UDC 004.021

D.Holovko (MITI)
O.Holovko (A0334)
PhD R.Bieliakov (MITI)
I.Karlovskiy (MITI)

THE METHOD OF AUTOMATED ANALYSIS OF WEB-SITES FOR THE PURPOSE CONTROL POINT DEVIATION OF SIGHT ON THE BASIS OF ASSOCIATIVE ALGORITHM

In the article method of the automated analysis of web-sites is presented for the purpose deviation from the control point of sight on the basis of associative algorithm. The got methodology is the synthesis of existent methods of analysis of web-sites on the basis of protocol of WHOIS, analysis of initial HTML-code, analysis of attachment of the unique extracted lines - id on the example of Google Add Words. The methodology can be applied for finding out the web-sites of terrorist and other illegal organizations.

Головко Д.О., Головко О.О., Бєляков Р.О., Карловський І.В. Метод автоматизованого аналізу сайтів на предмет відхилення від контрольної точки зору на основі запропонованого асоціативного алгоритму. *Отримана методика являється синтезом існуючих методів аналізу сайтів на основі WHOIS – протоколу, аналізу вихідного HTML – коду, аналізу прив'язки виділених унікальних строк – id на прикладі Google Add Words. Розроблена методика може бути застосована для ідентифікації сайтів терористичних та інших неправомірних організацій.*

Головко Д.А., Головко А.А., Беляков Р.О., Карловский И.В. Метод автоматизированного анализа сайтов на предмет отклонения от контрольной точки зрения на основе ассоциативного алгоритма. *Полученная методика является синтезом существующих методов анализа сайтов на основе WHOIS – протокола, анализа исходного кода HTML, анализа привязки уникальных извлеченных строк -id на примере Google Add Words. Полученная методика может быть применена для обнаружения сайтов террористических и других неправомерных организаций.*

Key words: *HTML-code, WHOIS, associative algorithm.*

Formulation of a problem. In the modern world, the Internet becomes profoundly occupied by media content. Even though all benefits of highly available content desired by modern society, this content might pose raising threat. 24/7 available and accessible content from almost all parts of the world forces us to reconsider methods of propaganda and information operations. Obviously, a foremost tool of newly invented propaganda and information operations is websites, that spreading all kind of media and news content. Unfortunately, Ukraine faces this kind of activity that supporting separatist movement in Eastern Ukraine. Plenty of websites were instantly deployed in the early beginning of the pro-Russian separatist movement [1 – 3]. By analyzing provided by websites fakes, that blaming Ukrainian official government it is easy to come up with the idea of central control of these websites. Therefore, was presented a method of automated analysis of websites that allowing making a conclusion about the association between websites from a centralized control standpoint.

Obviously, considered method would be applicable for other fields. First of all, method of detecting centrally controlled websites might be used by military and defense agencies for counteracting information operations. Especially it can be useful in fighting terrorism because terroristic groups widely employ web technology to spread propaganda and potentially dangerous information. Detecting the set of websites controlled by the same group could benefit fighters in order to understand flows of information spread, information origins cooperation between different sources. Besides military/intelligence application considered method might be beneficial for the private sector. Private companies would use it for competitive intelligence on the Internet. For instance, understanding of sites run by a competitor could disclose for analyst competitor marketing strategy or product development direction. Additionally, forensics investigator may need to detect a group of sites controlled by the same person in cyberstalking cases or in cases of spreading prohibited information (fascism symbols). For instance, an investigator might associate several sites with the same person and this association would make detectives job much easier.

In websites industry which based on analysis of freelancers` websites, there are several groups of actors involved in websites lifecycle. This paper will consider following groups of actors [4 – 6].

Owner (investor) – person or group of persons that financially involved in websites lifecycle. Provided finance for another group in order to perform their jobs. The owner might also belong to one of other groups. Usually, owner gathers statistical/advertisement information from websites in order to control it. Provide directions for other groups.

SEO (search engine optimization) – experts responsible for maintaining site`s high ranking by search engines. Might provide directions for programmers, designers, authors (copywriters).

Programmers (administrators) – experts responsible for technical part of site`s lifecycle. They create software modules (so called CMS plugins), maintain webserver, etc.

Designers – experts mostly responsible for creating graphical content and page markup. They might create another media content. Usually, designers work in cooperation with programmers and SEO group.

Authors (copywriters) – experts responsible for creating of unique textual content for websites. They work in cooperation with SEO group in order to reach good ranking for specific search engines` queries.

Thus, described division of responsibilities between persons poses some difficulties in process of sites association. Because of involvement of many different people in the lifecycle, forensics or artificial intelligence method should be applied carefully. Well-known methods of authorship determination might generate false positive. If these methods will be applied to textual content they might determine an association between authors, however, one author can work for two not associated owners. A similar scenario also possible with source code analysis.

Main research task. In order to solve task of association of centrally controlled websites main association feature should be chosen.

As discussed above, it may be difficult to choose the right option. Thus, the main goal of the research is the automatic determination of the processes of automated analysis of web sites for targeted control of management based on the associative algorithm.

To solve the task, it is necessary to perform analysis of various functions and protocols. Let's look at some features or their sources that may be of interest for trusting relationships

Analysis of recent researches and publications. Such outstanding scientists as Hotto, Kotler, Emily, Georgia Frantzeskou are engaged in search of sites of malefactors, which in their works apply various hierarchical methods and techniques [6 – 9]. Also, obviously, these issues are dealt with by entire departments and security services, who prefer not to disclose their unique techniques. The main disadvantages are listed in the description below.

The methods and research materials which was used:

1. WHOIS – protocol. WHOIS was founded as Internet-wide service that supposed to provide information about site owners. Essentially, WHOIS provides information about domain or IP. Essentially domain should be a thing that belongs to owner. Each internet user recognize site by its name. For instance, google.com or time.com etc. Even though WHOIS service was created for purposes that are considered by our method, WHOIS frequently provides not real information [7]. One of the reasons is privacy protected. There is service on the Internet that provided fake email, phone number, person name for WHOIS services. So, stranger (regular user of Internet) cannot get true information about owner. Additionally, some security services for websites would like to replace original information in WHOIS databases by fake one. For instance, Cloud Flare services recommend it for top level of accounts.

Another reason for false information in WHOIS databases is registration of domain thorough hosting company. In this case hosting put owner credential in WHOIS services. Moreover, geographically distributed structure and weakly controller WHOIS database might contains outdated information. So all these reasons make a WHOIS-based information a bad clue for websites association (Fig. 1).

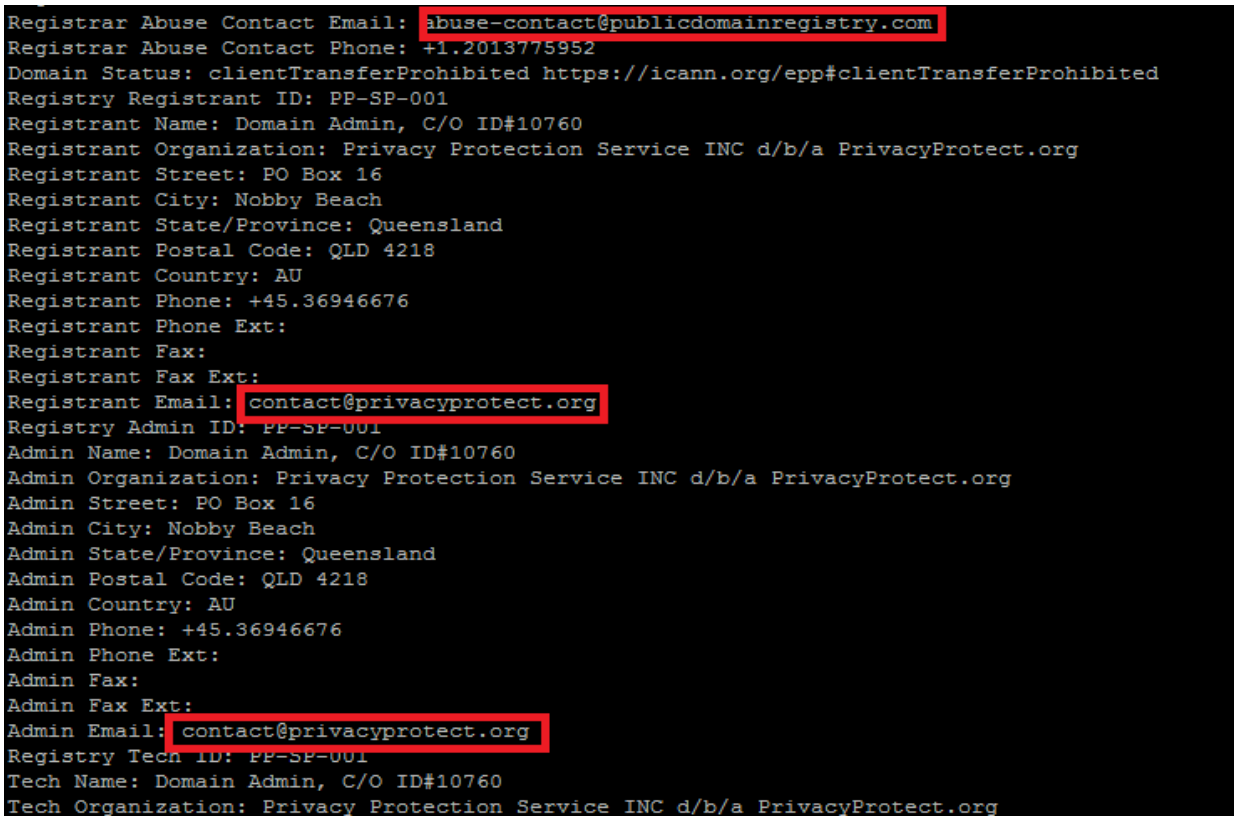

Figure. 1. WHOIS output for privacy protected domain

2. Source code analysis. Nowadays, regular websites are created using Content Management System (CMS).

Content Management Systems are type software that allowing websites creation in coding-less manner and extending a functionality using plugins and changing of design using themes. Most of the CMSs are distributed under open sources licenses. So, almost any person familiar with programming can contribute to their source code (backend and frontend) [8].

Persons familiar with programming might create plugins and theme for CMSs, as well. For instance, 27,7 % of all websites use WordPress as CMS and first TOP3 CMS on Internet are open source products [9].

These factors do not allow direct applying of source code based authorship methods in order to pursuit our task.

3. Links/directory structure and content authorship. Another source of significant features that might be considered for association of centrally controlled websites is analysis on hyperlinks or directory tree structure that might repeat from site to site for the same owner.

However, statistics mentioned in previous research proof that this approach would not be viable.

All websites that used the same CMS have same directory structure and similar links structure. Even plugins installed on CMS do not change a structure significantly. By taking into account that WordPress have 49,335 plugins, it was assuming that owner want to install already developed plugin rather than invest in development of new one. Even so called self-written CMS will be a hard task because of need to scan/brute force all pre-known (dictionary) and not-known directory names.

Thus, analysis of directory structure is not a good approach for our task.

As one of the approaches to reach desired objective is analysis of text/content of website`s articles. However, such analysis should balance (weight) somehow plenty of different pages of the same website and reduce them to one parameter. Additionally, text authorship analysis will take a long time to do because of necessity of indexing(downloading) of all website`s pages.

Therefore, applicability of this approach for large set of websites would be difficult. Moreover, copywriting markets, where website owner can buy a high quality SEO-oriented article, make unreasonable text based authorship analysis.

The main part.

The used approach was inspired by an idea that almost all websites try to gather visitor's statistic using services such as Google Analytics.

Google Analytics for this goal generated a special JavaScript that should be inserted on all pages of a website and this script contains special unique id. Thus, finding a website that has the same identification token (unique id) allows us to conclude that both sites have the same owner, at least owner of google analytics account. In pursuit of extending, a scope of search characterization of known Google analytics, Google AdWords and some other identifications tokens (strings) was performed. Thus, identification string or API token usually consist of:

– the id-string that contains English alphabet characters in low and high case and numbers;

– the id-string contains simultaneously both characters and numbers;

– the id-string is at least 14 characters long (Lower length might increase rate of false positive);

– the id-string is not end by known file extension (It is a sign of machine generated filename, might be content based generation);

– string contain at least 4 numbers.

So, the source code is considered as main websites attribute that would allow determination of association factor between websites by accurate metric selection.

Aforementioned characteristics of identification strings allow extraction of these strings from source code.

Implemented algorithm (software) search and extract the id-strings that match parameters. Because these strings usually inserted on all pages of a website, it is sufficient to analyze just an index page of a website. Therefore, it benefits a speed of proposed algorithm.

In order to detect centrally controlled websites following associative algorithm was proposed:

1. Download a source code (HTML) of main page of the WEB-site.
2. Extract identification strings using aforementioned characteristics.
3. Create an array of unique extracted id-strings for each WEB-sites.

Also, for calculating a distance between websites, another formula might be used dist=size(x∩y)/size(x∪y), where x and y are id-strings arrays of websites.

Even though two formulas were proposed, the meaning of distance is not so important for the fact of websites association. Because we a looking for strings that are unique machine generated strings fact of the match is more significant than distance.

However, in the case when two websites have a couple of id-strings matched it gives us a much better clue of association.

4. Compare id-strings arrays of sites again each other and calculate distance between WEB-sites using formula we can estimate distance between sites i and j, with counts n_i and n_j and intersection count n_{ij}, as $R_{ij} = \sqrt{n_i n_j / n_{ij}}$ [10].

In order evaluate the effectiveness of proposed method experiment was performed. For purpose of experiment, a list of websites related to the connected with the temporarily occupied Donetsk region was formed. There are, a written script leverages Google Search API (Enclosure 1).

Thus, 255 websites were gathered and listed. Another script (Enclosure 2) implements described algorithm and output of the table (Enclosure 3) that contains association factor for each pair of sites. If consider websites as nodes and table of association factors as edges we can build a graph.

Enclosure 1

Source code of websites extraction script

```
<?php
function getSites($website,$s)
{
                                                $json =
file_get_contents('https://www.googleapis.com/customsearch/v1?key=<API_ID>&q=related:'.$website.'&cx=<CUSTO
                M_SEARCH_ENGINE_ID>:iqvgd8yg8fy&start=1&fields=items(displayLink)');
        $s[$website] = 1;
        $sites = json_decode($json);
        if (count($sites) > 0)
        foreach ($sites->items as $val)
        {
                if(!array_key_exists($val->displayLink,$s))
                        $s[$val->displayLink] = 0;
        }
        return $s;
}
$all_sites = array();
$all_sites["dnr-news.com"] = 0;
while(count($all_sites) < 250){
        $w = array_search(0,$all_sites);
        $all_sites = getSites($w,$all_sites);
        echo count($all_sites);
        sleep(2);
}
//print_r($all_sites);
foreach(array_keys($all_sites) as $val)
```

Enclosure 2

Source code of the main module

```
<?php
$sites = file("sites.list",FILE_IGNORE_NEW_LINES);
$main = array();
foreach ($sites as $value) {
        echo "Requesting index page of ".$value.". ";
        $f = get_site($value);
        if ($f!=false){
                echo "Page for ".$value." successfully retrived.";
                preg_match_all("/([a-z0-9\-]{14,})((\.[a-z]{2,4}|[^\w]))/im",$f,$out);
                echo "Unique strings found:".count($out[0])."\n";
$strings = array();
                foreach ($out[1] as $key => $v2) {
                        if ((strpbrk($v2, '1234567890') == true) and (preg_match("/[a-z]/i",$v2) == 1) and
(strlen($out[3][$key])<2) and (contains_number($v2,4))){//
                                $strings[] = $v2;
                        }
                }
if (count($strings)!=0) {
                        $main[$value] = array_unique($strings);
                        }
        } else {
                echo "LoadingError[".$value."]\n";
        }
}
$matrix = array();
foreach ($main as $key => $value) {
```

Continuation of enclosure 2

```
		foreach ($main as $key2 => $value2) {
			if ($key == $key2){
				$matrix[$key][$key2] = 0;
			} elseif (!isset($matrix[$key2][$key])){
				$common = count(array_intersect($main[$key], $main[$key2]));
				//1st formula number_format($common / count($main[$key]+$main[$key2]),4)
				//*2nd fotmual
number_format(sqrt((count($main[$key])*count($main[$key2]))/$common),4)
				if ($common>0){
					$distance = sqrt((count($main[$key])*count($main[$key2]))/$common);
				} else {
					$distance = 0;
				}
				$matrix[$key][$key2] = array(number_format($distance,4),implode('|',
array_intersect($main[$key], $main[$key2])));
			}
		}
}
$fp=fopen('file_st.csv','w');
$fp2=fopen('file_st2.csv','w');
$list = array_keys($main);
$nodes = $list;
array_unshift($nodes, "Label");
file_put_contents("nodes", implode(";\n",$nodes));
fputs($fp,"Source;Target;Weight;Intersection;\n");
foreach ($matrix as $node1 => $value) {
		$num1 = array_search($node1, $list);
		foreach ($value as $node2 => $val2) {
			$weight = $val2[0];
			$intersect = $val2[1];
			$num2 = array_search($node2, $list);
			if ($node1!=$node2){
				fputs($fp,"$node1;$node2;$weight;$intersect;\n");
				fputs($fp2,"$num1 $num2 $weight\n");
			}
		}
}
fclose($fp);
fclose($fp2);
echo "Output: file_st.csv, file_st2.csv\n";
function contains_number($str,$count){
		$c = 0;
		for ($i=0; $i < strlen($str); $i++) {
			if (is_numeric($str[$i])){
				$c++;
			}
			if ($c>=$count){
				return true;
			}
		}
		return false;
}
function get_site($url){
		$curl_handle=curl_init();
		curl_setopt($curl_handle, CURLOPT_URL,$url);
		curl_setopt($curl_handle, CURLOPT_CONNECTTIMEOUT, 20);
		curl_setopt($curl_handle, CURLOPT_RETURNTRANSFER, 1);
		curl_setopt($curl_handle, CURLOPT_FOLLOWLOCATION, true);
		curl_setopt($curl_handle, CURLOPT_USERAGENT, 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/42.0.2311.135 Safari/537.36 Edge/12.246');
		curl_setopt($curl_handle,CURLOPT_ENCODING , "gzip");
		$query = curl_exec($curl_handle);
		curl_close($curl_handle);
		return $query;
}
?>
```

Enclosure 3

Edge table (reduced only not zero values presented)

Source	Target	Weight	Intersection
dnr24.com	dnr-board.com	4.4721	ca-pub-4270514290018072
sprotyv.info	dialogweb.ru	25.0998	eHBhY2tldCBiZWdpbj0i77u|IiBpZD0iVzVNME1wQ2VoaUh6cmVTek5UY3prYzlkIj8
www.polk.ru	www.amic.ru	4.8990	d27cdb6e-ae6d-11cf-96b8-444553540000
xn----ctbhoxecbwfl.ru-an.info	xn----ctbbwca3adfbdba1aiiai.ru-an.info	2.2361	ca-pub-9476904727309632|books555banner2|afterbooks555banner2|banner34er45tg56block|yaCounter23548864
anti-maidan.com	rls.tv	2.8284	t7VBWd2Ls-tXO77vTAR8|ca-pub-6338348788326520
ok.ru	mail.ru	15.6844	msapplication-square70x70logo|msapplication-square150x150logo|msapplication-wide310x150logo|msapplication-square310x310logo
ok.ru	www.yandex.ru	32.0156	msapplication-square70x70logo|msapplication-square150x150logo
podrobnosti.ua	www.1tv.ru	25.6905	evrovidenie-2017
ukranews.com	112.ua	31.0805	evrovydenye-2017
mincult.govdnr.ru	mzdnr.ru	13.0000	67fb34f6a866c40d0570
mincult.govdnr.ru	minjust.ru	24.9800	67fb34f6a866c40d0570
zpolk-org.livejournal.com	www.livejournal.com	5.6125	8v2h21h9V0H21z|15h-2v-4h-4V9h4V5h2v4h4v2h-4V15L16|11h10v2H10V11L10|7h10v2H10V7L10|33h16v-2H9V33z|5h18v-4h-18zM8|8v-4h4v4h-4zM26|18v-4h4v4h-4zM14|5h18v-4h-18v-0|flaticon--googleplus2015|flaticon--cross--w20--99bfcc|da7aa44a6827a9b38d22ad009135c8841719b239
zpolk-org.livejournal.com	www.evasiljeva.ru	36.0832	4NEyzWZVv7gepd-Hsmcddnda8pBrp3JDACLcB
www.ntv.ru	echo.msk.ru	44.1135	yaCounter42355849
www.youtube.com	www.google.ru	155.0161	AHpOoo-J3J0yqNDMPVrmQT6j-SBFfGx8oA
x-true.info	www.opennet.ru	10.1980	D27CDB6E-AE6D-11cf-96B8-444553540000
x-true.info	battlefront.ru	7.2111	D27CDB6E-AE6D-11cf-96B8-444553540000
mail.ru	www.yandex.ru	24.4949	msapplication-square70x70logo|msapplication-square150x150logo
mzdnr.ru	minjust.ru	24.9800	67fb34f6a866c40d0570
novorossia.tovus.info	militarymaps.narod.ru	8.7750	yaCounter39883200
novorossia.tovus.info	poisk-rubizhne.ucoz.ua	7.9373	yaCounter39883200
www.opennet.ru	battlefront.ru	2.8284	D27CDB6E-AE6D-11cf-96B8-444553540000
militarymaps.narod.ru	poisk-rubizhne.ucoz.ua	9.9499	yaCounter39883200.

In order to analyze graph, was applied visualization software (Gephi).

This software allows plotting of the graph. On the graph (Fig. 2), we can observe that algorithm found several groups of websites.

These groups usually are 2 – 3 sites big. In addition, by visiting the websites it is easy to recognize that websites connected by the same topic.

By manually analyzing and determining reasons for such connection. We can observe some false-positives.

For instance, a couple of nodes connected by matching string „evrovidenie-2017” would be a coincidence.

However, there are most of the websites that connected by strings, such as „ca-pub-4270514290018072” or „yaCounter39883200”.

Those strings are id-strings for online visitor`s analytics for search engines Google and Yandex. Thus, algorithm found 14 groups of connected of websites. It is about 13.7 % of all websites listed for research.

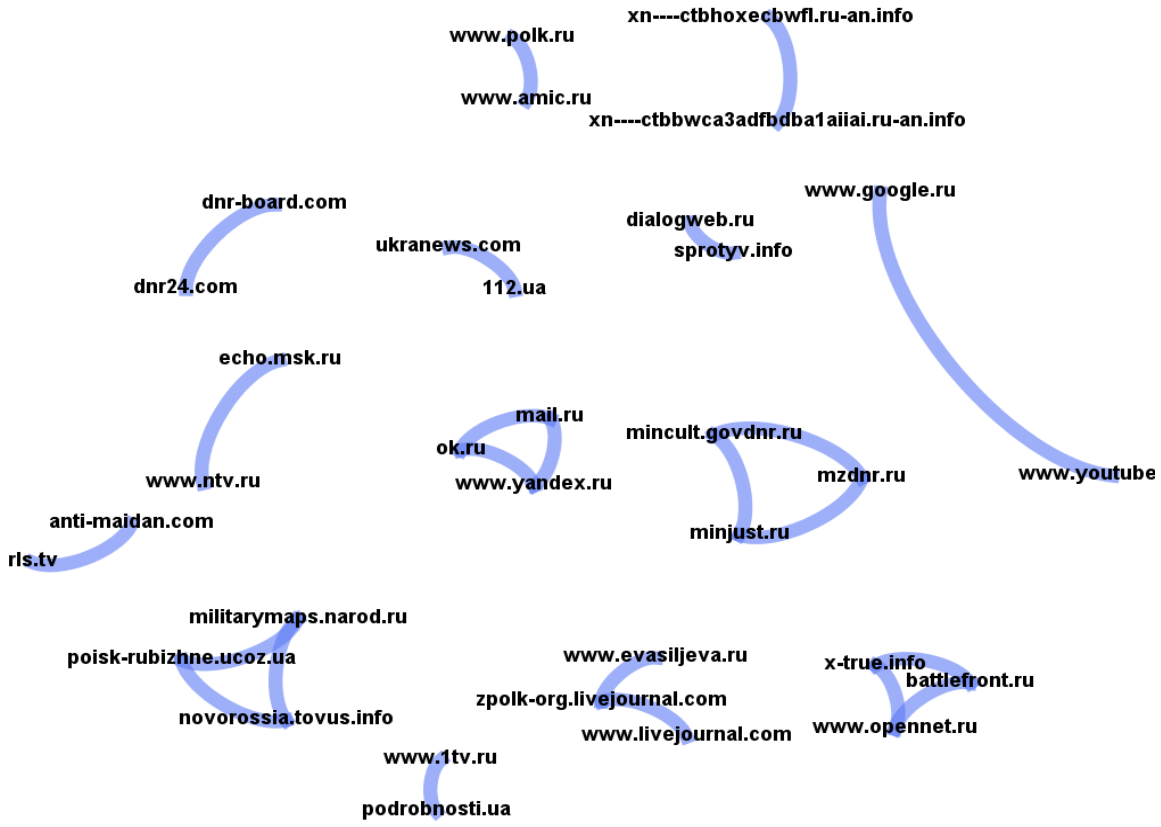

Figure. 2. Graph produced by Gephi for connected WEB – sites

Conclusions and direction of further researches

Developed approach have some pros and cons.

The drawback of approach is that it heavily depends on id-strings extraction algorithm for that. In addition, aforementioned characteristics are based on general observations and might be miss important data for an association. In order to solve this problem, we would gather a significant amount of id-strings and apply the machine-learning algorithm.

Machine learning will reduce false positive and rate of missed identification data.

Another drawback is in the trustworthiness/credibility of source code. Data can be unreliable for criminal investigation because it might be forged.

For instance, a person that familiar with described approach may want intentionally put some identification strings in pages’ source code.

So, in this way lead investigator in a wrong direction. Even though proposed approach has some drawbacks, it can be considered for some cases because it gives a good clue about central control of the websites.

So, ***the direction of further research*** is to improve the proposed method by integrating existing and developing new models and methods of artificial intelligence to fully automate the search for central control websites.

REFERENCES

1. P. Christopher., M. Matthews. The Russian „Firehose of Falsehood" Propaganda Model [https://www.rand.org/content/dam/rand/pubs/perspectives/PE100/PE198/RAND_PE198.pdf] – viewed 17.05.2018.

2. Olga Oliker, „Russia's New Military Doctrine: Same as the Old Doctrine, Mostly," Washington Post, January 15, 2015.

3. Giorgio Bertolin, „Conceptualizing Russian Information Operations: Info-War and Infiltration in the Context of Hybrid Warfare," IO Sphere, Summer 2015, p. 10.

4. Marko Bielliniaso. Razrabotka Web-prilogeniy v srede ASP.NET 2.0: zadacha – proekt - reshenie = ASP.NET 2.0 Website Programming: Problem – Design – Solution. – M.: „Dialectica", 2007. – P. 640. – ISBN 0-7645-8464-2.

5. Olishiuk Andrey. Razrabotka Web-prilogeniy na PHP 5. Professionalnaia rabota. – M.: „Viliams", 2006. – P. 352. – ISBN 5-8459-0944-9.

6. Hotto, Kotler Emily. Web-redisain, 2 isdanie. – SPB.: „Simvol", 2006. – P. 416. – ISBN 5-93286-082-0.

7. ICANN. About WHOIS. [https://whois.icann.org/en/about-whois] – viewed 17.05.2018.

8. Georgia Frantzeskou. Identifying Authorship by Byte-Level N-Grams: The Source Code Author Profile (SCAP) Method. International Journal of Digital Evidence. Spring 2007, Volume 6, Issue 1. [file:///C:/Users/User/Downloads/Identifying_Authorship_by_Byte-Level_N-Grams_The_S.pdf] – viewed 04.04.2018.

9. Web Technologies Survey. Usage of content management systems for websites. [https://w3techs.com/technologies/overview/content_management/all] –

viewed 04.04.2018.

10. Neil, Rowe. Computer Foresics CS4677. – Monterey: Computer Foresics, 2017. P. 25 – 29.