International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

Collecting and Analyzing Academic Publication with UTeMAIR

Zurina Saaya, Yogan Jaya Kumar, Lim Boon Hee

Abstract: ***Academic publication has been one of the main requirements in academic research and university ranking to seek research funding and scholarship. Mostly, the publication is published as academic journal article, book or thesis in offline and online distribution. There are variety of tools and services available today to understand academic publication data. However, existing tools is still very limited in the terms of specific institute and academics. In this study, we propose a practical tool, UTeMAIR which retrieve publication information for academic staff of specific institute from online publication repositories sites such as Google Scholar and Scopus. Specifically, UTeMAIR consists of three main components namely, crawling engine, statistical analysis and keywords analysis. The objectives of this system to retrieve and store publication record of academic staff, to continuously update the collected information and to analyze publication data for all academic staff. With the availability of an effective retrieval and analysis tool for publication data, the university can monitor scholarly information in a better way and plan towards increasing the publication index among academics and ultimately improve university visibility.***

Keywords : academic-focused web crawler, retrieval engine, academic publication.

I. INTRODUCTION

Academic electronic information archival has evolved far beyond the simple days of creating contents and storing them into local repository systems. Nowadays, most of this academic information is available in one of many online sites that indexes the full text or metadata of scholarly literature. For instance, information about an author's academic publications can be retrieved from Google Scholar website and Scopus. Google Scholar also keeps record on the citation index for each of the publication. The citation index can further give the h-index scores of the authors. h-index is the highest number h where h publications have at least h citations. Such information is essential to all academic institutions to measure the research competency at the institution level. The Malaysia Research Assessment Instrument (MyRA), is one such instrument used to evaluate the capacity of research in Higher Learning Institutes (HLI) in Malaysia [1].Thus, the academic institute also needs to gather and update all these records to keep track of its research progress and standings among other HLIs.

Currently, in University Teknikal Malaysia Melaka (UTeM), the publication record is acquired through an internal research information system, which is a platform that keeps record of all research details including publication information. However, this system requires staff to manually submit their publication details into the database. This could lead to several issues such as inefficient data collection, missing data, and delayed submission. At present, search engines and digital libraries are useful for searching publication records for all staff, however, they do not include additional analysis tools required to summarize their articles and publication performance. Tool with visualizations feature may help users to easily understand the data make effective decision, such as which research areas are having bright future directions for academic staff, or whether it is appropriate to increase or reduce funding by the research management center or even provide opportunities for investment by industries to invest in a particular research area.

In this study, we propose to build a tool called UTeM Academic Information Retrieval (UTeMAIR) to retrieve the academic publication information from online sites such as Google Scholar and Scopus then analyze these data to produce statistical results for a specific academic, department, faculty or university. With the availability of an effective retrieving tool for publication data, the university can monitor scholarly information in a better way and plan towards increasing the publication index among academics, attract more industries funding or investment and ultimately improve its ranking.

II. RELATED WORK

Academic publication is the main requirement in most academic research and scholarship. Mostly, the publication is published as academic journal article, book or thesis in offline and online distribution. Scopus is an example of database for journals, conference proceedings and books. Besides searching tool, Scopus also provides features to analyze and visualize research topics. Google Scholar is another example of academic publication search engine. It indexes the full text or metadata of scholarly publication from different types of publishing formats and disciplines.

Manuscript published on November 30, 2019.
* Correspondence Author
Zurina Saaya*, Center for Advanced Computing Technology (C-ACT), Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Malaysia, Email: zurina@utem.edu.my
Yogan Jaya Kumar, Center for Advanced Computing Technology (C-ACT), Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Malaysia, Email: yogan@utem.edu.my
Lim Boon Hee, Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Malaysia, Email: b031310167@student.utem.edu.my

These databases and search tools basically provide an organized list of articles based on search query entered by user [20] Furthermore, there are also domain-specific databases available such as PubMed for biomedical topics and life sciences. For Computer Science and Informatics domain, there are IEEE Xplore and ACM Digital Library. Most scholarly database systems provide publication searching and return the search result consists of list of articles, but not many can provide sufficient features to explore the search results. Normally, the search result is sorted and filtered based on publication date, author name, author affiliations, keywords, search term relevance and citation frequency.

Web crawling is one of the many techniques that can be used to collect and extract information from any web-based applications [9]. Web crawler is an automated robot software that traverse the Web and downloads its content. The content of webpages is then sorted and indexed. The two important challenges of web crawler are; to handle the large volume of web pages and its rate of change [5]. Crawling involves complicated reliability and performance issues since data on the Internet is massive. The crawler has to fetch all the relevant data and at the same time needs to ensure that the existing fetched data remains current [5]. Martijn Koster was one of the early inventors of web crawler that used the combination of human and automated effort to traverse the Web. The web crawler is called ALIWEB (Archie Like Indexing of the WEB) which was made available publicly in October 1993. Built base on ALIWEB, the next generation of robots began to evolve with fully automated features [6].

There are five main types of crawlers namely parallel crawler, distributed crawler, focused crawler, incremental crawler and hidden web crawler where each of them implements a different technology and algorithm. For example, parallel crawler is the technology where search engines run multiple processes in parallel so that its download rate is maximized [10]. UbiCrawler is one example of distributed crawler where every task for the crawler is distributed to several agents (crawlers)[9]. Specifically, each URL is assigned to a specific agent at any time, which is the only one responsible for it. Incremental Crawler in the type of crawler that selectively and incrementally updates its repositories, rather than refreshing from time to time the collection in bundle mode. Ideally, the incremental crawler ensure the collection is up to date and bring in new pages in a on time [10].

In this research, we developed a focused crawler to collect scholarly publication for academic staff i.e. an academic-focused web crawler. Focused crawler is a kind of web crawler that selectively traverse webpages that are related to a pre-defined domain [7]. It looks for the URLs that are relevant for crawling and ignores irrelevant parts of the Web. Specifically, we need to determine a list of seed URLS to the focused crawler. The seed URLs are the initial URLs from which a crawler starts to execute the crawling engine. There are several open source crawlers which can be implemented in different programming languages. It can be customized based on user requirements. One example of such kind of crawlers is the Citeseerxbot, which is a crawler for CiteSeerX search engine [8]. CiteSeerX is an online digital library and search engine that has focused primarily on the publication in computer and information science. CiteSeerXbot crawler traverse all web pages by research institutions, such as home pages with university domains and public online publishers.

Besides crawling publication data, analyzing the data can provide further necessary information. Academic publication usually includes lots of metadata data, such as article titles, authors, keywords and citations. There is a number of visualization approaches can be easily implemented on academic publication data, which allows research organization to easily understand the content and context of academic data sets and disclose hidden patterns in the publication data for their scholars. Liu et al. did a research on established visualization approaches for publication data [35]. They list number of prominent approaches such as chart, word cloud and maps. Fig. 1 shows an example of visualization for keywords data.

Fig. 1. Sample keyword visualization from carrot2 search engine

In relation to keywords analysis, in particular, recognizing keywords for an article, a method which can be applied is using text categorization or supervised approach. In this way, we can assume that all potential keywords can be categorized into a predefined controlled vocabulary [16]. The other method for keyword analysis is keyword extraction or unsupervised approach. It is not restricted to a set of suggested keywords from a selected vocabulary. Instead, any phrase in a new article can be extracted as a keyword. Natural Language Processing (NLP) technique can improve the analysis of a collection of documents. This technique can detemine frequency of terms or phrases, recognize topics, and identify main concepts. Document clustering and multi-document summarization can help users to understand collection of documents by providing an automated descriptions and topics [15] [20]. The extraction approach has been used by Gong and Liu in their text summarization technique using Latent Semantic Analysis (LSA) [13].

Basically, LSA is used to identify semantically important sentences, for summary creations from input documents.

Besides LSA, Latent Dirichlet Allocation (LDA) has been applied in topic exploring domain. Savage et al. use LDA in TopicXP, a tool that support developers during software maintenance tasks [17]. LDA is used to provide developers a brief of a software system under analysis by visualizing and extracting natural language topics from source code, variable names and comments. In another related study, Momtazi and Naumann used LDA approach in their expert finding research, where the LDA is used to extract the main topics from a collection of scholarly articles. Then the extracted topics is used to show the relationship between expert candidates and user queries [18].

Named Entity Recognition (NER) is another approach for information extraction tasks. It concerns with identifying keywords with respect to names of entities such as people, organizations, products, locations, numeric expressions including time, date, currency and percent expressions. During early days of NER approach, the focus is to extract information from unstructured text, such as articles from newspaper in order to form a structured information of organization activities and defense related activities [24]. Until today, NER still prominent approach in NLP research area. NER has been used in many domains for example in analyzing data form social media platform [19][25]. chemistry [26] and biomedical [27].

In this research, we have compared these three approaches as a way to find the most accurate way to analyze relevant keywords for individual academic staff based on their publication data. The input document for keywords analysis is the articles metadata i.e. titles and abstracts of the article written by the individual staff.

III. SYSTEM OVERVIEW

This section discusses the main system design and architecture of UTeMAIR. The main objectives of this system are as following (1) to retrieve and store publication record of academic staff; (2) to continuously update the collected information, (3) to analyze publication data for all academic staff. The database is setup to keep all the publication information of academic staff for a specific university; in this case UTeM. The data form this database is then used for publication analysis. Similar approach can also be used for other universities.

The desired deliverables of this research will be (1) scholarly publication database of academic staff; (2) focused crawler that will collect the desired information. Furthermore, it will ensure the database information is updated; (3) dashboard that is able to summarize and visualize the publication data that are gathered by the crawling tool. There are three main components in UTeMAIR, namely, crawling engine, statistical analysis and keywords analysis. Each of these components are described in the following sections.

A. Crawling Engine

UTeMAIR crawling engine is a focused web crawler which retrieve data from multiple online publication repositories namely Google Scholar and Scopus. The data is then analyzed using statistical and topic analysis approach to help the university research management centre to have better understanding in publication information among their academics. UTeMAIR differs from the existing focused web crawler namely Citeseerxbot, since it specifically crawls for requested list of authors affiliated to a specified academic institute. The crawled data is then organized into structured data of academic staff publication records which is stored in publication repository.

UTeMAIR system design in Fig. 2 shows that the crawling engine can download webpages from the Scopus and Google Scholar based on specified keywords. The keywords are a list of author id. These author ids are collected manually from all staff as we need a valid id from each staff for crawling purpose. There are situations where a staff has more than one author id, therefore we need to verify the author id and their publication to avoid undesired data replication.

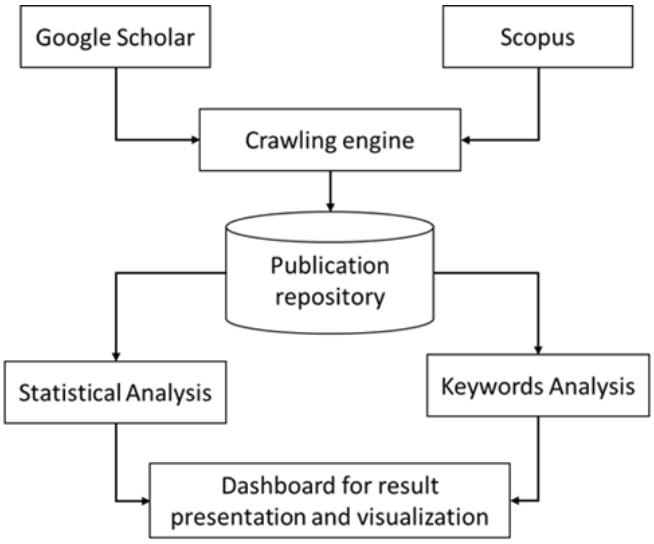

Fig. 2. UTeMAIR System Design

The scheduler will trigger the crawling engine to run on a timely basis to update the information in the local repository which consists of publication database. To avoid any denial of service or being blacklisted by the web server (data source), the crawler will be set to retrieve the data according to the robot exclusion standard. This standard is used as guidelines for the crawler to know which page can be processed and the pages that should not be processed or scanned. The information stored in the local repository will then be processed into structured data to be accessed by the dashboard for data analysis and reporting.

The crawler will generate the citation page of the given author id to retrieve the publication information. As shown in the flow chart in Fig. 3, the crawling engine will retrieve the citation page for each author id which is used as seed keyword. For each citation page, the crawler will download its content to gather the details of publication data for each author. The crawler will perform pre-processing on the retrieved pages to clean the extracted content and perform analysis to identify unique data to avoid duplications.

B. Publication Statistical Analysis

To have better understanding of the data, simple statistical engine is built to summarize all publication data.

For example, publication data is categorized based on author's department and author publication information is aggregated based on citation and h-Index. Furthermore, this statistical engine able to rank author based on publication and summarize departmental publication in the form of average h-Index and total publications.

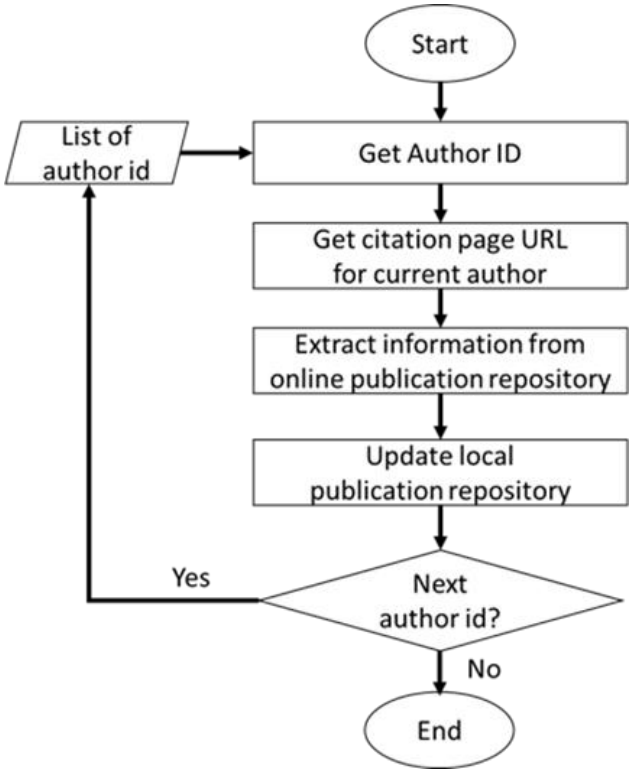

Fig. 3. UTeMAIR System Flow for Data Crawling

C. Publication Keywords Analysis

Topic analysis is a type of statistical approach to discover the abstract topics in a set of documents. Specifically, topic analysis on scholarly publication approach is used to determine prominent research areas among academics in the university. To identify the topics related to each staff, we apply several topic analysis approaches namely, Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA) and Named Entity Recognition (NER).

LDA and LSA both describe mathematical models that are designed to be used for information retrieval. LSA examines the words used in a document and looks for their relationships with other words. Basically, it is capable of capturing and modeling interrelationships among words so that it can semantically cluster it [13]. On the other hand, LDA is an extension of LSA where a set of significant words from a document are grouped into a number of topics to determine which topic is more relevant to the document [14]. There many open source LDA and LSI platform that are ready to use by researcher. For example, Stanford Topic Modeling Toolbox for LDA and carrot2 [33] for LSI.

NER is mostly used in information extraction process. It basically extracts significant information such as name of persons, organization, locations, quantities, expressions of times, currency values, percentages and many more. There are many notable platforms for NER implementation, namely, GATE (General Architecture for Text Engineering), OpenNLP, Stanford NER and Python Natural Language Toolkit (NLTK). In our LDA and LSI implementation, we choose to use genism, a topic modelling library in Python. For NER implementation we use NLTK which also provides library in Python.

IV. EVALUATION AND RESULTS

A. Crawling Performance

For the evaluation of crawling engine, we captured the duration of crawling for seven set of authors which in each set consists of 20 authors. From this experiment we can conclude the average of time or duration taken to crawl each author as each author has different number of publication as it will also reflect the duration to gather their information from the resources. The publication data gathered by the crawler is stored in local repository. The crawling duration captured during evaluation phase is depicted in Table 1. These durations are based on the crawling data for every 20 authors and during the crawling activity the engine is put into sleep mode for random n seconds [12]. Average duration of crawling for 20 authors is 1 hour 12 minutes or 3.6 minutes for every author.

Table 1. Crawling duration

No. of author	Start time (hh:mm)	End time (hh:mm)	Duration (hh:mm)
20	9:50:00	11:13:00	1:23
20	12:07:00	12:58:00	0:51
20	13:28:00	14:40:00	1:12
20	14:40:00	16:15:00	1:35
20	17:10:00	19:00:00	1:50
20	8:02:00	8:49:00	0:47
20	9:59:00	10:47:00	0:48
		Average	1:12

B. Statistical Analysis

Evaluation for statistical analysis is done manually by comparing the data from its resources, i.e. Google Scholar and Scopus to ensure that they are depicted correctly in the dashboard. The retrieval engine is used for retrieval and is used by the dashboard for analysis and reporting. For example, as shown in Fig. 4, the dashboard can generate the summary of publication data for all academic staffs. Particular in this scoreboard, it shows the total author, average h-Index and total publication for current year for the university.

Meanwhile, in Fig. 5 the publication data is analyzed to produce a scoreboard for individual authors. Besides that, the dashboard can also visualize the data using graphs and charts as in Fig. 6. and Fig. 7. As we can see in Fig. 6, total publication is shown based on department (faculty) and Fig. 7. shows the total publication based on year.

This way of presentation is useful for the university to monitor scholarly information in a better way and plan towards increasing the publication index among academics and ultimately improve the university ranking.

Fig. 4. Sample result (Scoreboard) from Statistical Analysis component

Author Name	Department	Citations	H-index	i10-index	Last Update
ABDUL KADIR	FTK	5368	24	33	2017-03-21
RABIAH AHMAD	FTMK	1289	18	25	2017-03-24
MOHD JAILANI MOHD NOR	FKM	1318	17	23	2017-03-18
M.Z.A ABD AZIZ	FKEKK	994	16	34	2017-03-20
KENNETH SUNDARAJ	FKEKK	793	15	24	2017-03-19
M.A. OTHMAN	FKEKK	1016	15	30	2017-03-21
MOHD RUDDIN AB GHANI	FKE	987	15	17	2017-03-24
ABDUL RANI OTHMAN	FKEKK	627	13	18	2017-03-24
CHIN KIM, GAN	FKE	457	13	14	2017-03-22
AUZANI JIDIN	FKE	578	12	16	2017-02-10

Fig. 5. Sample Result (publication summary data for each author)

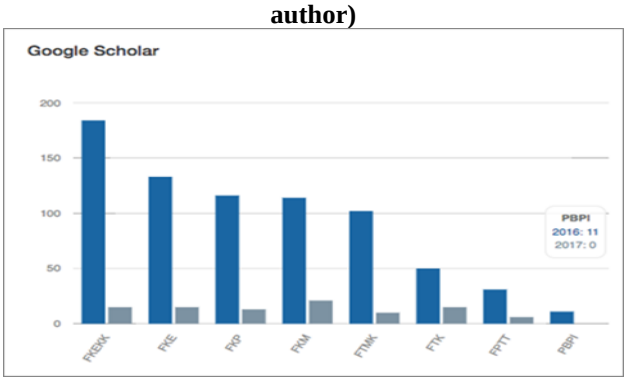

Fig. 6. Sample result (number of publications by faculty)

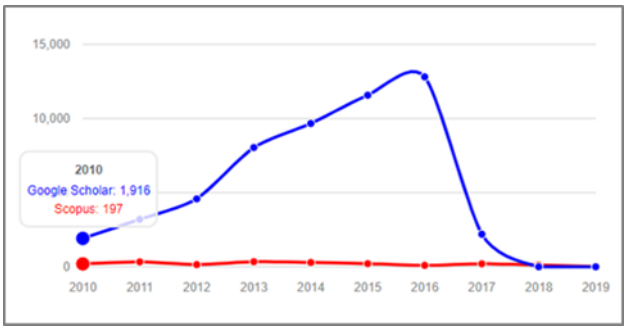

Fig. 7. Sample result (number of publications per year)

C. Evaluation on Keywords Analysis

The next evaluation is done for keywords analysis. There are three topic modelling approaches that has been used in this experiment namely LDA, LSA and NER. In particular, we evaluate the overall accuracy of keywords analysis. The data used for this evaluation is gathered from one of the faculties in UTeM. For the purpose of the evaluation, we limit our interest to those staff that have minimum one publication and have minimum 10 user-defined keywords for their publications, which we can use as the ground-truth against which to judge the keywords analysis approaches.

The test dataset covered 24 staffs and their list of publication which include publication titles, abstracts and user-defined keywords. Summary of dataset used in this experiment is depicted in Table 2. As shown Table 2 on average each author has 10 publication or articles and about 33 user-defined keywords. Maximum number of publications for each author in our test dataset is 29.

Table 2. Summary of dataset use in experiment

	No. of publications	No. of keywords
Average	10	33.5
Median	7	23.5
Standard deviation	8	25.5
Minimum	1	10
Maximum	29	100

For each author (staff) we gather all of their publication titles and abstract text used as input in keywords analysis engine. This engine will then produce list of prominent keywords that are related to the current author. Each author will be assigned to top 10, 15 and 20 keywords generated by keywords analysis engines. To evaluate each modelling approach, the keywords that are generated by keywords analysis engine will be compared against user-defined keywords. Next, we compute accuracy results or the number of time that user-defined keyword match with the generated keywords. The computation of accuracy results does not consider the order of keywords, but it just focuses on whether any of the generated keywords match with the user-defined keywords.

Fig. 8, shows the accuracy results of each modelling namely, NER, LDA and LSA. The experiment is divided into three set of keywords sizes, 10, 15 and 20. We see that all three approaches generate more than 60 percent accuracy for all sizes. Accuracy for size 10 for NER is the highest among other approaches which is 74 percent. Based on the result NER would be the best topic modeling and the suitable size of keywords is 10 for each author.

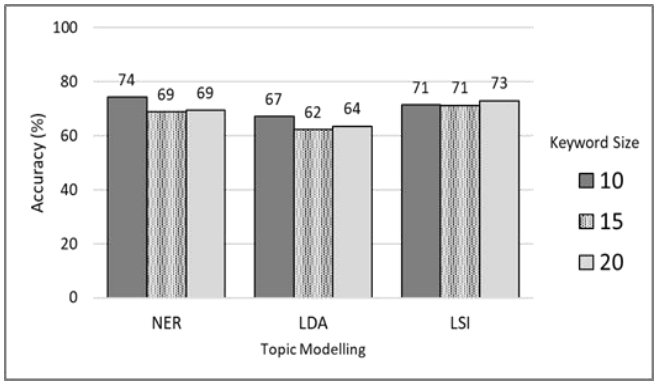

Fig. 8. Accuracy results

The output from keywords analysis is then displayed in UTeMAIR dashboard as in Fig. 9. From this figure we can see top 10 keywords that are prominent for each author.

The keywords can be viewed based on department (faculty) and depicted in Fig. 10. These findings can provide useful information regarding the main research areas of the university and its research direction.

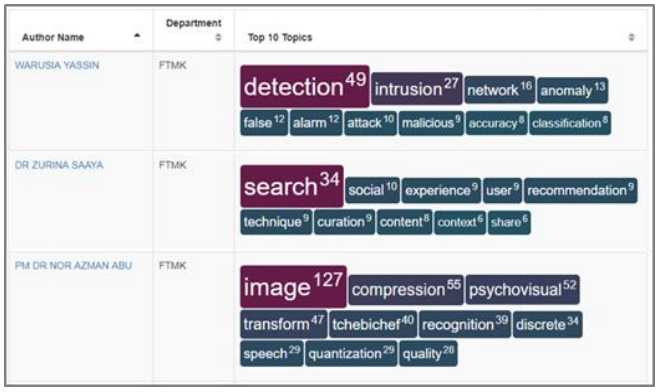

Fig. 9. List of prominent keywords for each author

Fig. 10. List of popular keywords from one department

V. CONCLUSION

We have described the development of UTeMAIR, tool to collect and analyze academic publication data. The publication data were extracted from two popular online publication repository sites i.e. Google Scholar and Scopus. In designing UTeMAIR, our primary goal is to gather publication data for a list of authors affiliated to an academic institute.

By employing the proposed crawler, universities can now keep track and update their academic publication information automatically to monitor the performance of their academic staff particularly for their research and publication assessment. Statistical analysis and keywords analysis module organize information through abstractions and aggregation as shown in Section IV. This can help university management i.e. research center unit finding patterns and important information regarding staff publication. With the availability of this information, the university can monitor scholarly information in a better way and plan towards increasing the publication index among academics and ultimately improve the university ranking.

The results from evaluation shows the performance of crawler engine with reasonable duration of crawling for individual author. Since crawlers are computer programs that traverse the web with the goal of automating specific tasks related to the web, there exist issues regarding denial of service. To avoid any denial of service by the web server, the crawler engine is set on sleep mode for a random n second. This approach is important so that the standards and ethics in web crawler are met. The "politeness" of the crawler has also been ensured to avoid blacklisting of the crawler's IP address that could cause future problems to the users in the network. However, UTeMAIR crawler must be scalable towards the growth in the number of academic staff. It must be able to retrieve a large amount of information as the number of staff increases. Therefore, future enhancement needs to be done to handle this issue.

As for the future work, beside improvement in crawling engine, we will consider validating the collection against other publication database such as IEEE Xplore and ACM Digital Library. Ultimately, we hope to develop an expert search engine based on author's publication data specifically for UTeM's academics.

ACKNOWLEDGMENT

The authors would like to thank to Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Malaysia (UTeM), Centre for Advanced Computing Technology (CACT) and Computational Intelligence and Technologies (CIT) research group for their incredible supports in this project.

REFERENCES

1. Yunus, A. S. M., & Pang, V. "Academic Promotion in Malaysia: Meeting Academics Expectation and Institutional Needs". *RIHE International Seminar Reports*. No. 23. Research Institute for Higher Education, Hiroshima University, 2015.
2. Google Scholar (2019), Retrieved from https://scholar.google.com/intl/en/scholar/about.html
3. Boldi, Paolo, et al. "Bubing: Massive crawling for the masses." *Proceedings of the 23rd International Conference on World Wide Web*. ACM, 2014.
4. Castillo, Carlos. "Effective web crawling." *ACM SIGIR Forum*. vol. 39, no. 1. ACM, 2005.
5. Yoke Chun, Tham. "World wide web robots: an overview." *Online and CD-Rom Review* vol. 23, no 3, pp. 135-142, 1999.
6. Kuyoro Shade, O., O. Okolie Samuel, and U. Kanu Richmond. "Trends in web-based search engine." *Journal of Emerging Trends in Computing and Information Sciences*, vol. 3, no. 6, 2012.
7. Cho, Junghoo, and Hector Garcia-Molina. "Parallel crawlers." *Proceedings of the 11th international conference on World Wide Web*. ACM, 2002.
8. Wu, Jian, et al. "The evolution of a crawling strategy for an academic document search engine: whitelists and blacklists." *Proceedings of the 4th Annual ACM Web Science Conference*. ACM, 2012.
9. Boldi, Paolo, et al. "Ubicrawler: A scalable fully distributed web crawler." *Software: Practice and Experience*, vol. 34, no. 8. pp. 711-726 2004.
10. Cho, Junghoo, and Hector Garcia-Molina. "The evolution of the web and implications for an incremental crawler". Stanford, 1999.
11. Chakrabarti, Soumen, Martin Van den Berg, and Byron Dom. "Focused crawling: a new approach to topic-specific Web resource discovery." *Computer Networks*, vol. 31, no. 11, pp.1623-1640, 1999.
12. Thelwall, Mike, and David Stuart. "Web crawling ethics revisited: Cost, privacy, and denial of service." *Journal of the American Society for Information Science and Technology*, vol. 57, no. 13, pp. 1771-1779, 2006.
13. Gong, Y., & Liu, X. Generic text summarization using relevance measure and latent semantic analysis. In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 19-25). ACM, September 2001.

14. Blei, D. M., Ng, A. Y., & Jordan, M. I. "Latent dirichlet allocation." *Journal Of Machine Learning Research*, 3(Jan), 993-1022, 2003
15. Agarwal, N., Gvr, K., Reddy, R. S., and Rosé, C. P. "Towards multi-document summarization of scientific articles: making interesting comparisons with SciSumm*". In Proceedings of the Workshop on Automatic Summarization for Different Genres, Media, and Languages*, pp. 8-15. Association for Computational Linguistics, June 2011.
16. Caragea, C., Bulgarov, F., & Mihalcea, R. "Co-training for topic classification of scholarly data." *In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. pp. 2357-2366, 2015.
17. Savage, T., Dit, B., Gethers, M., & Poshyvanyk, D. "Topic XP: Exploring topics in source code using latent Dirichlet allocation." *In 2010 IEEE International Conference on Software Maintenance*. pp. 1-6. IEEE. September 2010.
18. Momtazi, S., & Naumann, F. "Topic modeling for expert finding using latent Dirichlet allocation" *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(5), 346-353. 2013.
19. Derczynski, L., Maynard, D., Rizzo, G., Van Erp, M., Gorrell, G., Troncy, R. and Bontcheva, K. "Analysis of named entity recognition and linking for tweets", *Information Processing & Management*, 51(2), 32-49, 2015.
20. Dunne, C., Shneiderman, B., Gove, R., Klavans, J., & Dorr, B. "Rapid understanding of scientific paper collections: Integrating statistics, text analytics, and visualization", *Journal of the American Society for Information Science and Technology*, 63(12), 2351-2369. 2012.
21. United States National Library of Medicine. (2019). *PubMed*. Retrieved from http://ncbi.nlm.nih.gov/pubmed
22. Association for Computing Machinery. 2019. *ACM Digital Library*. Retrieved from https://dl.acm.org
23. Institute of Electrical and Electronics Engineers. 2019. *IEEE Xplore*. Retrieved from http://ieeexplore.ieee.org
24. Nadeau, David, and Satoshi Sekine. "A survey of named entity recognition and classification." *Lingvisticae Investigationes* 30.1 (2007): 3-26.
25. De Oliveira, Diego Marinho, Alberto HF Laender, Adriano Veloso, and Altigran S. da Silva. "FS-NER: a lightweight filter-stream approach to named entity recognition on twitter data." *In Proceedings of the 22nd International Conference on World Wide Web*, pp. 597-604. ACM, 2013.
26. Rocktäschel, Tim, Michael Weidlich, and Ulf Leser. "ChemSpot: a hybrid system for chemical named entity recognition." *Bioinformatics* 28.12 2012: 1633-1640.
27. Tang, Buzhou, et al. "Evaluating word representation features in biomedical named entity recognition tasks." *BioMed research international 2014*, 2014.
28. The University of Sheffield, *GATE (General Architecture for Text Engineering)*, 2019. Retrieved from https://gate.ac.uk
29. The Apache Software Foundation, *OpenNLP*, 2019. Retrieved from https://opennlp.apache.org
30. Stanford University, *Stanford Named Entity Recognizer (NER)*, 2019. Retrieved from https://nlp.stanford.edu/software/CRF-NER.html
31. NLTK Project, *Natural Language Toolkit*, 2019. Retrieved from https://www.nltk.org
32. Stanford University. *Stanford Topic Modeling Toolbox*, 2019. Retrieved from https://nlp.stanford.edu/software/tmt/tmt-0.4/
33. Stanislaw Osinski and Dawid Weiss, *carrot2 open source framework for building search clustering engines*, 2019, Retrieved from https://project.carrot2.org
34. Radim Řehůřek and Petr Sojka, *genism topic modelling for human*, 2019, Retrieved from https://radimrehurek.com/gensim
35. Liu, Jiaying, et al. "A survey of scholarly data visualization." *IEEE Access 6*, 2018: 19205-19221

AUTHORS PROFILE

Zurina Saaya is a senior lecturer in Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM) where she involved with teaching computer networking topics. She has been teaching for almost 15 years. She completed her PhD in Computer Science and Informatics from Universiy College Dublin, Ireland in 2014. Her research focuses on technologies for information retrieval, data mining and recommender systems. She has published more than 10 publication topics related to information retrieval, recommender system and sentiment analysis.

Yogan Jaya Kumar is a senior lecturer at the Department of Intelligent Computing and Analytic in the Faculty of Information and Communication Technology (FTMK), Universiti Teknikal Malaysia Melaka (UTeM). He completed his PhD studies at Universiti Teknologi Malaysia, in 2014 in the field of Computer Science. Currently, his research involves in the field of Text Mining, Information Extraction and AI applications.

Lim Boon Hee is an alumni of Universiti Teknikal Malaysia Melaka (UTeM). He has graduated from UTeM with Bachelor Degree in Computer Science majoring in computer networking in 2017. Currently he is working as software engineer in Kuala Lumpur Malaysia. Some part of this article is written based on his Undergraduate Final Year Project.