DEVELOPING METHODS BASED ON TEXT MINING TECHNOLOGY TO IMPROVE THE QUALITY AND SPEED OF AUTOMATIC CLUSTERING OF DOCUMENTS

Oleksii Vasilenko, Oleksandr Kuzomin, Artem Mertsalov

***Abstract**: The goal is to develop methods based on the use of Text Mining technology, which allows to improve the quality and speed of automatic clustering of documents.The object of the research is the intellectual analysis of the text array.The methods of numerical simulation and analytical substantiation are used. The research of methods of intellectual analysis of the text was conducted; Methods of preprocessing of text, selection of keywords and classification of documents are considered.As a result of implemented software implementation of the system analysis of the text array.*

The aim of the work is to study of ranking algorithms on data from Web Archives. A developed software use web archives and extracted data to create a graph of relations between web pages. A HITS algorithm applied on the graph, allows to find meaningful hubs among the pages. The hubs allow calculating authority of each page and ranking them by using the value.

Link pairs from German Web Archive Data are used as input data for graph (incoming and outgoing links), and German Wikipedia articles as search topics to evaluate results.

The results evaluated using survey with results of HITS algorithm in comparison with results of Bing search engine and competing algorithm PageRank.

The work use Java programming language for implemented algorithms and support software, the destination graph stored in-memory by using Redis. The data extracted from Web Archives by using Hadoop framework and stored in Hive database.

***Keywords**: web archives, web search, ranking, hyperlink induced topic search, page rank.*

***ITHEA Keywords**: NNP - proper noun, AJ - general adjective, DT - general determiner, NLP - natural language processing, E.2 Data Storage Representations, H.3.1 Content Analysis and Indexing.*

Introduction

Automated exclusion of knowledge from the text is one of the main tasks of artificial intelligence and is directly related to the understanding of texts in the natural language. Since the mid-50s of the last century, considerable efforts of scientists have been directed at the development of mathematical algorithms and computer programs for the processing of texts in the natural language [Bansal, 2014]. To automate the analysis and synthesis of texts, various patterns of text processing were created, as well as corresponding algorithms and data representation structures. Traditionally, the analysis of natural language texts was presented as a sequence of processes - morphological analysis, syntactic analysis, semantic analysis. For each of these steps, appropriate models and algorithms were created. For semantics of the text - the classical semantic networks and the framed models of Minsk, for the syntax of the sentence - the chronological grammar, the systemic grammar of the Hollywood, the trees of subordination and the system of components of Gladky, the expansion of the network of transitions; For morphological analysis, many different models have been developed that focus on specific language groups.

The task of automated analytical processing of text information is trying to solve many foreign and domestic scientists. In particular, in 1979 Kuzin N.T. [Ridings, 2002] described the methods of frequency subtraction of textual information, which were subsequently improved in the works of A. Broder and D.V. Lande [Bansal, 2014] In his manual A.A. Barseghyan and MS Kupriyanov summarized data on modern methods of automatic analysis of Data Mining and Text Mining. However, none of the described methods does not provide extraction from text information knowledge. In studies AI Vavilenkova provided a description of the main methods of Data Mining, highlighting their advantages and disadvantages, emphasizing that none of the described methods is capable of removing knowledge from the information. In this work the researcher demonstrated the work of the Robinson resolution method for comparing two simple sentences; the algorithm of comparison of logical-linguistic models of text information on the content is proposed.

Thus, one can distinguish the main problems associated with the need to optimize the modeling and development of methods for analyzing textual information:

- The rapid growth in the amount of information contained on the Internet is the cause of increasing and growing difficulties in finding the necessary documents and organizing them in structured, structured, content-based repositories;

- most technologies of work with text documents focus on the organization of convenient work with information for a person, but virtually no opportunity to convey the semantic content of the text, that is, there is no semantic indexing;

- To effectively address the problem of search, it is necessary to broaden the concept of a traditional document: the document must link the knowledge that allows to interpret and process the data that is retained in this document;

- Unstructured information is a significant part of modern electronic text documents.Web Archives contains web pages from the past years and save new pages for future researchers, historians, and the public. They are very important for learning how interned is developing. They allow to use knowledge from the past and apply it today to extract new knowledge and very popular for hundreds of research tasks as playground.

Search is the most demanded tool in today's web. There is a trillions of pages stored somewhere in the web and search engines are like road signs before the navigator was developed. By typing simple search term, they navigate to destination pages. A web search engine is a software system that is designed to search for information on the World Wide Web. The common approach for web search engines is to analyze content of the pages in the web and create index for them. Then, using that index and different retrieval methods, they select all pages that match a required search term and to return relevant results they apply ranking algorithms to order results. All the same is true for web archives, but in addition, they also contain some specific attributes, which include crawling information and a history of the page. That can be used to improve search results quality.The algorithms are improving with years and modern search engines use very advanced technology with Artificial Intelligence (AI) and machine learning. But unfortunately, those high-end algorithms only available for lead companies that specified on search engines.

Therefore, the goal of the work is to apply available ranking algorithms on Web Archives and investigate potential ability to improve search results by including new attributes to the algorithm.

Motivation and problem statement

Improving search results is interesting and demanded task, which can be applied on Web Archives for research. In the work, one of the popular algorithms for ranking search results is applied on German Web Archives and learned how archived data can change search results. Despite the fact that for the web search simple algorithms are no longer enough, nevertheless they are increasingly used for research, personal and low-middle level commercial purposes. They should have good enough results and lower maintenance costs.

As a problem statement we have got web archives that contains web pages (articles, news, etc.), which stored as natural language text (unstructured text). With the purpose of building an efficient search results, we are confronted with following problems:

1. Different formats of URLs and aliases.
3. Using diacritic characters and language specific characters in URLs.

2. A huge amount of data, which should be processed in real-time.
4. Slow processing time for Web Archives.

Topic of research

Recent development of the Internet and computing technologies makes the amount of information increasing rapidly. That is why it is necessary to retrieve the best of the web pages that are more relevant in terms of information for the query entered by the user in search engine. In recent years, semantic search for relevant documents on web has been an important topic of research. Many semantic web search engines have been developed that helps in searching meaningful documents presented on semantic web. To relate entities, texts and documents having same meaning, semantic similarity approach is used based on matching of the keywords, which are extracted from the documents. For example, groups of authors presented a new web ranking system by using Semantic Similarity and HITS algorithm along with AI technique [Bansal, 2014]. In this paper, author proposed Intelligent Search Method (ISM) - a ranking system with improved HITS and Semantic Similarity techniques. It is used to rates the web pages and also known as Hubs and Authorities. A good hub represented a page that pointed to many other pages and a good authority represented a page that was linked by many different hubs. Therefore, its authority value, which estimates the value of the content of **the page, and its hub value, which estimates the value of its links to other pages.**

Author developed new method to index the web pages using an intelligent search strategy in which meaning of the search query is interpreted and then indexed the web pages based on the interpretation. Comparison of HITS Algorithm, Semantic Similarity Algorithm and ISM method is shown in (Tab. 1).

Table 1 – Comparison of Techniques

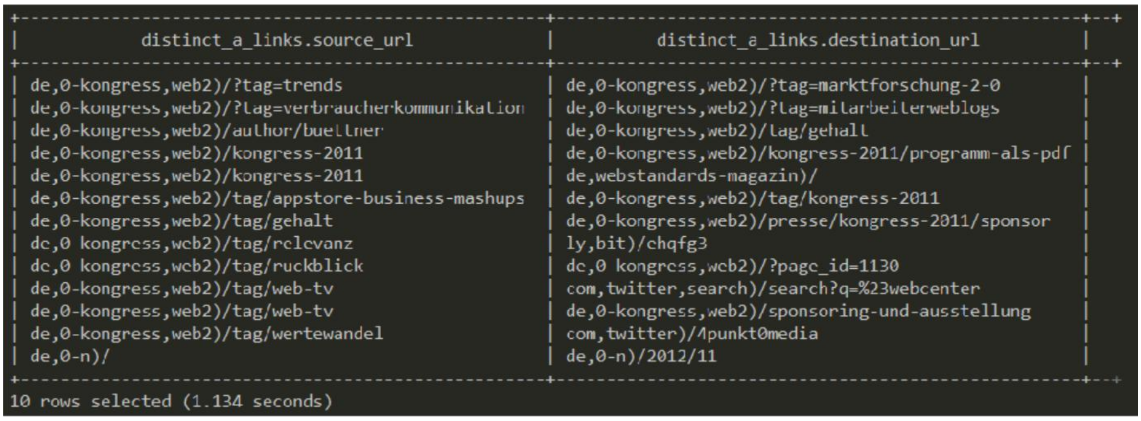
Parameter / Technique	HITS Algorithm	Semantic Similarity Algorithm	Proposed System
Time Efficiency	72%	87%	91%
Accuracy	79%	91%	95%
User specific Page Generation	No	No	Yes
Relevance Ratio	90%	92%	96%
High Relevance Ratio	30%	41%	51%

New ISM method can be integrated with any of the Page Ranking Algorithms to produce better and relevant search results.

Preparing dataset

On the input of the task, we have Hive table with two columns: source_url and desctination_url. Each row contains one edge of future graph (see Figure 1 for example of DB row).

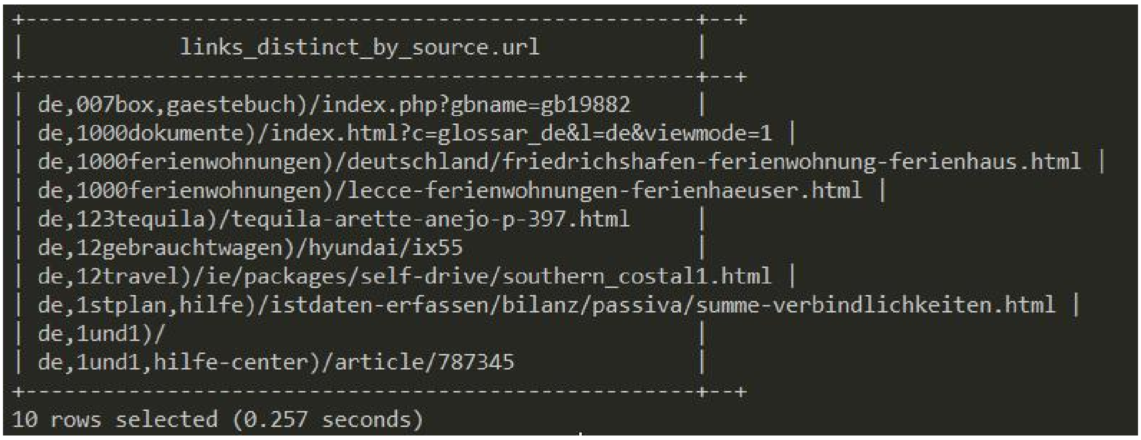
As the size of the graph is very big (more than 130 billion edges, about 10 terabyte size), first we need to optimize it. The first step was to extract all unique URLs from the table and replace them with IDs, then we will have much smaller graph where edges will be long to long instead of string to string.



```
+---------------------------------------------+---------------------------------------------+--+
|          distinct_a_links.source_url        |        distinct_a_links.destination_url     |
+---------------------------------------------+---------------------------------------------+--+
| de,0-kongress,web2)/?tag=trends              | de,0-kongress,web2)/?tag=marktforschung-2-0  |
| de,0-kongress,web2)/?tag=verbraucherkommunikation | de,0-kongress,web2)/?tag=mitarbeiterweblogs |
| de,0-kongress,web2)/author/buettner          | de,0-kongress,web2)/tag/gehalt               |
| de,0-kongress,web2)/kongress-2011            | de,0-kongress,web2)/kongress-2011/programm-als-pdf |
| de,0-kongress,web2)/kongress-2011            | de,webstandards-magazin)/                    |
| de,0-kongress,web2)/tag/appstore-business-mashups | de,0-kongress,web2)/tag/kongress-2011   |
| de,0-kongress,web2)/tag/gehalt               | de,0-kongress,web2)/presse/kongress-2011/sponsor |
| de,0-kongress,web2)/tag/relevanz             | ly,bit)/chqfg3                               |
| de,0-kongress,web2)/tag/ruckblick            | de,0-kongress,web2)/?page_id=1130            |
| de,0-kongress,web2)/tag/web-tv               | com,twitter,search)/search?q=%23webcenter    |
| de,0-kongress,web2)/tag/web-tv               | de,0-kongress,web2)/sponsoring-und-ausstellung |
| de,0-kongress,web2)/tag/wertewandel          | com,twitter)/1punkt0media                    |
| de,0-n)/                                     | de,0-n)/2012/11                              |
+---------------------------------------------+---------------------------------------------+--+
10 rows selected (1.134 seconds)
```

Figure 1. Sample results from distinct_a_links table

We extracted about 6 billion unique links (see Figure 2 for sample data), to rewrite out initial table from string to string to use short IDs of the pages we need very fast access to database which contains those IDs, therefore we decided to use in-memory databases.
Decoding URLs from SURT format.



```
+----------------------------------------------------+--+
|             links_distinct_by_source.url           |
+----------------------------------------------------+--+
| de,007box,gaestebuch)/index.php?gbname=gb19882     |
| de,1000dokumente)/index.html?c=glossar_de&l=de&viewmode=1 |
| de,1000ferienwohnungen)/deutschland/friedrichshafen-ferienwohnung-ferienhaus.html |
| de,1000ferienwohnungen)/lecce-ferienwohnungen-ferienhaeuser.html |
| de,123tequila)/tequila-arette-anejo-p-397.html     |
| de,12gebrauchtwagen)/hyundai/ix55                  |
| de,12travel)/ie/packages/self-drive/southern_costal1.html |
| de,1stplan,hilfe)/istdaten-erfassen/bilanz/passiva/summe-verbindlichkeiten.html |
| de,1und1)/                                         |
| de,1und1,hilfe-center)/article/787345              |
+----------------------------------------------------+--+
10 rows selected (0.257 seconds)
```

Figure 2. Sample results from distinct_links_all table

A However, 6 billion URLs still was a lot to fit in-memory (about 4 terabyte), we decided to store SHA-1 hash instead of URLs as key and ID long value as value for our Redis database.

During that work, we noticed that some of URLs has different formats, for example, some of them use http and some use https, some use www prefix and some not. We decided to remove such difference and added pre-processing of URLs which include:

1. Removing all unsafe ASCII characters, if they appears in domain names we replace them into Punycode domains and if such characters appears in path then replace them by using "%" followed by two hexadecimal digits.
2. Removing protocol prefixes, like: http, https.
3. Removing www prefixes.
4. Removing port numbers, like: 80, 443.

After replacing URLs to hashes, it has 21 153 collisions on our dataset. The hashes was extracted for investigation.

As we can see in table above, that is false negative results. In the middle column you can see URL as it stored in database, right column as it was normalized and left column is hash of the URL. As we can see, after normalization, we have same strings and as result same hashes. Also original URLs from database are not valid SURT format.

Tble 2. Hash collisions over links dataset

Hasah value	Original URL	Normalized URL
qEb3qCpvWQthRNLdkKTOiHInVmg=	de,merkspruch)/	merkspruch.de/
qEb3qCpvWQthRNLdkKTOiHInVmg=	de,merkspruch,)/	merkspruch.de/
ZLPnZaYBNeerff/5PH5ip3XXq40=	de,wuhletal,kirche)/	kirche.wuhletal.de/
ZLPnZaYBNeerff/5PH5ip3XXq40=	de,wuhletal,http://kirche)/	kirche.wuhletal.de/

Now when assigned short ID (long value) to each URL we need to update our whole dataset of pairs. That should significantly reduce size of it. Also, to create a graph we will need to go over all the pairs again. To decrease number of operations, we created module that read pairs, normalize URL and retrieve its ID from Redis database from previous task.

To create graph, we need to know incoming and outgoing links from each link. Unfortunately, hash table, which we used for previous task can’t store multiple values per single key. Therefore we used two new databases, one which store linkID and list of incoming linkIDs and other one with linkID and list of outgoing linkIDs.

The data stored in Hive database is unsorted, but to decrease number of operation on Redis server we need to order the database by source_url or destination_url depending of which table we want to fill. If the pairs are ordered for example by source_url, then during going through them we can collect all neighbors for same source_url (just compare if previous value is equal current) and merge destination_url values, put them to Redis in single operation.

As we already can access to incoming and outgoing links for any page, we can calculate hubs and authorities, which requires for HITS algorithm [Miller, 2001].

HITS ranking results

To evaluate HITS algorithm we use search results from Bing search engine. We have pre-saved results for all German Wikipedia articles. We selected most popular 3000 pages and used their title as search term. For each search term, we have about 100 search results from Bing.

For our HITS algorithm we use 100 pages from Bing as root set. Then we use base set of pages and all pages which linked or links to them as base set. For that base set we calculate authority and hubs values for thee steps. Then order pages from root set by authority and compare results with original Bing results. We also implemented PageRank algorithm for better evaluation of HITS results. It will add additional set of results to compare. The implementation of PR algorithm is very simple, we use 10 as default score for each page and split the score between all outgoing pages.

There are some limitations associated with Archived Data, for example the actives mainly contains German Internet pages (in .de domain zone), when Bing provides results regardless of the domain zone. But, we still have such pages in results because we have German pages that have outgoing links to different domain zones and we can calculate authority value for them.

The example results of applying HITS algorithm on search term “Kassel” you can see in the (Tab. 3), the comparison with PageRank score displays in (Tab. 4).

As we can see in (Tab. 4), the website of Kassel’s football team has lower PageRank score. That can be explained that in total the website has smaller number of incoming links, but the links is from better sources.

Table 3. Authority and hub values of HITS for search term “Kassel”

Link	Authority	Hub
Kassel Marketing | Tourismus-Informationen für Kassel kassel-marketing.de/	58929	148127497
Wetter Kassel - aktuelle Wettervorhersage wetteronline.de/wetter/kassel	46581	114631041
KSV Hessen Kassel e.V. - Die offizielle Homepage dasbesteausnordhessen.de/	45712	125635663
Kassel: Information für Kassel bei meinestadt.de home.meinestadt.de/kassel-documenta-stadt	45666	125741104

Table 4. HITS Authority and PageRank score

Link	HITS Authority	PageRank Score
Kassel Marketing | Tourismus-Informationen für Kassel kassel-marketing.de/	58929	0.09894597
Wetter Kassel - aktuelle Wettervorhersage wetteronline.de/wetter/kassel	46581	0.027334956
KSV Hessen Kassel e.V. - Die offizielle Homepage dasbesteausnordhessen.de/	45712	0.023083081
Stadtportal - Startseite www.kassel.de kassel.de/	45666	125741104
Kassel: Information für Kassel bei meinestadt.de home.meinestadt.de/kassel-documenta-stadt	45666	0.025142923
Stadtportal - Startseite www.kassel.de kassel.de/	45666	0.025142923

Some other results that illustrate the difference between HITS and PR are displayed in (Tab. 5). Also the results compared to Bing results in (Tab. 6).

Table 5. Results of HITS and PR for search term “Volkswagen AG”

Link	HITS Authority	PageRank Score
Volkswagen AG - Home - SSI SCHÄFER https://www.ssi-schaefer.com/de-de	184900	9.485999 (1)
VOLKSWAGEN AKTIEN News | 766403 Nachrichten… http://www.finanznachrichten.de/nachrichten-aktien/vo...	7462	0.10863955 (5)
Volkswagen Aktie | Aktienkurs | Chart | 766400 wallstreet-online.de/aktien/volkswagen-aktie	6456	0.025147859 (11)
Volkswagen Konzern Startseite volkswagenag.com/	2002	0.82785743 (2)
Volkswagen Personal volkswagen-karriere.de/de.html	1898	0.30051792 (3)

Table 6. Results of HITS and PR for search term “Volkswagen AG”

HITS results	Bing results
Volkswagen AG - Home - SSI SCHÄFER https://www.ssi-schaefer.com/de-de	Volkswagen Konzern Startseite volkswagenag.com/
VOLKSWAGEN AKTIEN News | 766403 Na… http://www.finanznachrichten.de/nachrichten-a...	Wie gut klingt das denn. volkswagen.de/de.html
Volkswagen Aktie | Aktienkurs | Chart | 766400 wallstreet-online.de/aktien/volkswagen-aktie	Volkswagen AG – Wikipedia de.wikipedia.org/wiki/Volkswagen_AG
Volkswagen Konzern Startseite volkswagenag.com/	Volkswagen Group Homepage volkswagenag.com/content/vwcorp/con…
Volkswagen Personal volkswagen-karriere.de/de.html	Volkswagen International de.volkswagen.com/de.html

Evaluating results

To evaluate results we created survey page, which contains ten results from Bing, and ten reordered results by using authority value of HITS algorithm. Also ten results of HITS with ten results of PageRank algorithm. The survey asks users to compare results which is more relative, as they think and make decision by clicking on one of the submit buttons on the bottom. Survey is available online for everyone, we asked some students to participate in and 31 people accept proposal. In average one-person answers for 50 topics, and 1541 in total. The results approximate expectations and Bing search engine provides better results than re-ranked results by using HITS algorithm. The results of that survey illustrated on (Fig. 3).

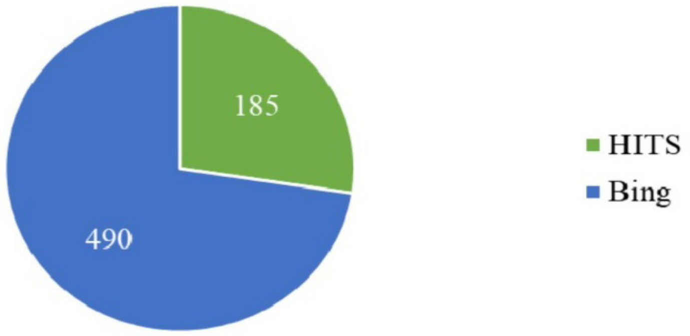


Figure 3. The results of survey HITS vs Bing results

Comparing results of HITS algorithm and PageRank algorithm (see Figure 4) give a little more points in favor of HITS algorithm.

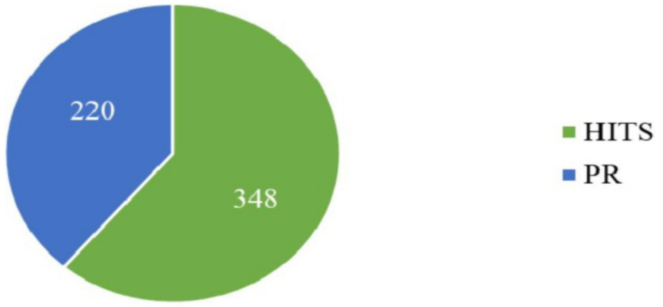


Figure 4. The results of survey HITS vs PR results

Comparing results of PageRank vs Bing (see Figure 5), gives most of the points to Bing results. It should be noted that in comparison with Bing results, HITS results take a bit more points than PageRank results.

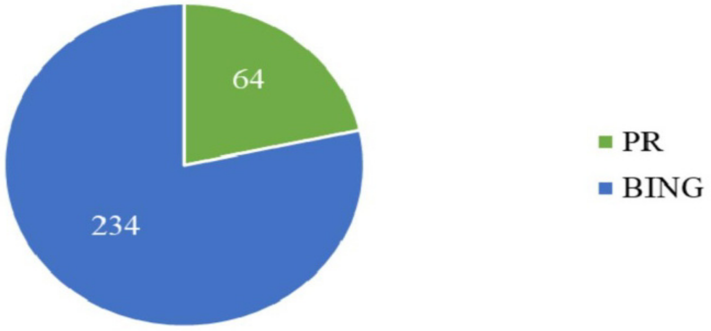


Figure 5. The results of survey PR vs Bing results

As we can see on table 6 the results of HITS and PageRank results worst results than Bing, six of ten top results contains pages with information about company shares. That pages appears on top due to specific content of the Web Archives that was used. For some other research purposes that archives contains a lot of pages with information about trades and shares.

In addition to that, some other results also contains very specific pages only for used Web Archives. Using a simple survey among unprepared users was not the best way to evaluate the quality of the results.

Conclusion

The aim of the work is to begin research in the direction and show some first results.

During the work was implemented two algorithms for ranking web results, the initial HITS algorithm was compared with PageRank algorithm.

Our results confirmed that modern search engines use very sophisticated technologies that include not only ranking algorithms, they also use AI and machine learning techniques to improve our daily Internet search experience.

Nevertheless, HITS algorithm that was developed slightly later than PageRank and using more depth scanning gives relatively better results. And that also gives us motivation to continue our work, we have plans to improve the results.

The amount of data that we have on input is very huge, and several first tries ware failed. We were need to experiment with different techniques and technologies to work with given data. Moreover, even now, when we have prepared relations graph, each iteration in the program must be justified, otherwise, everything works very slowly.

The search in Web Archives is not for everyday use and we do not expected that results will completely satisfy us. The key idea the work is research of additional attributes of web archives, unfortunately, we do not have enough time to present them in the work.

In the work, we finished only first part of our goal. Right now, we implemented simple HITS and PageRank algorithms, they allow us to make some small researches over retrieved data.

The next step will be to include first crawl date and last crawl date into HITS algorithm. Potentially we can find some hubs that existed before, but no longer exists today. By using those properties, we also can know age of the pages and we do not know how it change results.

Acknowledgement

The research was done with the help of L3S Research Center, Hannover, Germany.

Bibliography

[Bansal, 2014] N. Bansal, S. Paramjeet. Improved Web Page Ranking Algorithm Using Semantic Similarity and HITS Algorithm. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), 2014. pp. 346-348. http://www.ijettcs.org/Volume3Issue4/IJETTCS-2014-08-26-146.pdf

[Sharnagat, 2014] R. Sharnagat. Named Entity Recognition: A Literature Survey, 2014. 27 p. https://pdfs.semanticscholar.org/83fd/67f0c9e8e909dc7b90025e64bde0385a9a3a.pdf

[Ridings, 2002] C. Ridings, M. Shishigin. Pagerank Uncovered, Technical report, 2002. 56 p. http://www.voelspriet2.nl/PageRank.pdf

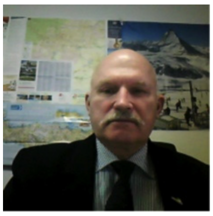
[Miller, 2001] J. C. Miller, G. Rae, F. Schaefer, L.A. Ward, T. LoFaro, & A. Farahat. Modifications of Kleinberg's HITS algorithm using matrix exponentiation and web log records. Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, 2001. pp. 444-445. https://dl.acm.org/citation.cfm?id=384086

Authors' Information

Oleksii Vasylenko *– Aspirant of Kharkiv National University of Radioelectronics; Kharkiv, Ukraine;*

e-mail*: ichbierste@gmail.com tel.: +380 63 841 66 23*

Major Fields of Scientific Research*: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*



Prof. Dr.-hab. Oleksandr Kuzomin *– Informatics chair of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine Ukraine;*

e-mail*: kuzy@daad-alumni.de tel.: +38(057)7021515*

Major Fields of Scientific Research*: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics.*

Artem Mertsalov *– Master student in Information and Communication technologies of Kharkiv National University of Radio Electronics; Kharkiv, Ukraine*

e-mail*: khripushinka@gmail.com*

Major Fields of Scientific Research*: General theoretical information research, Knowledge Discovery and Engineering, Business Informatics.*