Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 116 (2017) 645–651

Procedia Computer Science

www.elsevier.com/locate/procedia

2nd International Conference on Computer Science and Computational Intelligence 2017, ICCSCI 2017, 13-14 October 2017, Bali, Indonesia

Development of Review Rating and Reporting in Open Journal System

Willy[1], Wirya Samatha Priatna[1], Sonya Rapinta Manalu[1*], Arta Moro Sundjaja[2], Noerlina[2]

[1]*Computer Science Department, School of Computer Science, Bina Nusantara University, Jl. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta, Indonesia*

[2]*Information System Department, School of Information System, Bina Nusantara University, Jl. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta, Indonesia*

Abstract

This article describes the development of review rating and reporting features for Open Journal System (OJS), an open-source journal publishing platform used by more than 10,200 journals all over the world. Journal editors are able to receive rating recommendation automatically as a decision support in grading the review quality of peer reviewers. In addition, reporting features are also implemented to facilitate publishers in documenting their journals. Testing and evaluation show that both features are effective to be used in OJS-based journals.

Peer-review under responsibility of the scientific committee of the 2nd International Conference on Computer Science and Computational Intelligence 2017.

Keywords: peer review; rating; Open Journal System; report; recommender system

1. Introduction

In the world of academia, peer reviewing is a widely accepted means to evaluate a scientific work by experts in

* Corresponding author. Tel.: +62 21-534-5830; fax: +62 21 530-0244.

E-mail address: smanalu@binus.edu

1877-0509
Peer-review under responsibility of the scientific committee of the 2nd International Conference on Computer Science and Computational Intelligence 2017.
10.1016/j.procs.2017.10.035

the same field (peer reviewers), in order to determine and ensure that the submitted work is qualified to be published. A publication that has been peer reviewed is considered a relevant contribution to the field [1]. Hence, it is crucial to ensure that reviews are effective in assessing the quality of an article, and thus helping authors to improve their work. Open Journal System (OJS) is an open-source journal publishing platform created in 2002 by the Public Knowledge Project, headed by John Willinsky. OJS is capable of running the entire journal publishing workflow, starting from the article submissions process, peer reviewing, editing, to publications [2]. In reviewing a submission, OJS provides a way to assess the quality of peer reviewers by means of rating. Each review by peer reviewers will be scored by editors and be given a discrete rating between one and five. Journal publishers will then be able to use this information to selectively keeps competent peer reviewers, hence improving the journal's quality [1]. However, while some journals provide a basic guideline, no unified standard exists and peer reviewing practices still vary from one scientific journal to another [3]. Therefore, it's currently still a difficult process to assess a review objectively.

This study proposes a recommendation system for OJS, which automatically estimate the quality of reviews. Specifically, the system will show a summarization of the review and a score estimation of the review, complete with a breakdown analysis of scoring analysis. A summarization of review will be generated using the TextRank algorithm [4,5] to show editors a quick information regarding the quality of the article. The score estimation itself is generated by using structural features of the review text [6], adjusted using previously rated peer reviews as the standard. This estimation will be used as a decision support for editors to determine the review's final score. Finally, as more articles and reviews are submitted, journal publishers need a way to efficiently document those information for various purposes, such as the aforementioned peer reviewer selection, journal accreditation, et cetera. While OJS provides several default reporting tools, they aren't comprehensive and several significant information aren't included, such as reviewers' average rating performance. Therefore, new reporting features are also developed to provide an efficient way for OJS users to collect their journals' data. All features developed will then be able to be installed as a plugin application for OJS-based journals.

2. Methodology

The research steps taken in this paper are as follows: requirement gathering through direct observation, interviews, and Focus Group Discussion with the editorial team of Binus Business Review (BBR), an international journal hosted by the Research and Technology Transfer Office (RTTO) of Bina Nusantara University; researching various relevant literatures on the topic of peer reviewing; implementing the solution by developing new functionalities for the OJS; and finally, testing and evaluating the performance. The software development itself applied the Prototyping cycle [7], which focuses on rapid iteration of prototype development and reviews, which produces a hands-on application, ready to use by the editorial team. The OJS provides support for plugins which allows new features to be easily integrated into the software. Additionally, the plugins can be quickly installed to any OJS-based journals. In this research, development of the new application uses a collection of 50 article reviews, taken from BBR, as the standard used to estimate the review scoring.

2.1. Related Works

While various research attempts have been done to improve the peer reviewing process [3, 8, 9] no definitive model exists to assess the quality of a peer review. Ramachandran attempted to solve the problem of automatically assessing review quality using text mining and Natural Language Processing (NLP) to obtain metrics that determine the quality of review. It shows that review relevancy to the topic or field is the most important aspect in assessing the quality of review [10]. Xiong [11] has developed a helpfulness-guided review summarization, where useful reviews are summarized to help users obtain useful information effectively, based on an automated review helpfulness assessment that takes various textual features into account [12], such as: structural, lexical, syntactic, semantic, meta-data, etc. It shows that the structural feature achieves the highest performance in assessing peer review quality.

This study also uses TextRank to automatically summarize each review. While some successful methods of summarization employs supervised approach [13,14], they require large training data and unable to adapt new domains and/or language. TextRank uses unsupervised approach where no training data is required for the algorithm to function, independent of the language used. It is a graph-based ranking model, where text is represented as a graph

where text units are represented as nodes and the relationship between text units are represented as edges. While graph-based algorithm provides good results, it's running time is proportional to the complexity of the graph, therefore it's not a very efficient approach for summarizing large text [15].

2.2. Review Rating Recommendation

The review rating recommendation consists of two main algorithms. The first being an automatic summarization of review text using TextRank, an unsupervised, graph-based ranking method for extractive text summarization. It is a variant of Google's PageRank algorithm [16], which uses the concept of recommendation to calculate the score of a web page using graphs. This study uses a weighted graph and utilizes TextRank to calculate the score of each sentence in a text, and extracting sentences with highest score as the summary. The formula used to calculate the score of node V_i is as follows:

$$WS(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(Vj)} w_{jk}} WS(V_j) \quad (1)$$

where $In(V_i)$ denotes the set of nodes that point to V_i, $Out(V_j)$ denotes the set of nodes that V_j point to, w_{ij} denotes the weight of edge between V_i and V_j, and d signifies a randomizing factor, set between zero to one. The weight of edges is measured as the similarity value between two text units. Depending on the implementation, the text units and the form of similarity between sentences can vary. In this study, sentences are used to represent the text units, while the similarity is determined as the overlap of two sentences, S_i and S_j, formally defined as follows:

$$Similarity(S_i, S_j) = \frac{\left|\left\{w_k \,|\, w_k \in S_i \,\&\, w_k \in S_j\right\}\right|}{\log(|S_i|) + \log(|S_j|)} \quad (2)$$

where sentences are represented as a set of words, normalized by the length of both sentences. The TextRank algorithm will be used to compute the value of each nodes iteratively until convergence below a given threshold value is achieved [5]. The nodes will then be sorted, and best scoring sentences will be extracted to create the review summary, which will be shown to editors as a quick overview of the review's quality.

The second part of the review rating recommendation is the automatic review score estimation, which employs the structural features of review text as the factor to evaluate review text: the length of text as number of words, number of sentences, percentages of sentences ended with a question mark, and number of sentences ended with an exclamation mark [12]. The review text will be divided into several parts, such as the abstract, introduction, methodology, etc. with different weights to better show the review scores of each part. In this study, previously rated reviews are used to determine the standard to be used in the recommendation algorithm. Followings are the main steps used to implement the review rating recommendation:

- Retrieve the text from the uploaded review file submitted by peer reviewers.
- Preprocess the text as clean sentences, and create a graph using sentences as nodes and relation between each sentence as edges. Run the TextRank algorithm until convergence is achieved and build the summary.
- Calculate the score of each review part and the total score, using the structural features of the text.
- Show the summary of article review alongside the breakdown scores of each review part (including the value of each structural feature) and the final rating estimation between one and five.

2.3. Reporting

The additional reporting features will contain data regarding authors who have submitted article and reviewers who have reviewed. The reports are generated as a table in Comma-separated values (CSV) format. Since OJS is capable to host multiple journals in one OJS website instance, the reports created will only account for data in the

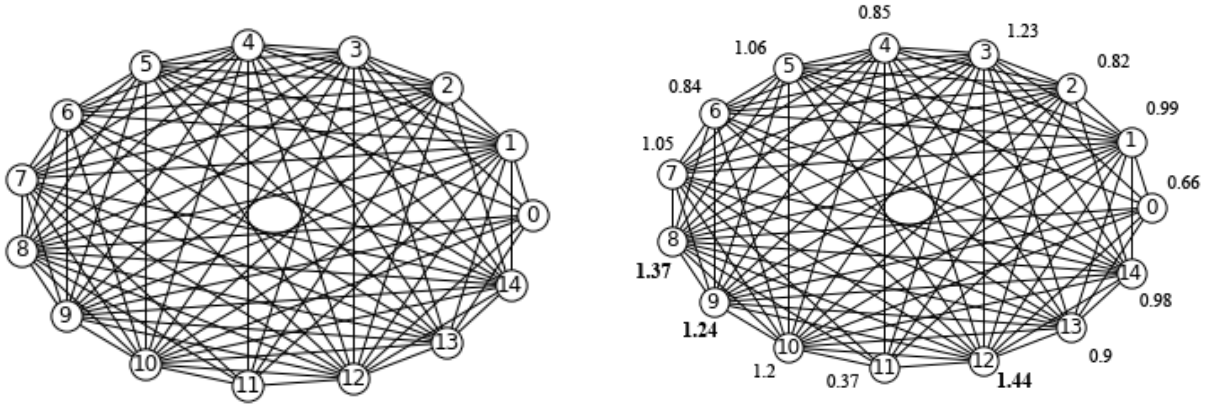
journal where the reporting features are called. For the author report, the table contains information regarding: author ID, author name, author's country, author affiliation, article ID, article title, journal issue volume, and journal issue number. Note that the records will be accounted per article, hence multiple authors with the similar identities but different ID might exist. For the reviewer report, the table contains: reviewer ID, reviewer name, reviewer status (local, national, international), average rating, reviewer affiliation, and reviewer's country. Unlike the author report, each unique reviewer will only be listed once, since each reviewer must be registered on OJS as a user, while authors didn't. The followings are main steps taken to generate the reports:

- Retrieve the journal ID.
- Retrieve each author or reviewer ID, and its corresponding details in the journal.
- Generate a table report in CSV format for users to download from OJS.

3. Result And Discussion

The development and implementation of the review rating recommendation's main algorithm is done using Python script with help from Natural Language Toolkit (NLTK) (17) library for text manipulation, while the rest are implemented using PHP, as it's OJS' base programming language. The results are presented in form of plugins, ready to be installed to any OJS-based journals. Bina Nusantara University's OJS website is used as a study case.

3.1. Review Rating Recommendation

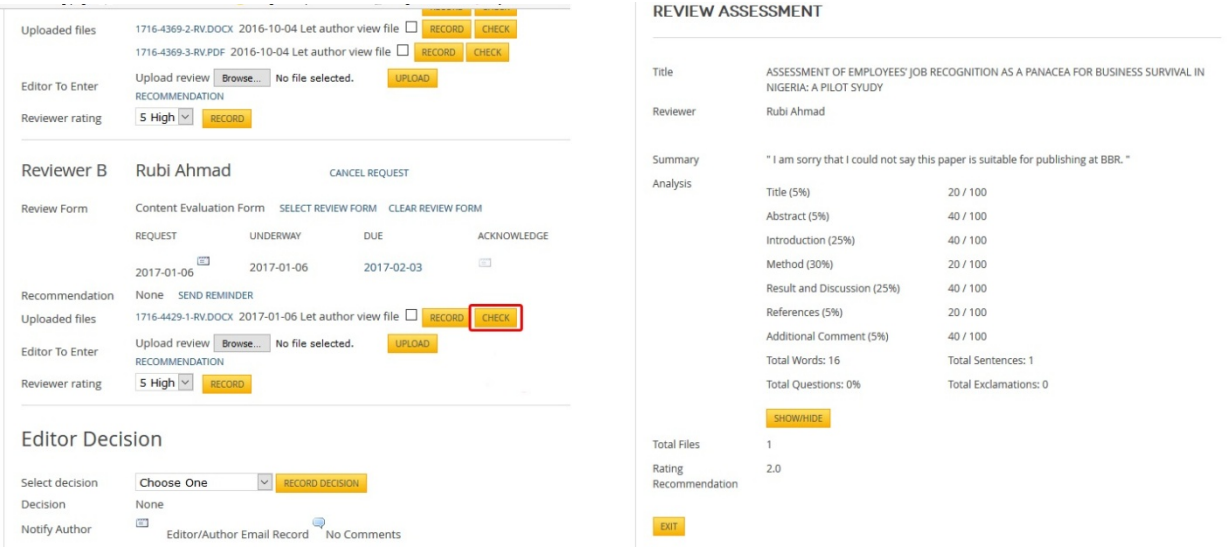


(a) (b)

Fig 1. (a) Initial graph; (b) Final graph

Fig. 1(a) shows the initial graph, created from the review text, with each node representing one sentence, while each edge represents a connection between the two connected nodes. In the first iteration of the algorithm, each node is given a trivial score of 1, while each edge will be given a weight value calculated with (2). The TextRank algorithm will be applied to calculate each node's value during an iteration. After the algorithm finally reached the convergence value each node will have a final score, as seen on Fig. 1(b). The highest scoring nodes (node 8, 9, and 12) will then be used to form the final summary. Fig. 2(b) shows an example of the result of the review summarization . This new function will be installed to OJS as a plugin, allowing editors to assess uploaded files by the peer reviewers.

Fig. 2(a) shows the article submission page, where reviewing process is done. Reviewers can upload review files to be checked by editors, before editors rate them and decide whether the article submission be revised, rejected, or approved. A new button "Check" is added at the end of each uploaded review file (denoted by the red square mark), ready to be used by editors to call the review rating recommendation feature, which will be shown in Fig. 2(b).



(a) (b)

Fig 2. (a) Article Submission Review Screenshot; (b) Review Rating Recommendation Screenshot

The review rating recommendation page will show: article title, reviewer's name, summary of the review, breakdown analysis of the review, total uploaded files, and the rating recommendation. As seen in Fig. 2(b), each part of the review will be scored and given different weights according to their importance: title (5%), abstract (5%), introduction (25%), methodology (30%), result and discussions (25%), references (5%), and additional comments (5%). In this study case, methodology is deemed the most important section of a review by the editorial team, where it accounts for almost a third of the total score. A total rating will then be calculated by taking each part into account to yield a discrete rating estimation between 1 (lowest) and 5 (highest). In addition to the score, the value of each structural feature is also shown. Finally, the summary and rating estimation will then be used by editors as an objective estimation for scoring the review more accurately.

3.2. Reporting

The reporting features are shown in Fig. 3(a) and Fig. 3(b). Each reporting plugins will be shown in the "Stats & Reports" page of the OJS journal. Users can call the features simply by clicking the report label, and a download prompt will appear. The CSV report will then be able to be downloaded, showing various data regarding authors or reviewers of the chosen journal.

Author ID	Name	Country	Affiliation	Article ID	Title	Volume	No.
1652	Nuraini Sari	ID	Bina Nusantara University	944	Analysis Method of Transfer	6	3
1653	Ririn Susanti Hunar	ID	Bina Nusantara University	944	Analysis Method of Transfer	6	3
1654	Dian Kurnianingrum	ID	Bina Nusantara University	945	The Influence of Bi Rate to th	6	3
1655	Dominica Rufina	ID	Bina Nusantara University	946	Analysis of Factors That Affec	6	3
1656	Stefanus Ariyanto	ID	Bina Nusantara University	946	Analysis of Factors That Affec	6	3
1657	Theresia Lesmana	ID	Bina Nusantara University	946	Analysis of Factors That Affec	6	3
1658	Dianka Wahyuningtias	ID	Bina Nusantara University	947	The Application of Dragon Fr	6	3
1659	Rini Kurnia Sari	ID	Bina Nusantara University	948	Analysis of Factors That Affec	6	3
1660	Agung Gita Subakti	ID	Bina Nusantara University	949	The Effects of Management I	6	3
1662	Maria Pia Adiati	ID	Bina Nusantara University	949	The Effects of Management I	6	3
1661	Gen Norman Thomas	ID	Bina Nusantara University	950	The Analysis of Different Per	6	3
1663	Bambang Leo Handoko	ID	Bina Nusantara University	951	The Effect of Institutional Ow	6	3
1664	Tinjung Desy Nursanti	ID	Bina Nusantara University	952	The Role of Person-Environm	6	3
1665	Masruroh Masruroh	ID	Bina Nusantara University	952	The Role of Person-Environm	6	3
1666	Andela Putri Maharani	ID	Bina Nusantara University	952	The Role of Person-Environm	6	3
1667	Mulyono Mulyono	ID	Bina Nusantara University	953	The Effects of Financial Ratio	6	3
1668	Lidiyawati Lidiyawati	ID	Bina Nusantara University	954	The Effect of Ratio, Issuance	6	3
1669	Ratih Wulandari	ID	Bina Nusantara University	954	The Effect of Ratio, Issuance	6	3
1670	Sasya Sabrina	ID	Bina Nusantara University	955	Analysis of Intellectual Capit	6	3
1671	Seri Nurmala	ID	Bina Nusantara University	956	Business Strategy of CV Jaya	6	3
1672	Hartiwi Hartiwi	ID	Bina Nusantara University	956	Business Strategy of CV Jaya	6	3
1673	Darjat Sudrajat	ID	Bina Nusantara University	957	The Relationships among Lea	6	3
1674	Muhammad Tony Nawawi	ID	Tarumanegara University	958	The Factors That Influence Cc	6	3
1675	Muhammad Safiqi Ikhaz	ID	Tarumanegara University	958	The Factors That Influence Cc	6	3

(a)

Reviewer ID	Name	Status	Rating	Affiliation	Country
638	Yasintha Soelasih	National	2	Atmajaya University	ID
675	Mohamad Heykal	National	1.5	Bina Nusantara University	ID
677	Erika Takidah	National	2	State University of Jakarta	ID
684	Terry Shevels	International	2	Newcastle College	GB
685	Rodrigo F. Malaquias	International	5	Universidade Federal de Uberlandia	BR
742	In Sue Kim	International	3.5	Korea Advanced Institute of Science and Technology (KAIST)	KR
754	Sevenpri Candra	International	2	Bina Nusantara University	ID
844	Annetta Gunawan	International	2	Bina Nusantara University	ID
963	Elia Ardyan	National	3.5	Surakarta Economics College	ID
980	Hendry Hartono	International	3	Bina Nusantara University	ID
982	Enggal Sriwardiningsih	International	2.5	Bina Nusantara University	ID
983	Rindang Widuri	National	2	Bina Nusantara University	ID
984	Bachtiar H. Simamora	International	5	Bina Nusantara University	ID
985	Ina Melati	Local	1	Bina Nusantara University	ID
987	Tulus Suryanto	National	1.6667	National Institute of Islamic Studies	ID
988	Iskandar Putong	International	2	Bina Nusantara University	ID
989	Agus Ridwan	Local	2	Bina Nusantara University	ID
991	Yen Sun	International	2	Bina Nusantara University	ID
994	Nugroho Juli Setiadi	International	2	Widyatama University	ID

(b)

Fig 3. (a) Author Reporting Screenshot; (b) Reviewer Reporting Screenshot

Each feature is tested and evaluated by the editorial team of RTTO Bina Nusantara University. Result shows that all functions perform well according to the expectation.

4. Conclusion

This article discusses the development of an effective peer review rating recommendation and reporting system for the Open Journal System. As it currently stands, no definitive method has been defined to accurately assess the quality of a peer review. This study attempts to improve the objectivity of review assessment done by journal editors, by employing the structural features of the review text as the deciding factor for estimating the review rating. A summarization of the review text is also implemented using the TextRank algorithm to give editors a quick insight as to the content of the review, which will reduce the time and resources used to assess the review, which improves the objective accuracy of the review scoring. Finally, reporting tools to retrieve author and reviewer data

are also developed to lessen the amount of work, time, and resources that journal publishers have to spend in documenting their journal data. Testings and evaluations confirmed that both features are running and functioning effectively as expected.

In addition, this study also suggests that OJS as a journal publishing platform still has rooms for improvements, and future works could implement more support for the system, whether it be the reviewing process, reporting, and other parts of the publishing procedure. Further research should also be done to better improve the peer reviewing process.

References

1. Mayden KD. Peer Review: Publication's Gold Standard. J Adv Pract Oncol. 2012;3(2):117–22.
2. Edgar BD, Willinsky J. A Survey of the Scholarly Journals Using Open Journal Systems. Sch Res Commun. 2010;1(2):1–22.
3. Onitilo A a, Engel JM, Salzman-Scott S a, Stankowski R V, Doi S a R. Reliability of reviewer ratings in the manuscript peer review process: an opportunity for improvement. Account Res. 2013;20(4):270–84.
4. Manalu SR, Willy, Sundjaja AM, Noerlina. Review assessment support in Open Journal System using TextRank. J Phys Conf Ser. 2017;801(1).
5. Mihalcea R, Tarau P. TextRank: Bringing order into texts. In: Proceedings of EMNLP. 2004. p. 404–411.
6. Kim S-M, Pantel P, Chklovski T, Pennacchiotti M. Automatically Assessing Review Helpfulness. Proc 2006 Conf Empir Methods Nat Lang Process (EMNLP 2006). 2006;(July):423–30.
7. Roger S. Pressman PD. Software engineering : a practitioner's approach. McGraw-Hill; 2010.
8. Lehan T, Hussey H, Mika E. Reviewing the review: An assessment of dissertation reviewer feedback quality. J Univ Teach Learn Pract. 2016;13(1).
9. Dobele AR. Assessing the Quality of Feedback in the Peer-Review Process. High Educ Res Dev. 2015;34(5):853–68.
10. Ramachandran L. Automated Assessment of Reviews. North Carolina State University; 2013.
11. Xiong W. Helpfulness-Guided Review Summarization. Proc 2013 NAACL HLT Student Res Work. 2013;(June):77–83.
12. Xiong W, Litman D. Automatically predicting peer-review helpfulness. Acl2011. 2011;(2009):502–507.
13. Yeh JY, Ke HR, Yang WP, Meng IH. Text summarization using a trainable summarizer and latent semantic analysis. Inf Process Manag. 2005;41(1):75–95.
14. Hirao T, Sasaki Y, Isozaki H, Maeda E. Ntt's text summarization system for duc-2002. Proc Doc Underst Conf 2002. 2002;(1):104–7.
15. Ferreira R, De Souza Cabral L, Lins RD, Pereira E Silva G, Freitas F, Cavalcanti GDC, et al. Assessing sentence scoring techniques for extractive text summarization. Expert Syst Appl. 2013;40(14):5755–64.
16. Brin S, Page L. The anatomy of a large scale hypertextual Web search engine. Comput Networks ISDN Syst. 1998;30(1/7):107–17.
17. Bird S, Klein E, Loper E. Natural Language Processing with Python. Vol. 43, Text. 2009. 479 p.