An Improved Framework for Content-based Spamdexing Detection

Asim Shahzad[1], Hairulnizam Mahdin[*2], Nazri Mohd Nawi[3]
Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia

***Abstract*—To the modern Search Engines (SEs), one of the biggest threats to be considered is spamdexing. Nowadays spammers are using a wide range of techniques for content generation, they are using content spam to fill the Search Engine Result Pages (SERPs) with low-quality web pages. Generally, spam web pages are insufficient, irrelevant and improper results for users. Many researchers from academia and industry are working on spamdexing to identify the spam web pages. However, so far not even a single universally efficient method is developed for identification of all spam web pages. We believe that for tackling the content spam there must be improved methods. This article is an attempt in that direction, where a framework has been proposed for spam web pages identification. The framework uses Stop words, Keywords Density, Spam Keywords Database, Part of Speech (POS) ratio, and Copied Content algorithms. For conducting the experiments and obtaining threshold values WEBSPAM-UK2006 and WEBSPAM-UK2007 datasets have been used. An excellent and promising F-measure of 77.38% illustrates the effectiveness and applicability of proposed method.**

***Keywords*—Information retrieval; Web spam detection; content spam; pos ratio; search spam; Keywords stuffing; machine generated content detection**

I. Introduction

Spamdexing or web spam is described as " an intentional act that is intended to trigger illegally favorable importance or relevance for some page, considering the web page's true significance" [1]. Studies by different researchers in the area show that on Web at least twenty percent of hosts are spam [2]. Spamdexing is widely recognized as one of the most significant challenges to web SEs [3]. One of the important current problems of SEs is spamdexing because spamdexing heavily reduces the quality of the search engine's results. Many users get annoyed when they search for certain content and ended up with irrelevant content because of web spam. Due to the unprecedented growth of information on the World Wide Web (WWW), the available size of textual data has become very huge to any end user. According to the most recent survey by WorldWideWebSize, the web is consisting of 5.39 billion1 pages. To the web corpus, thousands of pages are being added every day and out of all these web pages several are either spam or duplicate web pages [3]. Web spammers are taking the benefits from internet users by dragging them to their web pages using several smart and creative spamming methods. The purpose of building a spam web page is to deceive the SE in such a way that it delivers those search results which are irrelevant and not beneficial to the web user. The ultimate aim of web spammers is to increase their web page's rank in SERPs. Besides that, spamdexing also has an economic impact because web pages with higher rank can get huge free advertisements and huge web traffic volume. During the past few years, researchers are working hard to develop the new advanced techniques for identification of fraudulent web pages but, spam techniques are evolving also, and web spammers are coming up with new spamming methods every day [4]. Research in the area of web spam detection and prevention has become an arms race to fight an opponent who consistently practices more advanced techniques [4]. If one can recognize and eliminate all spam web pages, then it is possible to build an efficient and robust Information Retrieval System (IRS). Efficient SEs are needed which can produce promising and high-quality results according to the user search query. The next task is to arrange retrieved pages by the content or semantic similarity between retrieved web pages and the search query entered by the user. Finally, the arranged pages are presented to the user. There are many adverse effects of spamdexing on both search engine and end user [5]. Spam web pages not only waste the time but also waste the storage space. As SE needs to store and index a huge number of web pages, so more storage space is required. Furthermore, when SE requires to search web pages on the bases of the user's query, it will search in the huge corpus and therefore more time is required. Due to this, it diminishes the effectiveness of the SE and reduces the trust of the end user on SE [6].

To get over the web spam attacks the improvement in anti-web-spam methods is essential. All the techniques which can be used to get an undeservedly high rank are kind of spamdexing or web spam. Generally, there are three types of spamdexing, Content Spam, Link Spam, and Cloaking. Cloaking is a spamdexing method in which the content offered to the end user is different from that presented to the SE spider [7]. However, the most common types of spamdexing are content and link spam. Content spam is the one studied in this research work. Davison [8] defined the link spam as ― the connections among various web pages that are present for a purpose other than merit. In link spam, web spammers are creating the link structure for taking the benefits from link-based ranking algorithms, for instance, PageRank, it will assign the higher rank to a web page if other highly ranked web pages are pointing to the web page with backlinks. Content spam is consisting of all those methods in which web spammers are changing the logical view that an SE has over

[1] https://www.worldwidewebsize.com/
*Corresponding Author

the web page contents [1], it is the most common type of spamdexing [9]. This spamdexing technique is popular among the web spammers because of several SEs are using the information retrieval models (IRM) for instance, statistical language model [10], vector space model [11], BM25 [12] and probabilistic model which are applied to the web page's content for ranking the web page. Spammers try to utilize the vulnerabilities of these models for manipulating the content of target web page [13]. For instance, using important keywords several times on a target web page and increasing the keywords frequencies, copying the content from various good web pages, using the machine-generated content on target web page, and putting all dictionary words on the page and then changing the color of text similar to the background color so users can not see the dictionary words on target page and only visible to SEs spiders are some methods which web spammers are using for getting higher page rank in the SERPs [14]. Generally, content spam can be divided into five subtypes based on the structure of the page. These subtypes are Body spam, Anchor text spam, Meta-tag spam, URL spam, and Title spam. There are a number of spamming methods which are targeting the different algorithms used in SEs [15].

The focus of this article is content spam detection, in this work, a framework has been proposed to detect spam web pages by using content-based techniques. In the proposed approach, stop words, keywords density, Keywords database, part of speech (POS) ratio test, and copied content algorithms are used to detect the spam web pages. For the experimental purpose, WEBSPAM- UK2007 and WEBSPAM-UK2006 datasets are used. The experimental results with an encouraging F-measure demonstrate the applicability and effectiveness of the proposed improved framework as compared to other already existing techniques.

II. Literature Review

In the most recent years, several content-based spamdexing identification methods are proposed by the researchers during the spamdexing challenge [13]. Ntoulas et. al [3] proposed some content features. Their research work showed the text compressibility and HTML based characteristics which identify the content spam from normal web pages. In the research work done by Piskorski et. al [16], they explored a huge number of linguistic features. For text classification tasks the Latent Dirichlet Allocation (LDA) [17] is well known. For spamdexing detection, Biro et al. modified the LDA, they did a lot of research and proposed the linked LDA [18] and multi-corpus LDA [19] models.

For content-based analysis Ntoulas et. al. [3] used the decision tree classifier for identification of spam web pages. Many features were proposed by them for instance, the average length of words, the number of words, anchor text amount and the portion of visible content within in web page. A group of researchers proposed the combinatorial feature fusion and semi-supervised method for identification of spam web pages [20]. For a host, their produced Term Frequency-Inverse Document Frequency (TF-IDF)[2] feature vectors over a hundred pages are all sparse vectors. For exploiting the unlabeled samples, they used the semi-supervised learning, and for creating the new features and reducing the TF-IDF content-based features they used the combinatorial feature fusion method. Empirical results prove the effectiveness of their method.

The most fundamental work on content-based spamdexing detection algorithms is done by Fetterly et al [3], [21]–[23]. In [23] they suggest that using statistical analysis the spam web pages can be classified easily. Because generally, web spammers are generating the spam web pages automatically by adopting the phrase stitching and weaving methods [1], these spam pages are designed for SE spiders instead of real human visitors, so these web pages show the abnormal behavior. Researchers identified that there are a huge number of dashes, dots, and digits in the URL of spam web pages and the length of URL is also exceptional. During their research work, they identify that out of 100 longest observed host-names 80 were pointing to the adult contents, and 11 of them were referring to the financial credit related web pages. They also observe that web pages themselves have the duplicating nature - spam web pages hosted by the same host almost contains the same content with very little variance in word count. Another very fascinating finding of this research was they identified that spammers are changing the content on these spam pages very quickly. Specifically, they kept a close eye on content changing feature and observed the changes on these spam pages for a specific host on weekly bases. They come up with the results that 97.2% of the most active spam hosts can be identified on the bases of this one feature only. They proposed several other features in this research work all other features can be seen in the research article [23].

In another study conducted by them [21], [22], they worked on content duplication and identified that bigger clusters with the identical content are actually spam. For the identification of duplicate content and such clusters they applied the shingling method [24] which is based on Rabin-finger-prints [25], [26]. Initially, they used a primitive polynomial (PA), to fingerprint every *n* words on a web page, secondly, they used a different primitive polynomial (PB), they fingerprint every token from the initial step using extension transformation and prefix deletion, then every string which is obtained from the second step they applied *m* different fingerprint functions on it and hold the tiniest of *n* resulting values for every *m* fingerprint functions. Lastly, the document is a container of *m* fingerprints, so they used transitive closure of the near-duplicate relationship to perform the clustering. In another research study [3], they did some more work and come up with a few more content-based features. Ultimately, they combined all these features in a classification model within bagging frameworks, C4.5 and boosting. For boosting of 10 C4.5 trees they reported 97.8% true negative and 86.2% true positive rates. Another work [27] done by a group of researchers studied the machine learning models and several features, in their research they defined thoroughly that how machine learning models and several features can help in spamdexing identification. They concluded excellent classification results using easy to calculate content features, RandomForest, LogitBoost, and state of the art learning models. They also revealed that global

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

and computationally demanding features such as PageRank (PR)[3] can help just a bit in quality enhancement. Thus, the researchers claimed that a proper and careful selection of a machine learning model is critical.

To identify the script generated spam web pages Urvoy et. al [28] introduces the features which are based on HTML page structure. They used a very different and non-traditional method for data preprocessing, they removed all the content of the web page and only kept the layout of web page. Therefore, they identified the web page duplication by examining the layout of web page instead of content. Fingerprinting method [25], [26] is applied by them, followed by clustering to identify the groups of spam web pages which are structurally near-duplicate. [29] proposed a method for spamdexing identification in blogs by matching the language models [30] for web pages and blog comments, linked from these comments.

III. Understanding the Content Spam

We believe that it is impossible to tackle the spamdexing without complete knowledge of its working mechanism. The easy target of content-based spamdexing is text relevance algorithms such as TF.IDF [1] and BM25 [12], spammers easily can exploit the weaknesses of these algorithms. These types of algorithms are an easy target for spammers because these algorithms are exposed to content-based spam due to a powerful correlation between document relevance and the number of query words present in the text [31]. Usually, web spammers are using the content-based spamdexing in doorways – websites and web pages which are purposely designed for redirecting and attracting the web traffic [32]. Doorways can only perform effectively if these pages reach the top of SERPs. Normally web spammers like to build hundreds of doorway web pages, and they optimize each doorway web page for a specific keyword to maximize the volume of web traffic collected [31]. To increase the effectiveness of content-based spamdexing there are some requirements[4] which content-based spamdexing must satisfy.

- Content spam pages must be created in hundreds or even thousands. As spammers need to design thousands of spam web pages so these pages can create the content automatically and it will have a lot of spellings error in it [33].
- Each spam web page should increase the text relevancy for a specific search keyword. Therefore, web spammers have a few options for producing the content for their spam web pages. So, spammer can copy the content from other websites [34].
- Each spam page must have designed for generating the profit. Because normally spam web pages are designed for advertisement only with very little and irrelevant content [34].
- Spam web pages do not provide the relevant content to the web users who are browsing these web pages, these spam web pages only targeting the SE spiders [35].
- These web pages must have several irrelevant links and keywords even if these pages are including the real contents [35].
- Must automatically redirect the web users to irrelevant web pages, for instance, a web page which is entirely different from what is expected based on URL, search results or/and anchor text. Sometimes, it can also offer Search Engine Optimization (SEO) services, affiliate links, and random link exchange [36].
- Spam web pages must hide the text using the cloaking techniques, and they should be optimized heavily for search engines.

All web pages can also be categorized as web spam which are only displaying the catalogs of some products but in reality, they are redirecting the users to other traders without giving the additional value. Generating the text automatically is a tough job and there is no satisfactory and good technique available for this yet [37]. There are multiple levels of consistency in natural text, therefore, it is very difficult to follow all at once [38]. Several characteristics of natural texts are distinguished by the researchers during automatic text generation tasks, for instance, automatic document summarization. In different experiments conducted by the researchers, they proved that even very good and highly specialized text generation algorithms [39] didn't perform well and score little in many of these measure [40]. The consistency levels consist of topical consistency, local coherence, logical structure of the document, local coherence etc. In the proposed framework different important components are used for spam identification and tried to make it harder for web spammers to generate the spam web pages.

IV. Data Preprocessing and Experiments for Calculating the Threshold Values

To obtain the suitable threshold values well-known spamdexing datasets WEBSPAM-UK2006 and WEBSPAM-UK2007[5] are used. The WEBSPAM-UK2006 and WEBSPAM-UK2007 collections are based on a crawl of the .uk domain made on May 2006 and May 2007 respectively by the Laboratory of Web Algorithmics[6], Università degli Studi di Milano with the support of the DELIS EU - FET research project. Both datasets are labeled by a group of volunteers. WEBSPAM-UK2006 is consist of 11,402 hosts in total, out of which only 7,473 are labeled. WEBSPAM-UK2007 is consist of 114,529 hosts in total, out of which only 6,479 are labeled. Both datasets divide the web pages into testing and training sets with both non-spam and spam labels. To get optimized threshold values we manually selected 5000 webpages labeled as non-spam/spam by humans. Some content-based features for example Stop words, Keywords, Spam Keywords, Part of Speech and Duplicate Content were extracted for content-based spamdexing identifications. Finally, through different experiments, the most appropriate threshold values were obtained that provide the fewest false positive ratios and high F-measure.

[3] https://www.geeksforgeeks.org/page-rank-algorithm-implementation/
[4] http://chato.cl/webspam/datasets/uk2007/guidelines/
[5] http://chato.cl/webspam/datasets/
[6] http://law.di.unimi.it/

A. Stop words Threshold Value

Commonly used words (for instance, "a", "the", "in", "that", "an") are known as Stop words. Usually, programmers programmed search engines to ignore Stop words during indexing the records for searching. Stop words are considered irrelevant for searching purpose because Stop words frequently occur in natural language. To save time and space Stop words are dropped during indexing and ignored at search time. Spammers are taking advantage of this, to get a higher rank on Search Engine Result Pages (SERPs) they are generating the machine-generated articles which contain a high frequency of repeated keywords with less or no Stop words in the article. Using Stop words spamdexing can be detected. We are considering this feature for content-based spamdexing detection. To calculate the threshold value, approximately three thousand human labeled non-spam web pages were selected manually. A script has been used for Stop words identification and counting on each non-spam web page chosen for this experiment. The ratio of stop words on each web page is calculated using equation (1).

$$RSW\ on\ Wpi = \frac{No\ of\ SW\ on\ Wp_i}{Total\ number\ of\ words\ in\ Wp_i} \quad (1)$$

Where *RSW* represents the Ratio of Stop words, Wp_i is any webpage, and *SW* represents the Stop words. Finally, the average ratio of all Stop words is calculated and used as a standard threshold value for Stop words. Average Ratio = Sum of ratios of all Stop words / Total number of pages.

$$Stopwords\ Threshold\ Value = \frac{RSW_1+RSW_2+\cdots+RSW_i}{Wp_1+Wp_2+\cdots+Wp_i} \quad (2)$$

B. Keywords Threshold Value

Keyword frequency represents the percentage of times a phrase or Keyword appears on a webpage compared to the total number of words on the webpage. In search engine optimization context, keyword frequency can be used to check if a web page is relevant to a specific keyword or phrase. Some of the search engine optimization experts are saying that the optimal keywords density is unknown until Google or any other big search engine reveals it. And several search engine optimization experts think that the best keyword frequency is one to three percent and more could be regarded as search spam. After reading different research articles, search engine's guidelines and consulting the SEO experts, we found a lot of different conflicting opinions on ideal keyword frequency ratio. So we decided to calculate our threshold value for keywords frequency ratio through experiments. Around three thousand humans labeled non-spam web pages were manually selected from the training data set for calculating the threshold value. A script has been used for identification and storing of Keywords from every webpage Wp_i on a file. After Keywords identification, Keywords frequency test is performed. This test is used to determine the frequency (in percentage) of each distinct Keyword *KW* on a webpage Wp_i, i.e., the percentage of occurrences of every distinct Keyword compared to all other keywords on a webpage Wp_i. The steps given below shows how Keywords frequency test is performed on a webpage Wp_i.

1) Identify the Keyword frequency of every distinct keyword on a webpage Wp_i. The Keyword frequency of a distinct Keyword KW_i on webpage Wp_i is defined as:

$$T_{kw} = KW_1 + KW_2 + KW_3 + \cdots KW_i$$

$$KW_f(KW_i) = \frac{No\ of\ times\ KW_i\ appears\ on\ Wp_i}{T_{kw}\ on\ Wp_i} \quad (3)$$

Whereas KW_i represents the Keyword, Wp_i represents Webpage, T_{kw} is total number of keywords on Wp_i, and KW_f is keyword frequency.

2) Identify the Keyword Phrase frequency of every Keyword Phrase on a webpage Wp_i. The Keyword Phrase frequency of a distinct Keyword Phrase KW_p on webpage Wp_i is defined as:

$$KW_f(KW_p) = \frac{No\ of\ times\ KW_i\ appears\ on\ Wp_i * N_{wp}}{T_{kw}\ on\ Wp_i} \quad (4)$$

Whereas KW_p represents the Keyword Phrase, and N_{wp} is the number of words in a phrase.

3) Finally, after obtaining the frequency of every Keyword on all webpages selected for the experiment, the average frequency of all keywords is calculated to get a standard threshold value.

$$Keywords\ Threshold\ Value = \frac{KW_f(KW_1)+KW_f(KW_2)+\cdots+KW_f(KW_i)}{Total\ number\ of\ KW_f} \quad (5)$$

C. Spam-Keywords Threshold Value

Spam Keywords can play a significant role in spamdexing detection and prevention. Web spammers are excessively using Spam Keywords on their websites, pages, newsletters, and emails. Spam Keywords can be categorized in different categories such as personal, general, financial, medical, free offers, sense of urgency, exaggerated claims, etc. The Spam Keywords are used for content-based spamdexing detection in the proposed improved framework. The steps given below shows how the threshold value for Spam Keywords is calculated.

1) The initial step was to create a database of Spam Keywords. Several researchers, SEO experts, and search engines already identified the Spam Keywords spammers are commonly using in their content. After consulting different experts in the field and checking the existing well-known Spam Keywords Databases[7, 8, 9] , a more significant Spam Keywords database was created.

2) For calculating the threshold value, we manually selected approximately three thousand human labeled non-spam web pages from the dataset. A script has been used to identify all Spam Keywords using Spam Keywords database on all webpages selected for calculating the threshold value.

[7] https://www.automational.com/spam-trigger-words-to-avoid/

[8] https://blog.hubspot.com/blog/tabid/6307/bid/30684/the-ultimate-list-of-email-spam-trigger-words.aspx

[9] https://help.emarsys.com/hc/en-us/articles/115005000225-Known-spam-keywords

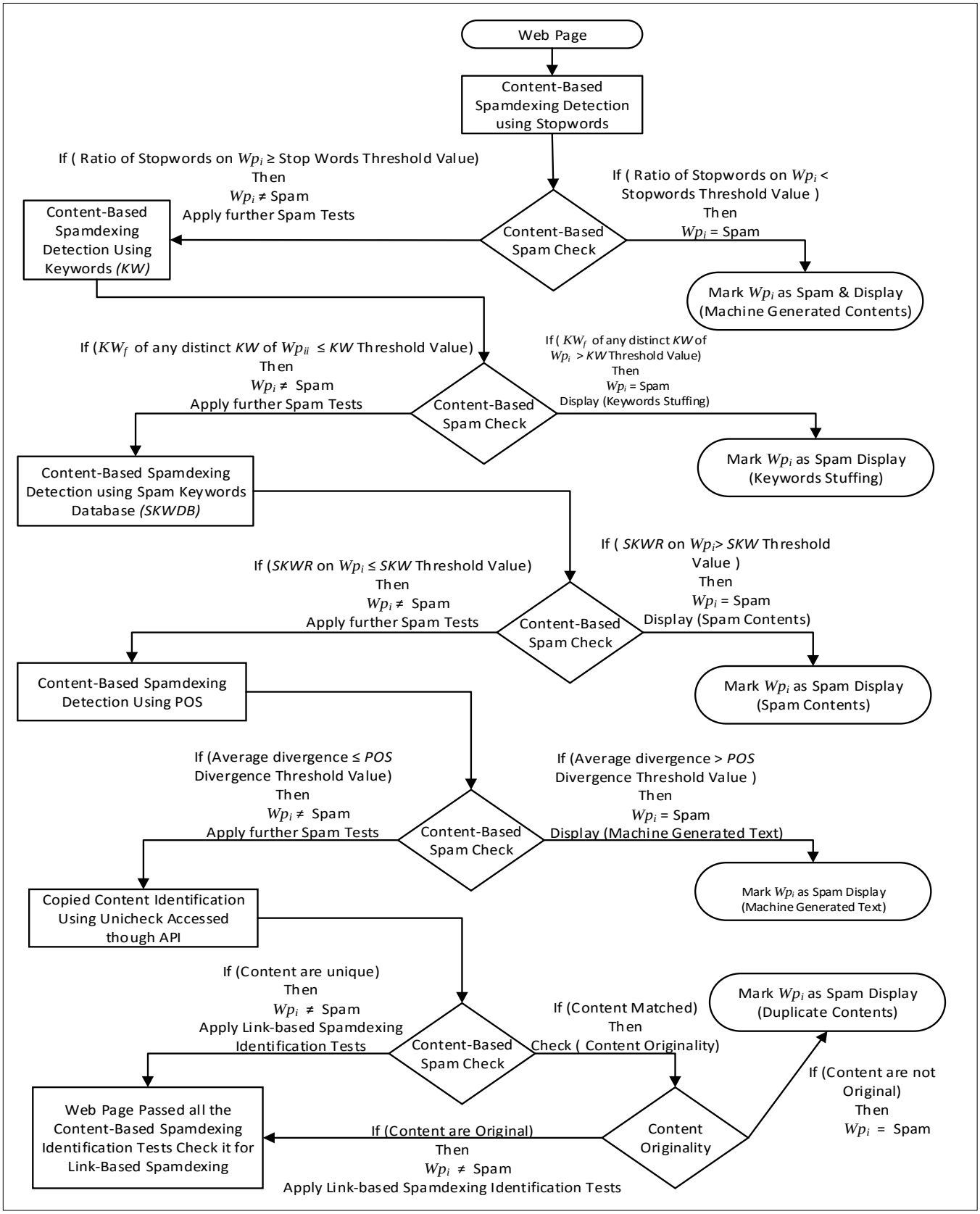

Fig. 1. Improved Framework for Content-based Spamdexing Detections.

3) After Spam Keywords identification, the ratio of Spam Keywords is calculated on each page using equation (6).

$$SKWR = \frac{T_{SKW}}{T_{kw\ on\ Wp_i}} \quad (6)$$

Where *SKWR* represents the Spam Keywords Ratio, and T_{SKW} represents the total number of spam keywords on webpage Wp_i.

4) Finally, after obtaining the Spam Keywords Ratio on all webpages selected for the experiment, the average Spam Keywords Ratio was calculated to get a standard threshold value.

$$Spam\ Keywords\ Threshold\ Value = \frac{SKWR_1 + SKWR_2 + \cdots + SKWR_i}{Total\ number\ of\ SKWR} \quad (7)$$

D. Part of Speech (POS) Threshold Value

Words are the smallest elements in the natural language that have unique meanings. These words can be categorized into various types based on their functions and use. In the English language, there are eight significant parts of speech: adjective, verb, adverb, interjection, conjunction, preposition, noun and pronoun. There is a standard ratio for each part of speech in the English Language. A researcher Eric Lease Morgan performed the experiments and calculated the standard ratio of occurrence of each grammatical form in English. For the standard ratio figures, the website[10] has been referred.

E. Copied Content

Copied content appears in more than one place on the Internet. For search engines, it is difficult to decide which version of the content is more suitable to the search query. To get high rank on SERPs usually, web spammers are copying the content from other good websites and using that content on their spam pages without or very little change in the content. A custom script has been used for the identification of copied content on the Internet. This script is using Unicheck API[11], which allowed to integrate Unicheck into the content workflow for instant identification of originality of the content as it enters the proposed system.

V. Content-based Spamdexing Detection Framework

In this section, an improved framework for content-based spamdexing detection is presented. Figure 1 is showing an overview of the proposed framework. This framework is using five different methods, and every method is using a unique feature for identification of content-based spamdexing. The methods are as follows:

1) Content-Based Spamdexing Detection using Stop words.

2) Content-Based Spamdexing Detection using Keywords.

3) Content-Based Spamdexing Detection using Spam Keywords Database (SKWDB).

4) Content-Based Spamdexing Detection Using POS.

5) Copied Content Identification Using Unicheck.

All five techniques are discussed in detail in the following subsections.

A. Spamdexing Detection using Stop Words

The first method is using stop words feature for identification of machine-generated text. A custom script has been used, which is accepting web page content as input and generating two separate output files (stop words and keywords file). Table 1 is showing an example of how this script works.

After obtaining the stop words file, it will count the number of stop words in the file for calculating the ratio of stop words on Wp_i using equation (1). Based on the stop words ratio on Wp_i, the Wp_i will classify into one of the two different categories specified below.

1) Category 1 (Low Stop Words Ratio): For generating the content quickly, web spammers are using the machine-generated content. After keywords research using different tools, they are producing the machine-generated articles [41] with no or very little, randomly inserted stop words in these articles. If the ratio of stop words on webpage Wp_i is lower than stop words threshold value (determined by experiment in sub-section 4.1) then web page $Wp_i \in spamdexing$.

By using the stop words ratio, the proposed method can efficiently identify the machine-generated web pages. Figure 2 is an example of this type of spam web page.

TABLE. I. The Example of the Working Mechanism of Custom Script

Input	*Output*	
Web Page Content	*Keywords file*	*Stop words file*
UTHM is the best University in Malaysia.	UTHM, best, University, Malaysia	is, the, in
It is the public sector University in Batu Pahat.	public, sector, University, Batu Pahat	it, is, the, in
The motto of UTHM is We produce professionals.	motto, UTHM, produce, professionals	the, of, is, we

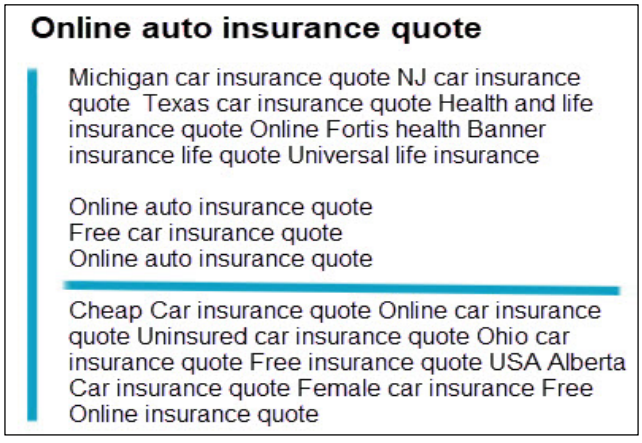

Fig. 2. Machine-Generated Webpage without Stop Words.

[10] http://infomotions.com/blog/2011/02/forays-into-parts-of-speech/
[11] https://unicheck.com/plagiarism-api-documentation

2) Category 2 (Equal or High): If the ratio of stop words on webpage Wpi is equal or higher than stop words threshold value (determined by experiment in sub-section 4.1) then web page Wpi passed the initial content-based spam test, and it will be submitted to the next web spam detection method for further tests. Figure 3 represents the algorithm for the content-based spamdexing detection using stop words.

B. Spamdexing Detection using Keywords

Keywords frequency test is the method to detect the keywords stuffing. This method is used to determine the frequency (in %) of every distinct keyword on webpage Wp_i. Keywords frequency test is done on webpage Wp_i using equations (3) and (4). Based on the keyword's frequency of every distinct keyword $KW \in Wp_i$, the web page Wp_i will classify into one of the two separate categories discussed below.

1) Category 1 (High Keyword Frequency): If the keyword frequency of any distinct keyword of web page Wp_i is higher than the keywords threshold value (determined by experiment in sub-section 4.2) then web page $Wp_i \in spamdexing$. All web pages belong to this category should be eliminated directly because it clearly shows that keyword stuffing is used which is content-based spamdexing technique. Keyword stuffing means, the ratio of different keywords in Wp_i is very less and the web spammer has repeatedly used some of the keywords too many times in Wp_i [42]. Keyword stuffing is done by repeating some specific distinct keyword/keywords again and again in several spots of Wp_i for instance in Alt attributes, content, comment tags, and Meta tags [42]. The primary target of a web spammer is to make search engine consider that the web page is relevant to different keywords present in the user's query and thus improve its page ranking. Therefore, Wp_i should be categorized as spam. Figure 4 is an example of keywords stuffing.

2) Category 2 (Low or Equal to Threshold Value): If the keyword frequencies of all different keywords of a web page Wp_i are less than KW threshold value (determined by experiment in sub-section 4.2) then web page Wp_i passed the second content-based spam test, and it will be submitted to the next web spam detection method for further tests. Figure 5 represents the algorithm for content-based spamdexing detection using keywords.

C. Spamdexing Detection using Spam Keywords Database

Content-based spamdexing can be detected using spam keywords database. A custom script has been used for identification of spam keywords on web page Wp_i. The script is matching every distinct spam keyword from spam keywords database with every distinct keyword in Wp_i for marking the spam keywords on Wp_i. After marking spam keywords, it calculates the spam keyword ratio *(SKWR)* on Wp_i using equation (6). Based on *SKWR* on Wp_i, the Wpi will classify into one of the two different categories specified.

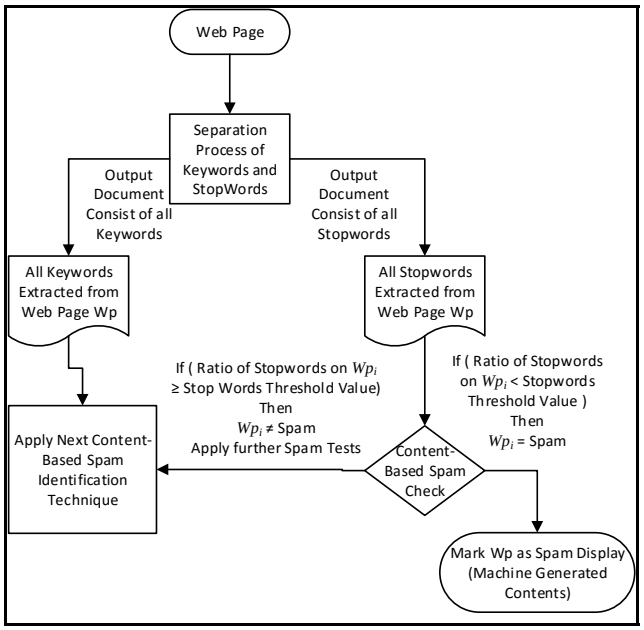

Fig. 3. Algorithm for Content-based Spamdexing Detection using Stop Words.

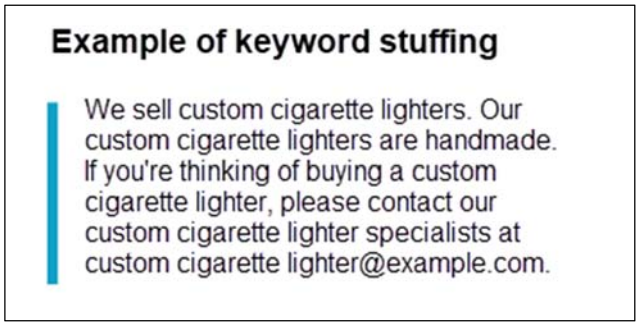

Fig. 4. Keyword Stuffing Example.

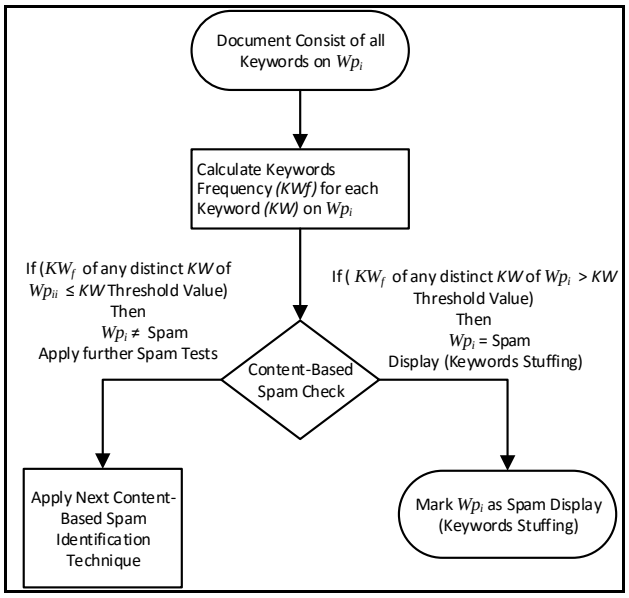

Fig. 5. Algorithm for Content-based Spamdexing Detection using Keyword Frequencies.

1) Category 1 (High Spam Keywords Ratio): If *SKWR* on Wp_i is higher than *SKW* threshold value (determined by experiment in sub-section 4.3) then web page *Wpi* $\in$ *spamdexing*. All pages belong to category 1 will be eliminated directly and marked as spam because it clearly shows the high usage of spam keywords which is a type of content-based spamdexing. Figure 6 is an example of spam keywords.

2) Category 2 (Low or Equal to Threshold Value): If *SKWR* on *Wpi* is less than or equal to *SKW* threshold value (determined by experiment in sub-section 4.3) then web page *Wpi* passed the third content-based spam test, and it will be submitted to the next web spam detection method. Figure 7 represents the algorithm for content-based spamdexing detection using spam keywords.

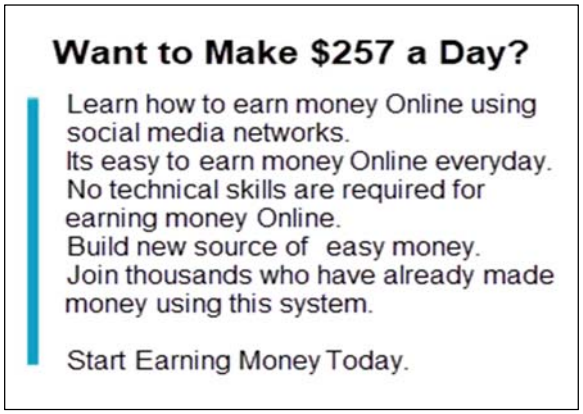

Fig. 6. Spam Keywords usage Example.

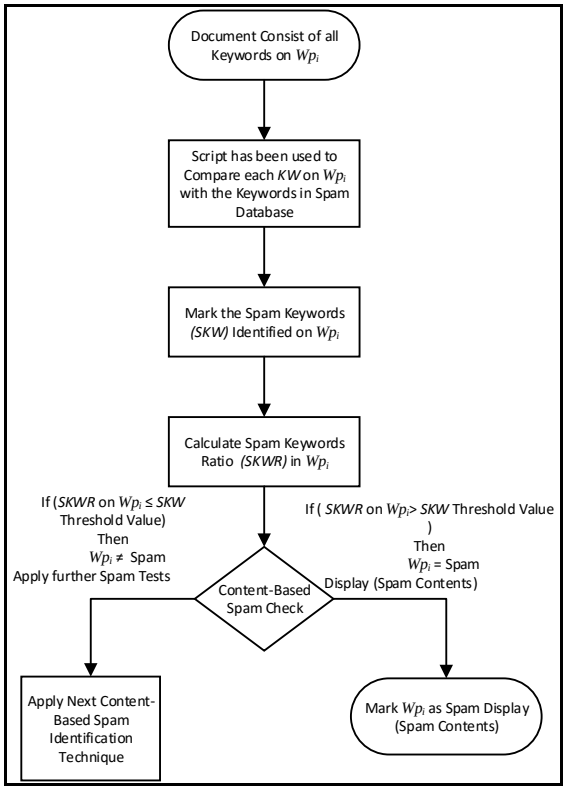

Fig. 7. Algorithm for Content-based Spamdexing Detection using SKWDB.

D. Spamdexing Detection using Part of Speech (POS)

Content repurposing can be detected using a linguistic technique known as part of speech (POS) [43]. Nowadays web spammers are using very advanced techniques for a content generation; machine-generated content is one of them. Some of the spam web pages on the internet are machine generated. Spammers are creating these spam web pages by combining large portions of a single page or by including various small sections of a page into a single web page. To detect the content repurposing on the web page linguistic features can be applied. This technique depends on the supposition that web spammers cannot replicate every aspect of natural language while producing machine generated content. The significance of utilizing broad ranges of linguistic features are discussed by Piskorski et al. [16] in their work. The primary purpose of applying these linguistic features is to recognize the originality and authorship of the content of a page. There are various grammatical forms (g_f) for instance, adjective, adverb, verb, pronoun, noun, conjunction, preposition, and interjection [44]. The ratio of every grammatical form is calculated to obtain the maximum information. The part of the speech ratio test is performed on web page *Wpi* as follows:

1) Finding and tagging the various g_f in Wpi. To tag each word in *Wpi*, Stanford Log-linear Part-Of-Speech Tagger [45] has been used.

2) Grammatical form ratio calculation: The ratio of every grammatical form, $g_f \in Wpi$ is calculated as follows:

$$Ratio\ (g_f) = \frac{x}{y} \tag{8}$$

Where x represents the number of occurrences of g_f in *Wpi* and y is the total number of words present in *Wpi*.

3) Calculation of divergence: initially the divergence of the ratio of g_f from the standard ratio of the existence of g_f available in standard English text is calculated. The standard value of each g_f can be seen on this website[10].

4) Calculation of average divergence: After computing the divergence of every $g_f \in Wpi$, the average divergence of *Wpi* is calculated using equation (9).

$$Average\ divergance = \frac{a}{b} \tag{9}$$

where a represents the sum of divergence of every $g_f \in Wpi$ and b is the total number of grammatical forms considered on *Wpi*.

5) Checking the spam status: Finally, it is time to perform the POS test. Based on the average divergence of *Wpi*, the *Wpi* will classify into one of the two different categories specified below.

a) Category 1 (average divergence is grater or equal to threshold value): If the average divergence is higher or equal to POS divergence threshold value (determined by experiments in sub-section 4.4) then web page Wpi $\in$ spamdexing, all web pages belong to this category will be eliminated directly and marked as spam because *Wpi* fails to qualify the part of the speech ratio test.

b) Category 2 (Average divergence is less than threshold value): If average divergence is less than POS divergence threshold value (determined by experiment in sub-section 4.4) then *Wpi* passed the fourth content-based spam test, and it will be submitted to the next content-based web spam detection technique. Figure 8 represents the algorithm for content-based spamdexing detection using POS.

E. Spamdexing Detection using Copied Content

Usually, the web spammers are copying the content from other similar pages and using copied content on their spam web pages. Content-based spamdexing can be identified through copied content. The copied content test is performed on web page *Wpi* is as follows:

1) A custom script has been used to access Unicheck for duplicate content identification. This integration with Unicheck is done through API.

2) The custom script accepts URL of a page as input and identifies all the pages on the internet with similar content. It returns the URLs of all pages having the duplicate content on it.

3) Based on the content of *Wpi*, the *Wpi* will classify into one of the two different categories specified.

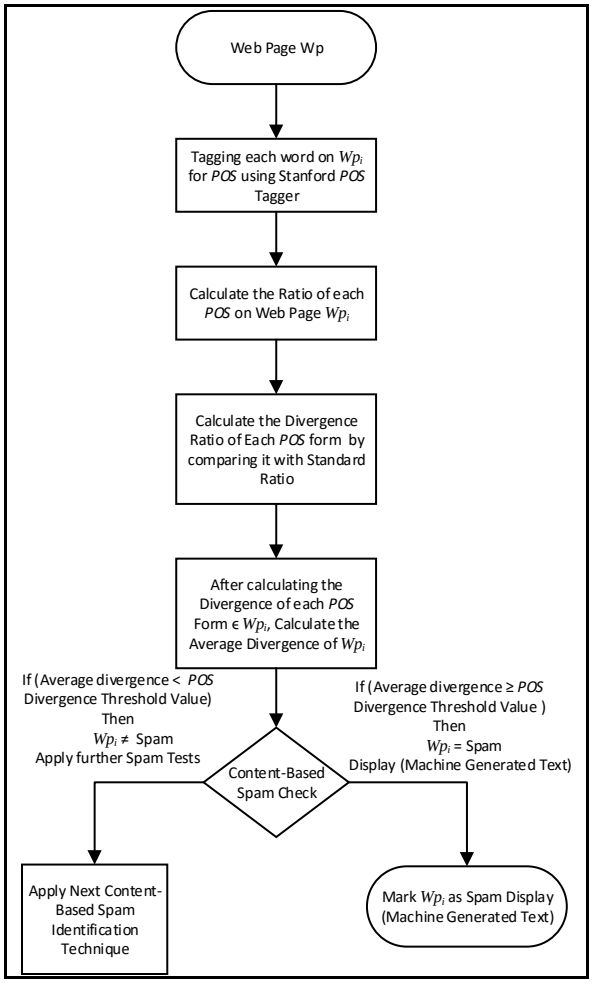

Fig. 8. Algorithm for Content-based Spamdexing Detection using POS.

a) Category 1 (Original Content): If the content of web page *Wpi* is unique and original, then *Wpi* passed all the content-based spamdexing tests, and *Wpi* is not a spam web page. It will be submitted to the link-based spam detection techniques for further testing. To save time, we first performed content-based spamdexing detections tests to identify the spam web pages (if any web page *Wpi* fails in any of the five tests then it is declared to be web spamming, and no further analysis is required). Initially applying the content-based identification techniques reduces the number of web pages for link-based tests.

b) Category 2 (Duplicate Content): If the content of web page *Wpi* is not unique, then check the originality and authorship of the content. For checking the originality and authorship, the publishing date is essential. A web page *Wpi* will be considered original if it is published before all the other duplicate web pages. Usually, the publishing date of a web page is available on the bottom or top of the same page. So, it can be easily identified that when the page was published. But there are some pages on the Internet without a publishing date on it, for finding the publishing date of such web pages we implemented a custom script using a small Google hack. Every web page published publicly on the Internet is having three different dates (publication, indexed and cache date) associated with it. To find the publication date of a web page our script is working as follows:

1) Open https://www.google.com and will paste the URL of web page *Wpi* in the search box with operator inurl: e.g. inurl:www.uthm.my/contact-us. Click search icon for searching.

2) After getting the search results for the URL above, go to the browser's address bar and at the end of Google search URL paste &as_qdr=y15 and click the search icon again for searching.

3) Google search engine will load the SERP again, and this time it will show the actual publication date of *Wpi* next to the title. Figure 9 shows the publication date of the URL.

After performing the originality and authorship test if the content is original then web page *Wpi* is not a spam page and it will be submitted to link-based techniques for further testing, and if the material is copied or not unique, then the web page is classified as spam, and no further testing are required. Figure 10 shows the algorithm for content-based spamdexing detection using copied content.

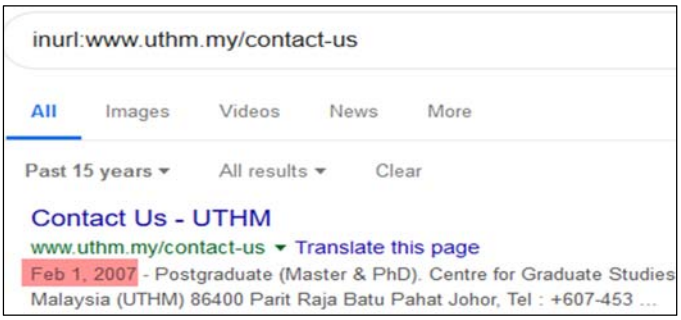

Fig. 9. Example of Publication Date.

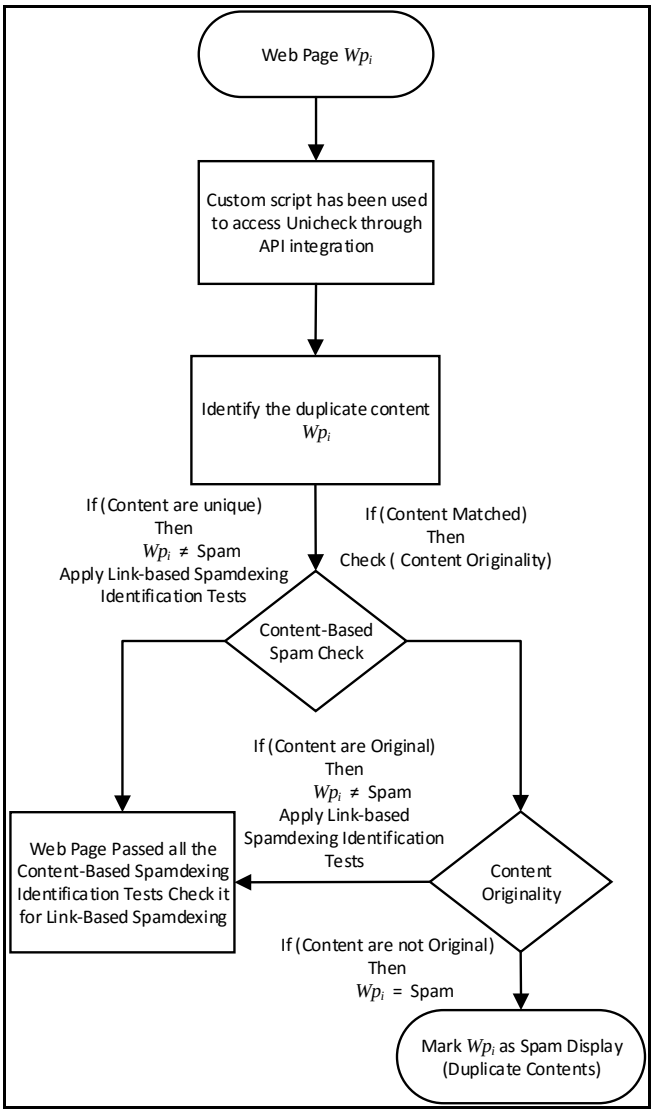

Fig. 10. Algorithm for Content-based Spamdexing Detection using Copied Content.

VI. Experimental Results

For conducting the experiments, the verification set is consisting of randomly chosen web pages which are labeled as non-spam and spam. These web pages are selected from dataset WEBSPAM-UK2006 and WEBSPAM-UK2007. These datasets are well-known and perfectly suited for web spam detection because of the following properties:

1) The datasets are consisting of several types of non-spam and spam web pages.

2) The dataset is freely available to all the researchers in the field and is used as a benchmark measure in the identification of spam web pages.

3) In the datasets, the sample web pages are uniform and random.

4) It includes different types of spam web pages which are produced by using several types of web spamdexing methods.

5) The datasets split the pages into testing and training sets with both non-spam and spam labels so these datasets can effectively be used for any link or content-based technique.

6) To get the optimized threshold values used for the proposed improved framework for content-based spamdexing detection these datasets has been utilized.

WEBSPAM-UK2006 is consist of 11,402 hosts in total, out of which only 7,473 are labeled. WEBSPAM-UK2007 is consist of 114,529 hosts in total, out of which only 6,479 are labeled. By practicing the below pre-processing methods, we obtained the dataset of five thousand pages.

1) Only those pages are considered which are labeled as non-spam or spam by real humans.

2) Among the human labeled pages, we only selected those web pages which are currently existing/working links.

3) We further filtered out the web pages and only selected those pages which are having at least 1KB content, which is necessary for our improved framework for content-based spamdexing detection.

4) Finally, we extracted the content of these web pages and stored the content in text file format.

Python is used for implementing the proposed framework, and a machine with 2x Intel Xeon E5-2670 V2 2.5GHz 10 Core, with 128GB DDR3 and operating system Ubuntu 14.04 has been used for the execution of algorithms. As F-measure is a standard approach for combining both precision and recall, so for comparison of the proposed work with other similar related works and for evaluation of the proposed algorithm we used the F-measure. The proposed improved framework for content-based spamdexing detection achieved the results shown in table 2.

TABLE. II. Performance Evaluation of Improved Framework

Technique	Precision (%)	Recall (%)	F-Measure (%)
The Proposed Framework for Content-Based Spamdexing Detection	78.3	75.6	77.4

VII. Comparison with Existing Approaches

The experimental results of the proposed improved content-based framework are compared with the following existing approaches. The comparison in Table 3 clearly shows that the proposed framework outperforms other spam detection methods. Figure 11 shows the comparison of all techniques.

1) Our proposed framework vs Roul et al [15]: The results of the proposed framework is compared with the research work of Roul et al [15]. For detecting the content-based spam web pages, they have used two features (keywords and POS). As per table 2 of Roul et al [15] they achieved an F-measure of about 70.2% and precision of approximately 71.3%, which is significantly less than our results.

TABLE. III. COMPARISON OF THE PROPOSED FRAMEWORK WITH OTHER STANDARD TECHNIQUES

Content-based Spam detection techniques	Precision	Recall	F-measure
The Proposed Improved Framework	78.3	75.6	77.38
Roul et al	71.3	69.3	70.2
Dai et al	65	44.3	52.7

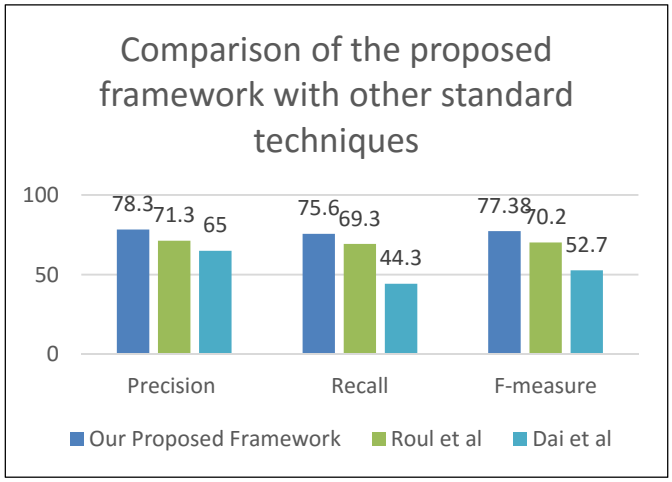

Fig. 11. Comparison of the Proposed Framework with other Standard Techniques.

2) Our proposed framework vs Dai et al [46]: Next, we compared our empirical results with Dia et al [46]. For spam identification they considered the historical web page information in their work. For improvement in spam classification, they have used the content features from the old version of pages. By applying the supervised learning techniques, they combined the classifiers based on the temporal characteristics and the current page content. With their method, they extracted several temporal features from archival copies of the web presented by Internet Archive's Way Back Machine. For their experiments, they have used the dataset WEBSPAM-UK2007. As per table 3 of Dai et al [46] they achieved an F-measure of about 52.7 and precision of approximately 65, which is less than our results.

VIII. CONCLUSION AND FUTURE WORK

In the lives of Internet users, the spamdexing is becoming a very big issue and causing big financial losses. Several techniques have been proposed to detect the spamdexing automatically and avoid this issue. In this article, we proposed an improved framework for content-based web spam detection. We explored five different techniques namely, stop words density, keywords density, spam keywords density, part of the speech ratio and copied content test to detect a web page as non-spam or spam. For this experimental work, we have used two datasets WEBSPAM-UK2006 and WEBSPAM-UK2007. An excellent and very promising F-measure of 77.38% compared to other existing approaches shows the robustness of our framework. We will extend this research work by adding the link-based spamdexing detection techniques to this framework. We believe that by using combine technique we can enhance the power of our framework to identify the wide range of web spam pages.

ACKNOWLEDGMENT

The authors would like to thank Ministry of Higher Education (MOHE) Malaysia and Universiti Tun Hussein Onn Malaysia (UTHM) for financially supporting this Research under Fundamental Research Grant (FRGS) vote 1611.

REFERENCES

[1] Z. Gyongyi and H. Garcia-Molina, "Web spam taxonomy," in First international workshop on adversarial information retrieval on the web (AIRWeb), 2005.

[2] M. R. Henzinger, R. Motwani, and C. Silverstein, "Challenges in web search engines," in IJCAI, 2003, vol. 3, pp. 1573–1579.

[3] 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.

[4] N. Z. J. MCA and P. Prakash, "Document content based web spam detection using cosine similarity measure," vol. 7, no. June, 2016.

[5] A. Shahzad, N. M. Nawi, E. Sutoyo, M. Naeem, A. Ullah, S. Naqeeb, and M. Aamir, "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.

[6] Y. Li, X. Nie, and R. Huang, "Web spam classification method based on deep belief networks," Expert Syst. Appl., vol. 96, pp. 261–270, 2018.

[7] Z. Guo and Y. Guan, "Active Probing-Based Schemes and Data Analytics for Investigating Malicious Fast-Flux Web-Cloaking Based Domains," in 2018 27th International Conference on Computer Communication and Networks (ICCCN), 2018, pp. 1–9.

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

[9] N. Spirin and J. Han, "Survey on web spam detection: principles and algorithms," Acm Sigkdd Explor. Newsl., vol. 13, no. 2, pp. 50–64, 2012.

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

[11] 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.

[12] 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.

[13] 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.

[14] A. Shahzad, N. M. Nawi, N. A. Hamid, S. N. Khan, M. Aamir, A. Ullah, and S. Abdullah, "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.

[15] 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.

[16] 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.

[17] 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.

[18] I. Bíró, D. Siklósi, J. Szabó, and A. A. Benczúr, "Linked latent dirichlet allocation in web spam filtering," in Proceedings of the 5th International Workshop on Adversarial Information Retrieval on the Web, 2009, pp. 37–40.

[19] I. Bíró, J. Szabó, and A. A. Benczúr, "Latent dirichlet allocation in web spam filtering," in Proceedings of the 4th international workshop on Adversarial information retrieval on the web, 2008, pp. 29–32.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] 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.

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

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

[27] M. Erdélyi, A. Garzó, and A. A. Benczúr, "Web spam classification: a few features worth more," in Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality, 2011, pp. 27–34.

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

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

[30] D. Hiemstra, "Language Models BT - Encyclopedia of Database Systems," L. LIU and M. T. ÖZSU, Eds. Boston, MA: Springer US, 2009, pp. 1591–1594.

[31] A. Pavlov and B. Dobrov, "Detecting content spam on the web through text diversity analysis," CEUR Workshop Proc., vol. 735, pp. 11–18, 2011.

[32] E. S. Swirsky, C. Michaels, S. Stuefen, and M. Halasz, "Hanging the digital shingle: Dental ethics and search engine optimization." Elsevier, 2018.

[33] "Attracting and analyzing spam postings," Nov. 2015.

[34] R. Hassanian-esfahani and M. Kargar, "Sectional MinHash for near-duplicate detection," Expert Syst. Appl., vol. 99, pp. 203–212, Jun. 2018, doi: 10.1016/J.ESWA.2018.01.014.

[35] R. Agrawal, "Controlling Unethical Practices in Web Designing by Search Engines," 2018.

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

[37] D. Pawade, A. Sakhapara, M. Jain, N. Jain, and K. Gada, "Story Scrambler–Automatic Text Generation Using Word Level RNN-LSTM," Int. J. Inf. Technol. Comput. Sci., vol. 10, no. 6, pp. 44–53, 2018.

[38] W. Li, "Consistency checking of natural language temporal requirements using answer-set programming," 2015.

[39] K. McKeown, Text generation. Cambridge University Press, 1992.

[40] H. T. Dang, "Overview of DUC 2005," in Proceedings of the document understanding conference, 2005, vol. 2005, pp. 1–12.

[41] A. Summerville, S. Snodgrass, M. Guzdial, C. Holmgard, A. K. Hoover, A. Isaksen, A. Nealen, et al., "Procedural Content Generation via Machine Learning (PCGML)," IEEE Trans. Games, vol. 10, no. 3, pp. 257–270, Sep. 2018.

[42] D. Roy, M. Mitra, and D. Ganguly, "To Clean or Not to Clean," J. Data Inf. Qual., vol. 10, no. 4, pp. 1–25, Oct. 2018.

[43] E. Sadredini, D. Guo, C. Bo, R. Rahimi, K. Skadron, and H. Wang, "A Scalable Solution for Rule-Based Part-of-Speech Tagging on Novel Hardware Accelerators," in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD '18, 2018, pp. 665–674.

[44] A. Keyaki and J. Miyazaki, "Part-of-speech tagging for web search queries using a large-scale web corpus," in Proceedings of the Symposium on Applied Computing - SAC '17, 2017, pp. 931–937.

[45] Y. S. Toutanova, Kristina, Dan Klein, Christopher Manning, "Feature-rich part-of-speech tagging with a cyclic dependency network," in In Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. Association for computational Linguistics, 2003, pp. 173–180.

[46] X. Dai, N., Davison, B. D., & Qi, "Looking into the past to better classify web spam," in In Proceedings of the 5th international workshop on adversarial information retrieval on the web, 2009, pp. 1–8.