An enhanced latent semantic indexing with term frequency-inverse document frequency variant for software traceability

IOP Conf. Series: Materials Science and Engineering **551** (2019) 012073 doi:10.1088/1757-899X/551/1/012073

An enhanced latent semantic indexing with term frequency-inverse document frequency variant for software traceability

R Tumeng[1], D N A Jawawi[2] and M A Isa[3]

[1,2,3]Software Engineering Research Group, School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia, 81310, Johor Bahru, Johor, Malaysia

[1]rooster2@live.utm.my

Abstract. This paper proposes an improved Latent Semantic Indexing (LSI) with Term Frequency-Inverse Document Frequency (TFIDF) variant for software traceability. The main advantage of the method is the simplicity and accuracy of the algorithm such that it can produce high precision and recall. The accuracy of tracing precise links between software artefacts is achieved by feeding the LSI with TF-IFDF variant that halves the TF and IDF values, thus increases the probability of yielding precise links. To test the accuracy of the proposed method, two case studies were evaluated, namely Mushroom Management System (MMS) and Robotic Wheelchair System (RWS). The superiority of the proposed method over a conventional LSI is confirmed by 409.09%, 900.00%, and 620.00% improvements for precision, recall, and harmonic mean scores, respectively, in MMS case study, at cosine threshold of 0.94. It is anticipated that the method could be beneficial in the design of a practical and accurate system for retrieving precise software traceability links.

1. Introduction

In Software Development Life Cycle (SDLC), traceability plays a crucial role in analyzing the impact of alteration across different software artefacts. Traceability is described as the ability to trace links between source artifact and target artifact [1].

Successful traceability relies on the ability to concisely trace source artifact to target artifact. In real world scenarios, SDLC is often plagued by an overloading of information as software information is continuously generated. Researchers are in agreement that an immense wealth of information may be useful to assist in successful traceability across various SDLC phases via formulating traceability problem through the lens of an information retrieval (IR) [2-3]. Researchers have utilized and proposed various IR methods in literature. Several instances include algebraic models [2], probabilistic models [4], and statistical language models [5]. Widely used algebraic models include Vector Space Model (VSM) [6] and Latent Semantic Indexing (LSI) [7]. Fundamental understanding of algebraic models involves computing similarity between documents to determine the extent of document relevance. Researchers have argued that conventional LSI suffers poor accuracy (i.e. yielding low precision and recall scores) [2]. In addition, combination of conventional LSI with regular TF-IDF in retrieving latent meanings in software documents often have poor accuracy as shown by studies in [2, 8]. Thus, a new method is needed which could improve the accuracy in tracing software trace links between software artefacts.

In this study, the authors evaluate conventional LSI and the proposed LSI with TF-IDF variant and report the experiments with two case studies. The evaluation indicated that the proposed method improved precision and recall. The paper is structured as follows. Section 2 elaborates conventional

LSI. Section 3 elaborates TF-IDF. Section 4 describes the proposed TF-IDF variant. Section 5 describes the case studies. Section 6 describes the evaluation measures. Section 7 elaborate the experiment results. Section 8 concludes the paper.

2. Latent Semantic Indexing (LSI)

LSI is an extension of Vector Space Model (VSM) that was proposed by [7] to solve synonymy and polysemy issues. LSI operates by reducing document noise via vector space dimension reduction and utilizes single value decomposition (SVD) to determine new dimensions. The k-dimensional LSI is used to compute similarities between vectors. A query vector, q_k, is mapped into its representation in the LSI space by the transformation as in Equation 1;

$$\overrightarrow{q_k} = \sum_k^{-1} U_k^T \vec{q} \tag{1}$$

Where $\overrightarrow{q_k}$ is query vector with respect to coefficient of low-rank approximation dimension, k, $\sum_k^{-1}$ is the inverse of diagonal matrix $\sum_k$, and U_k^T is transposed singular value decomposition term matrix with respect to *k*.

3. Term Frequency-Inverse Document Frequency (TF-IDF)

Term frequency, as expressed in Equation 2, refers to a normalized term weight occurring in document that is proportionate to term frequency [9];

$$tf(t,d) = \frac{f_{(t,d)}}{\max\{t' \epsilon d\} f_{t',d}} \tag{2}$$

Where $tf(t,d)$ is normalized term frequency of a term *t* that occurs in document *d*, $f_{t,d}$ is the raw count of frequency of term *t* in document *d*, and $\max\{t' \epsilon d\} f_{t',d}$

Meanwhile, inverse document frequency, as expressed in Equation 3, refers to specificity of terms quantifiable as inverse function of the number of documents in which the terms occur [10];

$$idf(t,D) = log \frac{N}{|\{d \in D : t \in d\}|} \tag{3}$$

Where $idf(t,D)$ refers to inverse document frequency of a term *t* that occurs in corpus *D*, *N* is the total number of documents in corpus, and $|\{d \in D : t \in d\}|$ is the number of documents where term *t* appears.

4. TF-IDF Variant

In this paper, the authors proposed to halve TF-IDF through removing logarithm in IDF and replacing it with square root. Resulting TF-IDF variant is expressed in Equation 4.

$$tf(t,d) \cdot idf(t,D) = \sqrt{\frac{f_{(t,d)}}{\max\{t' \epsilon d\} f_{t',d}}} \cdot \sqrt{\frac{N}{|\{d \in D : t \in d\}|}} \tag{4}$$

Where $\sqrt{\frac{f_{(t,d)}}{\max\{t' \epsilon d\} f_{t',d}}}$ is the square root of normalized frequency and $\sqrt{\frac{N}{|\{d \subset D : t \subset d\}|}}$ is the square root of the inverse document frequency.

5. Case Studies

The authors utilize Mushroom Management System (MMS) and Robotic Wheelchair System (RWS). MMS is a general software developed by final year students from the School of Computing, Universiti Teknologi Malaysia for a local mushroom producer. RWS is a safety-critical, wheelchair equipped

with AMD188ES microcontroller, parallel I/O, serial I/O, ADC/DAC, and an on/off switch [11]. The characteristics of case studies are listed in Table 1.

Table 1. Characteristics of case studies.

Characteristics	MMS	RWS
Domain	General	Safety-critical
Number of requirements	11	7
Number of test cases	38	29
Number of trace links	38	29

6. Evaluation Measures

Precision is defined in Equation 5, meanwhile Recall is defined in Equation 6. In addition, harmonic is also used, which is also referred to as F1 measure, as defined in Equation 7.

$$Precision = \frac{Correct}{Correct + Incorrect} \tag{5}$$

$$Recall = \frac{Correct}{Correct + Missed} \tag{6}$$

$$F1\ measure = 2 \bullet \frac{P \times R}{P + R} \tag{7}$$

Where *P* stands for *Precision,* while *R* stands *for Recall*.

7. Experiment Results

This section describes the results obtained from MMS and RWS case studies. Table 2 presents the results obtained from MMS case study. In general, the results in Table 2 revealed that the proposed LSI with TF-IDF variant had improved the precision, recall, and F1 measures significantly at cosine threshold of 0.94. Referring to Table 2, improvements of 409.09%, 900.00%, and 620.00% were observed for precision, recall, and F1 measures, respectively, for cosine threshold of 0.94, in comparison to conventional LSI. Meanwhile, a 100% improvement was observed for MMS' precision score at cosine threshold of 0.99. Such improvements indicate that the enhanced LSI with TF-IDF variant is capable of yielding higher accuracy at higher cosine thresholds. Meanwhile, at lower cosine thresholds (i.e. 0.80 and 0.60), no improvements but decrements of precision, recall, and F1 measures were observed for MMS, at cosine threshold of 0.60. This indicates that at lower cosine threshold, the proposed method was not able to return precise trace links for MMS case study.

Meanwhile, Table 3 presents the results obtained from RWS case study. Generally, precision and F1 measures improved significantly with values of 30.77% and 127.27%, respectively, when cosine threshold was 0.80. Meanwhile, high precision was maintained at cosine threshold of 0.90. The improvements at higher cosine thresholds were attributed to square-rooting the TF-IDF vector values, which yielded favorable results, particularly at cosine threshold of 0.94 for MMS case studies. Overall, in comparison with conventional LSI, the proposed LSI with TF-IDF variant produced unfavorable results for low cosine thresholds (i.e. 0.60). However, the findings could be inconclusive as the scale of the case studies were rather small.

In general, the results showed that the proposed LSI with TF-IDF variant is promising in improving precision to higher levels in all case studies, at higher cosine thresholds. In contrast, conventional LSI performed better at lower cosine threshold of 0.60, particularly in RWS. Based on this observation, the authors concluded that proposed LSI with TF-IDF variant has the most potential as a method to be used for automating software traceability at higher cosine thresholds.

Table 2. Characteristics of case studies.

MMS – Conventional LSI				
Cosine Threshold	0.99	0.94	0.80	0.60
Precision	0.50	0.11	0.00	0.57
Recall	1.00	0.10	0.00	0.25
F1	0.67	0.10	0.00	0.35
MMS – Proposed LSI with TF-IDF Variant				
Cosine Threshold	0.99	0.94	0.80	0.60
Precision	1.00	0.56	0.00	0.00
Recall	0.42	1.00	0.00	0.00
F1	0.59	0.72	0.00	0.00
Improvement				
Precision	100.00%	409.09%	N/A	-100%
Recall	-58.00%	900.00%	N/A	-100%
F1	-11.94%	620.00%	N/A	-100%

Table 3. Characteristics of case studies.

RWS – Conventional LSI				
Cosine Threshold	0.99	0.94	0.80	0.60
Precision	1.00	0.00	0.13	0.00
Recall	0.86	0.00	1.00	0.00
F1	0.92	0.00	0.11	0.00
RWS – Proposed LSI with TF-IDF Variant				
Cosine Threshold	0.99	0.94	0.80	0.60
Precision	1.00	0.55	0.17	0.17
Recall	0.31	0.55	0.44	1.00
F1	0.46	0.55	0.25	0.29
Improvement				
Precision	0.00%	N/A	30.77%	N/A
Recall	-63.95%	N/A	-56.00%	N/A
F1	-50.00%	N/A	127.27%	N/A

8. Conclusion

In this study, two IR methods have been investigated, conventional LSI and the proposed LSI with TF-IDF variant, across two case studies. The proposed method was targeted to solve the issue of poor accuracy of conventional LSI in capturing trace links between software requirements and test cases, encompassing precision, recall, and F1 scores. The findings of the experiment suggest that the proposed LSI with TF-IDF variant improved conventional LSI significantly at higher cosine thresholds (i.e. 0.99 and 0.94), particularly in MMS case study. Despite varied performances across the two case studies, the proposed method indicates a potential to improve the overall accuracy of IR-based software traceability systems.

As future works, further evaluation of benchmark datasets and larger case studies could be conducted. In addition, the future works could also consider comparing other IR-based techniques. Finally, hybridization of TF-IDF with other weighting schemes could also be investigated to produce precise IR-based software traceability systems.

References

[1] Center of Excellence for Software and Systems Traceability, [CoEST. Requirement Traceability. Retrieved 08 October 2018, 2018, from http://coest.org .
[2] Khatiwada S, Tushev M and Mahmoud A 2018 *Inform Software Tech* **93** 45-57
[3] Rempel P and Mäder P 2017 *IEEE TSE* **43** 777-797
[4] Turtle H and Croft W B 1990/2017 *SIGIR Forum* **51** 124-147
[5] Zhai C and Lafferty J 2017 *SIGIR Forum* **51** 268-276
[6] Salton G, Wong A and Yang C S 1975 *Commun ACM* **18**, 613-620
[7] Deerwester S, Dumais S T, Furnas G W, Landauer T K and Harshman R 1990 *J. Am. Soc. Inf. Sci.*, **41**, 391-407
[8] Mahmoud A and Niu N 2015 *RE*, **20**, 281-300
[9] Luhn H P 1957 *IBM J Res Dev.*, **1**, 309-317
[10] Spärck Jones K 1972 *JDoc.*, **28**, 11-21
[11] Jawawi D N A, Sabil S, Mamat R, Zaki M Z M, Talab M A S and Mohamad R A 2011 *In Mobile Robots-Control Architectures, Bio-Interfacing, Navigation, Multi Robot Motion Planning and Operator Training*

The authors fully acknowledge Universiti Teknologi Malaysia for Zamalah UTM Scholarship and UTM-TDR Grant Vot No. 06G23, and Ministry of Higher Education (MOHE) for FRGS 2019, which have made this research endeavor possible. Additionally, the authors would also like to express their sincere gratitude to members of Embedded Real Time Software Engineering lab for their continuous support and feedbacks.