links and words on web pages for tags to compute the cloaking score. Hua and Huaxiang assessed the content features of spam and non-spam web pages. There are several key statistical regularities that content on non-spam webpages tend to follow, whereas spam webpages typically only display a handful of such regularities. This is because spam webpages are often generated randomly and with a great deal of duplicate content in order to boost their position in search engine results pages. The researchers investigated the content elements of websites and found significant variations between the content of spam and non-spam pages. The authors looked at different content features to try and see what impacts a web page's ranking. These features included the number of words on the page, the number of words in the title, the anchor's text fraction, the average length of the word, corpus precision, the fraction of the visible text, corpus recall, compression rate, query precision, query recall, independent LH and entropy. Upon close examination of the aforementioned features, they found that many patterns and similarities exists in the content features for non-spam web pages, while only a few exist in spam web pages [95].

Automated article spinning tools are popular among web-spammers. They employ spinning tools to sidestep the detection of duplicate content. The spinning tool can replace words or phrases with synonyms to evade plagiarism detectors. The ease of use of spinning tools makes them attractive to spammers. With a few clicks, spammers can take a given article and spin it hundreds of times, then post their spam articles on target websites using web proxies [96]. Zhang et al. put forward a technique for the detection of spun content. Their method is closely linked with the method used by article spinning tools. The author's technique is based on expressions or words that don't change when the spinning tools generate spun content. The notion of spun articles led them to create the tool "Dspin." Their studies employed two collections of crawled articles to automatically identify spun articles and the spamming tendencies of spammers. [96]. During the web-spamming challenge [97], various methods for spamdexing detection were suggested by scholars. Important content features were recommended by researchers in [30]. They implemented HTML-based features and text compressibility to detect and discourage content spam. Piskorski et al. [93] investigated a multitude of linguistic features. Latent Dirichlet Allocation (LDA) [98] is a popular technique for text classification tasks. Biro et al. improved the LDA and created the linked LDA [100] and multi-corpus LDA [101] models to detect web spamming more effectively. To counter web pages laden with spam content, as well as to identify such web pages, [99] proposed a semi-supervised method that features combinatorial fusion. They employed semi-supervised learning to capitalize on unlabelled samples and utilized the combinatorial feature fusion technique to generate new features and to diminish the term frequency-inverse document frequency (TF-IDF) of content-based features. The experimental results showed that their technique was effective.

B. Combating Techniques for Linked-based Web Spam

There are five different categories of link-based algorithms for detecting web spam, based on how they work. The first category is about identifying suspicious nodes, links, and then down-weighting them [100]. In the second category, website-detection algorithms deal with the relationship between a set of web pages with known labels and with unknown labels. [18]. The third category of link-based web-spam-detection algorithms makes use of graph regularisation methods. [101]. The idea of improving labels based on web-graph topology is something that fourth category of link-based spam-detection algorithms use. [102]. The final category of algorithms extracts link-based features for every node and employs various machine-learning methods to detect web spam. [103] — linked-based web-spam-detection techniques discussed above are catalogued in Table 1. Comparison criteria are based on the working mechanism, algorithms used, complexity, and information type used. As most of the link-based techniques support the HITS and PR algorithms, PR and HITS are the most critical among all these algorithms.

1) Preliminaries for Web Spam Identification

Web Graph Model (WGM): The web graph represents the links among the WWW web pages. A graph, in general, is a set of edges and vertices. Spirin and Han modelled the Web as a graph G = *(V, E)* [38], where web pages are represented by *V* (vertices) and hyperlinks between the web pages are represented by *E* (directed weighted edges) [38]. If a page p_x has several hyperlinks to a web page p_y, they dropped all links in one edge $(x, y) \in E$. Self-loops are not allowed. They represent a group of web pages connected by a web page p_x as an *Out* (p_x) and a group of web pages indicating p_x as an *In* (p_x). Moreover, each edge $(x, y) \in E$ can hold an associated non-negative weight w_{xy}. A general weights-allocation strategy is $w_{xy} = \frac{1}{|Out\ (p_x)|}$, although different strategies are also possible [38]. For example, Abernethy et al. assigned a weight proportional to some links between web pages. A transition matrix M represents the web-graph model in a matrix notation defined as

$$M_{ij} = \begin{cases} w_{xy}, if\,(x,y) \in \mathcal{E} \\ 0, otherwise \end{cases} \qquad (5)$$

PageRank (PR): For ranking websites in Search Engine Results (SER), Google Search uses the PR algorithm. PR algorithm was named after one of Google's founders Larry Page [11]. PR is used to measure the significance of website pages. PR works by measuring the quality and counting the links to a web page to determine its importance. So the underlying assumption is that useful websites will get more incoming links from other relevant websites [5]. For ordering the SER, Google is not only using the PR algorithm, but it was the very first algorithm used by Google. PR is a link-analysis-based algorithm that allocates a numerical weighting to every element of a hyperlinked set of documents for calculating its relative significance within the set. The assigned numerical weight to a component A is known as the PageRank of A, represented by PR(A).

Additional factors such as author rank can also contribute to the significance of a component. Consider an example of a small website of only four web pages: *W*, *X*, *Y*, and *Z*. PR algorithm ignores several outgoing links from a single page to another individual page or if a web page is linking to itself. The latest version of the PR algorithm is based on the probability distribution between 0 and 1. Initially, it will assign the same values to all four web pages. Therefore, the initially assigned values for *W*, *X*, *Y*, and *Z* would be 0.25. Upon the next iteration, the PR, transferred from a web page to the targets of its outgoing links, is evenly distributed among all outgoing links. For instance, if web pages X, Y, and Z have an outgoing link to W, every link will transfer 0.25 PR to W on the next iteration, for a total of 0.75.

Table 1:Link-based Web spam Detection Techniques Comparison

	Techniques				
	Label Refinement Strategies	Feature-Based Strategies	Graph Regularisation Strategies	Link Propagation Strategies	Link pruning and Reweighting
Strategies used in web spam detection Algorithms	Clustering Algorithms	Truncated PageRank	PageRank	PageRank TrustPage	HITS PageRank
Working criteria for web spam detection Strategies	Using several machine learning algos to extract link-based features from each node	Graph regularisation technique used	Based on the web graph topology it is using the idea of label refinement	Technique takes advantage of the topological relationship between web pages	Identify the links and suspicious nodes and their subsequent down weighting
Mining Techniques Used	Web Content Mining	Web structure Mining	Web Structure Mining	Web Structure Mining	Web Structure Mining and Web Content Mining
Type of information used by spam detection strategy	Link base features of each node	Structural Patterns	URL	Topological Relationship	Down Weighting of links and nodes
Complexity	Limited data set are allowed	-	URL Classification	Internal Structure	Relationship between the nodes

$$PR(W) = PR(X) + PR(Y) + PR(Z) \qquad (6)$$

Instead, consider a bit more complicated example. If X had an outgoing link to Y and W, Y had an outgoing link to W, and Z had outgoing links to all three web pages. Upon initial iteration, X will transfer half (0.125) of its actual (0.25) value to W and the other half (0.125) to Y. While Y will transfer the whole existing value (0.25) to the only web page it links to, namely W. Since Z had outgoing links to all three web pages so Z will transfer one-third (0.083) of its actual value to W. After the completion of this iteration the approximate PR of W would be 0.458.

$$PR(A) = \frac{PR(X)}{2} + \frac{PR(Y)}{1} + \frac{PR(Z)}{3} \qquad (7)$$

Therefore, the PR conferred by an outgoing link = Document's own PR score / Number of Outgoing Links *L*

$$PR(W) = \frac{PR(X)}{L(X)} + \frac{PR(Y)}{L(Y)} + \frac{PR(Z)}{L(Z)} \qquad (8)$$

In general, the PR value of any web page *i* can be represented as:

$$PR(i) = \sum_{v \in X_i} \frac{PR(v)}{L(v)} \qquad (9)$$

For computing global importance scores for all web pages on the World Wide Web, Page et al. used the link information [104]. The fundamental underlying concept is that an outgoing link from a web page P_x to a web page P_y shows web

page P_x's confidence in P_y. The famous and the most straightforward way to present the PR is linear system formulation. In this way, the PR vector for every web page on the World Wide Web is represented as a solution of the following matrix equation.

$$\vec{\pi} = (1 - c).M^T \ \vec{\pi} + c.\vec{r} \qquad (10)$$

where *c* represents the damping factor and *r* represents the Static PR vector. (1/N,….,1/N) is a unit vector for non-personalised PR where N = |V|. It must meet the Perron-Frobenius Theorem conditions [105] and Markov Chain stationary distribution [32]. Personalised PageRank (PPR) is known as the solution for non-uniform static vector $\vec{r}$, where $\vec{r}$ is called random jump, personalisation or teleportation vector [106], [107]. There are some

essential properties of PR using which we can expand PPR as following.

$$PPR(\vec{r}) = \frac{1}{N} \sum_{v \in V} PPR(\chi_v) \qquad (11)$$

where χ_v represents the teleportation vector containing all zeros except the node (v) such that χ_v (v) = 1 [38]. PR has an interpretation as a probability to make *j* steps before stopping is equivalent to c · (1 − c) j and the representation followed is valid [108], [109],

$$PPR(\vec{r}) = c\vec{r}.\sum_{j=0}^{\infty} (1 - c)^j (M^T)^j \qquad (12)$$

HITS: It stands for Hyperlink-Induced Topic Search, also famously known as hubs and authorities. HITS is a link-analysis algorithm developed by Jon Kleinberg and is used for rating web pages. [44]. The basic concept of the authorities and hubs is derived from the particular perception of creating pages when the internet was initially forming. That is, specific pages are distinguished as hubs, which served as large directories, but the information these hubs were delivering is not authoritative and these pages are used as a catalogue of information that directed the visitors to other authoritative web pages. In simple words, the best hub represented a web page that leads to several other web pages and a competent authority is represented by a web page that is linked by several hubs [110]. Therefore, this system assigns two different scores for each web page, authority, and hub score. The value of the web page's content is estimated by authority and the value of the links to other web pages is estimated by the hub [9]. A series of iterations are executed in the HITS algorithm and every iteration consists of two necessary steps: authority update and hub update [111]. To start ranking, the following condition is assumed

$\forall_p, auth(p) = 1$ and $hub\ (p) = 1$.

Two types of update rules should be considered, hub-update rule and authority-update rule. For measuring every node's authority/hub scores, repeated iterations of the hub-update rule and authority-update rule are applied [112].

Authority-Update Rule

$\forall_p$, $auth(p)$ is updated to be the following summation:

$$auth(p) = \sum_{i=1}^{n} hub(i) \qquad (13)$$

where *n* represents the total number of web pages linked to *p* and *i* is a web page linked to *p*. Therefore, a page's authority score is the sum of all hub scores of web pages that lead to it.

Hub-Update Rule

$\forall_p$, $hub(p)$ is updated to be the following summation:

$$hub(p) = \sum_{i=1}^{n} auth(i) \qquad (14)$$

where *n* represents the total number of web pages linked to *p* and *i* is a web page linked to *p*. Therefore, a web page's hub score is a sum of all web pages' authority scores linked to it. After a specified number of repetitions of the algorithm, the final hub authority scores of all the nodes are calculated. Since an iterative and direct application of the authority-update rule and hub-update rule leads towards diverging values, the normalisation of the matrix is essential after every iteration. Therefore, the values gained from this method will ultimately converge.

2) Techniques based on Label Propagation

The primary motivation for using label-propagation-based algorithms is to examine a set of web pages for which labels are already known, and then using a series of propagation rules to determine labels for other nodes.

TrustRank (TR) is one of the earliest algorithms from this group. It uses PageRank which has been tailored specifically for trust to propagate it from a small, select set of extremely trustworthy pages. [88]. The commonly held belief is that well-crafted web pages tend to link to high-quality websites. The TrustRank algorithm relies on the principle that a good set of pages is relatively isolated. They recommended inverse PageRank for choosing the seed set of well-known pages; it functions on the graph with all connections reversed. After calculating the inverse PageRank values for every single web page on the World Wide Web, they selected the Top-P web pages and sought the opinion of human experts to decide the stature of these web pages. The personalisation vector is constructed by assigning a non-zero value to components that correspond to creditable web pages as judged by the user. Ultimately, the PPR score is computed. However, compared to PageRank, TrustRank has much better properties for indicating web spam status. Anti-TrustRank (ATR) is a follow-up to trust propagation [113]. [114] demonstrated how a web-spam-detection problem could be approached with a novel technique that adopts the ACO learning method to construct a rule-based classifier. The authors suggested three various methods: Distrust ACO, Trust-ACO and Combined ACO that is contingent on distrust and trust hypotheses. The first strategy is intended to produce the spam classifier that distinguishes the spam from non-spam web pages The second strategy is focused on creating the non-spam classifier which would be able to discern non-spam pages from spam pages. The third approach is to use both classifiers by combining and reordering all of the rules. Additionally, the authors also proposed an adaptive-learning method which would incentivize or penalize the ants based on their performance in order to improve their results. The method is assessed by means of two publicly accessible data sets, namely WEBSPAM-UK2006 and WEBSPAM-UK2007. After acquiring the experimental outcomes, the authors declared that their proposed method could identify web spam precisely.

The TrustRank is not the only technique for identifying spam web pages. Another approach, known as distrust propagation, is more accurate. In this approach, researchers start with a seed set of web pages with high PR values. From there, they identified spam web pages by following links on an inverted graph. This technique outperforms the TrustRank in terms of accuracy. Further research was conducted on Trust and Anti-TrustRank [115]–[117]. In [118], Different researchers have proposed different semi-automatic anti-spam algorithms, with the most recent being TDR. Their technique takes advantage of trust and mistrust propagations, as well as differential trust and distrust propagation.

Another method for determining the poor quality of a web page is called BadRank (BR) and it uses the inverse PR calculation[119]. The Wu et al. study illustrated that there is a relationship between PR and TR and BR and ATR. They further explored how distrust and trust can propagate together. [120]. First, the TR algorithm's trust propagation method is challenged, each heir receives the equivalent share of trust from a predecessor $.\frac{TR(p)}{|Out(p)|}$. Moreover, they suggested two more procedures:

Logarithmic splitting: Every child in a logarithmic split receives an equal share of the parent's score, normalized by the log of the number of children.

$$c.\frac{TR(p)}{\log(1+|Out(p)|)} \qquad (15)$$

Constant Splitting: In constant splitting, every child receives a similar discounted share of trust from parent *c* · TR(p) regardless of the number of children. Their investigation of several partial trust aggregation strategies revealed that TR only recapitulates every parent's trust score. They propose a linear combination of distrust and trust values as an improved trust aggregation strategy.:

$$TotalScore(p) = \eta.TR(p) - \beta.AntiTR(p) \qquad (16)$$

where η, β ∈ (0, 1). Their findings demonstrated that a hybrid of the two methods is more effective in web spam detection. Guha et al. explored the concepts of trust and distrust propagation in relation to reputation systems [121]. The two algorithms make use of the disintegration property of PR to identify the excessive PR coming from dubious pages [18], [122].

3) Techniques based on Link-based Features

In this type of algorithms, web pages are represented as feature vectors in order to carry out the clustering analysis or standard classification. This allows for a more accurate understanding of the data and how it can be best used. Amitay et al. worked on link-based features to categorise websites based on their functionality. They believe that websites that have similar layouts, such as the number of outgoing links per page or the average page level, have comparable functions on the internet [103]. The researchers used a dataset of 1,100 websites and applied cosine similarity to cluster them into 31 groups. They found that 183 of these websites were part of a web spam ring.

In [123], the authors compared various machine-learning classifiers that are currently used for web spam classification. The researchers looked at how well current machine-learning classifiers work and talked about new features that could help in web spam detection. In [124], Patil and Bhadane investigated various efficient spam-identification methods based on classifier that combine language models with novel link-based features. In [125], the authors proposed CFS-PSO, a Swarm-Based hybrid technique that consolidates the characteristics of the Particle Swarm Optimisation[11] (PSO) strategy and Correlation-Based Feature Selection (CFS). The key to success for Machine Learning and Data Mining is choosing the right features (a pre-processing strategy). The goal of feature selection is to create a simpler and more logical model that will improve performance by increasing accuracy and decreasing development time for the learning model. The authors assessed the efficacy of their technique on WEBSPAM-UK2006 with five classifiers. Their empirical results demonstrated a decrease in original features and an increase in F-measure up to 88% & 45.83% respectively.

4) Techniques based on Graph Regularisation

The spam detection algorithms in this group are more effective. These algorithms make use of the web graph for smoothing predicted labels. Several empirical analyses and studies have demonstrated that employing graph regularisation algorithms for web-spam detection is more effective. [126], [127] used the regularisation theory for spam detection, as described in [101]. The primary reason for this is that it addresses the fact that spammers are not placing the hyperlinks randomly to some extent, there is a similarity between linking web pages [128], [129]. The primary motivation for adding a regularizer to the objective function is to encourage precise predictions. Additionally, by using an approximate isolation of genuine web pages, it argues for an asymmetric regularizer. The objective function is as follows:

$$\Omega(\vec{w},\vec{z}) = \frac{1}{l}\sum_{i=1}^{l} L(\vec{w}^T \vec{x}_i + z_i, y_i) + \lambda_1\|\vec{w}\|^2 + \lambda_1 \|\vec{z}\|^2 + \gamma \sum_{(i,j)\in\varepsilon} a_{ij}\Phi(\vec{w}^T \vec{x}_i + z_i, \vec{w}^T\vec{x}_j + z_j), \tag{17}$$

where $\vec{w}$ expresses the vector of coefficients, $\vec{x}_i$ and y_i representing the features and a real label respectively, L (a, b) denotes the loss function, bias term is z_i, the weight of the link $(i, j) \in \mathcal{E}$ is represented by a_{ij}, and the regularisation function is $\Phi (a, b) = max\,[0, b - a]^2$. The authors offered two distinct methods to solve the optimisation issue: the conjugate gradient and alternating optimisation. The host graph's weight setting problem was also studied by these researchers who concluded that the logarithm of the number of links provides the best results. Finally, their experimental study proved that their algorithm was highly scalable. A discrete analogue of classification regularisation theory [126], [127], has been put forward by Zhou et al. They have determined discrete operators of Divergence, gradient, and Laplacian on the directed graphs and proposed an algorithm [130]. Initially, the inverse

[11] http://www.swarmintelligence.org/tutorials.php

weighted PR is computed with transition probabilities which are defined as $a_{ij} = \frac{w_{ji}}{In(p_i)}$. In the second step, they created the graph Laplacian.

$$L = \prod - \alpha \frac{\prod A + A^T \prod}{2}, \qquad (18)$$

Here α represents the user-specified parameter in [0, 1], the transition matrix is represented with A, and Π represents the diagonal matrix with the PR score over diagonal. Then, they solved the matrix equation below.

$$L\vec{\varphi} = \prod \vec{y} \qquad (19)$$

Several studies were conducted on the algorithms based on graph regularisation and every review comes up with different exciting results. Cheng et al. proposed extracting web spam URLs from discussion forums of search engine optimisation [131]. The underlying motivation behind this study was a critical observation of the author in SEO forums. Usually, spammers use SEO forums to share their website links and find other people to build a global link farm. They suggested a technique for solving the URL classification problem using extracted features from search engine optimisation discussion forums, websites, web graphs and regularising it with four terms obtained from the user's URL graph and link graph. After their experiments, they reported that their technique of spam identification could effectively detect legitimate looking spam pages.

5) Spam-Detection Techniques based on Label Refinement

Researchers used machine-learning techniques for label refinement to classify general problems [18]. Some of the researchers used this technique for spam detection also. In this section, we will discuss the algorithms which use this technique for web-spam identification. Benczur et al. proposed some web-graph-based techniques. Their proposed algorithm works in two different stages [17]. Initially, they assigned the labels using the spam detection algorithm discussed by Becchetti et al. [16]. In the second stage, they improved the labels using one of the three different strategies. Propagation with random walks is used in the first strategy for label refinement [111]. Initializing the personalisation vector $\vec{r}$ in personalised page rank by normalising the foresight of base algorithm: $r_p = \frac{s(p)}{\sum_{p \in v} s(p)}$, where the base algorithm prediction is represented by *s(p)* and *(rp)* is the component of *r* (vector) that corresponds to p (page). The other approach is based on web-graph clustering [132], which improves the labels by applying the following rules. If most web pages are predicted as spam in a cluster, it will mark all the web pages in a cluster as spam. The base algorithm's predictions are assumed in [0,1] and finally, the cluster's average value is calculated and compared against the threshold value [133]. A similar technique is used for the prediction of non-spam. Finally, Stacked Graphical Learning (SGL) [134] strategy is used. The concept behind this strategy is to use the machine-learning algorithm again after extending the original feature representation of the object with a feature that is an average prediction for related web pages in the graph. After two cycles of stacked learning, they reported a 3% improvement over the baseline. [102], [135] proposed a few more relabelling techniques. For reducing the size of the training dataset in web-spam detection, Geng et al. applied self-training. They also suggested the use of feature re-extraction technique using propagation, clustering, and neighbour-graph analysis [136].

6) Techniques based on reweighting and Link Pruning

Another category of web spam detection and prevention is link-pruning and reweighting-based algorithms. The working mechanism of algorithms belonging to this group is finding the suspicious and unreliable links followed by demoting those links. Bharat et al. [137] identified issues in HITS algorithm [44], such as neighbour-graph topic drift and dominance of a mutually reinforcing relationship. They augmented content analysis with link analysis and proposed techniques for solving these issues. Nomura et al. [138] worked on the same problem and they offered the projection-based approach for computing the authority scores. In the HITS algorithm, they modified the eigenvector part to get the best results. A group of researchers introduced the Idea of the tightly-knit community (TKC) and proposed the SALSA algorithm [139]. To calculate the hub and authority values for web pages, SALSA performs two random walks in a sub-graph retrieved initially by the keyword-based search. It is worthwhile to mention that inverted and original sub-graphs are supposed to get two different scores. Roberts and Rosenthal extended the same work; they used clustering structure on web pages. They analysed the linkage patterns of web pages to down-weight the suspicious and lousy links [140]. They used the fundamental idea to identify and count the number of cluster-points to a web page instead of counting the number of individual web pages. Therefore, a page's authority can be defined as follows.

$$a_j = \sum_{k:j\in l(k)} \frac{1}{\sum_{i:j\in l(i)} S_{ik}} \qquad (20)$$

where $S_{ik} = \frac{|l(i)\cap l(k)|}{|l(i)\cup l(k)|}$ and *l(i)* represent sets of web pages linked to web page p_i. This method works like a popularity-ranking technique discussed in articles [17], [72]. The "Small-in-large-out" problem of the Hyperlink-Induced Topic Search algorithm is studied by Li et al. [141] who suggested reweighting the outgoing and incoming links for web pages from the root set. Davison introduced the new concept of nepotistic links. These are links that are present for some reasons instead of quality, for example, links among the web pages in a link pyramid or links for navigation on a website [142]. Davison recognised the nepotistic links using the C4.5 algorithm. He used 75 distinct binary features, for instance, IsSimilarHost, IsSimilarHeaders. Finally, he suggested the down-weighting or pruning nepotistic links. In another work Wu and Davison, they analysed the links in link farms [100]. The algorithm works in three steps. Initially, it selects a group of lousy seed web pages. It extends the group of lousy web pages using the idea that web pages should be considered harmful if pointing to several flawed web pages from the seed set. Finally, the links between the extended group of lousy web pages are down-weighted or eliminated and any ranking algorithm (rank-based) can be used now. da Costa et al. [143] studied similar techniques on the host level. In another work, Wu and Davison proposed a complete hyperlink [144], a link copulated with associated anchor text. An entire hyperlink is used for the identification of web pages having the same suspicious linkage pattern. [145] noticed that the PR score of web pages that obtained high page ranks by using the link-spamming methods correlate with the damping factor c. They used this observation for the identification of suspicious nodes. A more general analysis of several collusion topologies is performed by Baeza-Yates et al. [146]. They showed that PR's improvement is negligible for top-ranked web pages because of the power-law distribution of PR [147]. A similar kind of work is done by [9], [45].

C. Spam Detection Based on Combined Approach

This section discusses the techniques which use the combined approach for spamdexing detection. Abernethy et al. [101] suggested a system that combined link- and content-based features to identify web spam. Their research studies proposed an algorithm known as WITCH, to identify web spamming.

They used support vector machine (SVM) in conjunction with a graph regularisation classifier (GRC). Egele et al. [148] introduced a new technique to distinguish between authentic web pages and spam web pages. They used a j48 decision-tree classifier as part of their technique. By decreasing the false positive rate to zero, they were able to detect one spam page out of five spam pages. A technique called SAAD (Spam Analyser and Detector), proposed by Prieto et al. [149] is based on a heuristic set and is designed to detect spam. The researchers utilized two publicly accessible datasets, Email Spam and Yahoo!, to compare and test the SAAD against other existing benchmarking techniques. After obtaining the experimental results, it was announced that the technique proposed by them can protect users from attacks and secure the client environment. Goh et al. [150] proposed a link-based technique for detecting spam pages using weight properties. They classified weight properties as the impact of one web node on the other. The WEBSPAM-UK2007 dataset was used for their experiments and their results surpassed the benchmark algorithms by 6.11 percent at the page level and by 30.5 percent at the host level. Roul et al. [151] suggested a blended approach for content- and link-based spam detection that can identify various kinds of web spam. They used term density and parts of speech (POS) ratio to detect content-based spam and investigated the personalised PageRank to classify web pages as spam or non-spam. The WEBSPAM-UK2006 dataset was used for their experiments and the results were compared to those of existing methods. Their approach is highly effective, as demonstrated by their excellent F-measure of 75.2 per cent. The importance of historical web page information for spam identification was first explored by Dia et al. [152]. In order to improve spam classification, they utilized the content features from the previous version of pages. The supervised learning techniques were used to combine the classifiers based on the temporal characteristics and the current page content. They extracted various temporal features from archival copies of the Web that are available on the Internet Archive's Way Back Machine. For their research, they utilized the WEBSPAM-UK2007 dataset.

D. Spam Detection Based on Non-Traditional Techniques

This section will discuss the spamdexing detection algorithms that use non-traditional features and techniques to identify spam.

1) HTTP Analysis and Real-Time Spam Detection

Based on the working mechanism, the techniques used for HTTP analysis and real-time spam detection can be sub-categorised further into two groups: server-side and client-side. Server-side techniques are more accurate because they can include additional real-time information, while client-side techniques use insufficient information and usually do not need learning and are less accurate. Webb et al. proposed a lightweight client-side spamdexing detection technique [75]. In the study, they considered the HTTP-session information instead of investigating the link-based and content-based features of a web page and obtained competitive outcomes. Each of the associated sessions and pages is represented as a vector of features, such as, IP address or request header's words in a "bag of words" model. Using several machine-learning algorithms, they came up with a classification. In another study, the same group of researchers introduced the technique for creating a massive web spam detection dataset. The dataset can be created by obtaining URLs from email spam messages [153]. However, the dataset is not completely clean and includes about 350000 spam pages.

Furthermore, researchers also analysed the HTTP sessions for the detection of malicious redirections issues. Chellapilla and Maykov performed an extensive study of the issue [53] and categorised all spam redirection methods into three types: JavaScript redirection, HTTP redirection, and META Refresh.

Furthermore, at the end of their studies, they introduced a lightweight technique for JavaScript-redirection detection. JavaScript-redirection detection is the most common type of redirection, but its identification is challenging. Svore et al. presented the idea of using rank-time features apart from query-independent features [69]. They significantly resolved the issue of the demotion of spam web pages after the query was issued based on the rule that web spammers mislead the page rank algorithms and obtain the top ranks using prohibited techniques instead of genuinely related web pages. For that reason, the spam web pages should be outliers. In general, researchers used 344 rank-time features, for instance, the number of web pages containing the search term, headings and subheadings of the web page containing the search term, the number of search terms in the page title, and total frequency of a search term on a web page. According to their analysis, by adding rank-time features, precision can be increased by 25% at the same recall levels. During the same research, they found that the overfitting problem in web-spam identification can be fixed by keeping the testing- and training-data domains separated. Otherwise, the testing error can be 40 per cent smaller compared to the real error.

Spam content and links affect the desktop web pages and affect the performance of the mobile version of web pages and social media networks. So, it is necessary to solve the spamdexing issues in the mobile version of web pages and social media networks. Researchers are also analysing these issues carefully and proposing different spam identification techniques using HTTP analysis and real-time detection techniques. The mobile version of websites is significantly different from their desktop version in terms of layout, functionality, and content. Therefore, existing methods and techniques for identifying spam websites are not performing well for such web pages. Spam content on social media networks (Facebook, Twitter, Reddit) is also a crucial problem nowadays. For social-network-spam detection, current research focuses on using machine-learning techniques. Twitter uses tweets' statistical features for spam identification. In [154], the authors studied a similar type of problem. They observed that in the labelled tweets dataset, the spam tweets' statistical features change with time, which is one of the primary reasons behind low performance in existing machine-learning-based classifiers. This problem is known as Twitter Spam Drift. To overcome this issue, the authors analysed one million non-spam and one million spam tweets and then proposed a new Lfun technique. Their proposed method can find the change in spam tweets from unlabelled tweets and include it in the classifiers' training process. The authors evaluated their technique by several experiments and proved that the Lfun approach could improve spam identification accuracy in real-world scenarios.

2) Web Spam Detection using Evidence Theory

As we know, search engines are the primary sources for searching information on the web. Determining the liability of search results presented by the search engine is a considerable challenge due to the presence of web spam. Spam techniques are evolving and new types of spam are introduced every day, making the determination of the accuracy of the results more challenging. In the presence of uncertainty, the issue is the reasoning problem. The authors introduced a technique for predicting web spam in which they formulated the spamicity of spam as a reasoning problem. Their method is based on evidence theory, a mathematical model based on DST (Dempster-Shafer Theory). The main advantage of their spam-detection technique is the ability of Dempster-Shafer Theory to deal with the uncertainty. Usually, when a new type of spam is introduced, the existing spam detection systems require proper prior knowledge for spam detection, while Dempster-Shafer Theory does not need the previous information for newly introduced spam detection. In last, they statistically evaluated their proposed technique and reported 99.27% accuracy in web spam detection [155].

3) Unsupervised Web-Spam Detection

[163] – [165] performed different studies on unsupervised spam-detection techniques. They introduced the concept of spamicity and developed the online client-side algorithm for detecting the web spam. A unique feature of their method is that the training data is not required. A previous study introduced a (θ, k)-page farm model [168] and used this model as a base for further studies. The working mechanism of the algorithm proposed by them is that it greedily selects the web pages for a given web page from the k-neighbourhood that shares the most PR score by using equation (21).

$$PRContrib(v,p) = \begin{cases} PR[p,\mathcal{G}] - PR[p,\mathcal{G}(V-\{v\})], if\ v \neq p, \\ \frac{1-c}{N},\ otherwise, \end{cases} \qquad (21)$$

Moreover, it calculates the score for link-spamicity at every iteration as the observed PR contribution ratio from chosen web pages over the optimal PR contribution. The property of Monotonicity is used in the algorithm for limiting the number of supporters under consideration. Subsequently, it identifies the web page as doubtful if the whole k-neighbourhood of a selected web page is processed and the threshold score is still less than the link-spamicity score. [159] introduces an unsupervised technique for the identification of spam content. The authors used a stochastic approach for link-structure-based algorithms to classify spam content. Dataset was obtained from the famous Dutch social networking website and tested with several performance measures, for instance, time of execution, true positive rate, accuracy, and false-positive rate. They reported that their technique outperforms the currently existing unsupervised spam-detection models based on HITS. Different unsupervised and semi-supervised techniques are used to fight against web spam in [160]–[162].

4) Click-Spam Detection

The click-spam's main target is to intentionally add malicious noise in the query log to corrupt the data of the ranking function. Most of the click-spam detection techniques investigate methods to enhance the learning algorithms so these algorithms can prevent malicious noise. Identifying the instances of click-fraud methods using different systems and techniques are no more a secret now. For example, click fraud happens when a malware, user, or bot click on paid search listings in SERPs or click on an advertisement based on pay per click to generate fake traffic. These kinds of fraudulent clicks can cost a massive amount of money to advertising sponsors. Fraudulent click-detection systems extract statistical information from an event database, for instance, the ratio of unpaid clicks to pay per click. As a reference dataset for analysing the local dataset's statistical information, the global dataset's statistical data is used. No fraudulent clicks are considered to have occurred if the statistical datasets match relatively and if it does not match relatively well then the click fraud is considered to have occurred [163]. Radlinski proposed a fascinating method for click-spam prevention [67]. To prevent click-spam manipulation, he recommended using the personalised ranking functions. Author presented an interesting utility-based framework and performed an empirical study to demonstrate that spammer's manipulations can be reduced using personalised ranking. Dou et al. investigates the quality of the standard click-through-based ranking function's construction process and concludes that it is effective against fraudulent clicks [68]. Immorlica et al. examine the online advertising platform's click-fraud issue and mainly discuss the issue of competitor bankruptcy [71]. The authors presented a click-based family of ads pricing models and proved theoretically that such a model could limit the economic benefits for web spammers to perform malicious activity. [164] thoroughly investigates the complete ecosystem of spammers' by introducing the spam double-funnel model. This model explains the communication between advertisers and spam publishers through web page redirections. Bhattacharjee and Goel introduced the incentive-based ranking model [165], which includes users in ranking construction and provides a valuable economic opportunity for users to fix the system's imprecision.

Li, Xin, et al. proposed a new method for automatic session-level identification of click spam. Initially, they used a triple sequence for modelling the user's session. Then they constructed the pattern-session bipartite graph and user-session bipartite graph for describing the relationship between sessions and users as well as sessions and patterns. Finally, after the detection of cheating sessions, they distributed the cheating score on the bipartite graph to improve their work's recall and precision. They claimed that, compared to traditional session-level click-spam-identification methods, their method performed better with an accuracy of 97% [166]. Some other researchers in the field have done similar research work on click spam detection [167]–[170].

5) Semantic-Based Spam Detection

To overcome the drawbacks of content-based web-spam detection, semantic-based spam-detection techniques are used, where the semantics of the website is analysed instead of content. Wan et al. proposed a technique for detecting spam web pages based on the web page's topic and semantics. The authors used two categories of features: statistics and semantics. Initially, they performed the topic modelling over web pages' contents, with the mapped contents of a web page into the topic space. Secondly, they did the semantic analysis and calculation according to the topic space's distribution of topics. The classification of the web pages extracted the semantic features by combining them with the statistical features. Finally, in their results, they showed that they achieved better results [171]. Hu et al. proposed a new Chinese-spam identification method using the technology based on semantics-based text classification. The authors selected the related feature terms from the semantic meanings of text content. They selected feature terms and extracted semantic meanings by adding notes on the text layer by layer. After experimenting with their proposed method on a public Chinese-spam corpus[12], they identified that their proposed method performed well [172].

6) User's Browsing-Behaviour-Based Algorithms

Liu et al. developed the concept that users' browsing data can be used for web spam identification [77]. In their approach, they built a browsing graph G = (V, E, T, σ) where nodes (V) represented the web pages, transitions between web pages were represented with edges (E), *T* is the staying time on a web page, and the random jump probability is denoted with σ. In last, the significance for each web page is computed using the PR-Like algorithm. A unique aspect of their proposed technique is that as the time information is included in the user browsing data, their proposed solution uses the continuous-time Markov process as an underlying model. The working mechanism of the algorithm is as follows. Initially, for a web page *i*, it uses staying times $Z_1, \ldots, Zm_i$, which identify the diagonal element q_{ii} of the matrix Q = P′(t) as a solution to the optimisation issue:

$$[\left(\overline{Z}+\frac{1}{q_{ii}}\right)-\frac{1}{2}\left(S^2-\frac{1}{q_{ii}^2}\right)]^2 \longrightarrow min_{q_{ii}}, s.t. q_{ii} < 0 \qquad (22)$$

[12] https://en.wikipedia.org/wiki/Text_corpus

For Matrix Q, the non-diagonal elements are estimated as:

$$-\frac{q_{ij}}{q_{ii}} = \begin{cases} c\frac{\widetilde{w_{ij}}}{\sum_{k=1}^{N+1}\widetilde{w_{ij}}} + (1-c)\sigma_j, i \in V, j \in \tilde{V} \\ \sigma_j, \qquad i = N+1, j \in V \end{cases} \tag{23}$$

The other pseudo vertex N + 1 is added to graph G for modelling the teleportation effect and random jump probability is represented with σj. In the next step, using the matrix Q, the fixed distribution is calculated, which is proved to correspond to the fixed distribution for the continuous-time Markov process. A similar idea is studied by [173] in which a hyperlink-click graph is defined, which is a union of a query-page click-log-based and standard web graph and for performing the web pages ranking, they used random walks. Liu et al. also used user-behaviour data for spam detection [78]. Their strategy is based on two key observations. First, the spammers' main target is to achieve high ranking in a search-engine's result pages. Therefore, a significant amount of traffic is coming from search engines to spam websites. Second, users can quickly identify spam websites. Consequently, they are not spending much time on these spam websites and leave them considerably fast. Based on these two observations, three new features are proposed by the authors: the number of clicks and pageviews on a website per visit and the ratio of the visits from search engines. A machine-learning technique is proposed by Neria et al. for the identification of risky browsing behaviour and risky web pages. They used their interaction analysis method between two modules, naive users and risky web pages [174]. The feedback loop is implemented between these modules so that if a web page is opened to massive traffic from risky users, the "risk score" of the web page will increase.

Similarly, if the user is open to risky web pages, the user's "risk score" will increase. The authors obtained the real-world HTTP-logs dataset from the American toolbar company and tested their technique. After the experimental results, the authors claimed that involving the web pages and users in the feedback-learning process can increase the scoring accuracy and help identify spam web pages. Cooley also used the website browsing data for web spam identification [175].

VI. Comparison of Current Techniques

Every web spamming technique adds particular keywords to web pages for cheating the ranking algorithms of the search engines. In link-based spamming strategies, the spammers attempt to cheat the ranking algorithms by adding several links to particular pages designed for backlinks or they might place links from outstanding high-rank web pages. In hiding strategies, for instance, cloaking, the spammers provide the contents to regular visitors of site and web crawler; recognising this type of spam is a bit difficult because, in some scenarios, cloaking is legal. In one of these techniques, spammers utilise the web page features to fool search engines. Among all these techniques, defending against the hiding techniques is harder since spammers present different contents to web browsers and web crawlers. The comparison between the content submitted to the regular users and the content submitted to the web crawler is required to identify cloaking. This comparison is costly and time-consuming. Content-based spamming techniques negatively influence the search engines at most. Because spammers quickly add special terms or keywords invisibly or visibly to web pages and this technique is famous among the spammers. Moreover, if we search a keyword in any search engine (Google, Bing, Yahoo), we will first encounter the content-based spamdexing techniques.

VII. THE CHALLENGES IN SPAM IDENTIFICATION TECHNIQUES

In the literature review, we have explored many researchers from industry and academia who are working on techniques of identification and prevention of web spam. They proposed some outstanding strategies for anti-web-spam. However, there are still some open challenges to these anti-web-spam techniques. Some of the critical challenges are highlighted below:

- Due to the dynamic structure of the web, i.e., rapidly changing technology, spamming techniques are evolving and coming up with new spamming strategies every day by exploiting the weaknesses of currently used anti-web-spam techniques. Further research is required for the development of AI-based strategies that can detect and prevent the newly evolved spamming techniques.
- Most of the currently used spam-detection techniques currently try to overcome the spamdexing issues, but they mainly focus on link-based or content-based techniques. Further research is needed to develop hybrid or combined strategies to identify both link and content-based problems effectively.
- In the trust-modelling system, the users' trust is changing rapidly over time due to the involvement of social media networks and the user's experience. Hardly a few techniques are dealing with the dynamics of trust by differentiating between old and new tags. Further research considering trust dynamics can lead to much better modelling in a real-world application.
- Many existing techniques work with textual information and assume the monolingual environment. However, many people from different countries are developing websites and blogs in other languages and social media networks are also used by many people from different parts of the world. Therefore, many languages appear simultaneously on a web page, in comments and tags. There are high chances that some non-spam text might be considered spam in such conditions because of the language spam. Therefore, multilingualism is required to be incorporated in trust modelling to fix this issue.
- Most of the currently used techniques focus only on user-profile analysis and textual processing for spam detection. Simultaneously, useful information about the relevancy of the content can be retrieved from multimedia content features such as visual and audio content features. So, combining multimedia-content analysis with user-profile analysis and conventional-tag processing could be challenging.

VIII. BASIC PRINCIPLES FOR CONSTRUCTING THE SPAM DETECTION ALGORITHM

After the literature review and investigation of existing web spam detection algorithms, we identified certain rules commonly used for the construction of algorithms for spam detection and prevention. Some of the recognised rules are highlighted below:

- Usually, most of the content generated by the spammers are machine-generated in nature with the primary aim of manipulating search engines and getting the top rank in search engine results pages. This machine-generated spam content contains abnormal properties, for instance, meaningless articles without any useful information with many famous search terms used in it, a considerable number of duplicate links and content and rapid change in keywords, links, and other content.
- To exploit the weaknesses of ranking algorithms, web spammers usually build their link farms and link pyramids to boost their page rank and get the top position in a search-engine's result pages. Link forms and link pyramids have specific topologies that can be analysed using a web graph.

- Spammers mostly do the keywords research using different online free available tools, for instance, Google Trends, Keyword Shitter, and Google Correlate. Using these tools, they can quickly identify and target popular and high advertising value search terms.
- Based on different experiments, it has been observed that useful web pages often link to legitimate web pages. In contrast, spam web pages can connect to both. It has been seen that some level of semantic similarity exists among connected web pages, therefore, using a web graph for label smoothing is a beneficial strategy.
- Various spam-detection algorithms use the concept of trust and distrust propagation using several similarity measures, seed selection heuristics, and propagation strategies.
- The excess of nepotistic links negatively influences the performance of link-mining algorithms, so, there is an essential concept of link down weighting and removal.
- Page content and links are not the only sources of information and there are several other sources, for instance, HTTP requests, user behaviour, and semantics. For web-spam detection and prevention, smart feature engineering is essential.
- Smart feature engineering can increase the performance of spam detection and prevention, but the decent selection and proper training of a machine-learning model are also important.

IX. CONCLUSION

In this research work, we analysed the existing techniques used for the detection and prevention of web spam. This paper discussed all types of web spamming methods spammers are using for web spamming. We also discussed how web spam affects search engine providers and compel them to waste their useful processing and storage resources and how web spamming is affecting the user's trust. We tried to motivate academic and industrial researchers by highlighting the need for research in spam detection and prevention. Finally, we discussed the existing techniques fighting web spamming, critical challenges in spam identification and prevention, and the basic concepts behind the construction of various algorithms. In this research work, we tried to show the complete evolution of spam-detection algorithms describing how the research on spam detection started from a simple content-based spam-detection technique to more advanced traditional and non-traditional web-spam-detection strategies available today. During our work, we also found that most of the researchers came up with excellent spam-detection techniques, but research in the area cannot stop here due to critical challenges in the field and the evolving nature of web spamming. Researchers are still working in the field, and we are hopeful that they will come up with new techniques and strategies that will neutralise the adverse effects of web spam on people and businesses in the future.

REFERENCES

[1] A. Shahzad *et al.*, “Search Engine Optimization Techniques for Malaysian University Websites : A Comparative Analysis on Google and Bing Search Engine,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 8, no. 4, pp. 1262–1269, 2018.

[2] A. Somani and U. Suman, “Counter measures against evolving search engine spamming techniques,” in *ICECT 2011 - 2011 3rd International Conference on Electronics Computer Technology*, 2011, vol. 6, pp. 214–217, doi: 10.1109/ICECTECH.2011.5942084.

[3] E. Convey, “Porn sneaks way back on web,” *Bost. Her.*, vol. 28, 1996.

[4] M. Henzinger, R. Motwani, C. S.-A. S. Forum, and undefined 2002, “Challenges in web search engines,” *dl.acm.org*.

[5] J. Abernethy, O. Chapelle, … C. C. the 4th I. W. on, and U. 2008, "Witch: A new approach to web spam detection," *Citeseer*, 2008.

[6] M. Cutts, "https://googleblog.blogspot.com/2011/01/google-search-and-search-engine-spam.html, 2018," 2011. .

[7] "https://searchenginewatch.com/tag/adversarial-information-retrieval/, 2018." .

[8] Z. Gyongyi and H. Garcia-Molina, "Web spam taxonomy," 2005.

[9] Z. Gyöngyi, … H. G.-M. the 31st international conference on, and undefined 2005, "Link spam alliances," *dl.acm.org*.

[10] A. Shahzad *et al.*, "The Impact of Search Engine Optimization on The Visibility of Research Paper and Citations," *JOIV Int. J. Informatics Vis.*, vol. 1, no. 4–2, pp. 195–198, 2017.

[11] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the web.," 1999.

[12] N. Eiron, K. McCurley, J. T.-P. of the 13th international, and undefined 2004, "Ranking the web frontier," *dl.acm.org*.

[13] R. Jennings, "The global economic impact of spam," *Ferris Res.*, 2005.

[14] C. Silverstein, H. Marais, M. Henzinger, and M. Moricz, "Analysis of a very large web search engine query log," in *ACm SIGIR Forum*, 1999, vol. 33, no. 1, pp. 6–12.

[15] T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay, "Accurately interpreting clickthrough data as implicit feedback," in *ACM SIGIR Forum*, 2017, vol. 51, no. 1, pp. 4–11.

[16] L. Becchetti, C. Castillo, D. Donato, S. Leonardi, and R. A. Baeza-Yates, "Link-based characterization and detection of web spam.," in *AIRWeb*, 2006, pp. 1–8.

[17] C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri, "Know your neighbors: Web spam detection using the web topology," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007, pp. 423–430.

[18] A. A. Benczur, K. Csalogany, T. Sarlos, and M. Uher, "Spamrank–fully automatic link spam detection work in progress," in *Proceedings of the first international workshop on adversarial information retrieval on the web*, 2005, pp. 1–14.

[19] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proceedings of the 2008 international conference on web search and data mining*, 2008, pp. 219–230.

[20] S. Zhang and N. Cabage, "Search engine optimization: Comparison of link building and social sharing," *J. Comput. Inf. Syst.*, vol. 57, no. 2, pp. 148–159, 2017, doi: 10.1080/08874417.2016.1183447.

[21] T. R. Carraher and J. Palmer, "Search engine optimization using page anchors." Google Patents, Apr. 2017.

[22] S. Agrawal and O.- Page, "Discernment of Search Engine Spamming and Counter Measure for It," vol. 147, no. 8, pp. 8–11, 2016.

[23] D. Regalado *et al.*, *Gray Hat Hacking The Ethical Hacker's Handbook*. McGraw-Hill Education Group, 2015.

[24] D. Giomelakis and A. Veglis, "Investigating search engine optimization factors in media websites: the case of Greece," *Digit. Journal.*, vol. 4, no. 3, pp. 379–400, 2016.

[25] C. Castillo *et al.*, "A reference collection for web spam," *ACM SIGIR Forum*, vol. 40, no. 2, pp. 11–24, 2006, doi: 10.1145/1189702.1189703.

[26] A. H. Keyhanipour and B. Moshiri, "Designing a web spam classifier based on feature fusion in the Layered Multi-population Genetic Programming framework," *Proc. 16th Int. Conf. Inf. Fusion, FUSION 2013*, pp. 53–60, 2013.

[27] G. Collins, "Latest search engine spam techniques, Aug. 2004," *Online at http://www. sitepoint. com/article/search-enginespam-techniques*.

[28] A. Perkins, "White paper: The classification of search engine spam," *Online http//www. silverdisc. co. uk/articles/spam-classification*, 2001.

[29] S. Chhabra, R. Mittal, and D. Sarkar, "Inducing factors for search engine optimization techniques: A comparative analysis," in *Information Processing (IICIP), 2016 1st India International Conference on*, 2016, pp.

1–4.

[30] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly, “Detecting spam web pages through content analysis,” *Proc. 15th Int. Conf. World Wide Web - WWW ’06*, p. 83, 2006, doi: 10.1145/1135777.1135794.

[31] S. Robertson, H. Zaragoza, and M. Taylor, “Simple BM25 extension to multiple weighted fields,” in *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, 2004, pp. 42–49.

[32] G. Salton, A. Wong, and C.-S. Yang, “A vector space model for automatic indexing,” *Commun. ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[33] C. Zhai, “Statistical language models for information retrieval,” *Synth. Lect. Hum. Lang. Technol.*, vol. 1, no. 1, pp. 1–141, 2008.

[34] J. L. Neto, A. D. Santos, C. A. A. Kaestner, N. Alexandre, and D. Santos, “Document clustering and text summarization,” 2000.

[35] J. Ramos, “Using tf-idf to determine word relevance in document queries,” in *Proceedings of the first instructional conference on machine learning*, 2003, vol. 242, pp. 133–142.

[36] H. Zuze and M. Weideman, “Keyword stuffing and the big three search engines,” *Online Inf. Rev.*, 2013.

[37] M. D. Oskuie and S. N. Razavi, “A Survey of Web Spam Detection Techniques,” *Int. J. Comput. Appl. Technol. Res.*, vol. 3, no. 3, pp. 180–185, 2014, doi: 10.7753/IJCATR0303.1010.

[38] N. Spirin and J. Han, “Survey on web spam detection,” *ACM SIGKDD Explor. Newsl.*, vol. 13, no. 2, p. 50, 2012, doi: 10.1145/2207243.2207252.

[39] I. Drost and T. Scheffer, “Thwarting the nigritude ultramarine: Learning to identify link spam,” in *European Conference on Machine Learning*, 2005, pp. 96–107.

[40] N. Marres and E. Weltevrede, “Scraping the social? Issues in live social research,” *J. Cult. Econ.*, vol. 6, no. 3, pp. 313–335, 2013.

[41] J. A. Malcolm and P. C. R. Lane, “An approach to detecting article spinning,” 2008.

[42] D. M. Christopher, R. Prabhakar, and S. Hinrich, “Introduction to information retrieval,” *An Introd. To Inf. Retr.*, vol. 151, no. 177, p. 5, 2008.

[43] O. A. McBryan, “GENVL and WWWW: Tools for taming the web,” in *Proceedings of the first international world wide web conference*, 1994, vol. 341.

[44] J. M. Kleinberg, “Authoritative sources in a hyperlinked environment,” *J. ACM*, vol. 46, no. 5, pp. 604–632, 1999.

[45] S. Adalı, T. Liu, and M. Magdon-Ismail, “Optimal Link Bombs are Uncoordinated,” *Advers. Inf. Retr. Web*, p. 58.

[46] A. Mathur, “Spam Detection Techniques : Issues and Challenges,” *Int. J. Appl. Inf. Syst. (IJAIS)-ISSN 2249-0868.*, vol. 2013, no. Icwac, pp. 39–41, 2013.

[47] B. Wu and K. Chellapilla, “Extracting link spam using biased random walks from spam seed sets,” in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, 2007, pp. 37–44.

[48] V. N. Gudivada, D. Rao, and J. Paris, “Understanding search-engine optimization,” *Computer (Long. Beach. Calif).*, vol. 48, no. 10, pp. 43–52, 2015.

[49] K. Chellapilla and D. M. Chickering, “Improving Cloaking Detection using Search Query Popularity and Monetizability.,” in *AIRWeb*, 2006, pp. 17–23.

[50] B. Wu and B. Davison, “Cloaking and Redirection: A Preliminary Study.,” in *AIRWeb*, 2005, pp. 7–16.

[51] B. Wu and B. D. Davison, “Detecting semantic cloaking on the web,” in *Proceedings of the 15th international conference on World Wide Web*, 2006, pp. 819–828.

[52] J.-L. Lin, “Detection of cloaked web spam by using tag-based methods,” *Expert Syst. Appl.*, vol. 36, no. 4, pp. 7493–7499, 2009.

[53] K. Chellapilla and A. Maykov, “A taxonomy of JavaScript redirection spam,” in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, 2007, pp. 81–88.

[54] R. Smička, “Optimalizace pro vyhledávače-SEO,” *Jak zvýšit návštěvnost webu. Dubany Jasmínka*, 2004.

[55] N. Chachra, "Understanding URL Abuse for Profit." UC San Diego, 2015.

[56] C. Adams, K. LaRiviere, and J. Jones, "Method and system of optimizing a web page for search engines." Google Patents, Apr. 2017.

[57] R. a. Malaga, "Worst practices in search engine optimization," *Commun. ACM*, vol. 51, no. 12, p. 147, 2008, doi: 10.1145/1409360.1409388.

[58] K. Du, H. Yang, Z. Li, H.-X. Duan, and K. Zhang, "The Ever-Changing Labyrinth: A Large-Scale Analysis of Wildcard DNS Powered Blackhat SEO.," in *USENIX Security Symposium*, 2016, pp. 245–262.

[59] H. Gao, J. Hu, T. Huang, J. Wang, and Y. Chen, "Security issues in online social networks," *IEEE Internet Comput.*, vol. 15, no. 4, pp. 56–63, 2011.

[60] T. Urvoy, T. Lavergne, and P. Filoche, "Tracking Web Spam with Hidden Style Similarity.," in *AIRWeb*, 2006, pp. 25–31.

[61] A. Alarifi and M. Alsaleh, "Web spam: A study of the page language effect on the spam detection features," in *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, 2012, vol. 2, pp. 216–221.

[62] C.-V. Boutet, L. Quoniam, and W. S. R. Smith, "Towards active seo (search engine optimization) 2.0," *JISTEM-Journal Inf. Syst. Technol. Manag.*, vol. 9, no. 3, pp. 443–458, 2012.

[63] D. Giomelakis and A. Veglis, "Employing search engine optimization techniques in online news articles," *Stud. Media Commun.*, vol. 3, no. 1, pp. 22–33, 2015.

[64] A. V Ershov, "Context-based search visualization and context management using neural networks." Google Patents, Jan. 2009.

[65] P. Kent, *Search engine optimization for dummies*. John Wiley & Sons, 2012.

[66] P. Hayati, V. Potdar, A. Talevski, N. Firoozeh, S. Sarenche, and E. A. Yeganeh, "Definition of spam 2.0: New spamming boom," in *Digital Ecosystems and Technologies (DEST), 2010 4th IEEE International Conference on*, 2010, pp. 580–584.

[67] F. Radlinski, "Addressing malicious noise in clickthrough data," in *Learning to rank for information retrieval workshop at SIGIR*, 2007, vol. 2007.

[68] Z. Dou, R. Song, X. Yuan, and J.-R. Wen, "Are click-through data adequate for learning web search rankings?," in *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008, pp. 73–82.

[69] N. Daswani and M. Stoppelman, "The anatomy of Clickbot. A," in *Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets*, 2007, p. 11.

[70] Y. Peng, L. Zhang, J. M. Chang, and Y. Guan, "An effective method for combating malicious scripts clickbots," in *European Symposium on Research in Computer Security*, 2009, pp. 523–538.

[71] N. Immorlica, K. Jain, M. Mahdian, and K. Talwar, "Click fraud resistant methods for learning click-through rates," in *International Workshop on Internet and Network Economics*, 2005, pp. 34–45.

[72] P. Heymann, G. Koutrika, and H. Garcia-Molina, "Fighting spam on social web sites: A survey of approaches and future challenges," *IEEE Internet Comput.*, vol. 11, no. 6, 2007.

[73] A. Chandra, M. Suaib, and R. Beg, "Google Search Algorithm updates against web spam," *Dep. Comput. Sci. Eng. Integr. Univ.*, pp. 1–10.

[74] A. Benczúr, I. Bíró, K. Csalogány, and T. Sarlós, "Web spam detection via commercial intent analysis," in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, 2007, pp. 89–92.

[75] S. Webb, J. Caverlee, and C. Pu, "Predicting Web Spam with HTTP Session Information," *17th ACM Conf. Inf. Knowl. Manag.*, pp. 339–348, 2008, doi: 10.1145/1458082.1458129.

[76] M. Kolhe and D. Bhukte, "Data Mining for Web Spam Detection Analysis of Techniques," vol. 5, no. 10, pp. 2013–2016, 2016.

[77] Y. Liu *et al.*, "BrowseRank: letting web users vote for page importance," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 2008, pp. 451–458.

[78] Y. Liu, M. Zhang, S. Ma, and L. Ru, "User Behavior Oriented Web Spam Detection," pp. 1039–1040, 2008.

[79] S. Aggarwal, "Web Spam Detection using Timer with Ranking Technique," vol. 71, no. 18, pp. 17–20, 2013.

[80] J. Fdez-Glez, D. Ruano-Ordás, R. Laza, J. R. Méndez, R. Pavón, and F. Fdez-Riverola, "WSF2: A Novel Framework for Filtering Web Spam," *Sci. Program.*, vol. 2016, 2016, doi: 10.1155/2016/6091385.

[81] R. V Guha, "Detecting spam search results for context processed search queries." Google Patents, May 2013.

[82] D. Fetterly, M. Manasse, and M. Najork, "Detecting phrase-level duplication on the world wide web," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005, pp. 170–177.

[83] D. Fetterly, M. Manasse, and M. Najork, "On the evolution of clusters of near-duplicate web pages," in *Web Congress, 2003. Proceedings. First Latin American*, 2003, pp. 37–45.

[84] D. Fetterly, M. Manasse, and M. Najork, "Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages," in *Proceedings of the 7th International Workshop on the Web and Databases: colocated with ACM SIGMOD/PODS 2004*, 2004, pp. 1–6.

[85] Z. Gyongyi, ... H. G.-M. adversarial information retrieval, and undefined 2005, "Web spam taxonomy," *ilpubs.stanford.edu*.

[86] A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig, "Syntactic clustering of the web," *Comput. Networks ISDN Syst.*, vol. 29, no. 8, pp. 1157–1166, 1997.

[87] M. O. Rabin, "Fingerprinting by random polynomials," *Tech. Rep.*, 1981.

[88] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen, "Combating web spam with trustrank," in *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, 2004, pp. 576–587.

[89] A. Z. Broder, "Some applications of Rabin's fingerprinting method," in *Sequences II*, Springer, 1993, pp. 143–152.

[90] G. Mishne, D. Carmel, and R. Lempel, "Blocking Blog Spam with Language Model Disagreement.," in *AIRWeb*, 2005, vol. 5, pp. 1–6.

[91] D. Hiemstra, "Language models," *ACM Trans. Inf. Syst.*, pp. 1–6, 1980.

[92] M. Sydow, J. Piskorski, D. Weiss, and C. Castillo, "Application of machine learning in combating web spam." Submitted for publication in IOS Press, 2007.

[93] J. Piskorski, M. Sydow, and D. Weiss, "Exploring linguistic features for web spam detection: a preliminary study," in *Proceedings of the 4th international workshop on Adversarial information retrieval on the web*, 2008, pp. 25–28.

[94] A. Shahzad, H. Mahdin, and N. M. Nawi, "An Improved Framework for Content-based Spamdexing Detection."

[95] H. Ji and H. Zhang, "Analysis on the content features and their correlation of web pages for spam detection," *China Commun.*, vol. 12, no. 3, pp. 84–94, 2015, doi: 10.1109/CC.2015.7084367.

[96] Q. Zhang, D. Y. Wang, and G. M. Voelker, "DSpin: Detecting Automatically Spun Content on the Web," *Proc. 2014 Netw. Distrib. Syst. Secur. Symp.*, no. February, pp. 23–26, 2014, doi: 10.14722/ndss.2014.23004.

[97] N. El-Mawass and S. Alaboodi, "Data Quality Challenges in Social Spam Research," *J. Data Inf. Qual.*, vol. 9, no. 1, pp. 4:1--4:4, 2017, doi: 10.1145/3090057.

[98] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, no. Jan, pp. 993–1022, 2003.

[99] Y. Tian, G. M. Weiss, and Q. Ma, "A semi-supervised approach for web spam detection using combinatorial feature-fusion," in *Proceedings of the Graph Labelling Workshop and Web Spam Challenge at the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery*, 2007, pp. 16–23.

[100] B. Wu and B. D. Davison, "Identifying link farm spam pages," in *Special interest tracks and posters of the 14th international conference on World Wide Web*, 2005, pp. 820–829.

[101] J. Abernethy, O. Chapelle, and C. Castillo, "Graph regularization methods for Web spam detection," *Mach. Learn.*, vol. 81, no. 2, pp. 207–225, 2010, doi: 10.1007/s10994-010-5171-1.

[102] Q. Gan and T. Suel, “Improving web spam classifiers using link structure,” in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, 2007, pp. 17–20.

[103] E. Amitay, D. Carmel, A. Darlow, R. Lempel, and A. Soffer, “The connectivity sonar: detecting site functionality by structural patterns,” in *Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, 2003, pp. 38–47.

[104] R. Jennings, “Cost of spam is flattening–our 2009 predictions,” *Ferris Res.*, 2009.

[105] A. Berman and R. J. Plemmons, *Nonnegative matrices in the mathematical sciences*, vol. 9. Siam, 1994.

[106] D. Fogaras and B. Rácz, “Towards scaling fully personalized pagerank,” in *International Workshop on Algorithms and Models for the Web-Graph*, 2004, pp. 105–117.

[107] T. H. Haveliwala, “Topic-sensitive pagerank,” in *Proceedings of the 11th international conference on World Wide Web*, 2002, pp. 517–526.

[108] G. Jeh and J. Widom, “Scaling personalized web search,” in *Proceedings of the 12th international conference on World Wide Web*, 2003, pp. 271–279.

[109] D. Fogaras, “Where to start browsing the web?,” in *International Workshop on Innovative Internet Community Systems*, 2003, pp. 65–79.

[110] A. Y. Ng, A. X. Zheng, and M. I. Jordan, “Link analysis, eigenvectors and stability,” in *International Joint Conference on Artificial Intelligence*, 2001, vol. 17, no. 1, pp. 903–910.

[111] M. Bianchini, M. Gori, and F. Scarselli, “Inside pagerank,” *ACM Trans. Internet Technol.*, vol. 5, no. 1, pp. 92–128, 2005.

[112] A. N. Langville and C. D. Meyer, “Deeper inside pagerank,” *Internet Math.*, vol. 1, no. 3, pp. 335–380, 2004.

[113] V. Krishnan and R. Raj, “Web Spam Detection with Anti-Trust Rank.,” *AIRWeb*, vol. 6, pp. 37–40, 2006, [Online]. Available: http://www.ra.ethz.ch/cdstore/www2008/airweb.cse.lehigh.edu/2006/proceedings.pdf#page=45.

[114] B. Manaskasemsak and A. Rungsawang, “Web spam detection using trust and distrust-based ant colony optimization learning,” *Int. J. Web Inf. Syst.*, vol. 11, no. 2, pp. 142–161, 2015, doi: 10.1108/IJWIS-12-2014-0047.

[115] J. J. Whang, Y. S. Jeong, I. S. Dhillon, S. Kang, and J. Lee, “Fast Asynchronous Anti-TrustRank for Web Spam Detection,” 2018.

[116] L. Smitha, “Topical and Trust Based Page Ranking Using Automatic seed selection,” no. 2006, pp. 0–3, 2017, doi: 10.1109/IACC.2017.156.

[117] A. G. K. Leng, A. K. Singh, P. R. Kumar, and A. Mohan, “TPRank: Contend with web spam using trust propagation,” *Cybern. Syst.*, vol. 45, no. 4, pp. 307–323, 2014, doi: 10.1080/01969722.2014.887938.

[118] X. Zhang, Y. Wang, N. Mou, and W. Liang, “Propagating Both Trust and Distrust with Target Differentiation for Combating Link-Based Web Spam,” *ACM Trans. Web*, vol. 8, no. 3, pp. 1–33, 2014, doi: 10.1145/2628440.

[119] M. Sobek, “Pr0-Google’s Pagerank 0 Penalty. Accessed 25 February.” 2002.

[120] B. Davison, “Propagating trust and distrust to demote web spam,” 2006.

[121] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, “Propagation of trust and distrust,” in *Proceedings of the 13th international conference on World Wide Web*, 2004, pp. 403–412.

[122] Z. Gyongyi, P. Berkhin, H. Garcia-Molina, and J. Pedersen, “Link spam detection based on mass estimation,” in *Proceedings of the 32nd international conference on Very large data bases*, 2006, pp. 439–450.

[123] M. Iqbal, M. M. Abid, U. Waheed, and S. H. Alam Kazmi, “Classification of Malicious Web Pages through a J48 Decision Tree, aNaïve Bayes, a RBF Network and a Random Forest Classifier forWebSpam Detection,” 2017.

[124] M. R. C. Patil and M. V. R. Bhadane, “Survey on Web Spam Detection using Link and Content Based Features,” *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 4, no. 6, pp. 467–470, 2016.

[125] S. Singh and A. K. Singh, “Web-Spam Features Selection Using CFS-PSO,” *Procedia Comput. Sci.*, vol. 125, pp. 568–575, 2018.

[126] A. N. Tikhonov, V. I. Arsenin, and F. John, "Solutions of ill-posed problems (Vol. 14)," *Washington, DC Winst.*, 1977.

[127] G. Wahba, "Spline models for observational data. Society for Industrial and Applied Mathematics," 1990.

[128] B. Davison, "Topical locality in the web," in *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, 2000, pp. 272–279.

[129] S. Chakrabarti, *Mining the Web: Discovering knowledge from hypertext data*. Elsevier, 2002.

[130] D. Zhou, C. J. C. Burges, and T. Tao, "Transductive link spam detection," in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, 2007, pp. 21–28.

[131] Z. Cheng, B. Gao, C. Sun, Y. Jiang, and T.-Y. Liu, "Let web spammers expose themselves," in *Proceedings of the fourth ACM international conference on Web search and data mining*, 2011, pp. 525–534.

[132] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 359–392, 1998.

[133] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in neural information processing systems*, 2004, pp. 321–328.

[134] Z. Kou and W. W. Cohen, "Stacked graphical models for efficient inference in markov random fields," in *Proceedings of the 2007 SIAM International Conference on Data Mining*, 2007, pp. 533–538.

[135] G.-G. Geng, Q. Li, and X. Zhang, "Link based small sample learning for web spam detection," *Proc. 18th Int. Conf. World wide web*, pp. 1185–1186, 2009, doi: 10.1145/1526709.1526920.

[136] G. Geng, C. Wang, and Q. Li, "Improving web spam detection with re-extracted features," in *Proceedings of the 17th international conference on World Wide Web*, 2008, pp. 1119–1120.

[137] K. Bharat and M. R. Henzinger, "Improved Algorithms for Topic Distillation in a Hyperlinked Environment," in *ACM SIGIR Forum*, 2017, vol. 51, no. 2, pp. 194–201.

[138] S. Nomura, S. Oyama, T. Hayamizu, and T. Ishida, "Analysis and improvement of hits algorithm for detecting web communities," *Syst. Comput. Japan*, vol. 35, no. 13, pp. 32–42, 2004.

[139] R. Lempel and S. Moran, "SALSA: the stochastic approach for link-structure analysis," *ACM Trans. Inf. Syst.*, vol. 19, no. 2, pp. 131–160, 2001.

[140] G. O. Roberts and J. S. Rosenthal, "Downweighting tightly knit communities in world wide web rankings," *Adv. Appl. Stat.*, vol. 3, pp. 199–216, 2003.

[141] L. Li, Y. Shang, and W. Zhang, "Improvement of HITS-based algorithms on web documents," in *Proceedings of the 11th international conference on World Wide Web*, 2002, pp. 527–535.

[142] B. Davison, "Recognizing nepotistic links on the web," *Artif. Intell. Web Search*, pp. 23–28, 2000.

[143] W. da Costa Carvalho, A. L., Chirita, P. A., De Moura, E. S., Calado, P., & Nejdl, "Identifying link farm spam pages," in *In Proceedings of the 15th international conference on World Wide Web*, pp. 73–82.

[144] B. Wu and B. D. Davison, "Undue influence: Eliminating the impact of link plagiarism on web search rankings," in *Proceedings of the 2006 ACM symposium on Applied computing*, 2006, pp. 1099–1104.

[145] H. Zhang, A. Goel, R. Govindan, K. Mason, and B. Van Roy, "Making eigenvector-based reputation systems robust to collusion," in *International Workshop on Algorithms and Models for the Web-Graph*, 2004, pp. 92–104.

[146] R. A. Baeza-Yates, C. Castillo, V. López, and C. Telefónica, "Pagerank Increase under Different Collusion Topologies.," in *AIRWeb*, 2005, vol. 5, pp. 25–32.

[147] G. Pandurangan, P. Raghavan, and E. Upfal, "Using pagerank to characterize web structure," *Internet Math.*, vol. 3, no. 1, pp. 1–20, 2006.

[148] M. Egele, C. Kolbitsch, and C. Platzer, "Removing web spam links from search engine results," *J. Comput. Virol.*, vol. 7, no. 1, pp. 51–62, 2011, doi: 10.1007/s11416-009-0132-6.

[149] V. M. Prieto, M. Álvarez, and F. Cacheda, "SAAD, a content based Web Spam Analyzer and Detector," *J. Syst. Softw.*, vol. 86, no. 11, pp. 2906–2918, 2013, doi: 10.1016/j.jss.2013.07.007.

[150] K. L. Goh, R. K. Patchmuthu, and A. K. Singh, "Link-based web spam detection using weight properties," *J. Intell. Inf. Syst.*, vol. 43, no. 1, pp. 129–145, 2014, doi: 10.1007/s10844-014-0310-y.

[151] R. K. Roul, S. R. Asthana, and M. I. T. Shah, “Detection of spam web page using content and link-based techniques : A combined approach,” vol. 41, no. 2, pp. 193–202, 2016.

[152] N. Dai, B. Davison, X. Q.-P. of the 5th international workshop, and undefined 2009, “Looking into the past to better classify web spam,” *dl.acm.org*.

[153] S. Webb, J. Caverlee, and C. Pu, “Characterizing Web Spam Using Content and HTTP Session Analysis.,” 2007.

[154] C. Chen, Y. Wang, J. Zhang, Y. Xiang, W. Zhou, and G. Min, “Statistical features-based real-time detection of drifted twitter spam,” *IEEE Trans. Inf. Forensics Secur.*, vol. 12, no. 4, pp. 914–925, 2017.

[155] M. Chatterjee and A. S. Namin, “Detecting Web Spams Using Evidence Theory,” *2018 IEEE 42nd Annu. Comput. Softw. Appl. Conf.*, vol. 01, pp. 695–700, 2018, doi: 10.1109/COMPSAC.2018.10321.

[156] B. Zhou, J. Pei, and Z. Tang, “A Spamicity Approach to Web Spam Detection,” *Proc. 2008 SIAM Int. Conf. Data Min.*, pp. 277–288, 2008, doi: 10.1137/1.9781611972788.25.

[157] B. Zhou and J. Pei, “Link spam target detection using page farms,” *ACM Trans. Knowl. Discov. from Data*, vol. 3, no. 3, p. 13, 2009.

[158] B. Zhou and J. Pei, “OSD: An online web spam detection system,” in *In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD*, 2009, vol. 9.

[159] M. Agrawal and R. L. Velusamy, “Unsupervised spam detection in hyves using SALSA,” in *Proceedings of the 4th International Conference on Frontiers in Intelligent Computing: Theory and Applications (FICTA) 2015*, 2016, pp. 517–526.

[160] S. Wei and Y. Zhu, “Cleaning Out Web Spam by Entropy-Based Cascade Outlier Detection,” in *International Conference on Database and Expert Systems Applications*, 2017, pp. 232–246.

[161] C.-C. Hsu, Y.-A. Lai, W.-H. Chen, M.-H. Feng, and S.-D. Lin, “Unsupervised Ranking using Graph Structures and Node Attributes,” in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, 2017, pp. 771–779.

[162] S. Sedhai and A. Sun, “Semi-supervised spam detection in Twitter stream,” *IEEE Trans. Comput. Soc. Syst.*, vol. 5, no. 1, pp. 169–175, 2018.

[163] R. K. Zwicky, “Click fraud detection.” Google Patents, Oct. 2015.

[164] Y. Wang, M. Ma, Y. Niu, and H. Chen, “Spam double-funnel: Connecting web spammers with advertisers,” *InProceedings 16th Int. Conf. World Wide Web*, pp. 291–300, 2007, doi: 10.1145/1242572.1242612.

[165] R. Bhattacharjee and A. Goel, “Algorithms and incentives for robust ranking,” in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, 2007, pp. 425–433.

[166] X. Li, M. Zhang, Y. Liu, S. Ma, Y. Jin, and L. Ru, “Search engine click spam detection based on bipartite graph propagation,” in *Proceedings of the 7th ACM international conference on Web search and data mining - WSDM ’14*, 2014, pp. 93–102, doi: 10.1145/2556195.2556214.

[167] D. Vasumati, M. S. Vani, R. Bhramaramba, and O. Y. Babu, “Data Mining Approach to Filter Click-spam in Mobile Ad Networks,” in *Int’l Conference on Computer Science, Data Mining & Mechanical Engg.(ICCDMME’2015) April*, 2015, pp. 20–21.

[168] H. Xu, D. Liu, A. Koehl, H. Wang, and A. Stavrou, “Click fraud detection on the advertiser side,” in *European Symposium on Research in Computer Security*, 2014, pp. 419–438.

[169] E. M. Redmiles, N. Chachra, and B. Waismeyer, “Examining the Demand for Spam: Who Clicks?,” *Proc. CHI*, pp. 1–10, 2018, doi: 10.1145/3173574.3173786.

[170] R. Oentaryo *et al.*, “Detecting click fraud in online advertising: a data mining approach,” *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 99–140, 2014.

[171] J. Wan, M. Liu, J. Yi, and X. Zhang, “Detecting spam webpages through topic and semantics analysis,” *GSCIT 2015 - Glob. Summit Comput. Inf. Technol. - Proc.*, 2015, doi: 10.1109/GSCIT.2015.7353328.

[172] W. Hu, J. Du, and Y. Xing, “Spam filtering by semantics-based text classification,” in *Advanced Computational Intelligence (ICACI), 2016 Eighth International Conference on*, 2016, pp. 89–94.

[173] B. Poblete, C. Castillo, and A. Gionis, “Dr. searcher and mr. browser: a unified hyperlink-click graph,” in

Proceedings of the 17th ACM conference on Information and knowledge management, 2008, pp. 1123–1132.

[174] M. Ben Neria, N.-S. Yacovzada, and I. Ben-Gal, "A Risk-Scoring Feedback Model for Webpages and Web Users Based on Browsing Behavior," ACM *Trans. Intell. Syst. Technol.*, vol. 8, no. 4, pp. 1–21, 2017, doi: 10.1145/2928274.

[175] S. Cooley, "Systems and methods for associating website browsing behavior with a spam mailing list." Google Patents, Jan. 2015.