CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Assessment of Author Ranking Indices based on Multi-authorship

by

Muhammad Salman

A thesis submitted in partial fulfillment for the degree of Master of Science

in the

Faculty of Computing
Department of Computer Science

2020

I dedicate my dissertation work to my family, teachers and friends. Special feeling of gratitude to my loving parents for their love, endless support and prayers

CERTIFICATE OF APPROVAL

Assessment of Author Ranking Indices based on Multi-authorship

by

Muhammad Salman

(MCS171017)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Basit Shahzad	NUML, Islamabad
(b)	Internal Examiner	Dr. Abdul Basit Siddique	CUST, Islamabad
(c)	Supervisor	Dr. Mohammad Masroor Ahmed	CUST, Islamabad

Dr. Mohammad Masroor Ahmed
Thesis Supervisor
July, 2020

Dr. Nayyer Masood
Head
Dept. of Computer Science
July, 2020

Dr. Muhammad Abdul Qadir
Dean
Faculty of Computing
July, 2020

Author's Declaration

I, **Muhammad Salman** hereby state that my MS thesis titled "**Assessment of Author Ranking Indices based on Multi-authorship**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MS Degree.

(Muhammad Salman)

Registration No: MCS171017

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "**Assessment of Author Ranking Indices based on Multi-authorship**" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been dully acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS Degree, the University reserves the right to withdraw/revoke my MS degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

(Muhammad Salman)

Registration No: MCS171017

Acknowledgements

All worship and praise is for ALLAH (S.W.T), the creator of whole worlds. First and leading, I would like to say thanks to Him for providing me the strength, knowledge and blessings to complete this research work. I would also like to pay gratitude to my parents and family members for their endless support and prayers.

Special thanks to my supervisor **Dr. Mohammad Masroor Ahmed** and **Dr. Tanvir Afzal** for their assistance, valuable time and guidance and to my friend Mr. Fazal Ur Rehman for being a source of inspiration and motivation.

(Muhammad Salman)

Registration No: MCS171017

Abstract

The ranking of authors in the scientific community has gained an extensive importance. The research based decisions concerning tenure, promotion, nomination of awards, remunerations and scholarships largely depend upon the ranking of authors. There are various parameters to rank authors i.e. publication count, citation count, h-index and the variants of h-index. Currently, the h-index and its variants are being frequently used for the ranking of authors. These variants include the indices that take into account carrier length of authors, citation intensity, self-citations, field-dependence and multi-authorship. At present moment, the collaborations are growing larger and larger and the multi-authorship trend is enhancing day by day. The scientific community is focusing on the indices that consider multi-authorship in the research. However, there is a discussion in the scientific community that which multi-authorship index performs better for the ranking of authors. The current reports indicate that the multi-authorship indices are assessed on very small datasets making it challenging to identify the actual performance of these indices. Furthermore, the multi-authorship indices are assessed on the datasets of different domains, as a concequence of which, the comparison of indices and identification of most contributing index is difficult. There is a strong need for the assessment of these indices on a comprehensive dataset in a single domain. This thesis emphasizes on the assessment of multi-authorship based indices on a comprehensive dataset from a single domain. The assessment of h_m-index, g_m-index, h_i-index, h_f-index, g_f-index, w-norm, k-norm and g_F-index is performed on a comprehensive dataset from the Civil Engineering domain. The results obtained from these indices are further investigated to find the correlation between the ranked lists obtained by these indices. It is observed that there exists strong, very strong or moderate correlation between multi-authorship indices. Overall, it is observed that the indices having strong or very strong correlations exceed the indices having moderate correlations and none of the index has shown weak or negative correlation. Furthermore, the occurrence of awardees is checked in the ranked

list of each index for the determination of most contributing multi-authorship index. The award winners of four renowned society in Civil Engineering are considered for comparison. In top 10% of the ranked list, g_f-index and g_m-index remained successful in bringing most of the awardees i.e. around 67% of total awardees. Overall, none of the index remained successful in bringing 100% of awardees in top ranks.

Contents

List of Figures

List of Tables

Abbreviations

ACI	American Concrete Institute
ASCE	American Society of Civil Engineering
CSCE	Canadian Society of Civil Engineering
CEDB	Civil Engineering Database
ICE	Institute of Civil Engineering

Chapter 1

Introduction

1.1 Background

The ranking of authors in the scientific community has achieved an extensive importance. There are various benefits that can be achieved by the ranking of authors. For instance, it can assist the organizers of journals/conferences to locate a suitable reviewer for the evaluation of a scientific publication, it can facilitates the student community to find a relevant and suitable supervisor for meeting their research objectives. Furthermore, it can be helpful in responding to the questions i.e. who should be employed; who can get promotion; who can be considered for awards; and who should get scholarships?

There has been a variety of research assessment techniques proposed so far in the literature. The conventional technique was publication count [1]. The number of publications was counted to find the impact of an authors work. This technique though performed well but still needed the information of citations associated with the research work. Therefore, to address this issue a new technique was introduced by the name of citation count [2]. This technique primarily takes into account the citation count of a researcher. However, the proposed technique was found to be inefficient regarding the number of publications associated with the citations. In order to gauge the true performance of a researcher, both publication count and citation count were needed. Therefore, in 2005, Jorge Hirsh proposed a technique named h-index, in which he combined the citation count and the number of publications in a single number [3]. This technique became very popular and opened a new area of research for the scientific community.

In 2006, Egghe critically analyzed the h-index and highlighted the advantages and the drawbacks of h-index [4]. Talking about the advantage of h-index, he mentioned that h-index is not sensitive towards a set of papers that are lowly cited and towards a single or small number of publications that are very highly cited. Highlighting the drawback of h-index, although the h-index is not sensitive towards the tail of lowly cited papers but in the case of one or few highly cited papers, once the amount of papers meet the required citations, the rest of the citations become unimportant i.e. if these papers have 20 or 50 or 100 more citations, these citations become unimportant for the h-index. Considering this drawback of the h-index, Egghe proposed g-index as the enhancement of h-index while keeping all the advantages of h-index [4]. The g-index can be calculated as "the highest number g of papers that together receive g2 or more citations". Now that the several highly cited papers are the advantage for a researcher, this means that higher the citation count a researcher has in the list of papers, higher will be g-index.

After the proposal of h-index and g-index, a research started by the research community to add improvements in h- and g-indices by considering various factors of the contribution of authors in the research. The indices proposed later were termed as the variants and extensions of the h-index. These variants and extensions include the indices that consider the citation intensity i.e. A-index [5] and AR-index [6], self-citations i.e. ch-index [7] and carrier length or age of publication i.e. contemporary h-index [8], f-index [9] and hg-index [10] etc. These indexing methods have covered most of the aspects of scientific contribution of authors but they are insensitive towards the number of co-authors working in the research. The problem of how to assign credit to the co-authors working in the research i.e. each author in six-authored paper gets the credit of 1 (absolute counting) or gets the credit of 1/6 (fractionalized counting) [11]. Considering this limitation in co-authored research papers, the scientific community has proposed some techniques namely the variants and extensions of h-index that consider multi-authorship [12]. The indices of Multi-authorship category include h_i-index [13], fractional h and g indices [11], h_m-index [14], g_m-index [15], k-norm and w-norm [16] etc. These techniques consider the co-authorship factor in the research and assign the credit proportionally to the number of co-authors working in the research.

We performed analysis of multi-authorship based indices, there calculations on the datasets and the comparison of their results with other multi-authorship indices. We found that the multi-authorship indices are mostly assessed on very small datasets [13], [11], [14], [15]. Furthermore, the assessments are carried out on the datasets of different domains due to which, the comparison between multi-authorship indices and identification of most appropriate index is difficult [11], [16]. There is a strong need to assess these indices on a comprehensive dataset from a single domain in order to find the most contributing multi-authorship index for the ranking of authors.

This thesis emphasizes on the assessment of indices that take into account multi-authorship in the research including h_m-index, g_m-index, h_i-index, h_f-index, g_f-index, w-norm, k-norm and h_F-index. We intend to assess these indices on a single comprehensive dataset in the field of Civil Engineering. After the calculation of these indices, the correlation between the indices is found and the most contributing index in multi-authorship is identified. To compare the results obtained from multi-authorship indices, the data set of international award winners belonging to four renowned societies in the domain of Civil Engineering are considered for comparison. These societies include ASCE (American Society of Civil Engineering), ACI (American Concrete Institute), CSCE (Canadian Society of Civil Engineering) and ICE (Institute of Civil Engineering).

1.2 Purpose

The goal of this study is to investigate the multi-authorship based indices and assess these indices on a single comprehensive dataset in the domain of Civil Engineering. After evaluation, the most appropriate index is identified and correlation between the indices is found.

1.3 Problem Statement

It has been observed that the multi-authorship based indices are mostly assessed on very small datasets making it difficult to find the actual performance of these multi-authorship indices. Furthermore, the assessments are carried out on the datasets of different domains, as a consequence of which, the comparison of indices and identification of most contributing multi-authorship index is difficult. There is a strong need to evaluate these indices on a single comprehensive dataset from a specific domain in order to find the most contributing multi-authorship index for the ranking of authors.

1.4 Research Questions

The research questions around which the present study revolves can be described as follows:

Research Question 1: Is there any correlation among the ranked lists obtained by the author ranking indices based on multi-authorship?

Research Question 2: Does the international awardees lie in top of the ranked list and which multi-authorship index contributes the most in bringing the international awardees at the top?

1.5 Scope

The parameters proposed by the scientific community for the ranking of authors that includes publications count, citation count, h-index and variants of h-index.

1.6 Significance of the Solution

The indices discussed in our study are mostly calculated for small datasets for their assessment. We have taken an actual dataset in the field of Civil Engineering. Secondly, the indices are applied on the dataset of different domains. There needs to be an assessment of these indices on a wide-ranging dataset from a single domain. Our dataset is a comprehensive dataset that contains almost thirty-six thousand authors from the domain of Civil Engineering.

1.7 Thesis Organization

The following sections will explain the content of each chapter:
Chapter 2: Emphasizes on the literature review and the techniques proposed by different authors.
Chapter 3: Discusses the methodology.
Chapter 4: Emphasizes on the results and discussion.
Chapter 5: Focuses of the conclusion and future work.

Chapter 2

Literature Review

2.1 Introduction

Today, the assessment of an author is performed for various purposes. The authors are assessed for promotion, employment, nomination for awards, scholarships, finding a reviewer for conference/journal or a supervisor for thesis etc. The employment and tenure has been dependent for a time being upon the publication productivity of a scientist. The professionals had to achieve the required number of publications and citations to fulfill the criteria of teaching or other obligations.

An American scientist used the term publish and perish in 1942 in his article. The term means that a professor or an author has a pressure to publish in a conference to continue his/her tenure. Authors who do not publish frequently have less chance to be promoted and avail the advantages provided by an institution. Since the World War II, the doors of research were opened and the investments were also done to enhance the research. Due to large amount of publications thereafter, the management was main issue. The automation of publications was done with the help of some bibliographic tools and the earliest was Medical Literature Analysis and Retrieval System (MEDLARS), a database introduced in 1964 [16].

A scientist named Alan Pritchard introduced the term bibliometric in 1969, which refers to the statistical analysis of the articles, books and publications etc. Francis Narin outlined many bibliometric tools to measure the authors productivity [17]. Most of the authors try to publish their publications in renowned journals and some of the authors also publish in journals that do not consider too much quality. So, the number of publications of an author was not a convinced method to be found [1].

As the number of publications increased over time, and there was a need for the databases to store the data of publications. So, many databases were developed by the experts in order to store the data of publications in a sequence to make it easy for the public to access the data. These databases provide the tools to export the data for further use, generate graphs, and develop citation maps etc. [18]. These databases are also used to find the productivity of the publications. Some examples of these databases include IEEE Xplore, Scopus, CiteSeer, SpringerLink, Google Scholar, PubMed, and ASCE Library etc.

Publication data parameters have been developed by the experts for wide range of motives. Some of the parameters are used to find the productivity of the publication and some are used to find the influence and impact of the publications. In early times, the productivity of the research was analyzed by the number of citations received by the publications but after the advancement of assessment parameters, a series of quantitative analysis tools have been developed to demonstrate the productivity or influence of the publishers [19]. Examples include highly cited papers, size of the research group, type of the publication, conference types to which the publications are submitted etc.

2.2 Parameters based on Scholarly Productivity

One of the most basic types of parameter is an authors publication count since the start of publications of an author. This parameter is based on the document-level analysis that includes books, chapters of the book, conference publications, journal publications and others. The time frames i.e. the time since the first publication helps an author to receive more publication count. The award honoring societies give awards to the authors with more number of publications. One can predict how much an author can do publications in the future by observing the past number of publications [20].

The authorship position plays a vital role in the assessment of an author. An author can be the first author, last author or a sole author. The authorship position is recognized by the award honoring societies for honoring awards. The grants, scholarships, and promotions are also dependent upon the authors performance [21]. The first and the last author have a major role in the development of the publication. The publications with multiple authors show the collaborative work done by the authors of a paper, which can be used to show the productivity [22]. Some authors are undeserved i.e. who do not play a role in the publication but they are recognized by the parameters that rely solely on the number of publications.

2.3 Publication Parameters based on the Impact

The impact of an authors publication also depends upon the journal impact factor score of a journal. The research documents published in high impact factor journals are directly proportional to the tenure/promotion of an author [23]. The most commonly used parameter for the journal impact factor score is the Journal Citation Report (JCR). The JCR was developed in 1960 by Irvin Sher and Eugene Garfield and is calculated yearly since 1975. It can be calculated by dividing the citation count of a journal by the number of research articles published by that journal since two years. The impact factor value of 1.0 means the publications is cited one time in previous two years in journals. The JCR Impact Factor score is an easy to find parameter but it also has some drawbacks [24].

The JCR cannot be used to find the impact of a single author or publication rather it is used to find the impact of a journal as a whole. The JCR Impact Factor score is limited to only the journals ranked by the Web of Science. The ranking of a journal depends upon the citations made by other journals indexed by the WOS. Another flaw is that the JCR can be changed by making self-citations made by the journal authors and suggesting other peers to add them to their reference list from the same journals indexed by the Web of Science [25].

Although the JCR has been used as a parameter to assess the impact of a publication but the trend is changing. The San Francisco Declaration on Research Assessment (DORA) has suggested avoiding the use of JCR as a parameter rather to use the article-level parameters to evaluate the impact of the scientific publication.

2.4 Parameters based on Citations

The citation analysis is a parameter to evaluate the impact of a publication by observing the citations made by the later publications. The publications that are worthy enough can be easily assessed with the high citation counts. Citation counts of a publication also demonstrate the worth of its author. The older publications have long duration to acquire the citations than the newer ones and they receive high citations than the recent ones [26]. Some of the publications are highly cited and the scientific community has expectations from the cited publication authors but the authors do not come up with the newer publications. The phenomenon is termed in the literature as the Mendel effect [27]. Some limitations of this parameter include the lack of information about the quantity of publications made by an author through out the career. Self-citation is also an issue with the citation count. The citations can be earned through deliberately making citations by an author in his or her own publication or through a colleague author [28]. A study conducted by a journal of Science shows that there is a strong relationship between the number of citations received by the publication and the number of references at the end of the publication [29].

2.5 Beyond the Citation Count

A useful information can be extracted from the cited publication by carefully reviewing the publications to find out various questions. Who is citing the publication? What are the affiliations between cited and the citing one? Why is the publication being cited? What region does citing and the cited one belongs? If the affiliation is found then there is a chance of influence [1]. The funding body can also be checked as if the reference of publication includes the funding body. If there is any affiliation found between publication and the funding body, it represents the influence.

2.6 The h-index

The h-index is a parameter that combines the number of publications and citations in a single count. Jorge Hirsh developed h-index in 2005, using the formula that the maximum number of publications that have been cited at least h times [3]. An author has h-index 10, it means that he has 10 publications in his carrier with at least 10 citations each. This index is widely used to assess the impact of publications of an author or an academic institution. It is easy to be found, because many databases use h-index to assess the individuals including Scopus, Google Scholar and Web of Science. Among the limitations of h-index is that it varies across the databases [30]. For instance, an author can have different h-index if measured from Scopus and Google Scholar. The self-citations or deliberately made citations can increase the h-index. The h-index is also disregard of field-dependence and multi-authorship.

2.7 Other Parameters

After the h-index was introduced, Hirsh himself introduced the m-quotient [31], which can be obtained by simply dividing the h-index by length of carrier of an author. For example if the length of the carrier of an author is 10 years and the h-index is 20 then the m-quotient will be 2. In 2006, a scientist named Egghe critically analyzed the h-index and highlighted the advantages and the drawbacks of h-index. Talking about the advantage of the h-index, he mentioned that the h-index is not sensitive towards papers that are lowly cited and towards a single or small amount of papers that are very highly cited [4]. Highlighting the drawback of the h-index, he says that although the h-index is not sensitive towards the tail of lowly cited papers but in the case of one or few highly cited papers, once the amount of papers meet the required citations, the rest of the citations become unimportant i.e. if these papers have 20 or 50 or 100 more citations, these citations become unimportant for the h-index. Considering this drawback of h-index, the g-index was proposed by Egghe as the enhancement of h-index while keeping all the advantages of the h-index. The calculation of g-index can be performed as the highest number g of publications that together receive g^2 or more citations. Now that the several highly cited papers are the advantage for a researcher, this means that higher the amount of citations a researcher has in the list of papers, higher will be the g-index [4].

The A-index is also like the g-index but slightly different, as it also considers the fact that the citations should be considered if there is more citations achieve by the author in some paper, which are not considered in case of h-index [32]. The A-index considers the average number of citations of an author in h core. The precision problem of A-index is same as that of the h-index but h-index is less than or equal to the A-index always. The A-index has a problem that it involves the division with h so it may punish the scientist who has a greater h-index. This problem is solved by finding the square root of the sum and it is named as the R-index. R is equal to number of square root of the product of A and h. The A- and R-index do not consider the age of publication, so another index was proposed by Jin as AR-index [6], which consider the age of publication. As an example, publication record of an author named as B.C. Brookes. His h-index remains 12 from 2002 to 2007, hence his R-index is increased by 5% but his AR-index is decreased by 5%. The combination of both the indices A- and R- as AR-index decrease the disadvantage of h-index especially when AR-index considers age of publication.

The h- and g-index have different properties i.e. the h-index considers the h amount of publication that have h or above citations but once the h-index meets the required citations, the rest of the citations become unnecessary while the g-index can be calculated as g amount of publications that together receive g^2 or more citations that means higher the number of citations, higher is the g-index. Alonso et al. suggested an index named as hg-index that takes the advantages of both the parameters and minimizes their drawbacks [10]. The hg-index can be calculated as mean of h- and g-index, which balances both the indices. Once the h- and g-index are computed, the hg-index can be very easily computed. The value of hg-index is between h-index and the g-index but hg-index value remains closer to h-index as compare to the g-index i.e. the h-index of Egghe is 13 and the g-index is 19, the hg-index becomes 15.72, which is closer to the h-index. The hg-index provides a fine-grained measurement through which the scientists can be assessed more efficiently.

The A-index is used to measure the citation intensity of an author, AR-Index considers the age of publications, the ch-index tries to not include the self-citations etc. These indexing methods have covered most of the aspects of scientific contribution of authors but they are insensitive towards the number of co-authors working in the research. The problem of how to assign credit to the co-authors working in the research i.e. each author in six-authored paper gets the credit of 1 or gets the credit of 1/6 [11]. Considering this limitation, the scientific community has proposed some techniques namely the variants and extensions of h-index that consider Multi-authorship [12].

In 2018, Raheel et al have performed an assessment analysis of 11 indices that take into account citation intensity and age of publication [33]. The assessment is performed using a comprehensive dataset in the Civil Engineering domain. The correlation is found in the ranked list of these indices and the most contributing index for the ranking of authors is found however, the assessment is performed on the indices other than multi-authorship based indices. We have considered this paper as our base line paper.

2.8 Multi-authorship

In 2011, Bornman et al. have performed a meta-analysis on 37 different variants and extensions of h-index and categorized these variants on the basis of their properties. These h-index variants are formerly categorized as indices that take into account the carrier length of an author, citation intensity, self-citations, field-dependence and multi-authorship [12].

Today, the collaborations are growing larger and larger and the multi-authorship trend is enhancing day by day. The time has come for the assessment community to focus on the regulating indices that take into account the multi-authorship factor in the research articles for the assessment of authors. The categories other than multi-authorship assign the total number of citations of a paper to each of its author, giving full weightage to each co-author even when there are multiple authors in a paper. This makes it difficult for the assessment community to assess the scholars with different co-authorship pattern. Considering this limitation, some of the indices have been proposed in the literature that considers multi-authorship in the research articles. The indices include h_i-index [13], fractional h and g indices [11], h_m-index [14], g_m-index [15], k-norm and w-norm [16] etc. These are the indices that consider the multi-authorship in the research and assign the credit proportionally to the number of co-authors working in the research.

2.9 Critical Analysis

In 2006, Batista et al. have proposed an index named h_i-index [13]. This index is achieved by dividing h-index by the number of co-authors in the h publications. This index is less biased as compare to indices other than multi-authorship due to the consideration of multi-authorship effect for the assessment of authors. They have compared h-index with hi index for the top 10% of the Brazilian researchers in 4 departments of science [13]. They found that there is too much difference in the value of h-index across these 4 departments. E.g. the h-index for the authors of Physics is much more than the authors of other subjects. The reason behind this is that the authors of physics have larger collaboration than the authors of other subjects. Likewise the departments having larger collaboration have the high value of h-index. After that he has obtained the h_i-index for these authors and he came up with the outcome that The hi value for the selected authors of 4 departments is very closer to each other and the ranking plot is smoother as compared to that of h-index so it is found that the fractionalized way of ranking the researchers is more fair as compared to the absolute one. However, in this paper h_i-index is not compared with any other multi-authorship index in order to see the difference in the values obtained by the multi-authorship indices.

To address this issue M Schreiber has performed an assessment analysis of two multi-authorship indices h_i-index and h_m-index with h-index [14]. The assessment is performed using the publication record of eight prominent scientists in the field of physics. It has been seen that by using multi-authorship indices some scientists with small collaborations are moved upward in their ranking as compared to the ones who have worked with the larger collaboration. After that the correlation is also found between these indices. The value of correlation between h-index and h_m-index is (0.90), and between h-index and h_i-index is (0.55) and between h_i-index and h_m-index is (0.69). The correlation shows that the values of h_m-index are closer to h-index as compared to that of h_i-index. As the h_i-index is restricted to the h-core so not further paper can be included in the

h_i-core even if it is a single author paper but in the case of h_m-index by fractionally dividing each paper by the inverse of the number of co-authors more papers can be included in the h_m-core. Furthermore the value of h_i-index is excessively small as it fractionally divides the h value by the mean number of co-authors in the first h publications, and a single paper with very large collaboration e.g. 50 co-authors leads to a serious decrease in the value of the h_i-index but in the case of h_m-index the influence of such a paper is reasonably negligible. The validity of this index needs to be tested using a large dataset with even more multi-authorship indices in order to find the true significance of the multi-authorship indices but in this paper the dataset taken for the evaluation of the indices is very small.

Another study conducted by Egghe in 2008 to outline the calculation of h- and g-indexes when using a fractional crediting system [11]. They have constructed two types of multi-authorship indices. One is the fractional h- and g-indexes that takes into account fractional citation counts and the second one is the fractional h- and g-indexes that takes into account fractional paper counts. The author has presented the inequalities between the fractional h-index and the un-weighted h-index. The same is performed for the g-index, and the inequalities are seen in both fractional crediting systems. however, it is observed that the indices are calculated on only one single imaginary author with 6 publications and then the indices are calculated on the dataset of the author himself.

In 2009, Michael Schreiber demonstrated the role of g_m-index in contrast to g_a-index and g-index by taking two model datasets of an imaginary author and one dataset of Egghe, the author of g-index [15]. The first model dataset contains 6 publications with different co-authorship pattern. The second model dataset contains 12 publications with five co-authors each and the third dataset contains the publication record of L. Egghe that contains 30 publications in which mostly L.Egghe is a single author or has one co-author. The author has simply found the values of these indices and compared these values to see the differences. The author concludes that g_m is the better way of ranking the authors in a fair manner as the g_m-index is used not only for the determination of effective rank but also for the normalized citations.

Anania et al proposed the indices named as k-norm and w-norm [34]. These indices are used to assess the scientist based on the normalized citations. These indices consider the multi-authorship effect i.e. the publications of an author is divided by the number of co-authors to obtain the normalized citations. The value of h-norm is calculated first and after that, the k-norm and w-norm are calculated. The indices are calculated for 109 departments of 64 universities of Italy. If k-norm is used in place of h-norm, the ranking is changed by 10 positions for 13 out of 109 departments and if w-norm is used in place of h-norm, the ranking is changed by 10 positions.

In 2016, Tehmina et al. have compare the role of fractionalized h and g indices with the index that consider the number of co-authors as well as the variation factor in the citation history [35]. For the calculation of the indices the record of only two researchers has been used that have same publications count and citation count to see the difference in the values of fractional h and h-indices and their proposed method. The value of fractionalized h and g indices are same for both of the researchers as they have same number of publications and citations. After using the variation factor it is found that the indices that consider variation in the citation history perform better than the fractionalized h and g indices and the authors can be distinguished through their citation history record.

2.10 Gap Analysis

After performing the literature survey, it has been observed that the multi-authorship based indices are mostly assessed on very small datasets making it difficult to find the actual performance of these multi-authorship indices. Furthermore, the assessments are carried out on the datasets of different domains, as a consequence of which, the comparison of indices and the identification of most contributing multi-authorship index is difficult. There is a strong need to evaluate these indices on a single comprehensive dataset from a specific domain in order to find the most contributing multi-authorship index for the ranking of authors.

Chapter 3

Methodology

3.1 Introduction

This chapter deals with the details of proposed methodology. In the previous chapter it was seen that the multi-authorship based indices were mostly assessed on very small datasets. Furthermore, the assessments was performed on the datasets of different domains. We intend to assess the multi-authorship based indices on a single comprehensive dataset in the field of Civil Engineering. The following research questions have been focused in this study:

1. Is there any correlation among the ranked lists obtained by the author ranking indices based on multi-authorship?
2. Does the international awardees lie in top of the ranked list and which multi-authorship index contributes the most in bringing the international awardees at the top?

To answer these questions, the methodology has been proposed in this chapter. The architectural diagram of methodology is shown in Figure 3.1.

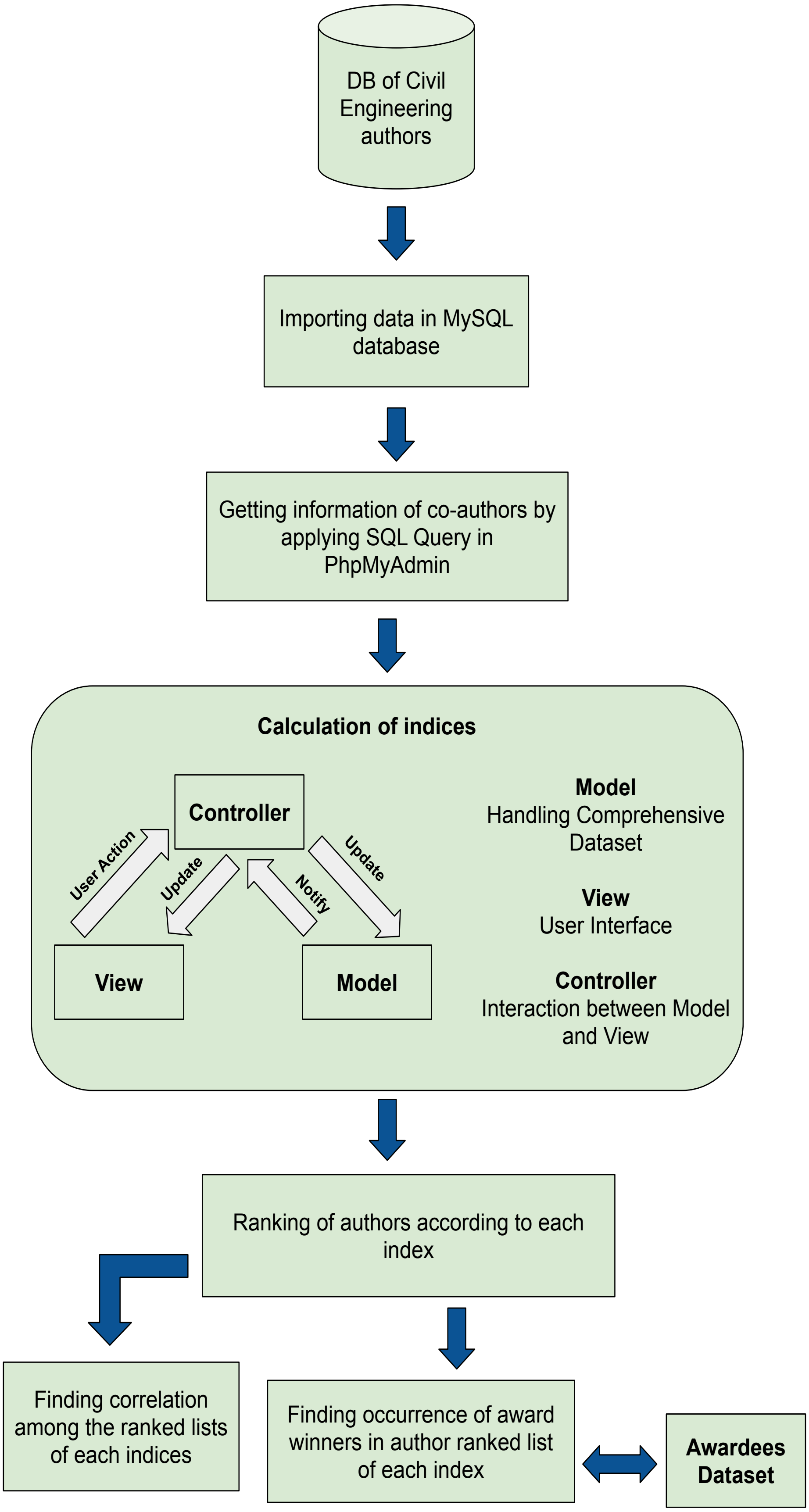

FIGURE 3.1: Architectural Diagram of Methodology

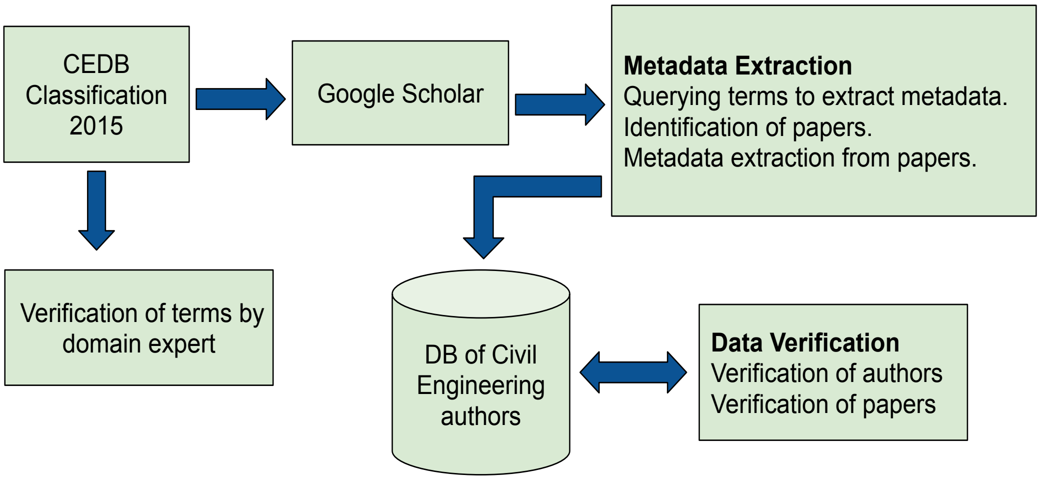

FIGURE 3.2: Steps of collection of acquired Dataset

3.2 Details of Acquired Dataset of Civil Engineering

For the assessment of selected indices, a comprehensive dataset in the field of Civil Engineering has been obtained from a former research [33]. This field covers most of the business in the world and still needs to be explored regarding the assessment of authors [36]. Furthermore, this field has large collaboration of authors and huge amount data to be presented. The information regarding the collection of acquired data set is mentioned in detail in the upcoming sections. The Figure 3.2 represents the steps of collection of acquired dataset.

3.2.1 Taxonomy Building

For the collection of data, a classification of this field was required. In this regard, a well known classification of Civil Engineering was used known as CEDB [1] (Civil Engineering Data Base). CEDB is the effort of a renowned society of Civil Engineering known as ASCE (American Society of Civil Engineering). This society publishes 37 journals in the field of Civil Engineering and the

[1] http://cedb.asce.org/CEDBsearch/

database of CEDB is updated every year. For the collection of dataset, the classification of 2015 was used as the collection was started in September 2015.

3.2.2 Search Engine

The search engine used to acquire the data of Civil Engineering against the categories of CEDB was Google Scholar. The reason to use Google Scholar is its large coverage of publications than other resources like Scopus and Web of Science. [37], [38]. It provides open access to data and the data is comprehensively available. Citation indexing is also a positive feature provided by Google Scholar. There is 13% growth of Google Scholar more than Web of Science. The citations in Google Scholar approximately increase by 1.5% per month and the database is updated every 2-3 months [39]. The citation noise in Google Scholar is less as compared to Web of Science [40] and Scopus [38]. It has been discussed in many studies that Google Scholar is the most suitable source of publications, citations and other metadata [38], [41], [42], [43], [44]. Therefore, Google Scholar was used to collect the data of Civil Engineering. The terms of CEDB categories were provided to Google Scholar by using a crawler. The crawler extracted the names of authors, title of papers, citation of papers, address of papers, and the year of publication.

It is discussed in a study about the extraction of record by Google Scholar against a given query [45]. It is found that when the query term is provided to Google Scholar the top results show more relevance to the query term and the relevance is directly proportional to the occurrence of query term in the title of paper. The results are ranked according to the query term and the number of received citations by the searched results.

The terms of classification were provided to crawler for the extraction of the data of authors and publications. It was observed that some of the terms of classification when provided to Google Scholar give irrelevant results. As an example, the problem is shown in Figure 3.3. The term "Construction" when

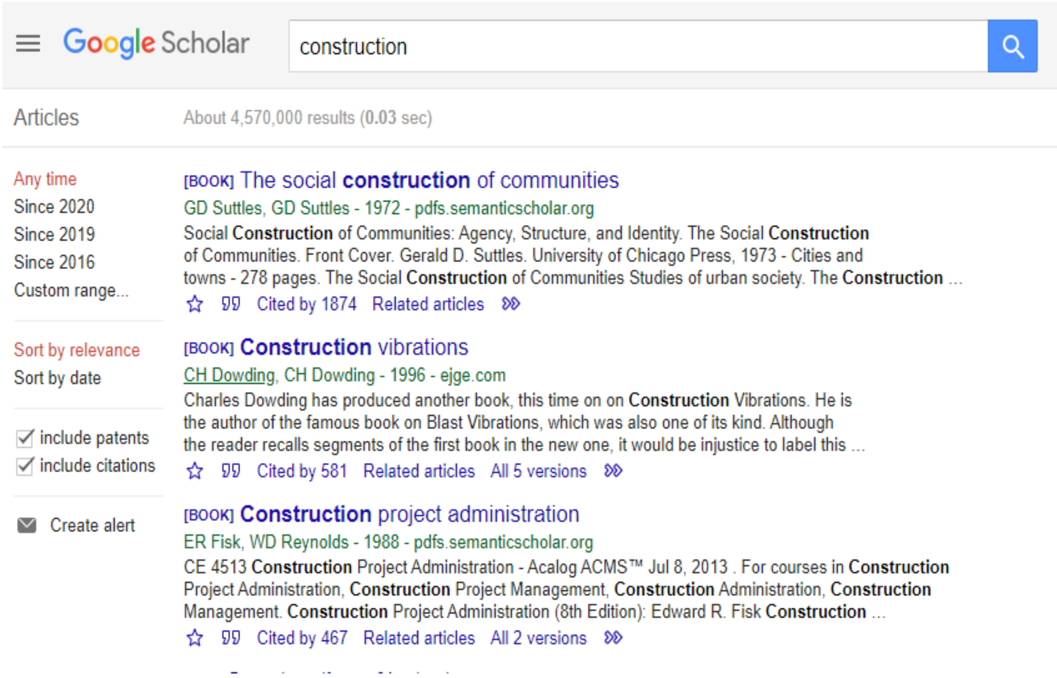

Figure 3.3: Irrelevant results against "Construction"

provided to Google Scholar, gives results irrelevant to "Construction" in Civil Engineering.

These terms were tuned by the domain experts and the terms that were only relevant to Civil Engineering were selected. After the tuning of these terms by the domain experts, it contained 152 terms and the data of these terms is given at http://cdsc-cust.org/Appendices/.

3.2.3 Crawling Metadata of Authors

The extraction of metadata of authors against the given query term is performed by a dedicated crawler. This crawler extracted the names of authors, title of papers, citation of papers, address of papers, conference or journal in which it was published and the year of publication. The data of subcategories were provided to crawler and it extracted the data against the given subcategories. Top 600 record was selected against the given query. The reason behind this is that Google Scholar gives irrelevant results beyond top 600 record. The records

Table 3.1: Dataset information before cleaning

Number of Authors	36,921
Number of Publication	20,307
Total Citations	2,184,638

were saved in a database and was maintained in SQL server. It was seen that for some queries, Google Scholar displayed the record below 600, so in that case only those records were maintained that were returned by Google Scholar. Table 3.1 shows the information of dataset collected through crawler.

3.2.4 Cleaning of Data

Many authors [46], [39], [47], [48] pointed that the data collected from Google Scholar must be cleaned as it has noise. Hence, the data was cleaned by first ensuring that the data belongs to the field of Civil Engineering. Secondly, it was ensured that there is disambiguation of authors in the record [49], [50]. In first case, the data was verified by following three steps:

1. Removal of invalid characters in the title of publications (%, $, ?, /, &, *, £).
2. Verify that the papers belong to the journals and conferences of Civil Engineering.
3. The verification was manually done for the remaining results.

After this process, some of the publications were removed and the remaining record is shown in Table 3.2.

In second step, the disambiguation of authors is performed by checking the duplication of the names of authors. For this purpose, it was checked whether the second name was shared with other authors and whether the first names were also distinct. If the first and second names were both same then it was

TABLE 3.2: Publications data set after verification

Descriptions	**Number of effected instances**
Publications with invalid keywords	34 Removed
Publications published in venues other than Civil Engineering	3250
Publications published in other venues and irrelevent to Civil Engineering	404 Removed
Total publications removed from dataset	438
Total publications after verification	19,869

considered as the duplication of authors. The profiles of those authors were manually checked for confirmation.

In the data set, there were 36,921 total authors. Among these authors, 17,589 authors had same last name and needed to be disambiguated. It was further found that there were 4130 distinct names that were shared by 17,589 authors. Among these 4130 names, some were shared by more than 100 authors and some were shared by only 2 authors. At the time of the collection of data of author, there were 2 cases in the name variations. In first case, both the first and last names of authors were same and in second case only the last names of authors are same. Both of these cases were handled to ensure that the authors are disambiguated and this task was accomplished manually.

In case 1, the publication record of 4130 authors was manually confirmed to ensure that the authors belong to same name or different name. It was found interestingly that there was not a single author having same last and first name. For case 2, the authors were again checked manually to ensure that they are different authors or same. It was detected that there were 88 variations as duplication of 45 authors which means that 45 authors were sharing the same last name but different first name. These duplications were removed from the database and after the cleaning of dataset, it contains 36,876 authors and 2,184,638 total number of citations. Table 3.3 shows the number of authors before and after the process of cleaning of dataset.

Table 3.3: Data of authors after verification

Descriptions	**Number of effected instances**
No. of authors before verification	36,921
Authors sharing same last name	17,589
Shared number of names	4130
Records having duplicated authors	88
No. of duplicated authors	45
Total authors after cleaning	36,876

3.3 Importing Data in MySQL Database

After the collection of data, it was saved in the form of Excel spread sheet. This spread sheet contains 5 tables named as "tbl_Authors", "tbl_Author_Papers", "tbl_Categories_CivilEngg", "tbl_Category_paper" and "tbl_Paper". The data required for the calculation of indices is the id of authors, publications of authors and the citations obtained by the publications. This information can be obtained from table "tbl_author_paper" and "tbl_papers". The table "tbl_Author_Papers" contains ap_id as primary key, author_id and paper_id while the table "tbl_papers" contains paper_id, paper_title, paper_url, conference and citation_count. From this table the required data is paper_id and citation_count only. In MySQL we created a new database named as "Indices" and in this database we created two tables as "tbl_author_paper" and "tbl_papers" and imported the required data in these two tables. PhpMyAdmin is the database administration tool that is used to handle this database. The subset of the data in these two tables is shown in Table 3.4 and Table 3.5 respectively.

The information of co-authors of each paper is necessary for the calculation of indices. Thus, to find the co-authors of each paper we have applied an SQL Query on the tables "tbl_author_paper" and "tbl_papers". This SQL Query extracts the information of co-authors of each paper. The query is illustrated in Listing 3.1

TABLE 3.4: Subset of Table "tbl_author_paper"

id	author_id	paper_id
1	1	1
2	2	2
3	3	2
4	4	3
5	5	3
6	6	3
7	7	4

TABLE 3.5: Subset of Table "tbl_papers"

id	paper_id	citation_count
1	1	335
2	2	256
3	3	257
4	4	300
5	5	211

TABLE 3.6: Subset of Table "tbl_coauthor_paper"

id	paper_id	co_authors	citation
1	1	1	335
2	2	2,3	256
3	3	4,5,6	257
4	4	7,8	300
5	5	9,10,11,12	211

```
SELECT a.paper_id, GROUP_CONCAT(a.author_id), b.citation_count
FROM tbl_author_paper a LEFT JOIN tbl_papers b ON
a.paper_id = b.paper_id GROUP BY a.paper_id,b.paper_id
```

LISTING 3.1: SQL Query for finding co-authors

TABLE 3.7: Tools and Technologies

Software Stack	WampServer Version 2.5
Web Server	Apache HTTP Server
RDBMS	MySQL
Source Code Editor	Sublime Text 3
Language	PHP
Spread Sheet	Microsoft Excel

Using the information of co-authors of each paper we created a third table named "tbl_coauthor_paper". This table contains paper_id, co-authors of each paper and the citations received by the papers. The information of this table is shown in Table 3.6.

3.4 Tools and Technologies

For the calculation of indices, the tools and technologies used are shown in Table 3.2. The programming language used to perform the calculation is PHP. Sublime Text 3 is used as a source code editor for writing the code in PHP language. A development framework known as Code-igniter is used which is based on model-view-controller (MVC) development pattern in which, Model contains queries that are applied on the database. View is used for user interaction and the Controller is used for interaction between Model and View. The logics are applied and different cases are handled in the controller section. Apache Web Server is used to execute the PHP files in web browser and these services (Apache, PHP and MySQL) are provided by a software stack known as Wamp Server.

3.5 Calculation of Indices

In MVC, the model Main_db is used to apply queries on the database. Hence, we have applied queries to access the data of authors and the publication data of each author. The query for accessing authors is represented in Listing 3.2 while the query for accessing publication data of authors is represented in Listing 3.3. The query for accessing authors returns the id of authors from the database and the query for accessing publication data of authors returns paper_id, citation count and coauthors of each paper against the id of authors. These two queries are used by the controller section to perform further calculation of indices. Hence the pseudocode of controller section is represented in the calculation process of each index. The calculated values of the indices are exported in Excel file using the functions of Excel libraries in the project.

```
function getAuthors()
{
$query = $this->db->query("SELECT DISTINCT(
tbl_author_paper.author_id) from tbl_author_paper
WHERE tbl_author_paper.author_id GROUP BY
tbl_author_paper.author_id ");
return $query->result();
}
```

LISTING 3.2: SQL Query for accessing authors

```
function getAuthorsData($id)
{
$query = $this->db->query("SELECT DISTINCT(author_id),
c.`paper_id`, c.`coauthors`, c.`citation` FROM
tbl_coauthor_paper c LEFT JOIN tbl_author_paper a ON
c.`paper_id` = a.`paper_id` WHERE a.author_id = $id
ORDER BY c.`citation` DESC");
return $query->result();
}
```

LISTING 3.3: SQL Query for accessing publication data of authors

3.5.1 g_m-index

The g_m-index is the modification of g-index. It considers multiple co-authorship in which each article is given fractional weight according to the number of co-authors. The effective rank is obtained by dividing 1 by the number of co-authors and adding the previous effective rank value to the next one. Afterwards the citation is normalized by dividing the citation count of each paper by the number of co-authors of each paper. The normalized citation count is sorted and the summation of normalized citation count is obtained. Equation 3.5.1 represents the formal description of g_m-index.

$$g_m \leq c_{eff}(g_m) \text{ Where } c_{eff}(r_{eff}) = 1\frac{}{r_{eff}} s_{eff}(r_{eff}) \text{ and } s_{eff}(r_{eff}) = \sum_{r'=1}^{r(r_{eff})} \frac{1}{a(r')} c(r') \quad (3.1)$$

The effective citation is obtained by dividing the value of normalized citation summation by the value of effective rank. The g_m-index is obtained by comparing effective citation by the effective rank. As the value of effective rank increases the value of effective citation, the comparison stops and the previous effective rank value is considered as g_m-index. If the effective rank value never increases the effective citation, the last effective rank value is considered as g_m-index. The pseudocode for finding g_m-index is shown in Listing 3.4 and the manual example for author_id 87 is shown in Table 3.8. The g_m-index for this example is 2.33.

```
procedure index()
   authors =  getAuthors()
   loop authors
      authorsData = getAuthorsData(author_id)
      paper_count = Count authorsData
```

TABLE 3.8: Manual example of g_m-index

rank	coauthors	citation	effRank	normCitation
1	2	49	0.5	24.5
2	4	45	0.75	11.25
3	2	14	1.25	7
4	4	13	1.5	3.25
5	3	13	1.83	4.33
6	4	5	2.08	1.25
7	4	4	2.33	1

Sort normalizedCitation	citationSummation	effectiveCitation
24.5	24.5	49
11.25	35.75	47.66
7	42.75	34.2
4.33	47.08	31.38
3.25	50.33	27.5
1.25	51.58	24.79
1	52.58	22.56

```
    normalizedCitationArray
    normalizedRankArray
    effectiveRankArray
    gm_index = 0
    effectiveRank = 0
    normalizedCitation = 0
    citationSummation = 0
    citationSummationPrev = 0
```

```
    effectiveCitation = 0
    loop authorsData
        coauthorsCount = count coauthors
        if paper_count = 1 then
            if coauthorsCount = 1 and citation > 0 then
                gm_index = 1
                return gm_index
                endif
            else if coauthorsCount = 1 and citation = 0 then
                gm_Index = 0
                return gm_index
                endif
            else if coauthorsCount > 1 and citation >= 1 then
                effectiveRank = 1 / coauthorsCount
                normalizedCitation = citation / coauthorsCount
                effectiveCitation = normalizedCitation /
                effectiveRank
            if effectiveRank <= effectiveCitation then
                gm_index = effectiveRank
                return gm_index
                endif
            else
                gm_index = 0
                return gm_index
                endif
        else if paperCount > 1 then
            if index of authorsData Array = 0 then
                effectiveRank = 1 / coauthorsCount
                endif
            else
                effectiveRank = effectiveRank + 1 /
                coauthorsCount
                normalizedCitation = citation / coauthorsCount
            if index of authorsData Array = 0 then
                citationSummation = normalizedCitation
```

```
                citationSummationPrev = citationSummation
                endif
            else
                citationSummation = normalizedCitation +
                citationSummationPrev
                citationSummationPrev = citationSummation
                effectiveCitation = citationSummation /
                effectiveRank
            Push values of effectiveRank in effectiveRankArray
            Push values of effectiveCitation in
            effectiveCitationArray
        endif
    end loop
    if effectiveRankArray is not empty
        previousValue = 0
        loop effectiveRankArray
            if value of effectiveCitationArray < value of
                effectiveRankArray
                gm_index = previousValue
                return gm_index
            endif
            else
                gm_index = value at index of effectiveRankArray
                previousValue = value at index of
                effectiveRankArray
                return gm_index
        end loop
    endif
  end loop
end procedure
```

LISTING 3.4: Pseudocode of g_m-index

3.5.2 h$_f$-index

This is a fractional counting method in which the publication rank remains unchanged and the citation is normalized by dividing the citation count by the number of co-authors of each paper. Equation 3.2 represents the formal description of h$_f$-index.

$$\frac{Yh_f}{\Phi(Yh_f)} \geq h_f \quad \text{Where y(i) = citation counts, } \Phi(i) \text{ = number of co-authors} \tag{3.2}$$

The normalized citation is sorted in descending order and h$_f$-index is obtained by comparing normalized citation with the adjacent rank of an author. As the value of rank increases the value of normalized citation, the comparison stops and the previous rank value is considered as h$_f$-index. If the rank value never increases the normalized citation then the last rank value is considered as h$_f$-index. The pseudocode for finding the h$_f$-index is shown in Listing 3.5 and the manual example for author_id 87 is shown in Table 3.9. The h$_f$-index value for this example is 4.

```
procedure index()
  authors = getAuthors()
  loop authors
     authorsData = getAuthorsData(author_id)
     paperCount = Count authorsData
     normalizedCitationArray
     hf_Index
     loop authorsData
        coauthorsCount = Count coauthors
        normalizedCitation = citation / coauthorsCount
        if  paperCount = 1 then
           if coauthorsCount = 1 and citation > 0 then
              hf_Index = 1
              return hf_Index
```

TABLE 3.9: Manual example of h_f-index

rank	coauthors	citation	normalizedCitation
1	2	49	24.5
2	4	45	11.25
3	2	14	7
4	4	13	3.25
5	3	13	4.33
6	4	5	1.25
7	4	4	1

Sort normalizedCitation	Old rank	New rank
24.5	1	1
11.25	2	2
7	3	3
4.33	5	4
3.25	4	5
1.25	6	6
1	7	7

```
            endif
        else if coauthorsCount = 1 and citation = 0 then
            hf_Index = 0
            return hf_index
            endif
        else if normalizedCitation >= 1 then
            hf_Index = 1
            return hf_index
```

```
                endif
            else
                hf-index = 0
                return hf_index
        endif
        else if paperCount > 1 then
            Push values of normalizedCitation in
            normalizedCitationArray
            endif
    end loop
    Sort normalizedCitationArray in reverse order
    if normalizedCitationArray is not empty
        loop normalizedCitationArray
            rank = index of array + 1
            if value at the index of array < rank
                hf_index = rank - 1
                return hf_index
                endif
            else
                hf_index = rank
                return hf_index
        end loop
    endif
  end loop
end procedure
```

LISTING 3.5: Pseudocode of $h_f - index$

3.5.3 g_f-index

This is a fractional counting method in which the publication rank remains unchanged and the citation count is normalized by dividing the citation count of each paper by the number of co-authors. The normalized citation count is sorted in descending order and the summation of normalized citation count is obtained. Equation 3.3 represents the formal description of g_f-index.

$$\sum_{i=1}^{g_f} \frac{Y_i}{\Phi(i)} \geq g_f^2 \quad \text{Where y(i) = citation counts, } \Phi(i) = \text{number of co-authors} \tag{3.3}$$

The ranks are squared and g_f-index is obtained by comparing the summation of normalized citation count with the square of the values of ranks. As the value of rank square increases the value of summation of normalized citation, the comparison stops and the previous rank value against rank square is considered as g_f-index. If the rank square value never increases the summation of normalized citation, the rank value against the last rank square is considered as g_f-index. The pseudocode for finding g_f-index is shown in Listing 3.6 and the manual example for author_id 87 is shown in Table 3.10. The g_f-index for this example is 7.

```
procedure index()
    authors =  getAuthors()
    loop authors
        authorsData = getAuthorsData(author_id)
        paper_count = Count authorsData
        normalizedCitationArray
        gf_index = 0
        normalizedCitationSummation = 0
        summationArray
        summationPrevious = 0
        loop authorsData
            coauthorsCount = count coauthors
            normalizedCitation = citation / coauthorsCount
            if paper_count = 1 then
                if coauthorsCount = 1 and citation > 0 then
                    gf_index = 1
                    return gf_index
                    endif
                else if coauthorsCount = 1 and citation = 0 then
                    gf_Index = 0
```

TABLE 3.10: Manual example of g_f-index

rank	coauthors	citation	normalizedCitation
1	2	49	24.5
2	4	45	11.25
3	2	14	7
4	4	13	3.25
5	3	13	4.33
6	4	5	1.25
7	4	4	1

Sort normalizedCitation	Old rank	New rank	rankSquare	citationSummation
24.5	1	1	1	24.5
11.25	2	2	4	35.75
7	3	3	9	42.75
4.33	5	4	16	47.08
3.25	4	5	25	50.33
1.25	6	6	36	51.58
1	7	7	49	52.58

```
        return gf_index
        endif
    else if normalizedCitation >= 1 then
        gf_Index = 1
        return gf_index
        endif
    else
        gf_index = 0
```

```
                return gf_index
            endif
            else if paperCount > 1 then
                Push values of normalizedCitation in
                normalizedCitationArray
                endif
        end loop
        Sort normalizedCitationArray in reverse order
        loop normalizedCitationArray
            if index of Array = 0
                normalizedCitationSummation = value at index
                of Array
                summationPrevious = normalizedCitationSummation
            else
                normalizedCitationSummation = value at index
                of Array + summationPrevious
                summationPrevious = normalizedCitationSummation
            Push values of normalizedCitationSummation in
            summationArray
        end loop
        if summationArray is not empty
            loop summationArray
                rank = index of Array + 1
                rankSquare = rank * rank
                if value at index of Array < rankSquare
                    gf_index = rank - 1
                    return gf_index
                    endif
                else
                    gf_index = rank
                    return gf_index
            end loop
        endif
    end loop
end procedure
```

LISTING 3.6: Pseudocode of g_f-index

3.5.4 g_F-index

This is a fractional counting method in which the citation count remains unchanged and the publication rank becomes the effective rank. The effective rank is achieved by dividing 1 by the number of co-authors foe each publication and adding the previous effective rank to the next one. Afterwards, the square of effective rank is also obtained. Equation 3.4 represents the formal description of g_F-index.

$$(\sum_{i=1}^{k} \frac{1}{\Phi(i)})^2 \leq \sum_{i=1}^{k} y_i \quad \text{Where} \quad g_F = \sum_{i=1}^{k} \frac{1}{\Phi(i)} \tag{3.4}$$

The citation counts are arranged and the summation of citation count is found. The g_F-index can be obtained by comparing the summation of citation count with the square of the effective rank. As the value of effective rank square increases the value of summation of normalized citation, the comparison stops and the previous rank value against rank square is considered as g_F-index. If the rank square value never increases the summation of normalized citation, the rank value against the last rank square is considered as g_F-index. The pseudocode for finding the g_F-index is shown in Listing 3.7 and the manual example for author_id 87 is shown in Table 3.11. The g_F-index for this example is 2.33.

```
procedure index()
    authors =  getAuthors()
    loop authors
        authorsData = getAuthorsData(author_id)
        paper_count = Count authorsData
        normalizedCitationArray
        gF_index = 0
```

Table 3.11: Manual example of g_F-index

rank	coauthors	citation	effRank
1	2	49	0.5
2	4	45	0.75
3	2	14	1.25
4	4	13	1.5
5	3	13	1.83
6	4	5	2.08
7	4	4	2.33

effRankSquare	citationSummation
0.25	49
0.56	94
1.56	108
2.25	121
3.34	134
4.32	139
5.42	143

```
    citationSummation = 0
    summationArray
    summationPrevious = 0
    normalizedRankArray
    normalizedRankSquareArray
    loop authorsData
        coauthorsCount = count coauthors
        normalizedRank = 1 / coauthorsCount
```

```
        if paper_count = 1 then
           if coauthorsCount = 1 and citation > 0 then
              gF_index = 1
              return gF_index
              endif
           else if coauthorsCount >= 1 and citation = 0 then
              gF_Index = 0
              return gF_index
              endif
           else if coauthorsCount > 1 and citation >= 1 then
              normalizedRankSquare = normalizedRank *
              normalizedRank
              if normalizedRankSquare <= citation then
                 gF_Index = normalizedRank
                 return gF_index
                 endif
              else
                 gF_index = 0
                 return gF_index
        endif

        else if paperCount > 1 then
           if index of Array of authorsData = 0
              normalizedRank = 1 / coauthorsCount
              endif
           else
              normalizedRank = normalizedRank + 1 /
              coauthorsCount
           if index of Array of authorsData = 0
              citationSummation = citation
              summationPrevious = citationSummation
              endif
           else
              citationSummation = citation +
              summationPrevious
```

```
                summationPrevious = citationSummation
            Push values of normalizedRank in
            normalizedRankArray
            Push values of normalizedCitationSummation in
            summationArray
            endif
    end loop
    Sort normalizedCitationArray in reverse order
    loop normalizedCitationArray
        if index of Array = 0
            normalizedCitationSummation = value at index
            of Array
            summationPrevious = normalizedCitationSummation
        else
            normalizedCitationSummation = value at index
            of Array + summationPrevious
            summationPrevious = normalizedCitationSummation
        Push values of normalizedCitationSummation in
        summationArray
    end loop
    if normalizedRankArray is not empty
        loop normalizedRankArray
            normalizedRankValue = value at the index
            of Array
            rankSquare = normalizedRankValue *
            normalizedRankValue
            if value at index of summationArray < rankSquare
                gF_index = rank - 1
                return gF_index
                endif
            else
                gF_index = rank
                return gF_index
        end loop
    endif
```

```
    end loop
end procedure
```

LISTING 3.7: Pseudocode of g_F-index

3.5.5 h_i-index

The h_i-index indicates the number of papers written in the carrier of an author with at least h_i citations if the author has written alone. Equation 3.5 represents the formal description of h_i-index.

$$h_i = \frac{h^2}{N_a^{(T)}} \text{ Where h represents h-index and } N_a^{(T)} \text{ represents authors in h papers} \tag{3.5}$$

This index is useful in a sense that it provides the contribution of an individual author even if multiple authors have written the publication. It can be obtained by dividing the square of h-index of an author by the total number of authors in considered h publications of an author. The pseudocode for finding the h_i-index is shown in Listing 3.8 and the manual example for author_id 87 is shown in Table 3.12.

```
procedure index()
    authors  = getAuthors()
    loop authors
        authorsData = getAuthorsData(author_id)
        paperCount = Count authorsData
        matchFlag = 0
        normalizedCitationArray
        hi_Index = 0
        h_Index = 0
        coauthorsCountSumm = 0
        loop authorsData
```

TABLE 3.12: Manual example of h_i-index

rank	coauthors	citation
1	2	49
2	4	45
3	2	14
4	4	13
5	3	13
6	4	5
7	4	4

```
        coauthorsCount = Count coauthors
        coauthorsCountSumm = coauthorsCount +
        coauthorsCountSumm
        end loop
    loop authorsData
        coauthorsCount = Count coauthors
        normalizedCitation = citation / coauthorsCount
        if  paperCount = 1 then
            if coauthorsCount = 1 and citation > 0 then
                hi_Index = 1
                return hi_Index
                endif
            else if coauthorsCount = 1 and citation = 0
            then
                hi_Index = 0
                return hi_Index
                endif
            else if coauthorsCount >= 1 then
                hi_Index = 1 / coauthorsCount
                return hi_Index
```

```
                    endif
                else
                    hi-index = 0
                    return hi_index
            endif
            else if paperCount > 1 then
                if citation < index of authorsData Array + 1
                then
                    matchFlag = 1
                    h_Index = index of authorsData Array
                    h_Index_square = h_Index * h_Index
                    hi_Index = h_Index_square / coauthorsCount
                    return hi_Index
                    endif
                else if matchFlag = 0 and index of authorsData
                = paperCount - 1 then
                    hi_Index = (paperCount * paperCount) /
                    coauthorsCount
                    return hi_Index
                endif
            endif
        end loop
    end loop
end procedure
```

LISTING 3.8: Pseudocode of h_i-index

3.5.6 h_m-index

This is the modification of h-index that takes multiple co-authorship into account by counting the papers fractionally according to inverse of the number of co-authors. Equation 3.5.6 represents the formal description of h_m-index.

$$r_{eff}(r) = \sum_{r'=1}^{r} \frac{1}{a(r')} \quad \text{then} \quad c(r(h_m)) \geq h_m \geq c(r(h_m)+1) \quad (3.6)$$

TABLE 3.13: Manual example of h_m-index

rank	coauthors	citation	effRank
1	2	49	0.5
2	4	45	0.75
3	2	14	1.25
4	4	13	1.5
5	3	13	1.83
6	4	5	2.08
7	4	4	2.33

The effective rank is achieved by dividing 1 by the number of co-authors and adding the previous effective rank to the next one. The h_m-index can be obtained by comparing the citation count with the effective rank. As the value of effective rank increases the value of citation count, the comparison stops and the previous effective rank value is considered as h_m-index. If the effective rank value never increases the citation count, the last effective rank value is considered as h_m-index. The pseudocode for finding the h_m-index is shown in Listing 3.9 and the manual example for author_id 87 is shown in Table 3.13. The h_m-index for this example is 2.33.

```
procedure index()
   authors =  getAuthors()
   loop authors
      authorsData = getAuthorsData(author_id)
      paper_count = Count authorsData
      citationArray
      effectiveRankArray
      hm_index = 0
      effectiveRank = 0
```

```
    loop authorsData
        coauthorsCount = count coauthors
        if paper_count = 1 then
            if coauthorsCount = 1 and citation >= 1 then
                hm_index = 1
                return hm_index
                endif
            else if coauthorsCount >= 1 and citation = 0
            then
                hm_Index = 0
                return hm_index
                endif
            else if coauthorsCount > 1 and citation >= 1
            then
                effectiveRank = 1 / coauthorsCount
                if effectiveRank <= citation then
                    hm_Index = effectiveRank
                    return hm_index
                    endif
                else
                    hm_index = 0
                    return hm_index
            endif
        endif
        else if paperCount > 1 then
            if index of authorsData Array = 0 then
                effectiveRank = 1 / coauthorsCount
                endif
            else
                effectiveRank = effectiveRank + 1 /
                coauthorsCount
            Push values of effectiveRank in
            effectiveRankArray
            Push values of citation in citationArray
        endif
```

```
        end loop
        if effectiveRankArray is not empty
            previousValue = 0
            loop effectiveRankArray
                if value of citationArray < value of
                    effectiveRankArray then
                    hm_index = previousValue
                    return hm_index
                    endif
                else
                    hm_index = value at index of
                    effectiveRankArray
                    previousValue = value at index of
                    effectiveRankArray
                    return hm_index
            end loop
        endif
    end loop
end procedure
```

LISTING 3.9: Pseudocode of h_m-index

3.5.7 k-norm

The k-norm index is the modification of k-index that takes into account the normalized citations rather than absolute citations. The normalized citations are obtained by dividing the citation counts by the number of co-authors. These normalized citations when compared with ranks give the value of h-norm. The value of h-norm is further utilized to obtain the value of k-norm. Equation 3.5.7 represents the formal description of k-norm.

k-norm = h-norm + (1 - (h-norm^2/ $\sum_{j=1,2,\ldots,h-norm} citnorm_j$)), $\forall\ h-norm > 1\ and\ k-norm = 0,\ \ if\ \ h-norm = 0, \quad (3.7)$

The value of h-norm represents the normalized individual h-index and the $citnorm_j$ represents the number of normalized citations obtained by the j^{th} publication included in the author's h-norm core. The pseudocode for finding the k-norm is shown in Listing 3.10 and the manual example for author_id 87 is shown in Table 3.14. The value of k-norm for the example shown in Table 3.14 can be obtained as: k-norm = 4+(1-(4^2/47.08)) = 4.66

```
procedure index()
   authors =  getAuthors()
   loop authors
      authorsData = getAuthorsData(author_id)
      paper_count = Count authorsData
      normalizedCitationArray
      citationSummationArray
      knorm = 0
      hnorm = 0
      summValue = 0
      summValue1 = 0
      normalizedCitation = 0
      citationSummation = 0
      citationSummationPrev = 0
      loop authorsData
         coauthorsCount = count coauthors
         if paper_count = 1 then
            if coauthorsCount = 1 and citation > 0 then
               hnorm = 1
               knorm = hnorm + (1-(hnorm * hnorm) / citation)
               return knorm
               endif
            else if coauthorsCount = 1 and citation = 0 then
               hnorm = 0
```

TABLE 3.14: Manual example of k-norm

rank	coauthors	citation	normalizedCitation
1	2	49	24.5
2	4	45	11.25
3	2	14	7
4	4	13	3.25
5	3	13	4.33
6	4	5	1.25
7	4	4	1

Sort normalizedCitation	Old rank	New rank	citationSummation
24.5	1	1	24.5
11.25	2	2	35.75
7	3	3	42.75
4.33	5	4	47.08
3.25	4	5	50.33
1.25	6	6	51.58
1	7	7	52.58

```
        return hnorm
        endif
    normalizedCitation = citation / coauthorsCount
    else if normalizedCitation >= 1 then
        hnorm = 1
        knorm = hnorm + (1-(hnorm * hnorm) /
        normalizedCitation)
        return knorm
```

```
                endif
            else
                knorm = 0
                return knorm
        endif
        else if paperCount > 1 then
            normalizedCitation = citation / coauthorsCount
            Push values of normalizedCitation in
            normalizedCitationArray
            endif
    end loop
    Sort normalizedCitationArray in reverse order
    loop normalizedCitationArray
        if index of authorsData Array = 0 then
            citationSummation = value of
            normalizedCitationArray
            citationSummationPrev = citationSummation
            endif
        else
            citationSummation = value of
            normalizedCitationArray + citationSummationPrev
            citationSummationPrev = citationSummation
        Push values of citationSummation in
        citationSummationArray
         end loop
    if normalizedCitationArray is not empty then
        loop normalizedCitationArray
            rank = index at normalizedCitationArray + 1
            if value of normalizedCitationArray < rank then
                hnorm = rank - 1
                endif
            else
                hnorm = rank
        end loop
    endif
```

```
        loop citationSummationArray
            if hnorm = index of citationSummationArray + 1 then
                summValue = value of citationSummationArray
                endif
            else
                summValue1 = value of citationSummationArray
        end loop
        if summValue > 0 then
            knorm = hnorm + (1-(hnorm * hnorm)/summValue)
            endif
        else
            knorm = 0
            return knorm
            endif
    end loop
end procedure
```

LISTING 3.10: Pseudocode of k-norm

3.5.8 w-norm

The w-norm index is the modification of w-index that takes into account the normalized citations rather than absolute citations. The normalized citations are obtained by dividing the citation counts by the number of co-authors. These normalized citations when compared with ranks give the value of h-norm. The value of h-norm is further utilized to obtain the value of w-norm. Equation 3.5.8 represents the formal description of w-norm.

$$\text{w-norm} = \text{h-norm} + (1 - \text{h-norm}^2/totcit-norm),\ \forall h-norm > 0\ \ and\ \ w-norm = totcit-norm/(1+totcit-norm)\ \ if\ \ h-norm = 0 \quad (3.8)$$

TABLE 3.15: Manual example of w-norm

rank	coauthors	citation	normalizedCitation
1	2	49	24.5
2	4	45	11.25
3	2	14	7
4	4	13	3.25
5	3	13	4.33
6	4	5	1.25
7	4	4	1

Sort normalizedCitation	Old rank	New rank	citationSummation
24.5	1	1	24.5
11.25	2	2	35.75
7	3	3	42.75
4.33	5	4	47.08
3.25	4	5	50.33
1.25	6	6	51.58
1	7	7	52.58

The value of h-norm represents the normalized individual h-index and the totcit-norm represents the total number of normalized citations obtained by the publications of an author. The process for finding the w-norm is shown in Listing 3.11 and the manual example for author_id 87 is shown in Table 3.15. The value of w-norm for the example shown in Table 3.15 can be obtained as:
w-norm = $4+(1-(4^2/52.58)) = 4.695$

```
procedure index()
   authors =  getAuthors()
   loop authors
      authorsData = getAuthorsData(author_id)
      paper_count = Count authorsData
      normalizedCitationArray
      citationSummationArray
      wnorm = 0
      hnorm = 0
      lastSumm = 0
      normalizedCitation = 0
      citationSummation = 0
      citationSummationPrev = 0
      loop authorsData
         coauthorsCount = count coauthors
         if paper_count = 1 then
            if coauthorsCount = 1 and citation > 0 then
               hnorm = 1
               wnorm = hnorm + (1-(hnorm * hnorm) / citation)
               return wnorm
               endif
            else if coauthorsCount = 1 and citation = 0 then
               hnorm = 0
               return hnorm
               endif
            normalizedCitation = citation / coauthorsCount
            else if normalizedCitation >= 1 then
               hnorm = 1
               wnorm = hnorm + (1-(hnorm * hnorm) /
               normalizedCitation)
               return wnorm
               endif
            else
               wnorm = normalizedCitation/(1+
               normalizedCitation)
```

```
                return wnorm
          endif
          else if paperCount > 1 then
             normalizedCitation = citation / coauthorsCount
             Push values of normalizedCitation in
             normalizedCitationArray
             endif
       end loop
       Sort normalizedCitationArray in reverse order
       loop normalizedCitationArray
          if index of authorsData Array = 0 then
             citationSummation = value of normalizedCitationArray
             citationSummationPrev = citationSummation
             endif
          else
             citationSummation = value of normalizedCitationArray
             + citationSummationPrev
             citationSummationPrev = citationSummation
          Push values of citationSummation in
          citationSummationArray
          lastSumm = last value of citationSummationArray
           end loop
       if normalizedCitationArray is not empty then
          loop normalizedCitationArray
             rank = index at normalizedCitationArray + 1
             if value of normalizedCitationArray < rank then
                hnorm = rank - 1
                endif
             else
                hnorm = rank
          end loop
          if lastSumm > 0 then
             wnorm = hnorm + (1-(hnorm * hnorm)/lastSumm)
             endif
          else
```

```
            wnorm = 0
            return wnorm
        endif
    end loop
end procedure
```

Listing 3.11: Pseudocode of w-norm

The authors are ranked separately according to each index and the ranked authors are assessed upon the research questions discussed in the study.

3.6 Awardees Dataset

The indices are tested using the data set of international award winners in Civil Engineering. The international awardees include the award winners of American Society of Civil Engineering (ASCE), American Concrete Institute (ACI), Canadian Society of Civil Engineering (CSCE) and Institute of Civil Engineering (ICE). These are the most renowned societies and give awards to the best authors of Civil Engineering [33]. The considered awards are those that are solely based on the research contributions. The awards based on teaching, planning, professional leadership, design or management are not considered. Furthermore, the award winners of 2016 are considered as the dataset of Civil Engineering contains the publication record till 2015. The list of award winners is shown in Table 3.16 and the data of awardees is represented in "Appendix B".

3.7 Correlation Among the Ranked Lists of Indices

The correlation is calculated in order to find the strength of association between the ranked lists of indices. The value of correlation varies between 1.0 and -1.0 i.e. 1.0 represents a complete positive correlation and -1.0 represents a complete negative correlation. In order to find the correlation between the ranked lists,

TABLE 3.16: Civil Engineering Societies and their Awards

Societies and their Awards	**Total Awardees**
ASCE	**56**
Maurice A. Biot Medal	1
Jack E. Cermak Medal	1
J. James R. Croes Medal	6
Rudolph Hering Medal	6
Karl Emil Hilgard Hydraulic Prize	3
Julian Hinds Award	1
Walter L. Huber Research Prizes	5
Wesley W. Horner Award	5
Samuel Arnold Greeley Award	6
Daniel W. Mead Prize	1
Thomas A. Middlebrooks Award	1
Moisseiff Award	5
Alfred Noble Prize	2
Norman Medal	4
Ralph B. Peck Award	1
Peurifoy Construction Research Award	1
Harold R. Peyton Award	1
Raymond C. Reese Research Prize	2
Thomas Fitch Rowland Prize	2
T.Y. Lin Award	2
ACI	**15**
Wason Medal for Most Meritorious Paper	2
Wason Medal for Material Research	3
Mete A. Sozen Award	4
ACI Construction Award	3
ACI Design Award	3
CSCE	**18**
Thomas C. Keefer Medal	3
Casimir Gzoski Medal	5
P.L. Pratley Award	3
Donald Stanley Award	5
Stephen G. Revay Award	2
ICE	**4**
James Alfred Ewing Medal	1
Telford Award	3
Grand Total	**93**

we have used Spearman's rank correlation as it suites the rank nature of indices [51], moreover used by the baseline paper [33]. The formula of Spearmans rank correlation coefficient is represented in Equation 3.9

$$R_s = 1 - (\frac{6 \sum d^2}{n(n^2 - 1)}) \tag{3.9}$$

The formula is applied in Microsoft Excel in order to find the correlation between ranked lists of indices discussed in the study.

3.7.1 Significance of Correlation in the Scientific Community

In order to evaluate the importance and significance of multi-authorship indices, an awardees dataset is developed on the basis of which the evaluation of indices is performed however before that it is necessary to critically observe the relationship between the indices. This is performed by finding the value of correlation among the indices. The correlation can help the scientific community in various aspects i.e. the indices can be interchangeably used when there is strong correlation between the indices and they bring more awardees in top positions. Secondly the indices with strong correlation can be further investigated to see which index has low computational cost or has easy access to the variables. The one with low computational cost or easy access to the variables can be preferably used for the evaluation purpose. Furthermore, if the correlation is strong and the indices are bringing less awardees in top positions then according to an altogether different philosophy, the scientific community can acknowledge to develop a new index that can bring more awardees in top ranks. In the case where there is week correlation between the indices it means that the indices are presenting different rankings. Now it is necessary to see that among these indices how much percentage of awardees are brought by the indices. If the correlation is week and they are bringing less awardees i.e. 20-30% awardees then it can assist in combining those indices to develop a better index so that it can bring maximum awardees in the top ranking positions.

3.8 Occurrence of Awardees in Author Ranked Lists

In this step, the occurrence of awardees is checked in the ranked list of each index discussed in the study. First of all the inclusion of awardees is checked in top 10% of the ranked lists and after that the awardees are checked in 11-20%, 21-30%, 31-40% and up 91-100% of the ranking lists. The occurrence of awardees is also checked in top 100, 500 and 1000 data of the ranked lists. First of all the indices are calculated for the comprehensive dataset of Civil Engineering and the authors are separately ranked according to each index. After that the location of award winners is identified in the ranked list. After this step, the number of awardees is found in the ranked lists as discussed above.

Chapter 4

Results and Discussion

This chapter focuses on the solution to research questions discussed in methodology.

4.1 Correlation Between the Ranked Lists

This section deals with the solution to first research question i.e. find the correlation between ranked lists obtained from author ranking indices that consider multi-authorship? The purpose of this evaluation is to find the strength of association or similarity between the ranked lists of multi-authorship indices. In this regard, the Spearman Rank correlation has been calculated between the ranked lists. The value of correlation lies between -1.0 and 1.0 i.e. -1.0 represent perfect negative correlation and 1.0 represents perfect positive correlation. The correlation found between the ranked lists of multi-authorship indices is shown in Table 4.1. We have followed the guidelines of Evens et al. regarding measurement of the strength of correlation. The correlation is very weak below 0.2, weak (0.2-0.39), moderate (0.4-0.59), high (0.6-0.79) and very high above 0.8 [52]. Figure 4.1 graphically represents the correlation found between the ranked lists of multi-authorship indices.

TABLE 4.1: Correlation between Indices

	h_f-index	g_f-index	g_m-index	h_i-index	k-norm	h_m-index	w-norm	g_F-index
h_f-index	1	0.980256	0.786906	0.649737	0.858788	0.789391	0.858901	0.785615
g_f-index		1	0.797141	0.519606	0.840816	0.791447	0.842187	0.795842
g_m-index			1	0.568329	0.698937	0.99515	0.700522	0.999264
h_i-index				1	0.570687	0.582821	0.569021	0.566627
k-norm					1	0.70148	0.999907	0.697643
h_m-index						1	0.702981	0.994412
w-norm							1	0.69931
g_F-index								1

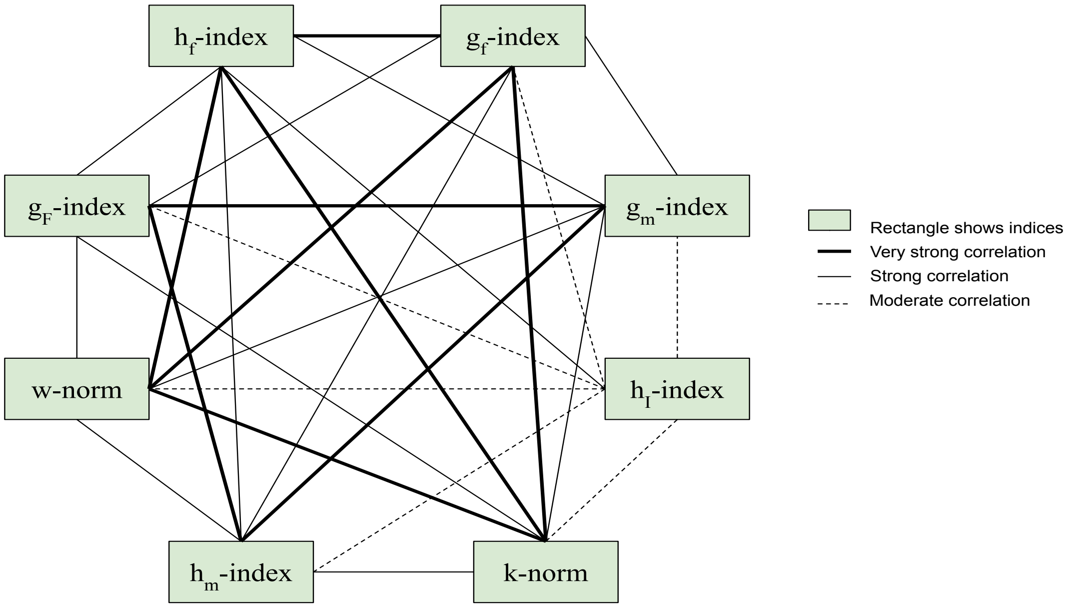

FIGURE 4.1: Strength of association between indices

The Table 4.1 shows correlation between the ranked lists of 8 multi-authorship based indices. The value of correlation nearer to 1 indicates that the indices are strongly correlated. The correlation with index itself is 1. In Figure 4.1 the rectangles represent the indices while the edges represent the strength of association between the indices. The results depict that the ranked lists of most of the indices have either strong or very strong correlation among each other. The dotted lines represent moderate correlation while narrow and bold lines represent strong and very strong correlation respectively. The indices with strong correlation and furthermore with more occurrence of award winners in the author ranked lists can be used interchangeably. Table 4.2 show the indices used in the study that can be interchangeably by the scientific community.

The frequency of correlation between the indices is shown in Table 4.3. It shows the frequency of strong, very strong and moderate correlations among indices. For instance, h_f-index has strong correlation with 4 indices while h_i-index has strong correlation with 1 index and very strong correlation with none

TABLE 4.2: Indices that can be interchangeably used

First Index	Second Index
h_m-index	g_F-index
k-norm	w-norm

of the indices. Overall, it is seen that the number of indices having strong or very strong correlations is greater than the number of indices having moderate correlations.

The results obtained by applying Spearman Rank correlation on multi-authorship based indices indicates that there is close match between the ranked lists of these indices. Except h_i-index all other indices have either strong or very strong correlations among each other. The h_i-index has moderate correlation with all other indices except h_f-index as it show strong correlation.

4.1.1 Correlation in Baseline Paper

Raheel et al. have found correlation among the indices that are based on citation intensity and age of publication [33]. The indices were calculated on the same data set as we are using i.e. the data set of Civil Engineering domain. They have applied Spearman Rank correlation on 11 indices and their results indicate that their is week correlation between most of the indices. The results are shown in Table 4.4. The Table 4.4 indicates that some of the indices have more strong correlations than week ones and some have more week correlations than strong ones. The case of negative correlation also exists as their are 2 indices i.e. A-index and raw h-rate that have negative correlation. Overall, it is seen that the indices week correlation prevail the cases of strong correlation. This indicates that the author ranking lists of indices considering citation intensity and age of

TABLE 4.3: Strength of association between indices

Index	Very strong correlation	Strong correlation	Moderate correlation
h_f-index	3	4	0
g_f-index	3	3	1
g_m-index	2	4	1
h_i-index	0	1	6
k-norm	3	3	1
h_m-index	2	4	1
w-norm	3	3	1
g_F-index	2	4	1

publication are not similar. In multi-authorship indices case we have seen that most of the indices show strong or very strong correlation. Hence, it can be said that the correlation trend in multi-authorship indices is different from the indices that consider citation intensity and age of publication.

4.2 Occurrence of Awardees in Author Ranked Lists

This section focuses on the second research question which is to assess the contribution of multi-authorship based indices in bringing the award winners at the top of ranked lists. To address this question, the occurrence of award winners is checked in top 10% of the ranked list of each index. The awardees are further checked in 11-20%, 21-30% and up to 91-100% of the ranked lists. The inclusion of awardees is also checked in top 100, 500 and 1000 of the ranked lists obtained by these indices.

Table 4.4: Correlation among indices based on citation intensity and age of publication

Index	Strong correlation	Weak correlation	Negative correlation
h-index	5	4	
Wu-index	5	3	
A-index	3	6	1
Maxprod-index	3	6	
Tepered h-index	5	2	
F-index	5	4	
T-index	5	4	
AR-index	3	4	
$Q^2 - index$	3	6	
raw h-rate	0	4	1
Contemporary h-index	5	3	

There are 93 total awardees that are present in the awardees data set. We have checked if there is any repetition in the names of these award but we found that there is no repetition and all the awardees are unique. Ideally, it was assumed that all of the awardees would lie in the data set however, we found that there are only 27 awardees that lie in the data set. Hence, the presence of these 27 awardees is checked in the ranked lists obtained by these indices.

As the award winners hold strong research background, therefore it was expected that all of the award winners would lie in top 10% of the ranked lists. Whereas, we came across with different results as illustrated in Figure 4.2. The Figure 4.2 illustrates that the maximum of 67% of awardees are found in top 10% of the ranked list. The indices that remained successful in bringing 67% of awardees in the top ranks are g_f-index followed by g_m-index. Comparatively

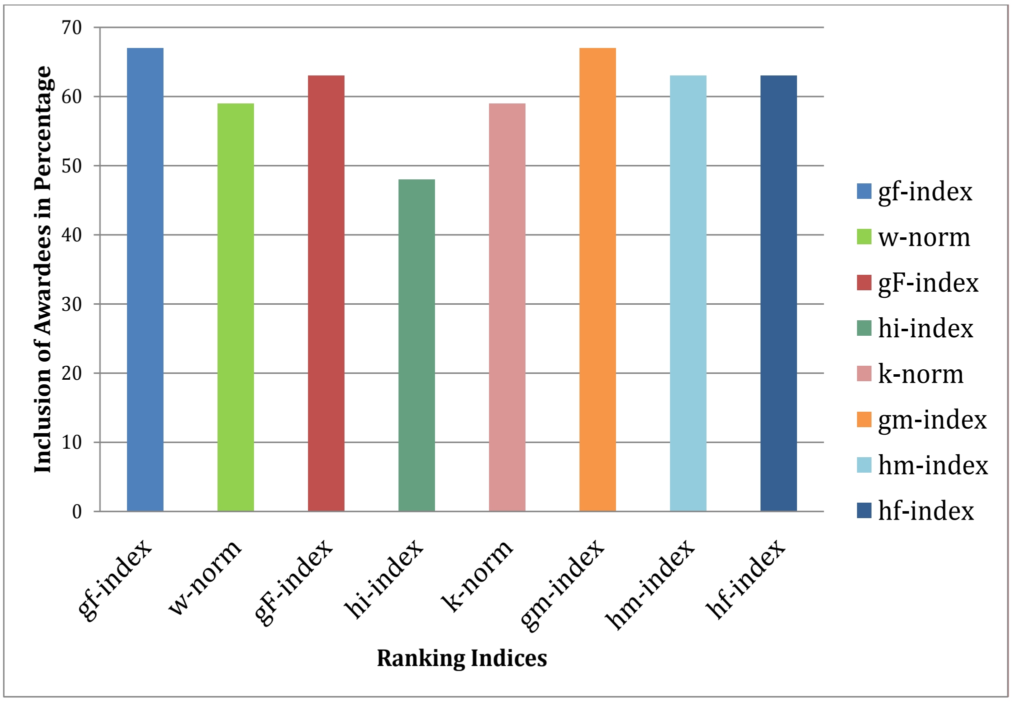

FIGURE 4.2: Percentage of occurrence of awardees in top 10% of ranked list

better results of these indices could be due to the fact that these indices handle the number of co-authors in more appropriate way [15].

The performance of g_F-index, h_m-index and h_f-index is around 63% in bringing awardees at the top ranks while k-norm and w-norm came up with around 59% awardees. The least number of awardees (Around 48%) are brought by h_i-index. This could be due to the fact that the value of h_i-index is mostly small as it fractionally divides the h value by the mean number of co-authors in the first h publications, and a single paper with very large collaboration e.g. 25 co-authors could lead to a serious decrease in the overall value of h_i-index. Whereas in the case of other indices like g_m-index and g_f-index, the influence of such a paper is reasonably negligible.

None of the index succeeded in bringing even 70% of awardees at the top ranks. Therefore, we decided to analyze the trend of awardees in 100% results. It was expected that there would be mere presence of awardees below the top 50%

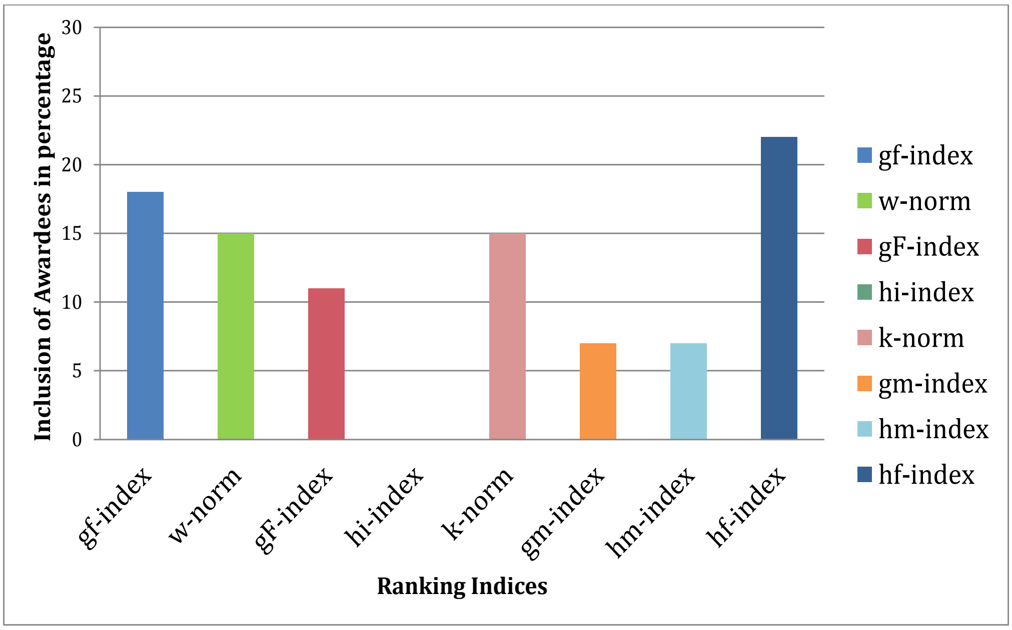

FIGURE 4.3: Percentage of occurrence of awardees in 11-20% of ranked list

whereas, the presence of awardees can be seen from top 10% to the least 10% of ranked lists.

The result of occurrence of awardees in 11-20% of the ranked list is shown in Figure 4.3. Furthermore, the awardees trend in 21-30%, 31-40% up to 91-100% can be seen in Figure 4.4, Figure 4.5, Figure 4.6, Figure 4.7, Figure 4.8, Figure 4.9, Figure 4.10 and Figure 4.11 respectively. It can be seen that the awardees also lie in 91-100% result.

In 11-20% of the ranked list, the maximum of awardees (Around 22%) are brought by h_f-index while none of the awardees can be seen in the case of h_i-index. In 21-30% of the ranked list, h_m-index remained successful in bringing most of the awardees while h_f-index and g_f-index came across none of the awardees. The awardess trend is different through out the results whereas in the least results it is observerd that h_i-index is bringing most of the awardees.

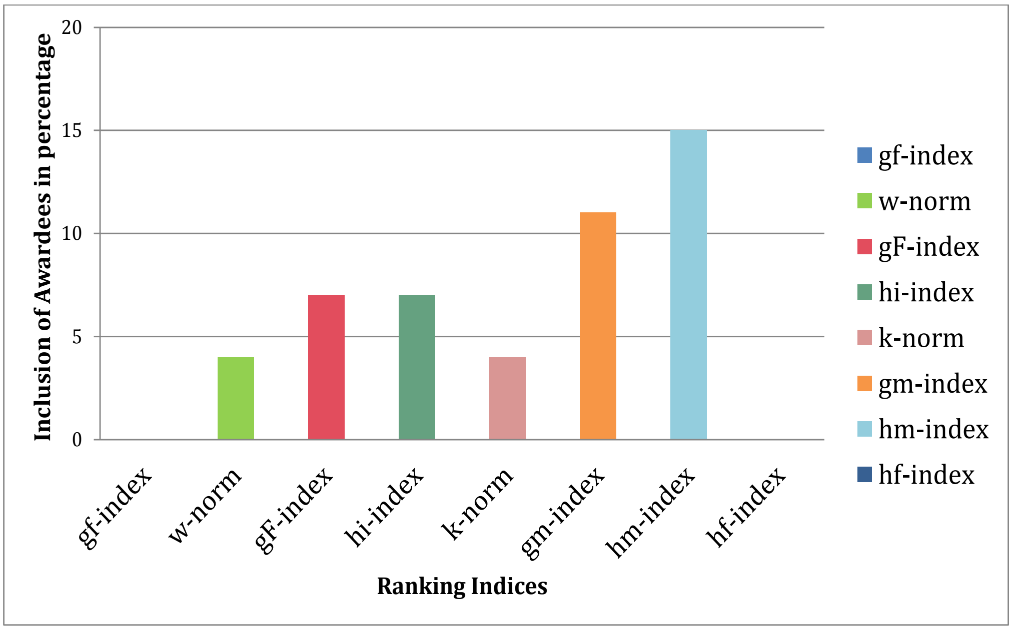

FIGURE 4.4: Percentage of occurrence of awardees in 21-30% of ranked list

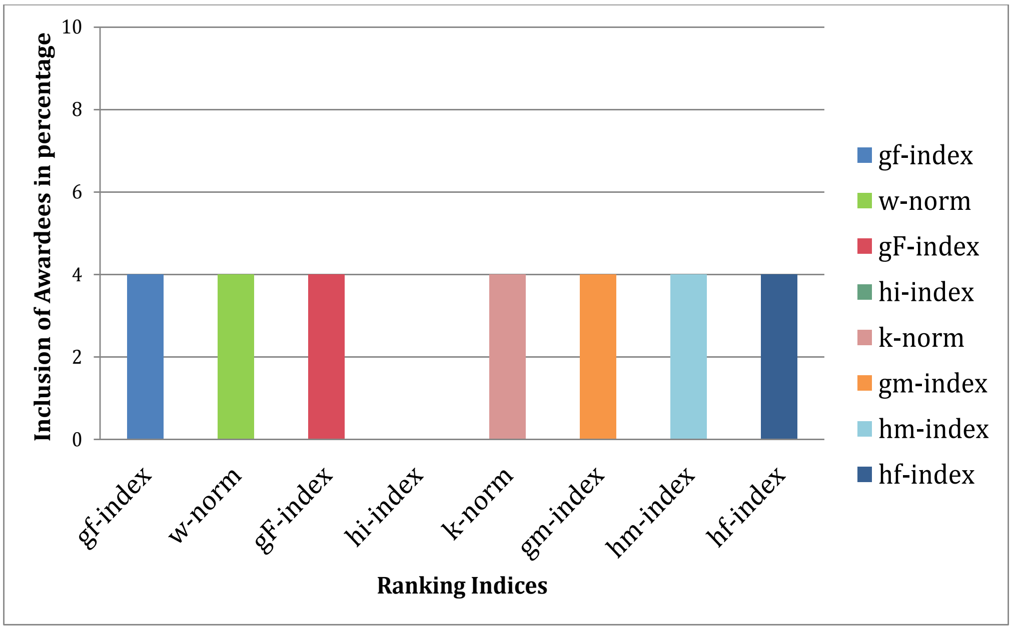

FIGURE 4.5: Percentage of occurrence of awardees in 31-40% of the ranked list

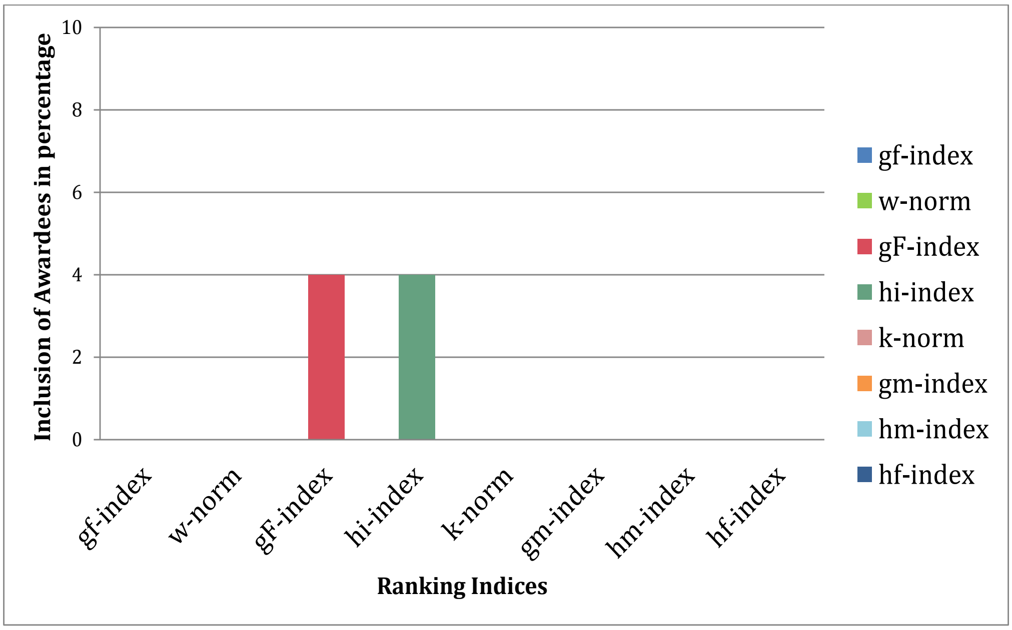

FIGURE 4.6: Percentage of occurrence of awardees in 41-50% of the ranked list

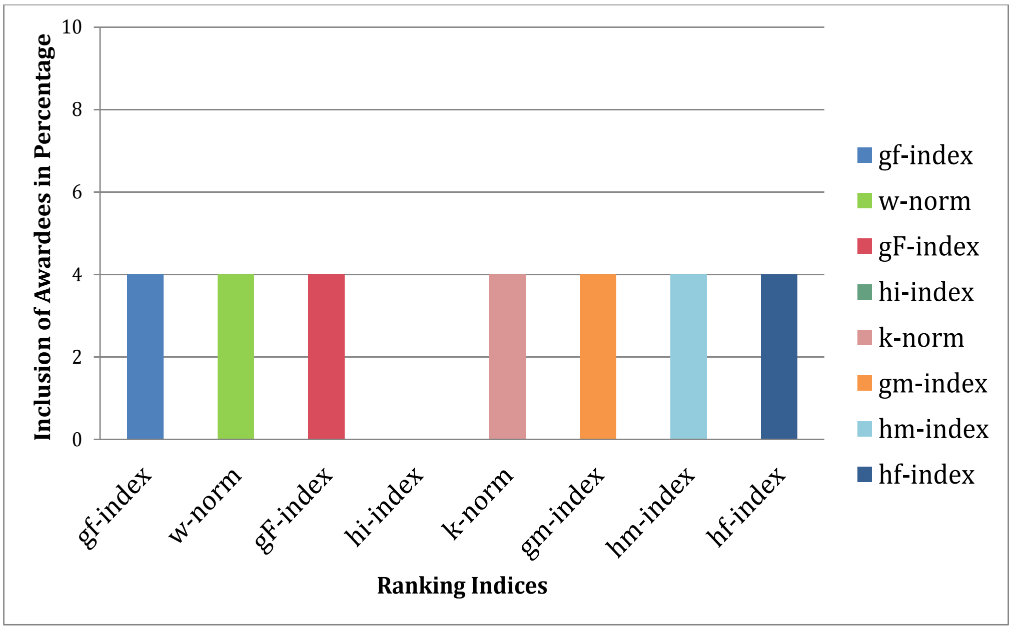

FIGURE 4.7: Percentage of occurrence of awardees in 51-60% of ranked list

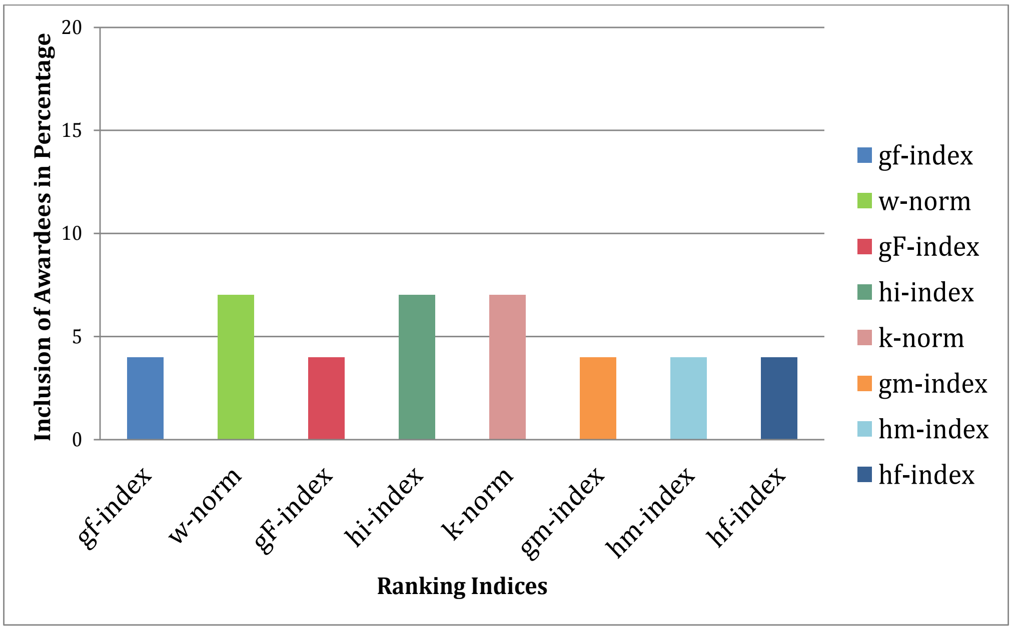

FIGURE 4.8: Percentage of occurrence of awardees in 61-70% of ranked list

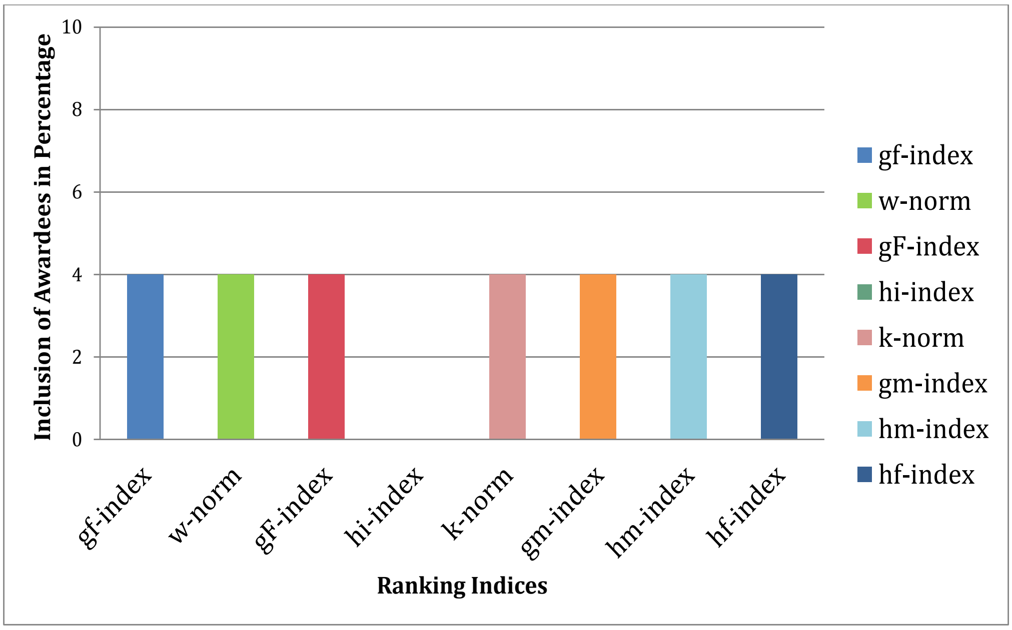

FIGURE 4.9: Percentage of occurrence of awardees in 71-80% of ranked list

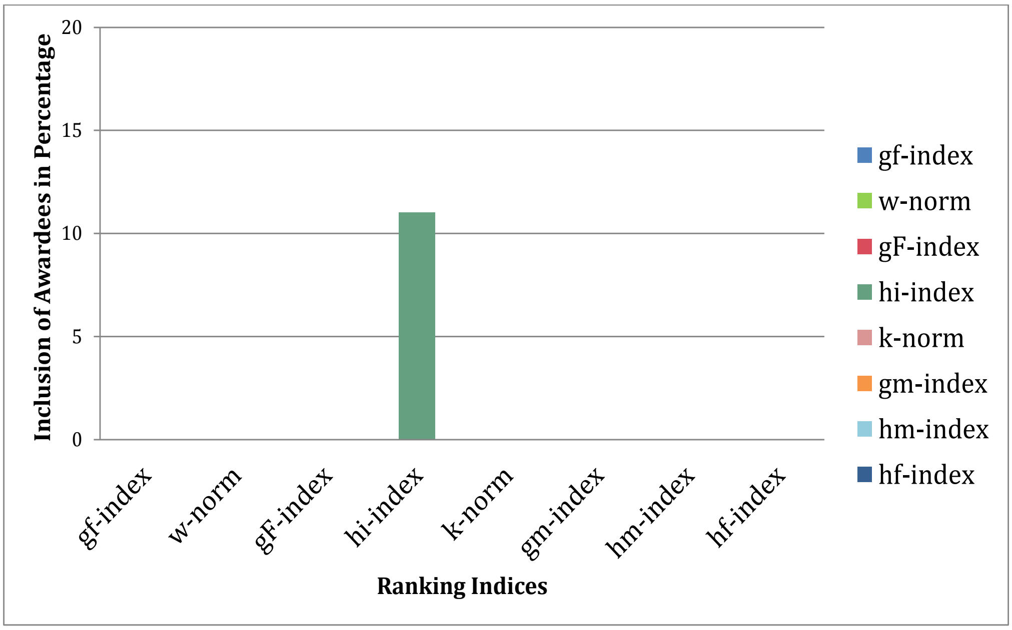

FIGURE 4.10: Percentage of occurrence of awardees in 81-90% of ranked list

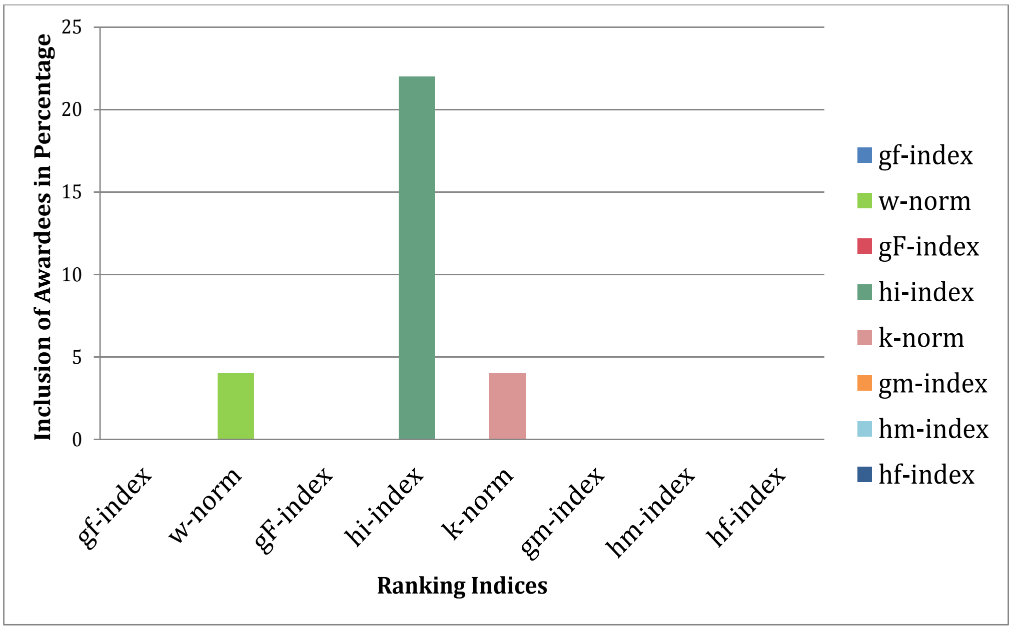

FIGURE 4.11: Percentage of occurrence of awardees in 91-100% of ranked list

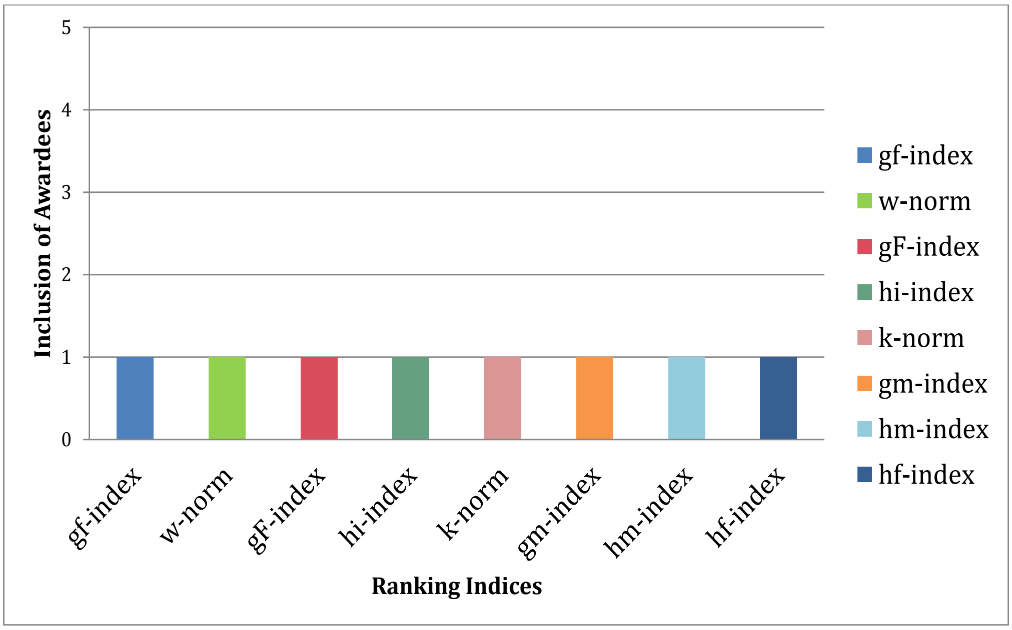

FIGURE 4.12: Occurrence of awardees in top 100 of the ranked list

4.2.1 Occurrence of Awardees in Top Ranked Authors

The results of the occurrence of awardees in top 100 of the list are shown in Figure 4.12. The Figure 4.12 illustrates that the performance of all the indices remained equal, which is 1 awardee in the top. The result of the inclusion of awardees in top 500 of the ranked list is shown in Figure 4.13. In top 500 of the ranked list, g_f-index remained successful in bringing most of the awardees, which is 7 awardees. The h_f-index and h_m-index have brought 6 awardees, k-norm, w-norm and h_i-index have brought 5 awardees while g_F-index and g_m-index have brought 4 awardees in the top 500 results.

The result of the occurrence of awardees in top 1000 of the list is shown in Figure 4.14. In top 1000 of the ranked list, g_f-index, w-norm and k-norm have brought the maximum number of awardees that is 10 awardees, h_m-index and h_f-index have brought 9 awardees while g_F-index, g_m-index and h_i-index have brought 8 awardees.

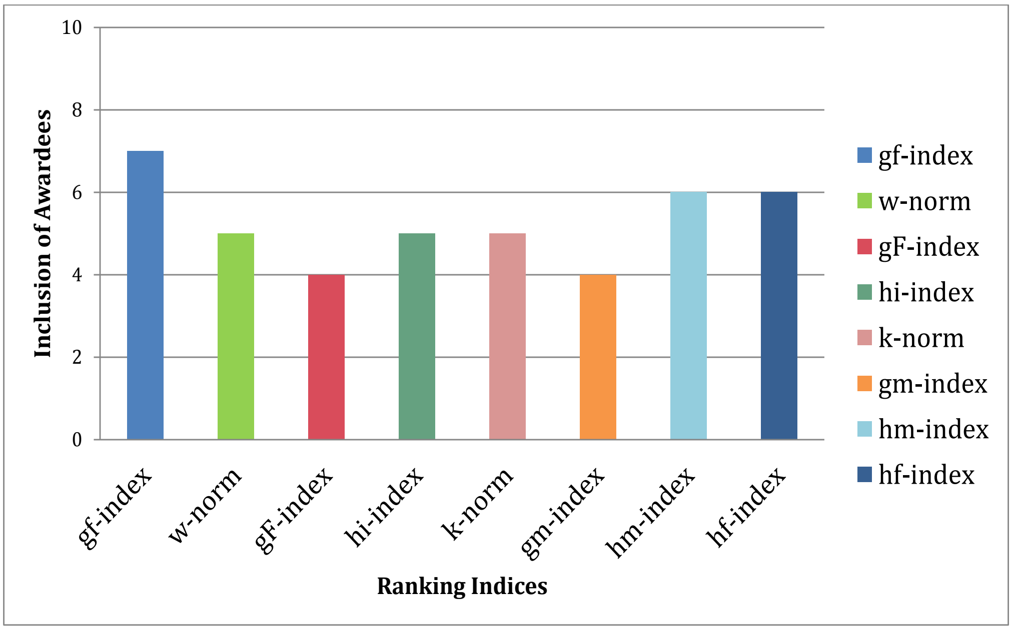

FIGURE 4.13: Occurrence of awardees in top 500 of the ranked list

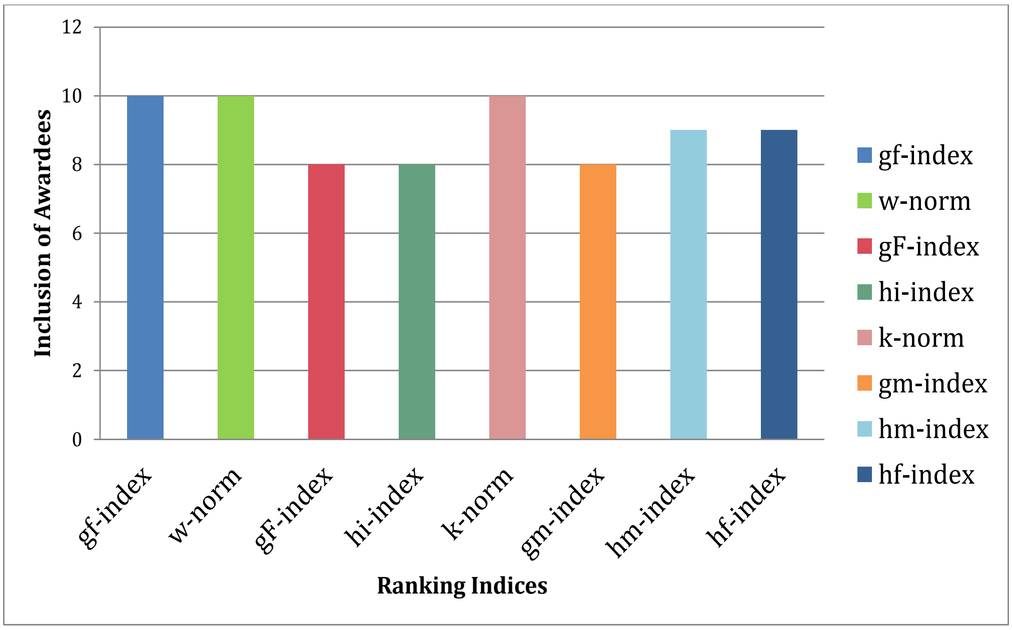

FIGURE 4.14: Occurrence of awardees in top 1000 of the ranked list

4.2.2 Index that Maps Better with International Awardees

The indices that remained successful in mapping most of the awardees in top 10% of the ranked list are g_f-index followed by g_m-index. These indices came up with 67% of awardees in top of the ranked list. Then the comparison is made in top of the ranked list in order to distinguish between these two indices. Thus the comparison is first made in top 100 of the ranked list. It is seen that both of these indices came across with 1 awardee in top 100, in top 500 g_f-index came up with 7 awardees and g_m-index came up with 4 awardees and in top 1000 of the ranked list g_f-index came up with 10 awardees while g_m-index came up with 8 awardees. So overall observation indicates that g_f-index is the index that maps better with the data of international awardees.

4.2.3 The Results of Occurrence of Awardees in Baseline Paper

The results of occurrence of awardees in top 10% of the baseline paper are illustrated in Figure 4.15. The Figure 4.15 illustrates the results of inclusion of awardees in top 10% of the ranked lists obtained by 11 indices based on citation intensity and age of publication. The Results indicate that the maximum inclusion of awardees is around 47% which is found in the top 10% list ranked by f-index followed by t-index and Wu-index. The indices that give more attention towards highly cited papers perform better i.e. Wu-index. Overall, the performance of indices i.e. contemporary h-index and tepered-index is around 45% while h-index brings 45% awardees. Rest of the indices show low values as the raw h-rate came up with 30% awardees while in multi-authorship based ranked lists of indices the maximum of 67% of awardees are found in top 10% of the ranked lists. The indices that remained successful in bringing 67% of awardees in the top ranks are g_f-index followed by g_m-index. Comparatively better results of these indices could be due to the fact that these indices handle the number of co-authors in more appropriate way [15]. The performance of g_F-index, h_m-index and h_f-index is around 63% in bringing awardees at the top

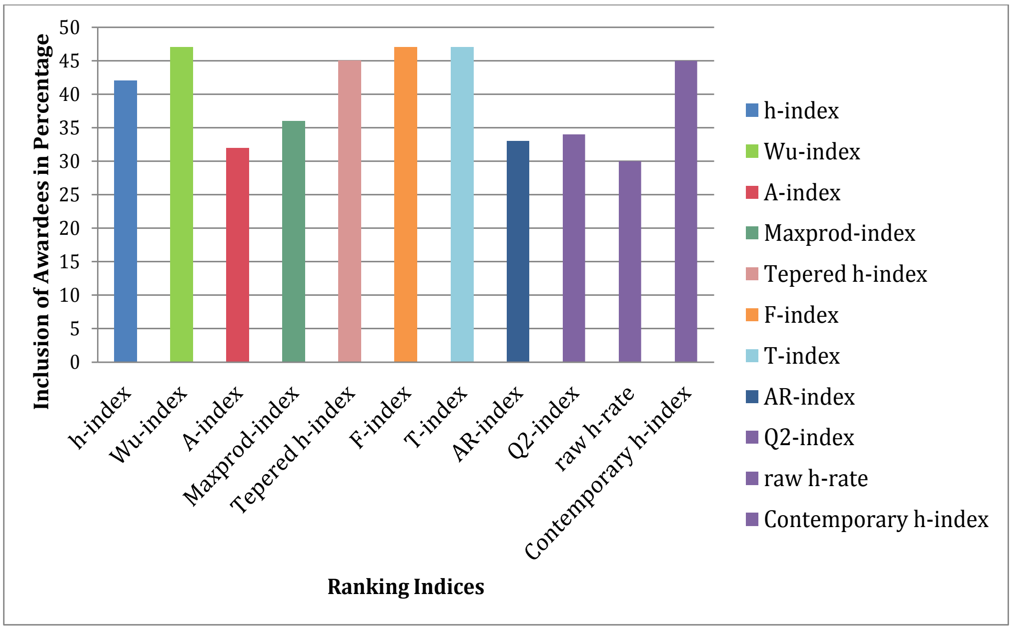

FIGURE 4.15: Percentage of occurrence of awardees in baseline paper

ranks while k-norm and w-norm came up with around 59% awardees. The least number of awardees (Around 48%) are brought by h_i-index.

Chapter 5

Conclusion and Future Work

Ranking of authors in the scientific community can facilitate in various aspects. It can assist in the research based decisions i.e. who should be employed, promoted, nominated for awards and scholarships? Furthermore, the ranking may assist the student community to find the most suitable supervisor for there thesis supervision, it may help the organizers of a conference or journal to hire a suitable reviewer for the evaluation of a paper, A university may select an employ from the list of candidates based on their research performance. Therefore, it can be stated that the ranking of authors play a significant in carrying out the decisions based on research.

For the ranking of authors there are parameter that cover various aspects of the scientific research i.e. publication count, citation count, h-index and the variants of h-index. We have performed analysis of h-index variants that consider multi-authorship in the research. As the multi-authorship trend is growing and the collaborations are increasing day by day, there is need to identify the most contributing multi-authorship index for the ranking of authors. We have performed analysis of these multi-authorship indices that take into account the number of co-authors in the research, there calculations on the dataset and comparisons of their results with other multi-authorship indices. We found that most of the indices are assessed on very small datasets making it challenging to identify the actual performance of indices. Furthermore, the indices are assessed on the datasets of different domains, as a consquence of which, the comparison of indices and identification of most contributing index is difficult.

In this study, we investigated the role of author ranking indices that consider multi-authorship in research, on a comprehensive dataset from single domain. The assessment of h_m-index, g_m-index, h_i-index, h_f-index, g_f-index, w-norm, k-norm and g_F-index is performed on a comprehensive dataset from the domain of Civil Engineering. The results obtained from these indices are further investigated to find the correlation between the ranked lists obtained by these indices. Furthermore, the occurrence of awardees is checked in the ranked list of each index for the determination of most contributing multi-authorship index in the ranking of authors. The data set of international award winners in Civil Engineering is considered for comparison.

To address the first research question, Spearman Rank correlation is found between the ranked lists of these multi-authorship indices. It is observed that the multi-authorship indices have strong, very strong or moderate correlation among each other. The h_f-index shows strong correlation with 4 indices while very strong correlation with 3 indices. H_i-index shows moderate correlation with 6 indices and strong correlation with 1 index. Furthermore, g_m-index shows strong correlation with 4 indices, very strong correlation with 2 and moderate in case of h_i-index. Overall, it is observed that the cases of strong or very strong correlations prevail the moderate correlations and none of the multi-authorship index has shown weak or negative correlation with other multi-authorship indices. These findings assist us to uncover the second research question i.e. to identify the most contributing index among these multi-authorship indices.

To adress the second research question, the occurrence of awardees is checked in the ranked list obtained from each multi-authorship index. In top 10% of the ranked list, g_f-index and g_m-index remained successful in bringing most of the awardees i.e. around 67% of total awardees. G_F-index, h_m-index and h_f-index have brought 63% awardees while k-norm and w-norm have brought 59% awardees in top of the ranked list. Overall, none of the index remained successful in bringing 100% of awardees in top of the ranking list. The least percentage of awardees is brought by h_i-index which is 48% of total awardees. Hence, the most contributing indices are g_f-index and g_m-index in bringing most of the awardees in top of the ranked list.

The indices used in this study play a vital role in assessing the quality of research considering multi-authorship. Apart from these indices, there are indices that also consider authorship position in multi-authorship. In future, those multi-authorship indices can be evaluated on the comprehensive data sets from different domains.

Bibliography

[1] D. C. C. Christopher R. Carpenter and C. C. Sarli, "Using publication metrics to highlight academic productivity and research impact," *Acad Emerg Med*, vol. 21, no. 10, pp. 1160–1172, 2014.

[2] D. Schoonbaert and G. Roelants, "Citation analysis for measuring the value of scientific publications: Quality assessment tool or comedy of errors?" *Blackwell Science Ltd*, vol. 1, no. 6, pp. 739–752, 1996.

[3] J. Hirsh, "An index to quantify an individuals scientific research output," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 46, pp. 16 569–16 572, 2005.

[4] L. Egghe, "An improvement of the h-index: The g-index," *ISSI Newsletter*, vol. 2, no. 1, pp. 8–9, 2005.

[5] B. Jin, "h-index: An evaluation indicator proposed by scientist," *Science Focus*, vol. 1, no. 1, pp. 8–9, 2006.

[6] Jin, "Ar-index: Complementing the h index," *ISSI Newsletter*, vol. 3, no. 1, p. 6, 2007.

[7] M. Kosmulski, "A new hirsch-type index saves time and works equally well as the original h-index," *International Society for Scientometrics and Informetrics*, vol. 2, no. 3, pp. 5–6, 2006.

[8] D. K. Antonis Sidiropoulos and Y. Manolopoulos, "Generalized hirsch h-index for disclosing latent facts in citation networks," *Scientometrics*, vol. 72, no. 2, pp. 253–280, 2007.

[9] R. S. J. Tol, "The h-index and its alternatives: An application to the 100 most prolific economists," *Scientometrics*, vol. 80, no. 2, pp. 317–324, 2009.

[10] E. H.-V. Sergio Alonso, Francisco Javier Cabrerizo and F. Herrera, "hg-index: A new index to characterize the scientic output of researchers based on the h- and g- indices," *Scientometrics*, vol. 82, no. 2, pp. 391–400, 2009.

[11] L. Egghe, "Mathematical theory of the h-and g-index in case of fractional counting of authorship," *Journal of the American Society for Information Science and Technology*, vol. 59, no. 10, pp. 1608–1616, 2008.

[12] R. M. Lutz Bornmann, "A multilevel meta-analysis of studies reporting correlations between the h index and 37 different h index variants," *Journal of Informetrics*, vol. 5, no. 3, p. 346359, 2011.

[13] O. K. Pablo D. Batista, Mnica G. Campiteli and A. S. Martinez, "Is it possible to compare researchers with different scientific interests?" *Scientometrics*, vol. 68, no. 1, pp. 179–189, 2006.

[14] M. Schreiber, "A modification of the h-index: The hm -index accounts for multi-authored manuscripts." *Journal of the American Society for Information Science and Technology*, vol. 60, no. 10, pp. 2145–2150, 2009.

[15] Schreiber, "Fractionalized counting of publications for the g-index," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 10, pp. 2145–2150, 2009.

[16] R. F., "The development of medlars," *Bull Med Libr Assoc*, vol. 52, pp. 150–151, 1964.

[17] F. Narin, "Evaluative bibliometrics: The use of publication and citation analysis in the evaluation of scientific activity," *Cherry Hill, N.J. : Computer Horizon*, 1976.

[18] I. J. Fanelli D, "Us studies may overestimate effect sizes in softer research," *ProcNatlAcadSci USA*, vol. 110, pp. 15 031–15 036, 2013.

[19] G. P. Durieux V, "Bibliometric indicators: quality measurements of scientific publication," *Radiology*, vol. 255, p. 342351, 2010.

[20] K. D. Rezek I, McDonald RJ, "Pre-residency publication rate strongly predicts future academic radiology potential," *AcadRadiol*, vol. 19, p. 632634, 2012.

[21] B. SB, "Authorship issues," *Lung India*, vol. 29, no. 632-634, 2012.

[22] F. WB, "Medical authorship: traditions, trends, and tribulations," *Ann Intern Med*, vol. 113, p. 317 325, 1990.

[23] M. B. Kumar V, Upadhyay S, "Impact of the impact factor in biomedical research: its use and misuse," *Singapore Med J*, vol. 50, p. 752755, 2009.

[24] C. DC, "Measuring the measurable: a commentary on impact factor," *AcadEmerg Med*, vol. 19, no. 12971299, 2012.

[25] S. PO, "Why the impact factor of journals should not be used for evaluating research," *BMJ*, vol. 314, pp. 498–502, 1997.

[26] S. H. Petersen AM, Wang F, "Methods for measuring the citations and productivity of scientists across time and discipline.phys rev e stat nonlin soft matter phys," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 81, pp. 2–3, 2010.

[27] v. R. A. Costas R, van Leeuwen T, "The mendel syndrome in science: durability of scientific literature and its effects on bibliometric analysis of individual scientists," *Scientometrics*, vol. 89, pp. 177–205, 2011.

[28] P. TJ, "A compendium of issues for citation analysis," *Scientometrics*, vol. 45, p. 117136, 1999.

[29] C. Z, "An easy way to boost a papers citations," *Nature News*, vol. doi:10.1038/news.2010.406, 2010.

[30] S. B, "The h-index outperforms other bibliometrics in the assessment of research performance in general surgery: a province-wide study," *Surgery*, vol. 153, p. 493501, 2013.

[31] J. E. Hirsch, "An index to quantify an individual's scientific research output," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 46, pp. 16 569–16 572, 2005.

[32] R. R. Jin BiHui, Liang LiMing and L. Egghe, "The r- and ar-indices: Complementing the h-index," *Chinese Science Bulletin*, vol. 52, no. 6, pp. 855–863, 2007.

[33] T. A. M. Raheel, S. Ayaz, "Evaluation of h-index, its variants and extensions based on publication age and citation intensity in civil engineering," *Scientometrics*, vol. 114, no. 3, pp. 1107–1127, 2018.

[34] G. Anania and A. Caruso, "Two simple new bibliometric indexes to better evaluate research in disciplines where publications typically receive less citations," *Scientometrics*, vol. 96, no. 2, pp. 617–631, 2013.

[35] M. A. S. N. N. Tehmina Amjad, Ishfaq Ahmad, "Variation in citation based fractional counting of authorship," *J. Engg. and Appl. Sci.*, vol. 35, no. 2, pp. 2518–4571, 2016.

[36] A. R. Lech Czarnecki, M.P. Kazmierkowski, "Doing hirsch proud; shaping h-index in engineering sciences," *BULLETIN OF THE POLISH ACADEMY OF SCIENCES TECHNICAL SCIENCES*, vol. 61, no. 1, 2013.

[37] M. G. P. G. Falagas ME, Pitsouni EI, "Comparison of pubmed, scopus, web of science, and google scholar: strengths and weaknesses," *The FASEB Journal*, vol. 22, no. 2, p. 338342, 2008.

[38] G. H. Henk F.Moed, Judit Bar-Ilan, "A new methodology for comparing google scholar and scopus," *Journal of Informetrics*, vol. 10, no. 2, p. 533551, 2016.

[39] A.-W. Harzing, "A longitudinal study of google scholar coverage between 2012 and 2013," *Scientometrics*, vol. 98, no. 1, p. 565575, 2014.

[40] B. R. K., "Scientific impact quantity and quality: Analysis of two sources of bibliographic data." 2005.

[41] K. Y. Lokman I. Meho, "Impact of data sources on citation counts and rankings of lis faculty: Web of science versus scopus and google scholar," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 13, p. 21052125, 2007.

[42] D. D. De Winter J. C., Zadpoor A. A., "The expansion of google scholar versus web of science: A longitudinal study," *Scientometrics*, vol. 98, no. 2, p. 15471565, 2014.

[43] N. A., "Google scholar: The new generation of citation indexes." *Libri*, vol. 55, no. 4, pp. 170–180, 2005.

[44] J. Henderson, "Google scholar: A source for clinicians?" *Canadian Medical Association Journal*, vol. 172, no. 12, pp. 1549–1550, 2005.

[45] E. W. Bela Gipp, Joeran Beel, "Academic search engine optimization (aseo): Optimizing scholarly literature for google scholar co," *Journal of Scholarly Publishing*, vol. 41, no. 2, pp. 176–190, 2009.

[46] M. L. I. K. Hongbo Deng, Jiawei Han, "Modeling and exploiting heterogeneous bibliographic networks for expertise ranking," *In Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 71–80, 2012.

[47] P. Jacso, "The pros and cons of computing the h-index using google scholar," *Online Information Review*, vol. 32, no. 3, pp. 437–452, 2008.

[48] M. T. A. Samreen Ayaz, "Identification of conversion factor for completing-h index for the field of mathematics." *Scientometrics*, vol. 109, no. 3, pp. 1511–1524, 2016.

[49] B. W. Daniela Rosenstreich, "Measuring the impact of accounting journals using google scholar and the g-index," *The British Accounting Review*, vol. 41, no. 4, p. 227239, 2009.

[50] I. F. Aguillo, "Is google scholar useful for bibliometrics? a webometric analysis," *Scientometrics*, vol. 91, no. 2, p. 343351, 2011.

[51] F. D. Corder GW, "Comparing variables of ordinal or dichotomous scales: Spearman rank-order, point-biserial, and biserial correlations," p. 122154, 2009.

[52] J. Evans, "Straightforward statistics for the behavioral sciences," *Pacific Grove: Brooks/Cole publishing*, 1996.

Appendix A

Indices	Calculation formulas
h$_m$-index	$r_{eff}(r) = \sum_{r'=1}^{r} \frac{1}{a(r')}$ then $c(r(h_m)) \geq h_m \geq c(r(h_m) + 1)$
g$_m$-index	g$_m \leq$ c$_{eff}(g_m)$ Where $c_{eff}(r_{eff}) = 1\frac{}{r_{eff}}s_{eff}(r_{eff})$ and $s_{eff}(r_{eff}) = \sum_{r'=1}^{r(r_{eff})} \frac{1}{a(r')} c(r')$
h$_f$-index	$\frac{Yh_f}{\Phi(Yh_f)} \geq h_f$ Where y(i) = citation counts, $\Phi(i)$ = Number of authors of paper
g$_f$-index	$\sum_{i=1}^{gf} \frac{Y_i}{\Phi(i)} \geq g_{f^2}$ Where y(i) = citation counts, $\Phi(i)$ = Number of authors
w-norm	W-norm = h-norm + (1-(h-norm)$^2/totcit-norm), \forall h-norm > 0$ $and\ W-norm = totcit-norm/(1+totcit-norm)\ if\ h-norm = 0$
k-norm	K-norm = h-norm + (1-(h-norm)$^2 / \sum_{j=1,2,...,h-norm} citnorm_j, \forall h-norm > 1\ and\ K-norm = 0\ if\ h-norm = 0$
g$_F$-index	$(\sum_{i=1}^{k} \frac{1}{\Phi(i)})^2 \leq \sum_{i=1}^{k} y_i$ Where $g_F = \sum_{i=1}^{k} \frac{1}{\Phi(i)}$
hi-index	$h_I = \frac{h^2}{N_a^{(T)}}$ Where h is h-index and $N_a^{(T)}$ is the authors in h papers

Appendix B

TABLE B.1: Award winners of ASCE 2016

First Name	Last Name	Award
Ronaldo I.	Borja	Maurice A. Biot Medal
Qiu S.	Li	Jack E. Cermak Medal
Kei	Ishida	J. James R. Croes Medal
M. Levent	Kavvas	J. James R. Croes Medal
Su-Hyung	Jang	J. James R. Croes Medal
Zhiqiang R.	Chen	J. James R. Croes Medal
Noriaki	Ohara	J. James R. Croes Medal
Michael L.	Anderson	J. James R. Croes Medal
Rene A.	Camacho-Rincon	Samuel Arnold Greeley Award
James L.	Martin	Samuel Arnold Greeley Award
Brian	Watson	Samuel Arnold Greeley Award
Michael J.	Paul	Samuel Arnold Greeley Award
Lei	Zheng	Samuel Arnold Greeley Award
James B.	Stribling	Samuel Arnold Greeley Award

Continued on next page

Table B.1 – continued from previous page

First Name	Last Name	Award
Meng	Hu	Rudolph Hering Medal
Tian C.	Zhang	Rudolph Hering Medal
John	Stansbury	Rudolph Hering Medal
You	Zhou	Rudolph Hering Medal
Han	Chen	Rudolph Hering Medal
Jill	Neal	Rudolph Hering Medal
Lindell	Ormsbee	Julian Hinds Award
Steven C.	Chapra	Wesley W. Horner Award
Rasika K.	Gawde	Wesley W. Horner Award
Martin T.	Auer	Wesley W. Horner Award
Rakesh K.	Gelda	Wesley W. Horner Award
Noel R.	Urban	Wesley W. Horner Award
Seung H.	Hong	Karl E. Hilgard Prize
Terry W.	Sturrm	Karl E. Hilgard Prize
Thorsten	Stoesser	Karl E. Hilgard Prize
Alexandria B.	Boehm	Walter L. Huber Prize
Claudia K.	Gunsch	Walter L. Huber Prize
Amit	Kanvinde	Walter L. Huber Prize
John S.	Mccartney	Walter L. Huber Prize
Narayanan	Neithalath	Walter L. Huber Prize

Continued on next page

Table B.1 – continued from previous page

First Name	Last Name	Award
Daniel P.	Loscalzo	Daniel W. Mead Prize
Jean-Louis	Briaud	Thomas A. Middlebrooks Award
Matthew R.	Eatherton	Moisseiff Award
Xiang	Ma	Moisseiff Award
Helmut	Krawinkler	Moisseiff Award
Gregory G.	Deierlein	Moisseiff Award

Continued on next page

Table B.1 – continued from previous page

First Name	Last Name	Award
Jerome F.	Hajjar	Moisseiff Award
Teng	Wu	Alfred Noble Prize
Ahsan	Kareem	Alfred Noble Prize
Brett W.	Maurer	Norman Medal
Russell A.	Green	Norman Medal
Misko	Cubrinovski	Norman Medal
Brendon A.	Bradley	Norman Medal
Ross	Boulanger	Ralph B. Peck Award
George E.	Gibson	Peurifoy Construction Award
Douglas L.	Kane	Harold R. Peyton Award
Ronny	Purba	Raymond C. Reese Prize
Michel	Bruneau	Raymond C. Reese Prize
Vitaliy	Priven	Thomas Fitch Rowland Prize
Rafael	Sacks	Thomas Fitch Rowland Prize
M. Ataur	Rahman	T.Y. Lin Award
Sri	Sritharan	T.Y. Lin Award

TABLE B.2: Award winners of CSCE 2016

First Name	Last Name	Award

Continued on next page

Table B.2 – continued from previous page

First Name	**Last Name**	**Award**
Hadi	Ghofrani	Cazimir Gzowski Medal
Gail M.	Atkinson	Cazimir Gzowski Medal
Luc	Chouinard	Cazimir Gzowski Medal
Philippe	Rosset	Cazimir Gzowski Medal
Kristy F.	Tiampo	Cazimir Gzowski Medal
Steven	Daly	Thomas C. Keefer Medal
Brian	Morse	Thomas C. Keefer Medal
Richard	Martin	Thomas C. Keefer Medal
Samy M.	Reza	P.L. Pratley Award
M. Shahria	Alam	P.L. Pratley Award
Solomon	Tesfamariam	P.L. Pratley Award
Gregory	Courtice	Donald Stanley Award
Abul Basar M.	Baki	Donald Stanley Award
David Z.	Zhu	Donald Stanley Award
Christopher	Cahill	Donald Stanley Award
William M.	Tonn	Donald Stanley Award
Farnaz	Sadeghpour	Stephen G. Revay Award
Mohsen	Andayesh	Stephen G. Revay Award

Table B.3: Award winners of ACI 2016

First Name	Last Name	Award
Ahmed	Osman	Aci Construction Award
Whitney	Morris	Aci Construction Award
Ahmad M. El	Magdoub	Aci Construction Award
Roberto T.	Leon	Aci Design Award
Weng Y.	Kam	Aci Design Award
Stefano	Pampanin	Aci Design Award
H. S.	Lew	Mete A. Sozen Award
Yihai	Bao	Mete A. Sozen Award
Santiago	Pujol	Mete A. Sozen Award
Mete A.	Sozen	Mete A. Sozen Award
Hugh H.	Wang	Wason Medal For Materials Research
Delia D.	Guajardo	Wason Medal For Materials Research
Hamid	Farzam	Wason Medal For Materials Research
Rmy D.	Lequesne	Wason Medal For Most Meritorious Paper
Jos A.	Pincheira	Wason Medal For Most Meritorious Paper

Table B.4: Award winners of ICE 2016

First Name	Last Name	Award

Continued on next page

Table B.4 – continued from previous page

First Name	Last Name	Award
Alba	Yerro	Telford Medal
Eduardo E.	Alonso	Telford Medal
Nuria M.	Pinyol	Telford Medal
Peter G.	Brewer	James Alfred Ewing Medal