Online ISSN : 0975-4172
Print ISSN : 0975-4350

# Global Journal of Computer Science and Technology : C

## Software and Data Engineering

DISCOVERING THOUGHTS AND INVENTING FUTURE

HIGHLIGHTS

Instructive of Ooze Information

A Simulation Based Approach

Online Attendance System

Based on Clustering Technique

Datacentre

Volume 12 | Issue 13 | Version 1.0 ENG

Global Journal of Computer Science and Technology: C
Software & Data Engineering

Global Journal of Computer Science and Technology: C
Software & Data Engineering

Volume 12 Issue 13 (Ver. 1.0)

Open Association of Research Society

**Incorporation No.: 0423089**
**License No.: 42125/022010/1186**
**Registration No.: 430374**
**Import-Export Code: 1109007027**
**Employer Identification Number (EIN):**
**USA Tax ID: 98-0673427**

Global Journals Inc.

*(A Delaware USA Incorporation with "Good Standing"; Reg. Number: 0423089)*

*Sponsors:Global Association of Research*

*Open Scientific Standards*

*Publisher's Headquarters office*

Global Journals Inc., Headquarters Corporate Office, Cambridge Office Center, II Canal Park, Floor No. 5th, ***Cambridge (Massachusetts)***, Pin: MA 02141 United States

*USA Toll Free: +001-888-839-7392*

*USA Toll Free Fax: +001-888-839-7392*

*Offset Typesetting*

Global Association of Research, Marsh Road, Rainham, Essex, London RM13 8EU United Kingdom.

*Packaging & Continental Dispatching*

Global Journals, India

*Find a correspondence nodal officer near you*

To find nodal officer of your country, please email us at *local@globaljournals.org*

*eContacts*

Press Inquiries: *press@globaljournals.org*
Investor Inquiries: *investers@globaljournals.org*
Technical Support: *technology@globaljournals.org*
Media & Releases: *media@globaljournals.org*

*Pricing (Including by Air Parcel Charges):*

*For Authors:*
22 USD (B/W) & 50 USD (Color)
*Yearly Subscription (Personal & Institutional):*
200 USD (B/W) & 250 USD (Color)

# Editorial Board Members (HON.)

**Dr. Bart Lambrecht**
Director of Research in Accounting and FinanceProfessor of Finance
Lancaster University Management School
BA (Antwerp); MPhil, MA, PhD (Cambridge)

**Dr. Carlos García Pont**
Associate Professor of Marketing
IESE Business School, University of Navarra
Doctor of Philosophy (Management), Massachusetts Institute of Technology (MIT)
Master in Business Administration, IESE, University of Navarra
Degree in Industrial Engineering, Universitat Politècnica de Catalunya

**Dr. Fotini Labropulu**
Mathematics - Luther College
University of ReginaPh.D., M.Sc. in Mathematics
B.A. (Honors) in Mathematics
University of Windso

**Dr. Lynn Lim**
Reader in Business and Marketing
Roehampton University, London
BCom, PGDip, MBA (Distinction), PhD, FHEA

**Dr. Mihaly Mezei**
ASSOCIATE PROFESSOR
Department of Structural and Chemical Biology, Mount Sinai School of Medical Center
Ph.D., Etvs Lornd University
Postdoctoral Training,
New York University

**Dr. Söhnke M. Bartram**
Department of Accounting and FinanceLancaster University Management SchoolPh.D. (WHU Koblenz)
MBA/BBA (University of Saarbrücken)

**Dr. Miguel Angel Ariño**
Professor of Decision Sciences
IESE Business School
Barcelona, Spain (Universidad de Navarra)
CEIBS (China Europe International Business School).
Beijing, Shanghai and Shenzhen
Ph.D. in Mathematics
University of Barcelona
BA in Mathematics (Licenciatura)
University of Barcelona

**Philip G. Moscoso**
Technology and Operations Management
IESE Business School, University of Navarra
Ph.D in Industrial Engineering and Management, ETH Zurich
M.Sc. in Chemical Engineering, ETH Zurich

**Dr. Sanjay Dixit, M.D.**
Director, EP Laboratories, Philadelphia VA Medical Center
Cardiovascular Medicine - Cardiac Arrhythmia
Univ of Penn School of Medicine

**Dr. Han-Xiang Deng**
MD., Ph.D
Associate Professor and Research
Department Division of Neuromuscular Medicine
Davee Department of Neurology and Clinical NeuroscienceNorthwestern University
Feinberg School of Medicine

**Dr. Pina C. Sanelli**
Associate Professor of Public Health
Weill Cornell Medical College
Associate Attending Radiologist
NewYork-Presbyterian Hospital
MRI, MRA, CT, and CTA
Neuroradiology and Diagnostic Radiology
M.D., State University of New York at Buffalo,School of Medicine and Biomedical Sciences

**Dr. Roberto Sanchez**
Associate Professor
Department of Structural and Chemical Biology
Mount Sinai School of Medicine
Ph.D., The Rockefeller University

**Dr. Wen-Yih Sun**
Professor of Earth and Atmospheric SciencesPurdue University Director
National Center for Typhoon and Flooding Research, Taiwan
University Chair Professor
Department of Atmospheric Sciences, National Central University, Chung-Li, TaiwanUniversity Chair Professor
Institute of Environmental Engineering, National Chiao Tung University, Hsin-chu, Taiwan.Ph.D., MS The University of Chicago, Geophysical Sciences
BS National Taiwan University, Atmospheric Sciences
Associate Professor of Radiology

**Dr. Michael R. Rudnick**
M.D., FACP
Associate Professor of Medicine
Chief, Renal Electrolyte and Hypertension Division (PMC)
Penn Medicine, University of Pennsylvania
Presbyterian Medical Center, Philadelphia
Nephrology and Internal Medicine
Certified by the American Board of Internal Medicine

**Dr. Bassey Benjamin Esu**
B.Sc. Marketing; MBA Marketing; Ph.D Marketing
Lecturer, Department of Marketing, University of Calabar
Tourism Consultant, Cross River State Tourism Development Department
Co-ordinator , Sustainable Tourism Initiative, Calabar, Nigeria

**Dr. Aziz M. Barbar, Ph.D**.
IEEE Senior Member
Chairperson, Department of Computer Science
AUST - American University of Science & Technology
Alfred Naccash Avenue – Ashrafieh

# Contents of the Volume

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# Instructive of Ooze Information

By K. Sudheer Kumar & Ch.S.V.V.S.N.Murthy
*Sri Sai Aditya Institute of Science And Technology, Surampalem, Kakinada, Andhra Pradesh, India*

*Abstract -* We study the following problem: A data distributor has given sensitive data to a set of supposedly trusted agents (third parties). Some of the data are leaked and bring into being in an unconstitutional place (e.g., on the web or somebody's laptop). The distributor must evaluate the likelihood that the leaked data came from one or more agents, as opposed to having been independently gathered by other means. We propose data distribution strategies (across the agents) that improve the likelihood of identifying leakages. These methods do not rely on alterations of the released data (e.g., watermarks). In some cases, we can also inject "realistic but replica" data records to further improve our chances of detecting leakage and identifying the guilty party.

In the course of doing business, sometimes sensitive data must be handed over to supposedly trusted third parties. For example, a hospital may give patient records to Researchers who will devise new treatments. Similarly, a company may have partnerships with other companies that require sharing customer data. Another enterprise may outsource its data processing, so data must be given to various other companies. There always remains a risk of data getting leaked from the agent.

Perturbation is a very valuable technique where the data are modified and made "less sensitive" before being handed to agents. For example, one can add random noise to certain attributes, or one can replace exact values by ranges. But this technique requires modification of data. Leakage detection is handled by watermarking, e.g., a unique code is implanted in each distributed copy. If that copy is later discovered in the hands of an unconstitutional party, the leaker can be identified. But again it requires code modification. Watermarks can sometimes be destroyed if the data recipient is malicious.

*Keywords :* *Allocation strategies, data leakage, data privacy, fake records, leakage model.*

*GJCST-C Classification:* *E.0*


*Strictly as per the compliance and regulations of:*

# Instructive of Ooze Information

K. Sudheer Kumar [α] & Ch. S.V.V.S.N.Murthy [σ]

*Abstract -* We study the following problem: A data distributor has given sensitive data to a set of supposedly trusted agents (third parties). Some of the data are leaked and bring into being in an unconstitutional place (e.g., on the web or somebody's laptop). The distributor must evaluate the likelihood that the leaked data came from one or more agents, as opposed to having been independently gathered by other means. We propose data distribution strategies (across the agents) that improve the likelihood of identifying leakages. These methods do not rely on alterations of the released data (e.g., watermarks). In some cases, we can also inject "realistic but replica" data records to further improve our chances of detecting leakage and identifying the guilty party.

In the course of doing business, sometimes sensitive data must be handed over to supposedly trusted third parties. For example, a hospital may give patient records to Researchers who will devise new treatments. Similarly, a company may have partnerships with other companies that require sharing customer data. Another enterprise may outsource its data processing, so data must be given to various other companies. There always remains a risk of data getting leaked from the agent.

Perturbation is a very valuable technique where the data are modified and made "less sensitive" before being handed to agents. For example, one can add random noise to certain attributes, or one can replace exact values by ranges. But this technique requires modification of data. Leakage detection is handled by watermarking, e.g., a unique code is implanted in each distributed copy. If that copy is later discovered in the hands of an unconstitutional party, the leaker can be identified. But again it requires code modification. Watermarks can sometimes be destroyed if the data recipient is malicious.

*Keywords : Allocation strategies, data leakage, data privacy, fake records, leakage model.*

## I. Introduction

In the course of doing business, sometimes sensitive data must be handed over to supposedly trusted third parties. For example, a hospital may give patient records to researchers who will devise new treatments. Similarly, a company may have partnerships with other companies that require sharing customer data. Another enterprise may outsource its data processing, so data must be given to various other companies. We call the owner of the data the distributor and the supposedly trusted third parties the agents. Our goal is to detect when the distributor's sensitive data has been leaked by agents, and if possible to identify the agent that leaked the data.

We consider applications where the original sensitive data cannot be perturbed. Perturbation is a very useful technique where the data is modified and made "less sensitive" before being handed to agents. For example, one can add random noise to certain attributes, or one can replace exact values by ranges [18]. However, in some cases it is important not to alter the original distributor's data. For example, if an outsourcer is doing our payroll, he must have the exact salary and customer bank account numbers. If medical researchers will be treating patients (as opposed to simply computing statistics), they may need accurate data for the patients.

Traditionally, leakage detection is handled by watermarking, e.g., a unique code is embedded in each distributed copy. If that copy is later discovered in the hands of an unauthorized party, the leaker can be identified. Watermarks can be very useful in some cases, but again, involve some modification of the original data. Furthermore, watermarks can sometimes be destroyed if the data recipient is malicious. In this paper we study unobtrusive (Not attracting unnecessary attention) techniques for detecting leakage of a set of objects or records. Specifically, we study the following scenario: After giving a set of objects to agents, the distributor discovers some of those same objects in an unauthorized place. (For example, the data may be found on a web site, or may be obtained through a legal discovery process).

At this point the distributor can assess the likelihood that the leaked data came from one or more agents, as opposed to having been independently gathered by other means. If the distributor sees "enough evidence" that an agent leaked data, he may stop doing business with him, or may initiate legal proceedings.

In this paper we develop a model for assessing the "guilt" of agents. We also present algorithms for distributing Objects to agents, in a way that improves our chances of identifying a leaker. Finally, we also consider the option of adding "fake" objects to the distributed set. Such objects do not correspond to real entities but appear realistic to the agents. In a sense, the fake objects acts as a type of watermark for the entire set, without modifying any individual members. If it turns out an agent was given one or more fake objects that were leaked, then the distributor can be more confident that agent was guilty.

*Author α : Department of Computer Science & Engineering, Sri Sai Aditya Institute of Science And Technology, Surampalem, Kakinada, Andhra Pradesh, India. E-mail : sudheerkumarkotha@gmail.com*
*Author σ : Department of Information Technology, Sri Sai Aditya Institute of Science And Technology, Surampalem, Kakinada, Andhra Pradesh, India. E-mail : chsatyamurty@gmail.com*

## II. Problem Setup and Notation

Entities and Agents: A distributor owns a set T={t1, tm} of valuable data objects. The distributor wants to share some of the objects with a set of agents U1, U2, Un, but does not wish the objects be leaked to other third parties. The objects in T could be of any type and size, e.g., they could be tuples in a relation, or relations in a database. An agent Ui receives a subset of objects Ri ⊆ T, determined either by a sample request or an explicit request:

Sample request Ri = SAMPLE (T, mi): Any subset of Mi records from T can be given to Ui.

Explicit request Ri = EXPLICIT (T, condi): Agent Ui receives all the T objects that satisfy condi.

Type of data leakage: In order to implement the appropriate protective measures, we must first understand what we are protecting. Based on publicly disclosed Data Leakage breaches, the type of data leaked is broken down as follows:

Type of information leaked Percentage
Confidential information - 15%
Intellectual property - 4%
Customer data - 73%
Health records - 8%

Guilty Agents: Suppose that after giving objects to agents, the distributor discovers that a set S ⊆ T has leaked. This means that some third party called the target has been caught in possession of S. For example, this target may be displaying S on its web site, or perhaps as part of a legal discovery process, the target turned over S to the distributor.

Since the agents U1, Un has some of the data, it is reasonable to suspect them leaking the data. However, the agents can argue that they are innocent, and that the S data was obtained by the target through other means.

For example, say one of the objects in S represents a customer X. Perhaps X is also a customer of some other company, and that company provided the data to the target. Or perhaps X can be reconstructed from various publicly Available sources on the web.

Our goal is to estimate the likelihood that the leaked data came from the agents as opposed to other sources. Intuitively, the more data in S, the harder it is for the agents to argue they did not leak anything. Similarly, the "rarer" the objects, the harder it is to argue that the target obtained them through other means. For instance, if one of the S objects was only given to agent U1, while the other objects were given to all agents, we may suspect U1 more. The model we present next captures this intuition.

We say an agent Ui is guilty and if it contributes one or more objects to the target. We denote the event that agent Ui is guilty as Gi and the event that agent Ui is guilty for a given leaked set S as Gi|S. Our next step is to estimate Pr {Gi|S}, i.e., the probability that agent Ui is guilty given evidence S.

## III. Related Works

The guilt detection approach we present is related to the data provenance problem [3]: (whether it is genuine or not problem) tracing the lineage of S objects implies essentially the detection of the guilty agents. Tutorial [4] provides a good overview on the research conducted in this field. Suggested solutions are domain specific, such as lineage tracing for data warehouses [5], and assume some prior knowledge on the way a data view is created out of data sources.

Our problem formulation with objects and sets is more general and simplifies lineage tracing, since we do not consider any data transformation from Ri sets to S. As far as the data allocation strategies are concerned, our work is mostly relevant to watermarking that is used as a means of establishing original ownership of distributed objects. Watermarks were initially used in images [16], video [8] and audio data [6] whose digital representation includes considerable redundancy. Recently, [1], [17], [10], [7] and other works have also studied marks insertion to relational data. Our approach and watermarking are similar in the sense of providing agents with some kind of receiver-identifying information.

However, by its very nature, a watermark modifies the item being watermarked. If the object to be watermarked cannot be modified then a watermark cannot be inserted. In such cases methods that attach watermarks to the distributed data are not applicable. Finally, there are also lots of other works on mechanisms that allow only authorized users to access sensitive data through access control policies [9], [2]. Such approaches prevent in some sense data leakage by sharing information only with trusted parties. However, these policies are restrictive and may make it impossible to satisfy agents' requests.

## IV. Agent Guilt Model

To compute this Pr{Gi|S}, we need an estimate for the probability that values in S can be "guessed" by the target. For instance, say some of the objects in S are emails of individuals. We can conduct an experiment and ask a person with approximately the expertise and resources of the target to find the email of say 100 individuals. If this person can find say 90 emails, then we can reasonably guess that the probability of finding one email is 0.9. On the other hand, if the objects in question are bank account numbers, the person may only discover say 20, leading to an estimate of 0.2. Probability pt is analogous to the probabilities used in designing fault-tolerant systems. That is, to estimate how likely it is that a system will be operational throughout a given period, we need the probabilities that

individual components will or will not fail. A component failure in our case is the event that the target guesses an object of S. while we use the probability of guessing to identify agents that have leaked information.

The component failure probabilities are estimated based on experiments, just as we propose to estimate the pt's. Similarly, the component probabilities are usually conservative estimates, rather than exact numbers. For example, say we use a component failure probability that is higher than the actual probability, and we design our system to provide a desired high level of reliability. Then we will know that the actual system will have at least that level of reliability, but possibly higher. In the same way, if we use pt's that are higher than the true values, we will know that the agents will be guilty with at least the computed probabilities.

There are T={t1,t2,t3};R1={t1,t2};R2{t2,t3};S={t1,t2,t3}

In this case, all three of the distributor's objects have been leaked and appear in S. Let us first consider how the target may have obtained object t1, which was given to both agents. The target either guessed t1 or one of U1 or U2 leaked it. We know that the probability of the former event is p, so assuming that probability that each of the two agents leaked t1 is the same we have the following cases:

- The target guessed t1 with probability p;
- Agent U1 leaked t1 to S with probability (1 - p)/2;
- Agent U2 leaked t1 to S with probability (1 -p)/2;

Similarly, we find that agent U1 leaked t2 to S with Probability 1 - p since he is the only agent that has t2. Given these values, the probability that agent U1 is not Guilty, namely that U1 did not leak either object is:

$$(1 -(1 -p)/2) -(1 -(1 - p)); \quad (1)$$

And the probability that U1 is guilty is: 1 – Pr {G1}

Note that if did not hold, our analysis would be more complex because we would need to consider joint events, e.g., the target guesses t1 and at the same time one or two agents leak the value. In our simplified analysis we say that an agent is not guilty when the object can be guessed, regardless of whether the agent leaked the value. Since we are "not counting" instances when an agent leaks information, the Simplified analysis yields conservative values (smaller Probabilities).

To simplify the formulas that we present in the rest of the paper, we assume that all T objects have the same pt, which we call p. Our equations can be easily generalized to diverse pt's though they become cumbersome to display. Next, we make two assumptions regarding the relationship among the various leakage events. The first assumption simply states that an agent's decision to leak an object is not related to other objects. In [14] we study a scenario where the actions for different objects are related, and we study how our results are impacted by the different independence assumptions.

*Assumption 1 :For all t,t' $\epsilon$ S such that t $\neq$ t' the provenance of t is independent of the provenance of t'.*

The term "provenance" in this assumption statement refers to the sources of a value t that appears in the leaked set. The Source can be any of the agents who have t in their sets or the target itself (guessing). To simply our formulas, the following assumption states that join events have a negligible probability. As we argue in the example below, this assumption gives us more conservative estimates for the guilt of agents, which is consistent with our goals.

*Assumption 2: An Object t $\epsilon$ S can only be obtained by the target in one of the two ways as follows:*

- A single agent $U_i$ leaked t from its own $R_i$ set.
- The target guessed (or obtained through other means) t without the help of any of the n agents.

In other words, for all $t \epsilon S$, the event that the target guesses t and the events that agents $U_i(i=1.....n)$ leaks objects t are disjoint. Before we present the general formula for computing the probability Pr{Gi|S} that an agent Ui is guilty, we provide a simple example. Assume that the distributor set T, the agent sets R's and the target set S are:

T = {t1, t2, t3}, R1 = {t1, t2}, R2 = {t1, t3}, S = {t1, t2, t3}.

In this case, all three of the distributor's objects have been leaked and appear in S. Let us first consider how the target may have obtained object t1, which was given to both agents. From Assumption 2, the target either guessed t1 or one of U1 or U2 leaked it. We know that the probability of the former event is p, so assuming that probability that each of the two agents leaked t1 is the same we have the following cases:

- The target guessed t1 with probability p;
- Agent U1 leaked t1 to S with probability (1 . p)/2;
- Agent U2 leaked t1 to S with probability (1 . p)/2;

Similarly, we find that agent U1 leaked t2 to S with probability 1. p since he is the only agent that has t2. Given values, the probability U1 is guilty is:

$$\Pr\{G1|S\} = 1 - \Pr\{\bar{G}1\}. \quad (2)$$

Note that if Assumption 2 did not hold, our analysis would be more complex because we would need to consider joint events, e.g., the target guesses t1 and at the same time one or two agents leak the value. Since we are "not counting" instances when an agent leaks information, the simplified analysis yields conservative values (smaller probabilities). In the general case (with our assumptions), to find the probability that an agent Ui is guilty given a set S, first we compute the probability that he leaks a single object t to S. To compute this we define the set of agents

$Vt = \{Ui|t \in Ri\}$ that have t in their data sets. Then using Assumption 2 and known probability p, we have:

$$\Pr\{\text{some agent leaked t to S}\} = 1-p. \quad (3)$$

Assuming that all agents that belong to Vt can leak t to S with equal probability and using Assumption 2 we obtain:

$$\Pr\{Ui \text{ leaked t to S}\} = \{^{1-P}|Vt| \text{ , if } Ui \in Vt, 0, \text{otherwise} \quad (4)$$

Given that agent Ui is guilty if he leaks at least one value to S, with Assumption 1 and Equation 4 we can compute the probability Pr {Gi|S} that agent Ui is guilty:

$$\Pr\{Gi|S\} = 1 - \prod_{t \in S \cap Ri} \left(1 - \frac{1-P}{|Vt|}\right) \quad (5)$$

Fake Objects: The distributor may be able to add fake objects to the distributed data in order to improve his effectiveness in detecting guilty agents. However, fake objects may impact the correctness of what agents do, so they may not always be allowable. Perturbed, e.g., by adding random noise to sensitive salaries, or adding a watermark to an image. In our case, we are perturbing the set of distributor objects by adding fake elements. In some applications, fake objects may cause fewer problems that perturbing real objects.

## V. Future Enhancements

In this paper we are using multiple agents, for the purpose of security at the same time we are creating database for separate user, so through this we are strictly find out who is leaked information in internet.

## References Références Referencias


1. R. Agrawal and J. Kiernan, "Watermarking Relational Databases," Proc. 28th Int'l Conf. Very Large Data Bases (VLDB '02), VLDB Endowment, pp. 155-166, 2002.
2. P. Bonatti, S.D.C. di Vimercati, and P. Samarati, "An Algebra for Composing Access Control Policies," ACM Trans. Information and System Security, vol. 5, no. 1, pp. 1-35, 2002.
3. P. Buneman, S. Khanna, and W.C. Tan, "Why and Where: A Characterization of Data Provenance," Proc. Eighth Int'l Conf. Database Theory (ICDT '01), J.V. den Bussche and V. Vianu, eds., pp. 316-330, Jan. 2001.
4. P. Buneman and W.-C. Tan, "Provenance in Databases," Proc. ACM SIGMOD, pp. 1171-1173, 2007.
5. Y. Cui and J. Widom, "Lineage Tracing for General Data Warehouse Transformations," The VLDB J., vol. 12, pp. 41-58, 2003.
6. S. Czerwinski, R. Fromm, and T. Hodes, "Digital Music Distribution and Audio Watermarking," http://www.scientificcommons. org/43025658, 2007.
7. F. Guo, J. Wang, Z. Zhang, X. Ye, and D. Li, "An Improved Algorithm to Watermark Numeric Relational Data," Information Security Applications, pp. 138-149, Springer, 2006.
8. F. Hartung and B. Girod, "Watermarking of Uncompressed and Compressed Video," Signal Processing, vol. 66, no. 3, pp. 283-301, 1998.
9. S. Jajodia, P. Samarati, M.L. Sapino, and V.S. Subrahmanian, "Flexible Support for Multiple Access Control Policies," ACM Trans. Database Systems, vol. 26, no. 2, pp. 214-260, 2001
10. Y. Li, V. Swarup, and S. Jajodia, "Fingerprinting Relational Databases: Schemes and Specialties," IEEE Trans. Dependable and Secure Computing, vol. 2, no. 1, pp. 34-45, Jan.-Mar. 2005.
11. B. Mungamuru and H. Garcia-Molina, "Privacy, Preservation and Performance: The 3 P's of Distributed Data Management, "technical report, Stanford Univ., 2008.
12. V.N. Murty, "Counting the Integer Solutions of a Linear Equation with Unit Coefficients," Math. Magazine, vol. 54, no. 2, pp. 79-81, 1981.
13. S.U. Nabar, B. Marthi, K. Kenthapadi, N. Mishra, and R. Motwani, "Towards Robustness in Query Auditing," Proc. 32nd Int'l Conf. Very Large Data Bases (VLDB '06), VLDB Endowment, pp. 151-162, 2006.
14. P. Papadimitriou and H. Garcia-Molina, "Data Leakage Detection," technical report, Stanford Univ., 2008.
15. P.M. Pardalos and S.A. Vavasis, "Quadratic Programming with One Negative Eigen value Is NP-Hard," J. Global Optimization, vol. 1, no. 1, pp. 15-22, 1991.
16. J.J.K.O. Ruanaidh, W.J. Dowling, and F.M. Boland, "Watermarking Digital Images for Copyright Protection," IEE Proc. Vision, Signal and Image Processing, vol. 143, no. 4, pp. 250-256, 1996.
17. R. Sion, M. Atallah, and S. Prabhakar, "Rights Protection for Relational Data," Proc. ACM SIGMOD, pp. 98-109, 2003.
18. L. Sweeney, "Achieving K-Anonymity Privacy Protection Using Generalization and Suppression," http://en.scientificcommons.

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# An Effcient Algorithm for Mining Association Rules in Massive Datasets

By D. Gunaseelan & P. Uma
*JAZAN University, Kingdom of Saudi Arabia*

*Abstract -* Data mining, also known as Knowledge Discovery in Databases (KDD) is one of the most important and interesting research areas in 21st century. Frequent pattern discovery is one of the important techniques in data mining. The application includes Medicine, Telecommunications and World Wide Web. Nowadays frequent pattern discovery research focuses on finding co-occurrence relationships between items. Apriori algorithm is a classical algorithm for association rule mining. Lots of algorithms for mining association rules and their mutations are proposed on the basis of Apriori algorithm. Most of the previous algorithms Apriori-like algorithm which generates candidates and improving algorithm strategy and structure but at the same time many of the researchers not concentrate on the structure of database. In this research paper, it has been proposed an improved algorithm for mining frequent patterns in large datasets using transposition of the database with minor modification of the Apriori-like algorithm. The main advantage of the proposed method is the database stores in transposed form and in each iteration database is filtered and reduced by generating the transaction id for each pattern. The proposed method reduces the huge computing time and also decreases the database size. Several experiments on real-life data show that the proposed algorithm is very much faster than existing Apriori-like algorithms. Hence the proposed method is very much suitable for the discovering frequent patterns from large datasets.

*Keywords :* *Data mining, frequent pattern mining, transposition of database, Apriori algorithm.*

*GJCST-C Classification:* *H.2.8*


*Strictly as per the compliance and regulations of:*

# An Effcient Algorithm for Mining Association Rules in Massive Datasets

D. Gunaseelan [α] & P. Uma [σ]

*Abstract -* Data mining, also known as Knowledge Discovery in Databases (KDD) is one of the most important and interesting research areas in 21st century. Frequent pattern discovery is one of the important techniques in data mining. The application includes Medicine, Telecommunications and World Wide Web. Nowadays frequent pattern discovery research focuses on finding co-occurrence relationships between items. Apriori algorithm is a classical algorithm for association rule mining. Lots of algorithms for mining association rules and their mutations are proposed on the basis of Apriori algorithm. Most of the previous algorithms Apriori-like algorithm which generates candidates and improving algorithm strategy and structure but at the same time many of the researchers not concentrate on the structure of database. In this research paper, it has been proposed an improved algorithm for mining frequent patterns in large datasets using transposition of the database with minor modification of the Apriori-like algorithm. The main advantage of the proposed method is the database stores in transposed form and in each iteration database is filtered and reduced by generating the transaction id for each pattern. The proposed method reduces the huge computing time and also decreases the database size. Several experiments on real-life data show that the proposed algorithm is very much faster than existing Apriori-like algorithms. Hence the proposed method is very much suitable for the discovering frequent patterns from large datasets.

*Keywords : Data mining, frequent pattern mining, transposition of database, Apriori algorithm.*

## I. Introduction

Data mining is one of the most dynamic emerging research in today's database technology and Artificial Intelligent research; the main aim is to discover valuable patterns from a large collection of data for users. In the transaction database, mining association rule is one of the important research techniques in data mining field. The original problem addressed by association rule mining was to find the correlation among sales of different items from the analysis of a large set of super market data. Right now, association rule mining research work is motivated by an extensive range of application areas, such as banking, manufacturing, health care, medicine, and telecommunications. There are two key issues that need to be addressed when applying association analysis.

*Author α, σ : College of Computer & Information Systems JAZAN University, Kingdom of Saudi Arabia.*
*E-mail α : dgseela@yahoo.com*
*E-mail σ : prmluma@gmail.com*

The first one is that discovering patterns from a large dataset can be computationally expensive, thus efficient algorithms are needed. The second one is that some of the discovered patterns are potentially spurious because they may happen simply by chance. Hence, some evaluation criteria are required.

Agrawal and Srikant (1994) proposed the Apriori algorithm to solve the problem of mining frequent itemsets. Apriori uses a candidate generation method, such that the frequent (k+1)-itemset in one iteration can be used to construct candidate (k+1)-itemsets for the next iteration. Apriori terminates its process when no new candidate itemsets can be generated. It is a multi-pass algorithm.

Unlike Apriori, the FP-growth method was proposed by Han et al. (2000) uses an FP-tree to store the frequency information of the transaction database. Without candidate generation, FP-growth uses a recursive divide-and-conquer method and the database projection approach to find the frequent itemsets. However, the recursive mining process may decrease the mining performance and raise the memory requirement.

Most of the reviews are presented in Section 2.2.A lot of algorithms were proposed to optimize the performance of the Apriori-like algorithm. In this research paper it has been presented an efficient and improved frequent pattern algorithm for mining association rules in large datasets. It is a two-pass algorithm.

The remainder of the paper is organized as follows: In Section 2, it has been described in brief an Apriori algorithm, and the relative researches of association rules. In Section 3 provides definitions for the mining method, and detailed steps on the proposed algorithm in mining frequent itemsets. An illustration is demonstrated in Section 4. In Section 5, the design of the experiment and performance analysis is discussed; finally, in Section6 offers conclusions.

## II. Background

At first, the data mining technique for association rule mining is the support-confidence framework established by Agrawal et al. [AIS 93]. The most important time-consuming part of the association rule algorithm is to discover large itemsets, while the generation of association rules from the given large itemsets is straightforward. This paper has been

focused on the discovery of large itemsets. For description, some well-known methods and notions based on this framework is used throughout this paper. In this section it has been presented the formal statement of association rule mining and the description of Apriori algorithm and related research review.

*a) Formal statement of the problem*

The following is a formal statement of association rule mining for transactional databases.

Let $I = \{i_1, i_2, i_3, \ldots, i_n\}$ represents a set of 'n' distinct data items. Generally, a set of items is called an itemset, and an itemset with k items is denoted as a k-itemset. Database D is a set of transactions, where the $i^{th}$ transaction $T_i$ denotes a set of items, such as $T_i \subseteq I$. $|D|$ is the total number of transactions in D, and $|T_i|$ is the number of distinct items in transaction $T_i$. Each transaction is associated with a unique identifier, which is termed as TID. An association rule is an implication of the form $X \rightarrow Y$, where $X, Y \subseteq I$, and $X \cap Y = \phi$. There are measures of quality for each rule in support of itemset $X \cup Y$ and confidence of rule $X \rightarrow Y$. First, we need to calculate the support of itemset $X \cup Y$, which is the ratio (denoted by s%) of the number of transactions that contain the $X \cup Y$ to $|D|$. Next, the confidence of rule $X \rightarrow Y$ is the ratio (denoted by c%) of the number of transactions containing $X \cup Y$ to the number of transactions that contain X in database D. The problems of association mining rules from database D can be processed in two important steps: (1) locate all frequent itemsets whose supports are not less than the user-defined minimum support threshold $\xi$, where $\xi \in (0, 1)$, and, (2) obtain association rules directly from these frequent itemsets with confidences not less than the user-defined minimum confidence threshold. The most time-consuming part of mining association rules is to discover frequent itemsets.

*b) Review of Apriori algorithm*

In conventional Apriori-like methods, the level wise process of identifying sets of all frequent itemsets is in a combination of smaller, frequent itemsets. In the *k*th level, the Apriori algorithm identifies all frequent *k*-itemsets, denoted as $L_k$. $C_k$ is the set of candidate *k*-itemsets obtained from $L_{k-1}$, which are suspected frequent *k*-itemsets. For each transaction in *D*, the candidate *k*-itemsets in $C_k$ contained within the transaction are determined, and their support count is increased by $1/|D|$. Following scanning (reading) and contrasting with the entire *D*, when the supports of candidate *k*-itemsets are greater than or equal to user-defined minimum support threshold $\xi$, they immediately become frequent *k*-itemsets. At the end of level *k*, all frequent itemsets of length *k* or less have been discovered. During the execution, numerous candidate itemsets are generated from single itemsets, and each candidate itemset must perform contrasts on the entire database, level by level, while searching for frequent itemsets. However, the performance is significantly affected because the database is repeatedly read to contrast each candidate itemset with all transaction records of the database.

*c) Related researches of association rules*

In 1995, Savasere et al. proposed the partition algorithm to improve the efficiency of Apriori algorithm, it does so by efficiently reducing the number of scans in the database, however, considerable time is still wasted scanning infrequent candidate itemsets [3]. In 1996, Pork et al. proposed an efficient and fast algorithm called DHP (direct hashing and pruning) for the initial candidate set generation. This method efficiently controls the number of candidate 2-itemsets, pruning the size of the database [4]. In 1999, Han et al. proposed a top-down method, which investigates progressively deeper, into the data was developed for the efficient mining of multiple-level association rules from large transactional databases based on the classical Aprioir principle. In 1996, Toivonen proposed a sampling algorithm which reduces the number of database scan to a single scan, but still wastage considerable time on candidate itemsets [9]. In1996, Brid et al. proposed the dynamic itemset count (DIC) algorithm [5] for finding large itemsets, which uses fewer passes over the data than classical algorithms, and yet uses fewer candidate itemsets than methods based on sampling. In addition, in 1999, Dunkel et al. proposed a column-wise apriori algorithm for frequent itemsets and in 2001, Berzal et al. proposed a tree based association rule mining which transformed the storage structure of the data, to reduce the time needed for database scans, improving overall efficiency.

## III. Proposed Algorithm

The proposed algorithm improvement mainly concentrated on (1) for reducing frequent itemset and (2) for reducing storage space as well as the computing time. In the case of large datasets like Wal-Mart datasets, the proposed algorithm is very much useful for reducing the frequent patterns and also reducing the database size for every subsequent passes. For example, in the improved algorithm, the number of occurrence of frequent k-itemsets when k-itemsets are generated from (k-1)-itemsets is computed. If k is greater than the size of the database D, there is no need to scan database D which is generated by (k-1)-itemsets according to the Aprior property and it can be remove automatically.

**Transposition of database:** A given database as a relation between original and transposed representations of a database is defined in Table 1. The itemsets are $D = \{I_1, I_2, \ldots, I_n\}$ and transaction ids are $TID = \{T_1, T_2, \ldots, T_m\}$. A string notation for itemsets is used, for example, $I_1I_4I_5$ denotes the itemset $\{I_1, I_4, I_5\}$ and $T_2T_4$ denotes the transaction ids set $\{T_2, T_4\}$. This

dataset is used in all the examples between two sets: a set of items (attributes) and a set of transactions (tuples).

*Table 1 :* Database D and transposition Database $D^T$

D

| Transaction IDs | Items |
|---|---|
| T1 | I1, I2, I3 |
| T2 | I2, I3, I4 |
| T3 | I1, I3, I4 |
| .. | … |

$D^T$

| Items | Transaction IDs |
|---|---|
| I1 | Tl, T3 |
| I2 | Tl, T2 |
| I3 | T1, T2, T3 |
| I4 | T2, T3 |

*Table 2 :* Notations used

| Notations | Description |
|---|---|
| D | Given database |
| $D^T$ | Transposed database |
| CT | Candidate transaction IDs |
| $CT_1$ | Candidate transaction IDs of size-1 |
| $LT_1$ | Large transaction IDs of size-1 |
| $CT_{k-1}$ | Candidate transaction IDs of size-k-1 |
| $LT_{k-1}$ | Large transaction IDs of size-k-1 |
| s | Minimum support |
| c | Minimum confidence |
| Count | Frequency |

At first, the given transaction database file D is transposed to database $D^T$ and count the number of item and number of transaction string generated for each item and sort the item numbers. Now apply Apriori-like algorithm in which first calculate the frequent transactions $CT_1$. It reduces infrequent transactions and its item details. For the subsequent passes Apriori-gen has been applied and finds the subsequent frequent transactions.

**Lemma 1:** All the subsets of a frequent transaction must also frequent. In other words, all the supersets of a frequent transaction must also infrequent.

Improved Algorithmic steps are described as below:

1. First the function *apriori-gen*($LT_{k-1}$) is called and to generate candidate k-transaction set by frequent k-transactions.
2. Checking whether candidate transactions *CT* are joined into candidate k-transactions or not. It proceeds by calling function recursively *has_infrequent_transactions*(ct, $LT_{k-1}$). If it is true, it means the set of transactions are not frequent and should be removed. Otherwise, scan database $D^T$.
3. The occurrence of frequent k-transaction is computed by generating (k-1)-transactions from k-transactions. If k-transaction is greater than the size of database $D^T$, it is not needed to scan database $D^T$ which is generated by (k-1)-transactions based on the lemma 1, and it can be deleted.
4. If the size of database $D^T$ is greater than or equal to k, then call function *subset*($CT_k$, $d^t$), which computes frequent pattern using a subsequent iterative level-wise approach based on candidate generation.

Algorithm 1: *Improved Algorithm*

Input: A transposed database $D^T$ and the user defined minimum support threshold s.

Output: The complete set of frequent patterns

Step 1: Convert Database D into transpose form $D^T$

Step 2: Compute $CT_1$ candidate transaction sets of size-1 and finds the support count.

Step 3:Compute the large transaction sets ($LT$) of size1. (i.e., for all $CT_1$ is greater than or equal to minimum support.)

$LT_1$ = {Large 1-transaction set ($LT$)};
For (k=2; $LT_{k-1}$ = 0; k++) do
Begin
$CT_k$ = *Apriori-gen*($LT_{k-1}$, *ct*); // new candidate transaction sets
End
Return $LT = \cup_k LT_k$;

Algorithm 2: *Apriori-gen* ($LT_{k-1}$), Generate candidate sets
For all transactions p ∈ $LT_{k-1}$ do begin
For all transactions q ∈ $LT_{k-1}$ do begin
If p.transaction$_1$=q.transaction,…, p.transaction$_{k-2}$=q.transaction$_{k-2}$, p.transaction$_{k-1}$< q.transaction$_{k-1}$ then begin
ct=p ∞ q;
If *has_infrequent_subset*(*ct*, $LT_{k-1}$) then
delete ct;
Else
For all transaction set t∈ $D^T$ do begin
If count(t) <k then delete t;
Else begin
Ct=subset($CT_k$, t);
End
End
For all candidate transactions ct∈$CT_i$ do begin
*CT.count* = *CT.count* + 1;
End; End;
$LT_k$ = {*ct*∈$CT_k$ | *CT.count* ≥ *s*};
End; End;
End;
End;
Return $CT_k$;
Algorithms 3: *has_infrequent_subset*(*ct*, $LT_{k-1}$)

// checking the elements of candidate generation
For all (*k-1*)-sub transaction set of *ct* do

Begin
If $t \in LT_{k-1}$ then
return true;
Else
return false;
End.

The main advantage of the proposed algorithm for frequent patterns discovery are, it reduces the size of the database after second pass and, the storage space and saves the computing time.

## IV. Performance Evaluation

The following is an example shows the processing steps of the proposed algorithm

Figure 1 shows the original Database D and the transposed database $D^T$. There are 15 transaction IDs in the database $D^T$, that is $|D^T| = 9$ and minimum support $s = 20\%$. The improved algorithm for mining frequent patterns in $D^T$ is used.

1. Scan the database $D^T$ for support count of each candidate transactions.

In the first iteration of the improved algorithm, all transaction sets are the member of the set of candidate 1-transactions, $CT_1$. The proposed method scans all the itemsets in $D^T$ and count the number of occurrences of each itemset.

2. Compute the support count with minimum support.

The user defined minimum support *s* is 20%, that the required support count is 2. Based on the minimum support, we can determine the set of frequent set of 1-transaction IDs($LT_1$). That means all the candidate 1-transaction IDs are satisfied with user defined minimum support *s*.

3. Generate all candidate transactions of size-2 i.e., $CT_2$ from $LT_1$ and count the support count.

The algorithm generates candidate transactions $CT_2$ from large transaction set of size-1, $LT_1$. Compute the number of occurrences in each transaction set by scanning the database $D^T$. Accumulate the total number of sub-transaction IDs with their support count.

4. Compare the number of occurrences of candidate transaction IDs with their minimum support s.

The Large transaction ID sets of size-2, $LT_2$ are determined by computing the number of occurrences of each candidate transaction IDs $CT_2$ with the minimum support s. Based on $LT_2$, we can determined a new modified transposed database $D^T_2$.

5. Generate candidate transactions of size-3 from $LT_2$ by scanning new modified database $D^T_2$ and finds the support count of $CT_3$.

First, combine the large transactions of size-2, $LT_2$ with $LT_2$ to determine $CT_3$. Based on the lemma 1, we can determine the four letter candidate transaction IDs $C_3$ cannot possibly be frequent transactions and therefore prune from $CT_3$. This is one of the advantages of saving time to count the number of occurrence of transaction IDs unnecessarily during the subsequent scan of $D^T_2$ for finding $LT_3$.

6. Compare the support count of candidate transaction IDs with minimum support.

The modified database $D^T_2$ is scanned by computing $LT_3$. i.e., the large transaction IDsof size-3, $LT_3$ are determined by computing the number of occurrences of each candidate transaction IDs $CT_3$ with the minimum support s.

7. Repeat the steps 4 to 6 until no more candidate transaction IDs are generated.

That is the algorithm terminates, having found all of the frequent transaction IDs. Also, it creates the modified database $D^T_3$, $D^T_4$, etc., based the size of the transaction IDs.

The following are the explanation of the proposed algorithm with an example.

| Transaction ID | Item ID |
|---|---|
| T1 | 1, 14 |
| T2 | 2, 4, 6, 7, 13, 15 |
| T3 | 4, 6, 10, 11, 12, 14 |
| T4 | 2, 3, 6, 13 |
| T5 | 1, 3, 5, 8, 10, 11, 12, 14 |
| T6 | 1, 5, 7, 12, 13, 14 |
| T7 | 3, 5, 7, 10, 11, 12, 13, 14 |
| T8 | 1, 2, 9, 12 |
| T9 | 7, 15 |

Original Database D

Transposed Database $|D^T| = 15$

Now apply the improved algorithm;

| Item ID | Transaction ID |
|---|---|
| 1 | T1, T5, T6, T8 |
| 2 | T2, T4, T8 |
| 3 | T4, T5, T7 |
| 4 | T2, T3 |
| 5 | T5, T6, T7 |
| 6 | T2, T3, T4 |
| 7 | T2, T6, T7, T9 |
| 8 | T5 |
| 9 | T8 |
| 10 | T3, T5, T7 |
| 11 | T3, T5, T7 |
| 12 | T3, T5, T6, T7, T8 |
| 13 | T2, T4, T6, T7 |
| 14 | T1, T3, T4, T5, T6, T7 |
| 15 | T2, T9 |

Minimum Support (s)=20%

Pass 1: Generate candidates for k=1

$C_1$={ T1, T2, T3, T4, T5, T6, T7, T8, T9}

| $C_1$ | T1 | T2 | T3 | T4 | T5 | T6 | T7 | T8 | T9 |
|---|---|---|---|---|---|---|---|---|---|
| Support | 2 | 6 | 6 | 4 | 8 | 6 | 8 | 4 | 2 |

$L_1$ = {T2:6, T3:6, T4:4, T5:8, T6:6, T7:8, T8:4}

Pass 2: Generate candidates for k=2

$C_2$ = {(T2,T3), (T2,T4), (T2,T5), (T2,T6), (T2,T7), (T2,T8), (T3,T4), (T3,T5), (T3,T6), (T3,T7), (T3,T8), (T4,T5), (T4,T6), (T4,T7), (T4,T8), (T5,T6), (T5,T7), (T5,T8), (T6,T7), (T6,T8), (T7,T8) } - 21 candidate sets

After applying improved algorithm

| $C_1$ | $T_2T_3$ | $T_2T_4$ | $T_2T_5$ | $T_2T_6$ | $T_2T_7$ | $T_2T_8$ | $T_3T_4$ | $T_3T_5$ | $T_3T_6$ | $T_3T_7$ | $T_3T_8$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Sup | 2 | 3 | 0 | 2 | 2 | 1 | 2 | 4 | 2 | 4 | 1 |

| $C_1$ | $T_4T_5$ | $T_4T_6$ | $T_4T_7$ | $T_4T_8$ | $T_5T_6$ | $T_5T_7$ | $T_5T_8$ | $T_6T_7$ | $T_6T_8$ | $T_7T_8$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Sup | 2 | 2 | 3 | 1 | 4 | 3 | 2 | 4 | 2 | 1 |

$L_2$ = {(T2T4):3, (T3T5):4, (T3T7):4, (T4T7):3, (T5T6):4, (T5T7):3, (T6T7):3}

– 7 large transaction sets only

$|D^T|$ = 11

| Item ID | Transaction ID |
|---|---|
| 1 | T1, T5, T6, T8 |
| 2 | T2, T4, T8 |
| 3 | T5, T7 |
| 5 | T5, T6, T7 |
| 6 | T2, T3, T4 |
| 7 | T2, T6, T7, T9 |
| 10 | T3, T4, T5, T7 |
| 11 | T3, T4, T5, T7 |
| 12 | T3, T4, T5, T6, T7, T8 |
| 13 | T2, T6, T7 |
| 14 | T1, T3, T4, T5, T6, T7 |

Based on $L_2$, we can prune infrequent transaction sets from the transposed database $D^T$. After pruning, the new modified transposed database $D^T_2$ with number of itemsets is 11 only. Previously it was 15.

Pass 3: Generate candidates for k=3

$C_3$ = {(T3 T5 T7), (T5 T6 T7)}

| $C_3$ | T3 T5 T7 | T5 T6 T7 |
|---|---|---|
| Support | 4 | 3 |

$L_3$ = {(T3 T5 T7), (T5 T6 T7)}

Based on L3, we can prune infrequent transaction sets from the transposed database $D^T$. After pruning, the new modified transposed database $D^T_3$ with number of itemsets is 7 only. Previously it was 11.

$|D^T_3|$ = 8

| Item ID | Transaction ID |
|---|---|
| 1 | T1, T5, T6, T8 |
| 5 | T5, T6, T7 |
| 7 | T2, T6, T7, T9 |
| 10 | T3, T4, T5, T7 |
| 11 | T3, T4, T5, T7 |
| 12 | T3, T4, T5, T6, T7, T8 |
| 13 | T2,T6, T7 |
| 14 | T1, T3, T4, T5, T7 |

Pass 4: Generate candidates for k=4

C4 = { (T3 T5 T6 T7): 1 }

L4 = {ϕ}

## V. Experimental Results

To evaluate the efficiency and effectiveness of the improved algorithm, we performed an extensive study of two algorithms: Apriori-like and improved algorithm, on both real time synthetic data sets with different ranges. All the experiments were examined on Pentium IV machine 1GB RAM, running Microsoft Windows 7. Two algorithms, Apriori and Improved algorithm were implemented in Java 2.0.

Also we got the real time medical database with 2280 itemsets and 4200 elements. The running time comparison between improved algorithm and Apriori algorithm are shown in the Figure 1 with minimum support ranges from 1 percentage (%) to 5 percentages (%).

The importance of improved algorithm is to reduce the number of items in each and every scan and also reduce the size of the original dataset. There are

three aspects to make this algorithm better than the original one.

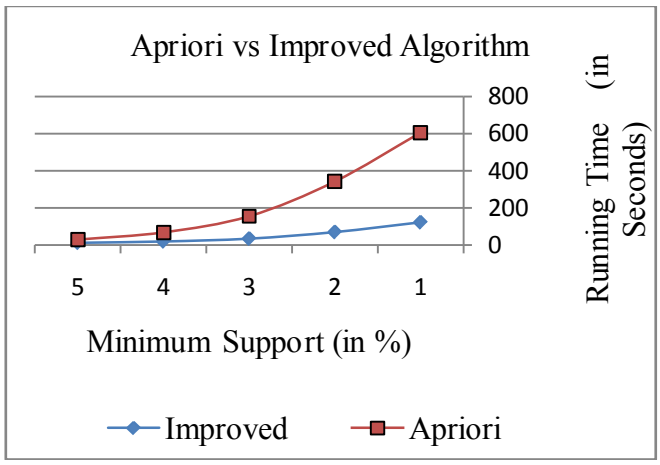



*Figure 1:* Running time between Apriori and improved algorithm

Firstly, when the candidates are being produced, instead of dealing with all the items of the previous large set, only the elements which having the same transaction ids are crossed. At the same time, generating frequent patterns, it may reduce the computing time dramatically and the size of the database is reduced. Secondly, by pruning, the number of elements in the candidate sets is decreased once more based on modified database. Finally, the computing time and storage space are saved.

## VI. Conclusion

In this research paper, it has been proposed an improved algorithm for mining frequent pattern based on Apriori-like algorithm. The main advantages of an improved algorithm are that it can reduce the number of scanning by the transposed database $D^T$, redundancy by the time of generating sub-transaction set tests and verifying them in the database. In order to discover frequent patterns in massive datasets with more columns than rows, it has been presented a complete framework for the transposition; the item set in the transposed database of the transposition of many classical transactions is given. Also it has been compared the classical Apriori algorithm with an improved algorithm. It has been presented the experimental results, using synthetic data, showing that the proposed algorithm always outperform Apriori algorithm. Hence, the proposed algorithm is very much suitable for a massive datasets.

## VII. Acknowledgements

The authors are extremely express gratitude to Dr. Omar Sayed Al-Mushayt, College Dean and Dr. Saeed Q Al-Khalidi, Vice Dean, College of Computer Science and Information Systems, JAZAN University, Kingdom of Saudi Arabia for having noble and continuous encouragement to complete this research. The special thanks also to the University President, JAZAN University, Kingdom of Saudi Arabia for inspiration and persistent support directly or indirectly for the completion of this research.

## References Références Referencias

1. Agrawal, R. and Srikant, R., 1994. Fast algorithms for mining association rules in large databases. *VLDB '94: Proceedings of the 20th International Conference on Very Large Data Bases*, San Francisco, USA, pp. 487-499.
2. Barabasi, A. and Albert, R., 1999. Emergence of scaling in random networks. *Science*, Vol. 286, pp. 509-512.
3. Bayardo, R. J., 1998. Efficiently mining long patterns from databases. *SIGMOD '98: Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, Seattle, USA, pp. 85-93.
4. Brin, S. et al, 1997. Dynamic itemset counting and implication rules for market basket data. *SIGMOD '97: Proceedings of the 1997 ACM SIGMOD international conference on Management of data*, Tucson, USA, pp. 255-264.
5. Cooper, C., 2006. The age specific degree distribution of web-graphs. *Combinatorics, Probability and Computing*, Vol. 15, No. 5, pp. 637-661.
6. Han, J. et al, 2000. Mining frequent patterns without candidate generation. SIGMOD '00: Proceedings of the 2000 ACM SIGMOD international conference on Management of data, Dallas, USA, pp. 1-12.
7. Han, J., Kamber, M and Pei. *Data Mining Concepts and Techniques*, The Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann Publishers, July 2011.
8. Purdom, P. W. et al, 2004. Average-case performance of the apriori algorithm. *SIAM J. on Comput.*, Vol. 33, pp. 1223-1260.
9. Watts, D. J. 2004. The "new" science of networks. *Annual Review of Sociology*, Vol. 30, pp. 243-270.
10. Zaki, M. J. and Ogihara, M., 1998. Theoretical foundations of association rules. *Proc. 3rd SIGMOD Worksh. Research Issues in Data Mining and Knowledge Discovery*, Seattle, USA, pp. 1-8.
11. Zheng, Z. et al, 2001. Real world performance of association rule algorithms. *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, San Francisco, USA, pp. 401-406.

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# Multidimensional Analysis Data to Create a Decision Support System Dedicated to the University Environment

By Latifa. Oubedda, Brahim. Erraha & Mohamed. Khalfaoui
*National School of Applied Sciences -Agadir,University Ibn Zohr*

*Abstract -* Our objective is to make proposals for the design of a SIS SID-quality and meet the needs of different stakeholders of the university. This is where we join (which is poorly modeled by the concept of data marts in the current tools of the market), namely the modeling of data resources. Often the documents are deposited on the information system of an organization without classification, without indexing, with all the information on their content, their purpose, their technical requirements and practices. The method of describing the properties of a document is a binding step involves an author and a culture of destruction of documents. Few users perform document properties they file on a system design and information. Then it is naturally more difficult to retrieve these information gaps which usually take the form of voids, it is still necessary that the input fields are provided adequate and appropriately organized, arranged and explained. Indeed, it often happens - for example on an intranet of an organization - the drop zones are not conducive to give relevant information on the properties of materials downloaded. In the best case, the documents are managed by their own systems, accessible through their own search engine or by federated search engines. Why we try to answer the question: how to reproduce a set of metadata specific to multidimensional databases specific to the decision-oriented universities.

*Keywords :* *Multidimensional databases, Metadata, data marts, design and information system.*

*GJCST-C Classification:* *E.1*


*Strictly as per the compliance and regulations of:*

# Multidimensional Analysis Data to Create a Decision Support System Dedicated to the University Environment

Latifa. Oubedda [α], Brahim. Erraha [α] & Mohamed. Khalfaoui [σ]

*Abstract -* Our objective is to make proposals for the design of a SIS SID-quality and meet the needs of different stakeholders of the university. This is where we join (which is poorly modeled by the concept of data marts in the current tools of the market), namely the modeling of data resources. Often the documents are deposited on the information system of an organization without classification, without indexing, with all the information on their content, their purpose, their technical requirements and practices. The method of describing the properties of a document is a binding step involves an author and a culture of destruction of documents. Few users perform document properties they file on a system design and information. Then it is naturally more difficult to retrieve these information gaps which usually take the form of voids, it is still necessary that the input fields are provided adequate and appropriately organized, arranged and explained. Indeed, it often happens - for example on an intranet of an organization - the drop zones are not conducive to give relevant information on the properties of materials downloaded. In the best case, the documents are managed by their own systems, accessible through their own search engine or by federated search engines. Why we try to answer the question: how to reproduce a set of metadata specific to multidimensional databases specific to the decision-oriented universities.

*Keywords : Multidimensional databases, Metadata, data marts, design and information system.*

## I. Introduction

The actors of the university have centered on the need Reporting, others need however to analyze more precisely the data (University maker). It is therefore to explain the anomalies and their origins, such as causing problems for the disappearance of students during their university life without any qualifications. It is also to highlight extreme events in the very structure of a numerical result.

An analysis of data reveals disparities and to explain phenomena apparently normal. In this logic, the Drill down is a method to visualize the detail information component, the opposite, the Drill up scrolls upward through the hierarchy of a dimension while the Drill through is to see other indicators to explain information. Starting from a 3D cube, it is possible to aggregate rotating along one dimension (pivot); we obtain a lattice of views (computable in SQL). The table below contains the main principles of the algebra of cubes. We will prove that multivariate analysis is to model data along several axes. The OLAP cube means the analytical technology that applies to this model of representation. This notion that rubs predictive analytics as designated by the Anglicism Data mining.

## II. Hypotheses

One starts by modeling [4] upstream actors taking into account the specifications and expectations of each of them, namely:

- From motivation to the job involvement: motivation
  - Thus appears as part of a directed behavior and completed (goal oriented).
- Training required by the actors in institutions in the university year.
- Etc.....

Given this situation, it is to correlate between the needs of University actors [1], [2] and those of the teacher and those of administration. Infect, we are faced with a situation of looking for satisfaction with a specific university. Indeed for a university, it is more about positioning and visibility of the organization. The company seeks a positioning performance level of its capital and the university aims to achieve a quality and a high ranking both domestically and internationally. The company seeks customer satisfaction; the university seeks to satisfy its stakeholders. Customer satisfaction in business is formalized in terms of costs. Satisfaction of the actors in university is renowned for meeting their needs.

The main objective of this work is to provide a simple, detailed and complete enough to meet the real needs of the university decision-maker (in [5], [6]) in terms of automatic adaptation needs and priorities of the indicator is to make a multidimensional model:

### *a) The model SIAG*

According to the model developed SIAG within our team; we observe several processes in the

*Author α : Laboratory of Industrial Engineering and Computer Science (LG2I), National School of Applied Sciences -Agadir, , University Ibn Zohr. E-mail : l.oubedda@gmail.com.*
*Author σ : Analysis Laboratory for Systems, Information Processing and Integrated Management, Superior School of Technology Sale, University Mohammed V Agdal.*

phenomenon of information retrieval which we leverage for our reflections. The model represents a situation EIAG information retrieval that involves cognitive phases following:

Discover the world of information => **Study**
The application of basic information => **Item**
The analysis of the basis of information => **Analysis**
Resolution based on different choices => **Gloss**

This model uses action verbs to describe different stages of information seeking: to investigate, item, analyze and gloss. These words evoke the underlying functionality of the information system, so as to satisfy the end user. We leverage this model to analyze the situations of our various stakeholders (policy makers and institutions of the presidency of the university) in a research or production information.

## III. Implementation

### a) *Actors in the University*

Given the scope of this project which combines university: students, professors, administrative domains and operating [4] in various disciplines in terms of their thematic, structural information we propose is based on the model [5]of a warehouse data, taking into account the different trades. For example a person may have different responsibilities: it can have the status of responsible teaching.

We discuss the data on different levels by actors. We distinguish three levels: the actor, the administrative level and educational level.

- The level player makes an initial typology of actors around 3 classes, showing students, teachers and administrators.
- The educational level is used to identify bases 'referents' correlated with previously identified actors: foundation courses geared towards the students, baselines for serving teachers and basic rules and regulations for the administration of destination.
- The administrative level census data on the administrative situation of the student actor, data on the administrative situation of the actor and teacher data from administrative and financial management of students, teachers and training relevant to the administrative actor. We illustrate by a diagram that data relating to the actors, supplemented by existing.

*After the consolidation of the formula 1, we obtain:*

The portfolio of the source (S) (in[1],[2])defines all the activities to be performed during one cycle by each university players. Category (C) defines the three actors of the university: Student, Teacher, and Administrator. Aggregation (A) defines the needs of each player for a graduate level.

| Activities (academic contribution to each player) | | | |
|---|---|---|---|
| 1 | Administer | | |
| | Prepare | 4 | 13 |
| 2 | Organize | 5 | 12 |
| | Learn | | 11 |
| | Train | 8 | |
| | Track | 6 | |
| | Correct | | 10 |
| 3 | information | 7 | |
| | present | | 9 |

| Aggregations (by contributing to university ea actor) | | | |
|---|---|---|---|
| | Regulation | 5 | 10 |
| | Budget | 1 | |
| 15 | Monitoring | | 11 |
| 13 | Training | | 8 |
| | Course | 6 | 9 |
| 12 | Registration | 2 | |
| 14 | Census | 3 | 7 |
| | Preparation | 4 | |
| 16 | Resources | | |

*Beginning of the University Cycle:*

- The portfolio administrative actors is the first actor at a time t:

Administrative actor (PA)= {Ci ($1 \le i \le 3$) ; Aj($1 \le j \le 6$)}

- The actor Teacher portfolio is the second player at time $t+\Delta t$:

Actor Teacher PE) = {Ci ($3 \le i \le 8$); Aj ($6 \le j \le 11$)}

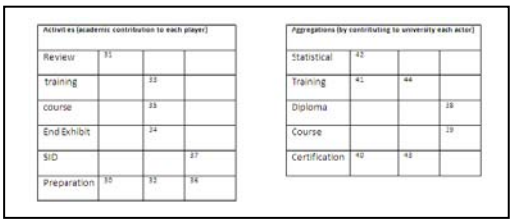
- The Student Portfolio actor is the No. 3 player at a time $t+\Delta t+1$ :Actor Student (PT)= {Si ($8 \le i \le 13$) ; Aj ($11 \le j \le 16$)}

This model is then obtained:

$$\text{Actor} \{ \sum_{i=1}^{i=13} S_i \ ; \ \sum_{j=1}^{j=16} A_j \}$$

End of University Cycle:

At the end of the cycle, the three actors are involved:



- The Teacher actor portfolio is the first speaker at the end of the academic cycle:

Teacher Actor (PE)={Si($29 \le i \le 31$) ;Aj ($37 \le j \le 39$)}

- The Student actor Portfolio is the second place at the end of the academic cycle:

Student Actor (PT)= {Si ($31 \le i \le 35$) ; Aj ($42 \le j \le 44$)}

- The administrative actors portfolio is the last speaker at the end of the academic cycle:

Administrator Actor (PA)= {Si ($35 \le i \le 37$) ; Aj ($39 \le j \le 42$)}

The following model is then obtained:

$$\text{Actor} \quad \{ ; \sum_{i=29}^{i=37} S_i ; \sum_{j=29}^{j=44} A_j \}$$

| Channels | Actors | Level actors | Roles | Activities cycle University | | | Aggregations cycle University | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Début | Milieu | Fin | Début | Milieu | Fin |
| Study English | Student | Students of the 2nd round | study | Present | Activities | Preparation | Registration | follow | preparation |
| | | | | Learn | Courses | | Registration | | Diplomar |
| | | | | Follau | | Review | Preparation | Preparation | |
| | | | | Organize | Participate | Endstage | Stage | | |
| | | | | Prepare | | Expare | Concur | Participate | |
| | | | | | Prepare | | | Review | |
| | | | | | Review | | | | |
| | | | | | Stage | | | | |
| | Teacher | Research team leader Professor | Teaching Administer | Prepare | Prepare | Preparation | Preparation | Training | Stage |
| | | | | Organize | | Training | Training | Correct | Training |
| | | | | Farmer | Inform | | Follau | | |
| | | | | Inform | Training | | Regulation | | |
| | | | | Corrected | | | Stage | | |
| | Administrative | University President Accountant Manager | Administer Manage Adviser | Administrer | Prepare | Training | Budget | Preparation | Attestation |
| | | | | | Inform | | Registration | | Diplomar |
| | | | | Organize | | SID | | Participer | Statistic |
| | | | | Inform | | | Ressources | Training | |
| | | | | | | | Regulation | | |
| | | | | | | | Follau | | |

*Table 1 :* Role, Activities, Aggregation of Actors

Model of application is to justify the balance between all the activities of all actors and their aggregations at the end of a graduate level.

In this context, we present, as an application of indicators defined by the makers of the university and programmed by technical information system making the institution in[4] order to improve the performance of each actor.

To better understand this approach, we are using a graphic to show the equilibrium relationship between each actor and their activities at an undergraduate level [6] and its aggregation, taking into account the multiple observations to develop our model.
In the middle of our development that we present the following scheme which provide an overview of both synthetic and cross.

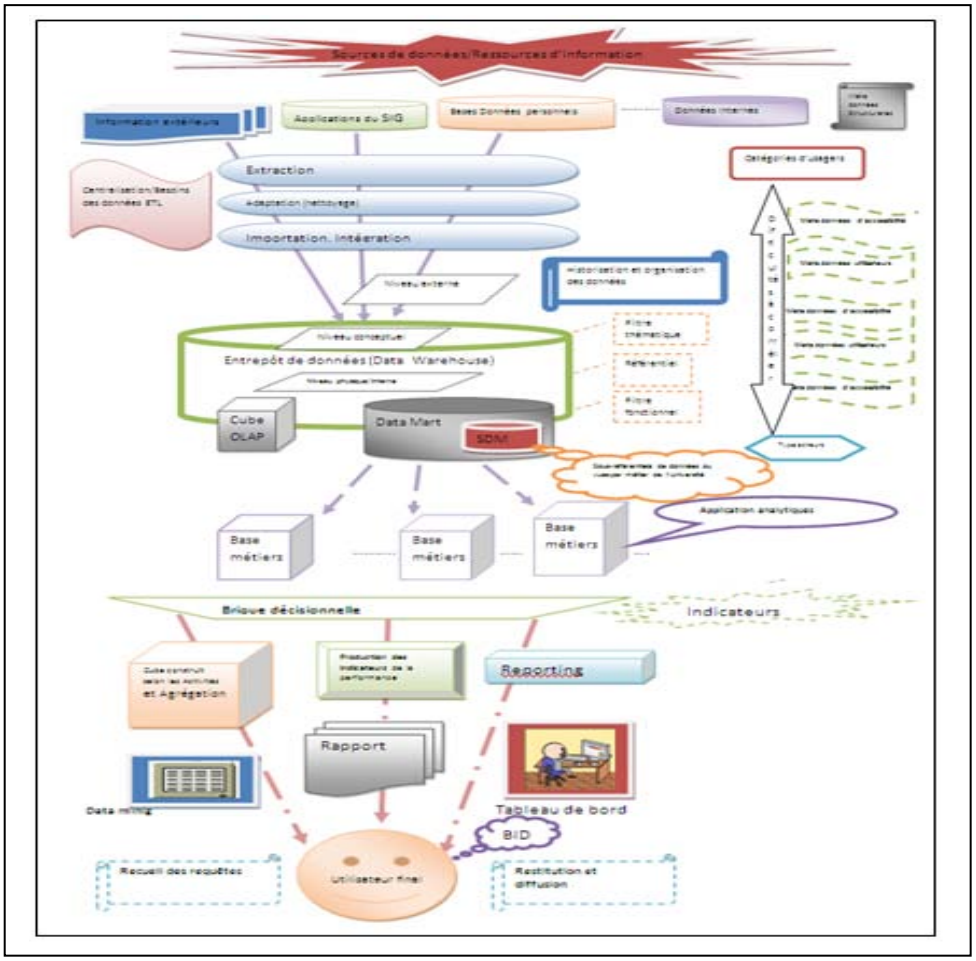



*Figure 1 :* Architecture of our model

*b) Data sources and feeding systems*

Remember that text documents are composed [6] of four main elements: the content (plain text), structure (logical organization of the text), context (meta data) and the layout (layout). The tables have a data structure formed of a series of data of the same type and the number of elements which is fixed a priori. They are both analytical tools and communication tools. Databases, when with them, are collections of data logically consistent with an intrinsic meaning. Each represents a 'mini world' or view data. They are managed by specific tools: the DBMS.

*c) Cleaning and monitoring of data quality*

For consistent results, we need every establishment of the university [8]do not neglect the quality of the original data, cleaning (Data Clearing) and management of reference data. This is called the Data Administration (DDA, design or data).

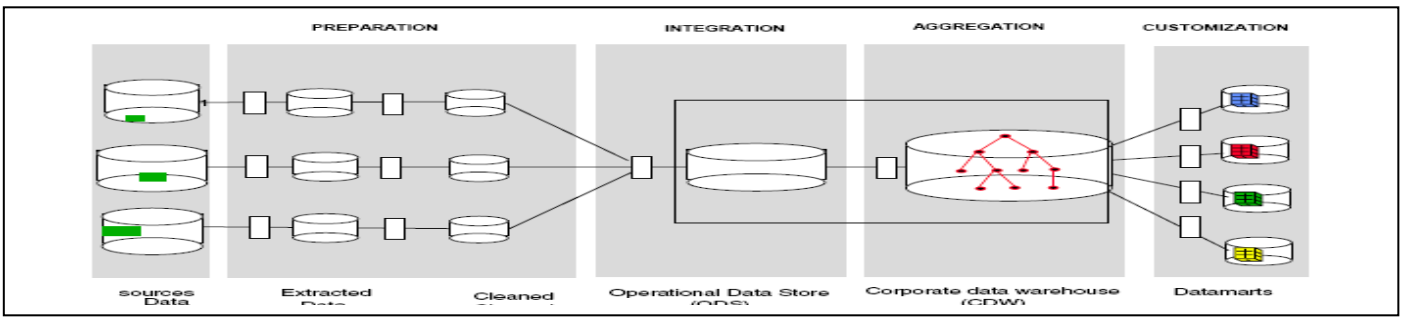



*Figure 2 :* Example of data integration, Source

The classic cases to verify data administration are redundancy, synonymy, duplicates (duplicates), the inconsistencies according to the origin or time, unreliability, failure to reuse and non-corporate knowledge. The problem of data quality [`7](inhomogeneous) has become central to the design of a data warehouse. The risks are to use data 'dirty', to make bad decisions, lack of relevant information, to misjudge the impact of a decision or fail to detect an abnormal situation. We must therefore ensure respect some essential criteria in order to have quality data.

- Completeness Expected values are present or not.
- Conformity Coherence, contradiction, format, syntax
- Correct Prediction, level of detail ...
- Credibility reputation, reliability ...
- Accessibility ,Availability of SI source, access rights, connectors ...
- Relevance, usefulness importance, value-added ...
- Freshness, news age, persistence / volatility ...
- Comprehension, interpretation understanding, meaning, origin ...

Emphasize that there is a difference between Data ware housing and Master Data Management. A data warehouse consolidates data from multiple sources to feed business intelligence applications, reporting and analysis. As MDM, Data Warehouse consolidates the data from source systems but conversely it is not intended to refer to these sources changed data. Only MDM ensures data synchronization between the repository and source systems / targets attached.

Any kind of information value-added used repeatedly in key processes and institutions shared by multiple applications can be included in the scope of MDM between the data and thus represents a candidate for Master Data Management (see the examples in Figure 7: Example the Repository Actor Single (UAR): a strategic MDM declination for the university).

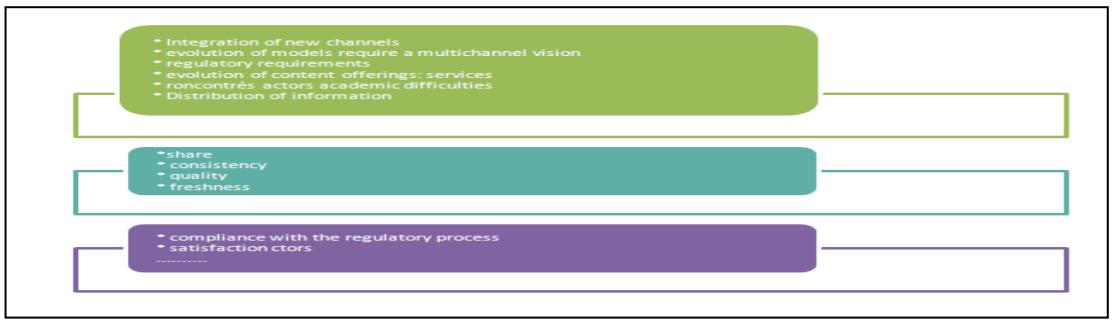



*Figure 3 :* Actor Single Repository (UAR) in Master Data Management

## IV. Conclusion

To implement this application, we went through three main phases. The first is the theoretical part that needs to have a model that is able to respond in an academic setting known for its complexity (different actors, the wealth of data, non-uniform data ...).

This requires a mathematical model defining simple relationships between the actors, their activities and their aggregations. The second phase focuses on collecting data and designing a multi-dimensional. The third used as 'data about data', or reference data in the context of data aggregation and facilitate cross-analyzes. These Meta data (accessibility) used to describe the data used in analysis and decision making as the exact definition of the data (semantics), the source data (date, origin), how they are calculated, aggregated (calculation rules), business rules relating thereto, the process of extraction, transformation and loading that has been implemented (ETL). In the case of intelligence, so there are tools for extracting and managing Meta data that are so flexible - that is to say, scalable - and play an important role in the establishment.

Data within data warehouses must be good quality, clean, but also described by meta data to be managed best by the Management System Database to provide the most relevant results possible.

## References Références Referencias

1. Oubedda L, Mir. A, Khalfaoui. M, "Modeling Resources Steering University ", acte SIL' 08, ENSA Marrakech 2008.
2. Oubedda L, Mir. A, Khalfaoui. M, «Human Resource Management of breast Moroccan Universities ", ENCG AGADIR 2009.
3. Olivier Bistorin, «Methods and tools for designing business process training systems ", these de Doctorat, University Paul Verlaine-Metz, décembre 2007
4. Olivier Bistorin, Thiband Monteiro, Claude Pourcel, " Process modeling of a training system ", Proceedings 1ère Conférence Internationale sur l'Ingénierie des Systémes de Formation, Carthagéne des Andes, Colombie, octobre 2007.
5. Peguiron F,David A,Thiery O, " Intelligence in academic settings including the user modeling ", IERA 2003,Nancy.

6. Bouaka N. et David A.," Model for the operation of a decision problem: a tool for decision support in a context of economic intelligence ", IERA 2003, Nancy.
7. Thiery O., Ducreau A., Bouaka N., David A., " Drive An Organization: Strategic Information To The Modeling of The User Application To The Field Of Hrm ", Grefige, 2004.
8. Samia Aitouche, Abdelghafour Kaanit And Kinza Mouss , Proposal of A Support Decision Support Using The Method Gimsi, For Better Reactivity In A Disturbed Environment, International Conference on Industrial Engineering and Manufacturing ICIEM'10, May, 9-10, 2010, Batna, Algeria.
9. Ann Chervenak, The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets, Journal of Network and Computer Applications Volume 23, Issue 3, July 2000, Pages 187–200

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# Performanace of Improved Minimum Spanning Tree Based on Clustering Technique

By P.Sampurnima, J Srinivas & Harikrishna
*NOVA college of engineering for women Vijayawada*

*Abstract -* Clustering technique is one of the most important and basic tool for data mining. Cluster algorithms have the ability to detect clusters with irregular boundaries, minimum spanning tree-based clustering algorithms have been widely used in practice. In such clustering algorithms, the search for nearest objects in the construction of minimum spanning trees is the main source of computation and the standard solutions take $O(N^2)$ time. In this paper, we present a fast minimum spanning tree-inspired clustering algorithm, which, by using an efficient implementation of the cut and the cycle property of the minimum spanning trees, can have much better performance than $O(N^2)$.

*Keywords :* *Clustering, graph algorithms, Minimum spanning tree, Divisive hierarchical clustering algorithm.*

*GJCST-C Classification:* *E.1*


*Strictly as per the compliance and regulations of:*

# Performanace of Improved Minimum Spanning Tree Based on Clustering Technique

P.Sampurnima [α], J Srinivas [σ] & Harikrishna [ρ]

*Abstract* - Clustering technique is one of the most important and basic tool for data mining. Cluster algorithms have the ability to detect clusters with irregular boundaries, minimum spanning tree-based clustering algorithms have been widely used in practice. In such clustering algorithms, the search for nearest objects in the construction of minimum spanning trees is the main source of computation and the standard solutions take $O(N^2)$ time. In this paper, we present a fast minimum spanning tree-inspired clustering algorithm, which, by using an efficient implementation of the cut and the cycle property of the minimum spanning trees, can have much better performance than $O(N^2)$.

*Keywords : Clustering, graph algorithms, Minimum spanning tree, Divisive hierarchical clustering algorithm.*

## I. Introduction

Given a set of data points and a distance measure, clustering is the process of partitioning the data set into subsets, called clusters, so that the data in each subset share some properties in common. Usually, the common properties are quantitatively evaluated by some measures of the optimality such as minimum intracluster distance or maximum intercluster distance, etc.

Clustering, as an important tool to explore the hidden structures of modern large databases, has been extensively studied and many algorithms have been proposed in the literature. Because of the huge variety of the problems and data distributions, different techniques, such as hierarchical, partitional, and density- and model-based approaches, have been developed and no techniques are completely satisfactory for all the cases.

For example, some classical algorithms rely on either the idea of grouping the data points around some "centers" or the idea of separating the data points using some regular geometric curves such as hyper planes. As a result, they generally do not work well when the boundaries of the clusters are irregular. Sufficient empirical evidences have shown that a minimum spanning tree representation is quite invariant to the detailed geometric changes in clusters' boundaries.

*Author α : M.Tech Student Department of Computer Science and Engineering NOVA college of engineering for women Vijayawada. E-mail : psampurnima@gmail.com*
*Author σ : Associate professor Department of Computer Science and Engineering NOVA college of engineering for women Vijayawada.*
*Author ρ : HOD Department of Computer Science and Engineering NOVA college of engineering for women Vijayawada.*

Therefore, the shape of a cluster has little impact on the performance of minimum spanning tree (MST)-based clustering algorithms, which allows us to overcome many of the problems faced by the classical clustering algorithms.

## II. An MST-Inspired Clustering Algorithm

Although MST-based clustering algorithms have been widely studied, in this section, we describe a new divide and- conquer scheme to facilitate efficient MST-based clustering in modern large databases. Basically, it follows the idea of the "Reverse Delete" algorithm. Before proceeding, we give a formal proof of its correctness.

**Theorem 1**. *Given a connected, edge-weighted graph, the "Reverse Delete" algorithm produces an MST.*

**Proof**. First, we show that the algorithm produces a spanning tree. This is because the graph is given connected at the beginning and, when deleting edges in the non increasing order, only the most expensive edge in any cycle is deleted, which does eliminate the cycles but not disconnect the graph, resulting in a connected graph containing no cycle at the end. To show that the obtained spanning tree is an MST, consider any edge removed by the algorithm. It can be observed that it must lie on some cycle (otherwise removing it would disconnect the graph) and it must be the most expensive one on it (otherwise retaining it would violate the cycle property). Hence, the "Reverse Delete" algorithm produces an MST.

For our MST-inspired clustering problem, it is straightforward that n=N and m=N (N-1)/2, and the standard solution has O ($N^2$logN) time complexity. However, m=O ($N^2$) is not always necessary. The design of a more efficient scheme is motivated by the following observations. First, the MST-based clustering algorithms can be more efficient if the longest edges of an MST can be identified quickly before most of the shorter ones are found. This is because, for some MST-based clustering problems, if we can find the longest edges in the MST very quickly, there is no need to compute the exact distance values associated with the shorter ones.

Second, for other MST-based clustering algorithms, if the longest edges can be found quickly,

the Prim's algorithm can be more efficiently applied to each individual size-reduced cluster. This divide-and-conquer approach will allow us to save the number of distance computations tremendously.

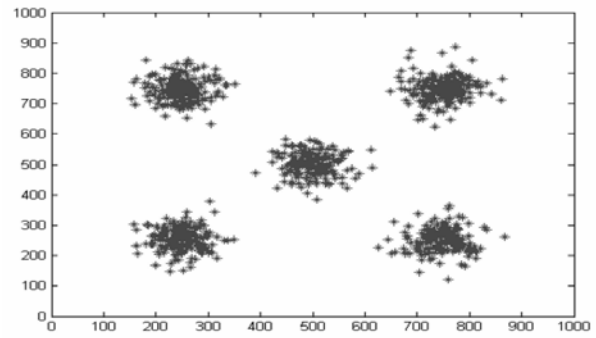


*Figure 1 :* A two-dimensional five-cluster data set

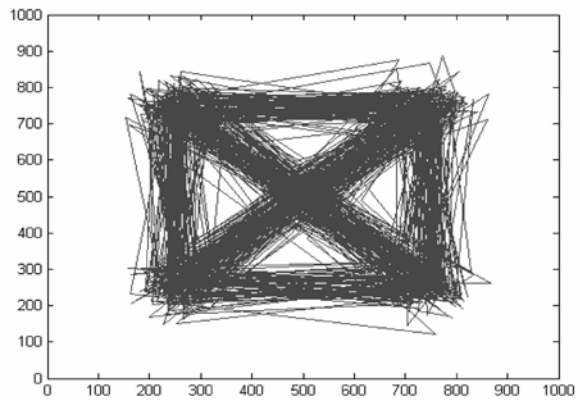


*Figure 2 :* Its spanning tree after the sequential initialization

*a) A Simple Idea*

Given a set of S-dimensional data, i.e., each data item is a point in the s-dimensional space, there exists a distance between every pair of the data items. To compute all the pairwise distances, the time complexity is O ($sN^2$), where N is the number of data items in the set. Suppose at the beginning, each data item is initialized to have a distance with another data item in the set. For example, since the data items are always stored sequentially, each data item can be assigned the distance between itself and its immediate predecessor—called a forward initialized tree—or successor—called a backward initialized tree. These initial distances, whatever they are, provide an upper bound for the distance of each data item to its neighbor in the MST.

In the implementation, the data structure consists of two arrays:

1. Distance array
2. Index array.

**Distance Array:**

The distance array is used to record the distance of each data point to some other data point in the sequentially stored data set.

**Index Array:**

The index array records the index of the data item at the other end of the distance in the distance array.

According to the working principle of the MST-based clustering algorithms, a database can be split into partitions by identifying and removing the longest inconsistent edges in the tree. Based on this finding, after the sequential initialization, we can do a search in the distance array (i.e., the current spanning tree) for the edge that has the largest distance value, which we call the potential longest edge candidate. Then the next step is to check whether or not there exists another edge with a smaller weight crossing the two partitions connected now by this potential longest edge candidate. If the result shows that this potential longest edge candidate is the edge with the smallest weight crossing the two partitions, we find the longest edge in the current spanning tree (ST) that agrees with the longest edge in the corresponding MST. Otherwise, we record the update and start another round of the potential longest edge candidate identification in the current ST.

It can be seen that the quality of our fast algorithm depends on the quality of the initialization to quickly expose the longest edges. Though the sequential initialization gives us a spanning tree, when the data are randomly stored, such a tree could be far from being optimal. This situation can be illustrated by a two-dimensional five cluster data set shown in Figure. 1. Shown in Figure.2 is its spanning tree after the sequential initialization (SI). In order to quickly identify the longest edges, we propose to follow the sequential initialization by multiple runs of a recursive procedure known as the divisive hierarchical clustering algorithm (DHCA).

*b) Divisive Hierarchical Clustering Algorithm*

Essentially, given a data set, the DHCA starts with k randomly selected centers and then assigns each point to its closest center, creating k partitions. At each stage in the iteration, for each of these k partitions, DHCA recursively selects k random centers and continues the clustering process within each partition to form at most $k^n$ partitions for the $n^{th}$ stage. In our implementation, the procedure continues until the number of elements in a partition is below k+2, at which time, the distance of each data item to other data items in that partition can be updated with a smaller value by a brute-force nearest neighbor search. Such a strategy ensures that points that are close to each other in space are likely to be collocated in the same partition. However, because any data point in a partition is closer to its cluster center (not its nearest neighbor) than to the center of any other partition (in case, the data point is equidistant to two or more centers, the partition to which the data point belongs is a random one), the data points in the cluster's boundaries can be misclassified into a

wrong partition. Fortunately, such possibilities can be greatly reduced by multiple runs of DHCA. To summarize, we believe that the advantage of DHCA is that, after multiple runs, each point will be very close to its true nearest neighbor in the data set.

To demonstrate this fact, one can think of this problem as a set of independent Bernoulli trials where one keeps running DHCA and classifying each data point to its closest randomly selected cluster center at each stage of the process, until it succeeds (i.e., it hits its nearest neighbor, or at least, its approximate nearest neighbor). Let p be the probability that a random data point hits its nearest neighbor. Let Y be the random variable representing the number of trials needed for a random data point to hit its nearest neighbor. The probability of obtaining a success on trial y is given by

$$P(Y=y)=q^{y-1}p,$$

Where q=1-p denotes the probability that a failure occurs. The relationship between p and P(Y=y) is plotted in from it, we can see that for a randomized process (i.e., p=0.5), at most 50 DHCAs are enough for most of the data points to meet their nearest neighbor.
For our purpose, after the sequential initialization, a spanning tree is constructed and each data item in the tree has already had a distance. During the divisive hierarchical clustering process, each data item will have multiple distance computations.

*c) MST-Inspired Clustering Algorithm*

Based on the methodology presented in the previous two sections, given a loose estimate of minimum and maximum numbers of data items in each cluster, an iterative approach for our MST-inspired clustering algorithm can be summarized in the following:

1. Start with a spanning tree built by the SI.
2. Calculate the mean and the standard deviation of the edge weights in the current distance array and use their sum as the threshold. Partially refine the spanning tree by running our DHCA multiple times until the percentage threshold difference between two consecutively updated distance arrays is below $10^{-6}$.
3. Identify and verify the longest edge candidates by running MDHCA until two consecutive longest edge distances converge to the same value at the same places.
4. Remove this longest edge.
5. If the number of clusters in the data set is preset or if the difference between two consecutively removed longest edges has a percentage decrement larger than 50 percent of the previous one, we stop. Otherwise go to Step 3.

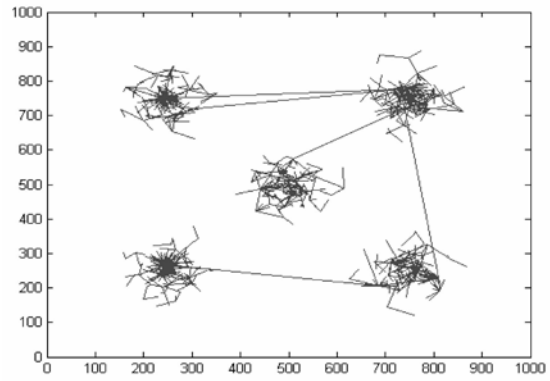



*Figure 3 :* Updated spanning tree using DHCAs

We stop Step 2 when the percentage threshold difference between two consecutive pruning thresholds, i.e., its percentage decrement, is below a threshold, say $10^{-6}$ in our implementation, because further DHCA-based distance upper bound updates will not bring us more gains which are worth the overhead of the DHCA. The spanning tree after the DHCA updates for the one shown in Figure. 2 is manifested in Figure. 3.

The terminating condition presented in the above MST inspired clustering algorithm is under the assumptions that the clusters are well separated and there are no outstanding outliers. However, in many real-world problems, the clusters are not always well separated and noise in the form of outliers often exists. For these cases, some of the longest edges do not correspond to any cluster separations or breaks but are associated with the outliers for such cases, we propose terminating

Conditions of that are adaption results from LM algorithm and the MSDR algorithm.

The advantage of the LM algorithm is the avoidance of unnecessary large number of small clusters. The advantage of the MSDR algorithm is that it can find the optimal cluster separations, particularly for cases where there exist some unknown hidden structures in the data set.

The adapted LM algorithm is the following:

1. Get a loose estimate of the maximum and Minimum number of data points for each cluster.
2. Always cut the largest subcluster and cut an edge only when the sizes of both clusters resulted by cutting that edge are larger than the minimum number of data points.
3. Terminates when the size of the largest cluster becomes smaller than the estimated maximum number of data points.

**The adapted LM algorithm is the following:**

1. Calculate the mean and the standard deviation of the edge weights in the distance array and use their sum as the threshold. 'Remove the longest edge that is larger than the threshold and that links either

a single point or a very small number of data points to the MST.

2. Continue Steps 1 and 2 until the edge is reached, by removing which, two large groups will form from the single largest group before that edge is cut.
3. Apply the MSDR algorithm on the denoised MST
4. Assign the removed data points the same cluster Label as their nearest neighbor's.

*d) Time Complexity Analysis*

From the description in the previous sections, it can be seen that our algorithm mainly consists of two phases. The first phase includes the sequential initialization and the DHCA spanning tree updating, and the second phase uses the MDHCA to locate the longest edges and partitions the obtained approximate minimum spanning tree to form sensible clusters. We expect the original DHCAs (i.e., no thresholding involved) to scale as O (fN logN), where f denotes the number of DHCA constructed before the terminating condition is satisfied. Since in our implementation, at each step of the spanning tree updating using the DHCA, before we assign a data item to a cluster center, if its current distance upper bound is smaller than the threshold (i.e., the sum of the mean and one standard deviation of the tree edge weights), we ignore it, the time complexity is actually (d(xN)log(xN)), where x is between 0 and 1.

Therefore, as long as x is small enough, the time complexity could be near linear on average. Though its worst time complexity could be $O(N^2)$, the average time complexity of the second phase is O(eN logN), where e denotes the number of MDHCA constructed before the terminating condition is satisfied. Since, on average, the number of longest edges is much smaller than the data set size N, as long as the spanning tree constructed in the first phase is very close to the true minimum spanning tree, we expect our MST-inspired algorithm to scale as O(log N).

*e) Pseudocode for Our Clustering Algorithm*

The implementation of the DHCA in our approach is through the design of a C++ data structure called Node. The Node data structure has several member variables that remember the indexes of the subset of the data items that are clustered into it from its parent level, the indexes of its randomly chosen k cluster centers from its own set for its descendants, and a main member function that generates k new nodes by clustering its own set into k sub clusters. The outputs of the Node data structure are at most k new Nodes as the descendents of the current one.

The divisive hierarchical clustering process starts with creating a Node instance, called the top Node. This top Node has every data item in the data set as its samples. From these samples, this top Node randomly chooses k data points as its clustering centers and assigns each sample to its nearest one, generating k data subsets in the form of k Nodes. Only when the number of samples in a Node is larger than a predefined cluster size will that Node be pushed to the back of the topNode, forming an array of Nodes. This process continues recursively. With the new Nodes being generated on the fly and pushed to the back of the Node array, they will be processed in order until no new Nodes are generated and the end of the existing Node array is reached.

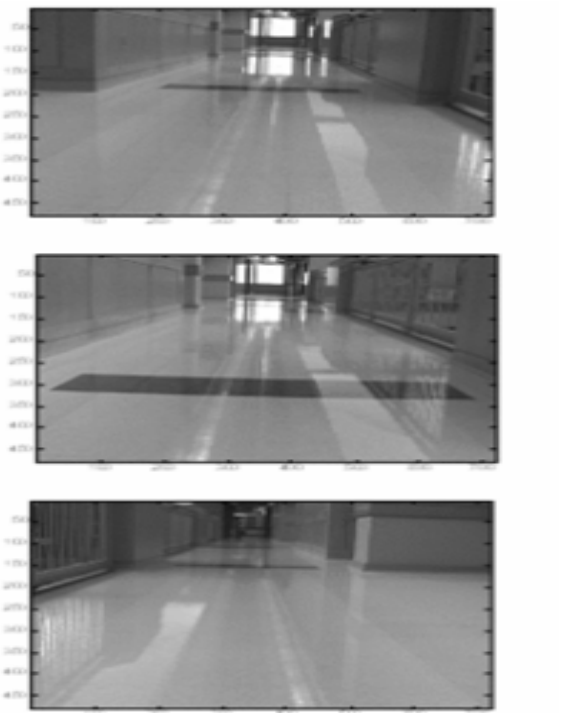
Totally, we need two variants of the DHCA procedure, DHCA for our spanning tree updating, and MDHCA for the cycle property implementation. The DHCA_ST procedure is given in Table 1. The DHCA_CYC procedure is the same as DHCA_ST except for the ways to choose cluster centers and will not be repeated here.

| Procedure Name | DHCA_ST |
|---|---|
| Input: | |
| Dist_st, edge_st array | The ST distance array and index |
| Dist_knn,edge_knn Nearest | The auxiliary arrays to remember k- |
| | Neighbours(kNN) for each data item |
| kNN | The no.of NNs of a data item |
| nodeArray | An array of the Node structures |
| currentNode | The current Node in the Node array |
| k | The number of clusters at each step |
| data | The input data set |
| maxclustersize | The maximum size of each clusters |
| threshold | The value used to filter |
| Output: | |
| Updated dist_st,edge_st,dist_knn,edge_knn and new generated<=k | |
| Nodes which are pushed to the back of nodeArray | |

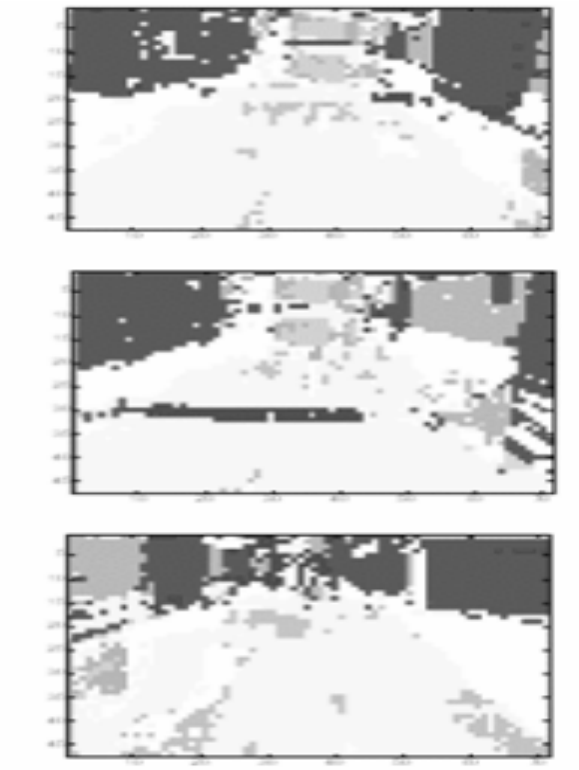
```
Begin
    Randomly select k centers from sampleNumbers of
    currentNode;
    Generate k newNodes;
    For each sample i in sampleNumbers of currentNode
that is not
    a center
    {
        find its nearest center j out of k;
if((dist_st[i]<distance(i , j)&&
                (sampleNumber[i]>sampleNumber[j]))
        {
            Update dist_st, edge_st;
        }
         if(dist_knn[i].max>distance(i,j))
        {
            Update dist_knn, edge_knn;
        }
        if(dist_st[i]>threshold)
        {
     assign sampleNumbers[i] to groups of center j;
   }
 }
   for each newNode j=1 to k
   {
        if(newNode[j].sampleNumbers.size()>maxclustersize)
```

```
        {
            Push newNode[j] to the end of nodeAyyay;
        }
    }
End
```

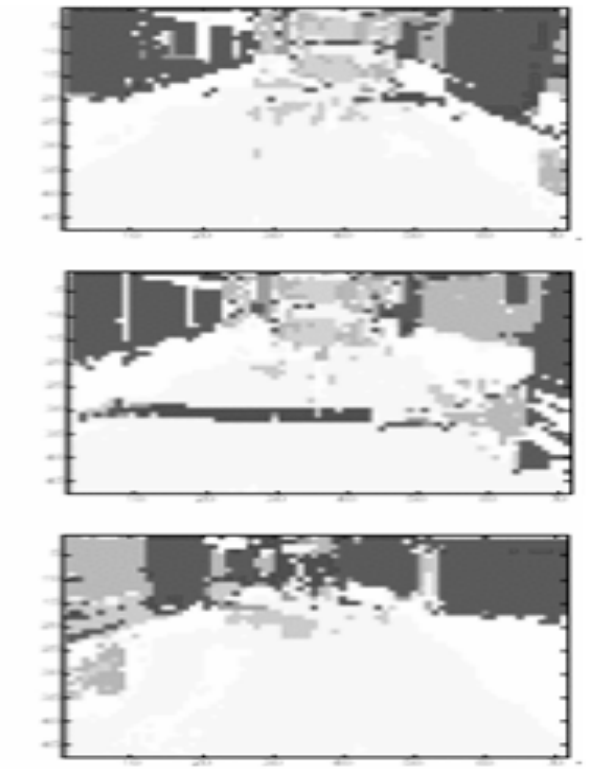
dist_knn[i].max is the kNNth nearest neighbor of data item i.



(a) Original image



(b) On our MST



(c) On Prim's MST

*Figure (a-c) :* Results of the adapted MSDR algorithm

We conducted extensive experiments to evaluate our algorithm against the k-means algorithm and two other state-of-the-art MST-based clustering algorithms on three standard synthetic data sets and two real data sets. The experimental results show that our proposed MST inspired clustering algorithm is very effective and stable when applied to various clustering problems. Since there often exist some structures in the data sets, our algorithm does not necessarily require but can automatically detrmine the desired number of clusters by itself.

In the future, we will further study the rich properties of the existing MST algorithms and adapt our proposed MST inspired clustering algorithm to more general and larger data sets, particularly when the whole data set cannot fit into the main memory.

## III. Conclusion

As a graph partition technique, the MST-based clustering algorithms are of growing importance in detecting the irregular boundaries. A central problem in such clustering algorithms is the classic quadratic time complexity on the construction of an MST. In this paper, we present a more efficient method that can quickly identify the longest edges in an MST so as to save some computations. Our contribution is the design of a new MST-inspired clustering algorithm for large data sets (however, without any specific requirements on the

distance measure used) by utilizing a DHCA in an efficient implementation of the cut and the cycle property.

## References Références Referencias


1. I. Katriel, P. Sanders, and J.L. Traff, "A Practical Minimum Spanning Tree Algorithm Using the Cycle Property," Proc. 11th European Symp. Algorithms (ESA '03), vol. 2832, pp. 679-690, 2003.
2. C.T. Zahn, "Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters," IEEE Trans. Computers, vol. 20, no. 1, pp. 68-86, Jan. 1971.
3. A. Vathy-Fogarassy, A. Kiss, and J. Abonyi, "Hybrid Minimal Spanning Tree and Mixture of Gaussians Based Clustering Algorithm," Foundations of Information and Knowledge Systems, pp. 313-330, Springer, 2006.
4. O. Grygorash, Y. Zhou, and Z. Jorgensen, "Minimum Spanning Tree-Based Clustering Algorithms," Proc. IEEE Int'l Conf. Tools with Artificial Intelligence, pp. 73-81, 2006.
5. R.C. Gonzalez and P. Wintz, Digital Image Processing, second ed. Addison-Wesley, 1987.
6. Y. Xu, V. Olman, and D. Xu, "Clustering Gene Expression Data Using a Graph-Theoretic Approach: An Application of Minimum Spanning Trees," Bioinformatics, vol. 18, no. 4, pp. 536-545, 2002.
7. J. Kleinberg and E. Tardos, *Algorithm Design,* pp.142-149. Pearson-Addison Wesley, 2005.
8. A. Ghoting, S. Parthasarathy, and M.E. Otey, "Fast Mining of Distance-Based Outliers in High Dimensional Data Sets," *Proc. SIAM Int'l Conf. Data Mining (SDM),* vol. 16, no. 3, pp.349-364, 2006.
9. I. Katriel, P. Sanders, and J.L. Traff, "A Practical Minimum Spanning Tree Algorithm Using the Cycle Property," *Proc. 11th European Symp. Algorithms (ESA '03),* vol. 2832, pp.679-690, 2003.
10. J. Lin, D. Ye, C. Chen, and M. Gao, "Minimum Spanning Tree-Based Spatial Outlier Mining and Its Applications," *Lecture Notes in Computer Science,* vol. 5009/2008, pp.508-515, Springer-Verlag, 2008.

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# System Design Principles – Reuse: Online Attendance System

By Ch V M K Hari, K S V Krishna Srikanth & N S S S Girish Kumar
*Institute of Technology GITAM University*

*Abstract -* Software engineering is an engineering approach for software development. In order to develop large software several phases has to be followed by the developer to achieve good quality software; cost effectively. System Design is the most important activity in software development which reflects reusability. System Design specifies what a new or modified system is going to do. To achieve good quality software, the primary characteristics of neat module decomposition are low coupling {data coupling}, high cohesion {functional cohesion} and top-down approach has to be followed. We applied these principles on developing Online Attendance System and observed reusability of code. The system has been successfully tested in our institute. Effective design principles always lead to an effective reusability which in turn benefited with Return on Investment (ROI).

*Keyterms : Design, Coupling, Cohesion, Reusability, Online Attendance System.*

*GJCST-C Classification: D.2.0*


*Strictly as per the compliance and regulations of:*

# System Design Principles – Reuse: Online Attendance System

Ch V M K Hari [α], K S V Krishna Srikanth [σ] & N S S S Girish Kumar [ρ]

*Abstract -* Software engineering is an engineering approach for software development. In order to develop large software several phases has to be followed by the developer to achieve good quality software; cost effectively. System Design is the most important activity in software development which reflects reusability. System Design specifies what a new or modified system is going to do. To achieve good quality software, the primary characteristics of neat module decomposition are low coupling {data coupling}, high cohesion {functional cohesion} and top-down approach has to be followed. We applied these principles on developing Online Attendance System and observed reusability of code. The system has been successfully tested in our institute. Effective design principles always lead to an effective reusability which in turn benefited with Return on Investment (ROI).

*Keyterms : Design, Coupling, Cohesion, Reusability, Online Attendance System.*

## I. Introduction

***Software engineering*** is the application of a *systematic, disciplined, quantifiable approach* to the development, operation, maintenance and retirement of software [1]. The use of the term *systematic approach* implies that methodologies are used for developing software. Software engineering includes process, managing techniques, technical methods, and use of tools. Software engineering deals with the problem of developing 'large' software. Software engineering helps to reduce the programming complexity. Software engineering principles or methodologies use two important techniques (1) abstraction and (2) decomposition to reduce problem complexity. The principle of *abstraction* implies that a problem can be simplified by omitting irrelevant details. The principle of *decomposition* states that a complex problem is divided into several smaller problems and then smaller problems are solved one by one.

The ***goal*** of software engineering is to develop high quality software with low cost i.e., within time and budget constraints. New software systems are built from the old ones and all must interoperate and cooperate with each other.

*Software* is meant to solve some problem of the *client* (the people whose needs are to be satisfied by the software). The problem is to develop software systematically to satisfy the needs of clients [2]. There are some factors for basic problem which affect the approaches selected to solve the problem and these factors are the primary forces that drive the progress and development in the field of software engineering.

***Software Development Life Cycle*** activities will have several stages where in one identifies the problem to be solved, develop a design, writes the code and so on. Software life cycle defines entry and exit criteria for every phase. A phase can start only if its phase-entry criteria have been satisfied. Without software life cycle it becomes difficult for software project managers to monitor the progress of the project. The software life-cycle [5, 6] consists of: feasibility study, requirements analysis, design, construction, testing (validation), deployment and maintenance. The development process tends to run iteratively through these phases rather than linearly.

Upon successfully demonstrating the feasibility of a project, the requirements analysis begins. The design starts after the requirements analysis is complete, and coding begins after the design is complete. Once the programming is completed, the code is integrated and testing is done. Upon successful completion of testing, the system is installed. After this, the regular operation and maintenance of the system takes place.

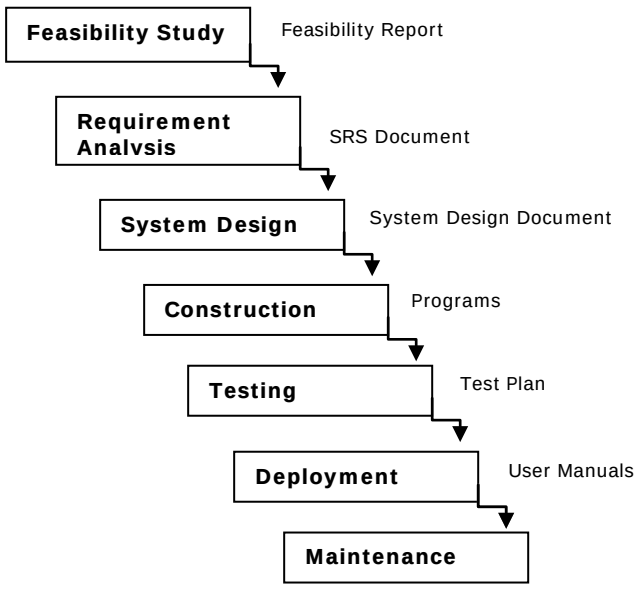



*Fig. 1 :* Software Life Cycle

*Author α : Department of IT GITAM Institute of Technology GITAM University. E-mail : kurmahari@gmail.com*
*Author σ : Department of IT GITAM Institute of Technology GITAM University. E-mail : ksvksrikanth@gmail.com*
*Author ρ : Department of IT GITAM Institute of Technology GITAM University. E-mail : girishnsss@hotmail.com*

***System Design*** is the process of defining architecture, modules, interfaces and data for a system to specify the requirements of proposed system. The design of a system is essentially a blueprint or a plan for a solution for the system. It specifies what components are needed for the system, their behavior and how they should be interconnected [2, 3]. The design activity begins when the SRS document is available. During design we further refine the architecture. The *goal* is to transform the requirements specified in the document into a structure that is suitable for implementation in some programming language.

Design focuses on the *module view*. A module of a system can be considered a system, with its own modules. A system as set of modules with defined behavior interacts with each other in a defined manner may produce some behavior or services for its environment. A good software design can be arrived infrequently by using single step procedure but rather through several iterations through a series of steps.

The design characteristics include the following:

1. *Top-down approach*: A top-down design approach starts by identifying the major components of the system, decomposing them into their lower-level components and iterating until the desired level of detail is achieved.
2. *Coupling*: Coupling between modules is the strength of interconnections between modules or a measure of the degree of interdependence between two modules. Classification of different types of coupling will help to estimate the degree between modules. The classification starts from *low to high*:
   a. *Data Coupling*: Two modules are said to be data coupled, if communication of modules is through a parameter.
   b. *Stamp Coupling*: Two modules are said to be stamp coupled provided, if communication of modules is through composite parameters.
   c. *Control Coupling*: Two modules are said to be control coupled when one module controls the execution behavior of another module.
   d. *Common Coupling*: Two modules are said to be common coupled, if they share data through some global data items.
   e. *Content Coupling*: Two modules are said to be content coupled provided one module refers to a piece of information defined in other module.
3. *Cohesion*: Cohesion of a module represents how the internal elements of the module are tightly bound to one another. Cohesion of a module gives the designer an idea about the different functions in it and how they belong together in the same module.

The different classes of cohesion that a module may possess from *high to low* are:

   a. *Functional Cohesion*: It is the highest. In this, all the elements of the module contribute to achieve a single function.
   b. *Sequential Cohesion*: When the elements are together in a module, the output of one element forms the input to another.
   c. *Communication Cohesion*: In this, the elements are together and they operate on the same input or output data.
   d. *Procedural Cohesion*: In this, a module contains number of functions in which certain sequences have to be carried out for achieving an objective.
   e. *Temporal Cohesion*: In this, elements of the module are executed in the same time span.
   f. *Logical Cohesion*: It occurs if all the elements of a module have some logical relationship between them and perform similar operations.
   g. *Coincidental Cohesion*: It is the lowest. A module is said to be coincidental cohesive, if it performs set of tasks that relate to each other very loosely and functions put in are out of pure coincidence without any design.
4. *Span of Control*: Number of subordinate modules under given modules.
5. *Size*: Indicates the overall code size.
6. *Sharability of modules:* Identify the commonalities with the program.

A module with high cohesion and low coupling is said to be functionally independent of other modules i.e., cohesive module performs a single task or function and has minimal interaction with other modules. Functional independence is a sign to a good design as it reduces error propagation; reuse of a module becomes possible; and the complexity of the design is reduced.

## II. Proposed Design Principles

### *a) Top-down Approach*

The top-down approach starts from the higher levels and decompose downwards to lower levels, identifying connections/collaborations at every stage. The top-down approach (also called stepwise design) starts from high level design description and break it down into different sub design or systems to gain observation into its composed sub systems. This gives good understanding of the problem. This starts with system specifications. It specifies/defines a module to implement the specifications. It specifies subordinate modules and then treats each specified module as the problem. Top-down design methods result in some form of elaboration where we reach to a level when no more refinement is needed and the design can be implemented directly. The top-down approach published by many researchers is found to be extremely useful for design. Most design methodologies are based on the top-down approach.

*b) Coupling*

Coupling is a measure of the relationship (i.e., dependency) between two modules. Coupling measures the degree to which each program module depends on each one of the other modules [3, 7]. Coupling is a measure of interconnection among modules in a program structure. Coupling captures the notion of dependence. Coupling tries to capture how strongly modules are interconnected. Coupling depends on type of information flow.

If two modules interchange large amounts of data, then they are highly independent. The degree of coupling depends on their interface complexity. The interface complexity is determined by number of types of parameters that are interchanged while invoking the functions of the module. Low coupling is often a note of good design as it supports goals of high readability and maintainability.

*Data Coupling :* Data coupling occurs between two modules when data are passed by parameters using a simple argument list and every item in the list is used. An example is an elementary data item (which should be problem related) passed as parameter between two modules. Example can be an integer, a character, a string etc.

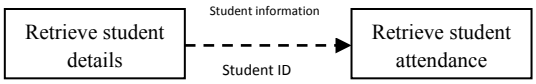



*Fig. 2 :* illustrates the module that retrieves student information using student id

*Strengths* of data coupling are: a module sees only the data elements it requires. *Weakness* of data coupling is, a module can be difficult to maintain if many data elements are passed.

*c) Cohesion*

Cohesion considers maximizing relationship between elements of same module. Cohesion is the measure of functional strength of a module [4, 7]. High cohesion is a mark for associating desirable features of software including robustness, reliability, reusability and understandability.

*Functional Cohesion :* Functional cohesion is the strongest cohesion. In a functionally bound module, all the elements of the module are related to performing a single function. By function, we mean modules accomplishing a single goal. A functionally cohesive module performs one and only one problem related task. Functionally cohesive modules may be simple and perform one task, such as Read Customer Record.

For example, a module containing all the functions required to manage employees' pay-roll exhibits functional cohesion. When a module exhibits functional cohesion, then we could be able to describe it using a single sentence.

*Strengths* of functional cohesion are functionally cohesive modules are good candidates for re-use, systems built with functionally cohesive modules are easily understood and, therefore, easier to maintain. *Weakness* of functional cohesion is designers should guard against designing over-simplified modules or methods. If functional cohesion is taken too far in structured design, the system design consists of hundreds of modules comprised of two or three lines of code.

*d) Span of Control*

Span of Control is a measure of the number of modules directly controlled by a higher-level routine. It is the number of sub-modules under a module. The number of subordinate modules for a project can be in 3 or 5 modules or levels.

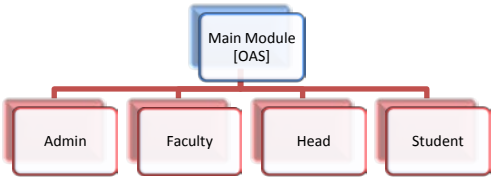



*Fig. 3 :* illustrates the span of control for OAS main module as 4

*e) Size*

The size indicates the overall code size. Example, 50 lines of code.

*f) Sharability of modules*

Identify the commonalities with the program and identify reuse components before development.

## III. Case Study: Online Attendance System

**Online Attendance System** is software developed for daily student attendance in colleges and institutes. It facilitates to access the attendance information of a particular student in a particular class. The attendance information is sorted by the system, which will be provided by the faculty for a particular class. This system will also help in evaluating attendance eligibility criteria of a student [8]. This helps faculty marks student's class attendance easily and quickly.

The **purpose** of developing online attendance system is to produce a computerized solution to manual attendance procedure as manual process is time consuming & mostly not effective and another purpose is to generate the report automatically at the end of the session, and also at end of the academic.

This attendance system lets faculty and administration do the following easily:

- Prints class attendance sheets when needed.
- Faculty checks student attendance instantly.

- The Head of the department or institution can check the attendance monthly, date-wise, at end of the academic and also check the summarized attendance of particular student when required.

The **scope** of the project is this system is intended for engineering institutions which is a web application. In this system there are mainly four entities; admin, faculty, HOD and student. The admin is the main secretary of the system who enters the data into the system.

*a) Admin Module*

The first entity is Admin who is the powerful in the system. The admin has the possibility to add new students, new faculty, new subjects and new courses; edit and delete the existing ones. The admin can update details of multiple students where students can be promoted to next class or semester easily.

The admin can allocate subjects to faculty for a particular class, so that the attendance registers are created dynamically. If allocation for a subject is updated with new faculty, the created registers are transferred to that particular faculty. The admin can view the allocations and registers as and when required.

For faculties to take attendance of an academic, the admin has the possibility to set attendance start date and end date. For new academic, the existing dates are deleted and added again.

To send the attendance report of the student to their parents, the admin gets the absentees' details of all the classes on the given date and details of the students whose attendance is less than the required percentage for the given month.

*b) Faculty Module*

The second entity is Faculty who plays an important role in the system. The Faculty takes the attendance of the students in this module.

The Faculty when selects the date, period and the register, navigates to the selected register where the attendance is taken to the students and saves the details which cannot be updated. When selecting the date, the faculty can give attendance from the mentioned start date and end date added by the admin.

To view the register of the subjects handled by the Faculty, the faculty selects the register, and views the attendance details of all the students till date. The Faculty can also view the overall number of classes, total attended classes and the percentage.

*c) HOD Module*

The HOD module also plays a major role in the system. This module is exceptionally used to view the attendance reports of the student. The user of this module can be Head of the Department or Head of the Institution.

The Reports include:

1) *Subject-Wise Report*, where faculty is selected, and obtains the list of registers of the particular faculty. This is similar to the register view of the faculty module.
2) *Student-Wise Report,* where the student ID is given to get the cumulative attendance report of the student for all subjects where report contains total classes, attended classes and the percentage of the attendance.
3) *Date-Wise Report,* where particular class and date are selected to view the attendance of the given date for all periods.
4) *Monthly Report,* where particular class and month are selected to view the attendance report in a cumulative format for all subjects with total classes, attended classes and percentage.
5) *Semester-Wise Report*, where particular class is selected to view the overall attendance report of the semester or academic up to the attendance end date. This report will be generated only after the attendance end date. This report is also similar to the cumulative format of monthly report.

*d) Student Module*

The final entity is the Student where he/she can get details that include the profile, and attendance report. The attendance details contain the total classes, attended classes. To get the percentage of the attendance to the classes attended, it is generated only after the attendance end date. This is also similar to Student-Wise report in the HOD module.

The four entities or modules mentioned above can access the features given to them in the system, where they have to login with their own username and password.

The Online Attendance System applies the proposed design principles, where the operations performed by each entity satisfy the characteristics of design low coupling and high cohesion which provides an efficient system for ease of usage. The top-down approach when used illustrated the software engineering principle decomposition where the system is broken into different sub-modules so that the module required to start is easily identified and implemented from that level. When applying system design principles to the application, application quality has improved and is operated at high level of efficiency and the requirements of the system specified are satisfied.

## IV. Screen Shots


*Fig. 4 :* Online Attendance System Home Screen


*Fig. 5 :* Faculty Module - Opening Attendance Register

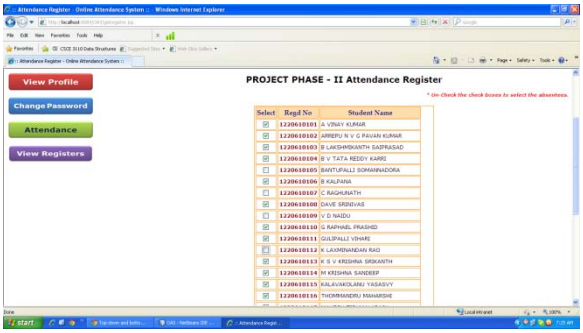



*Fig. 6 :* Faculty Module - Taking student Attendance for the selected Register

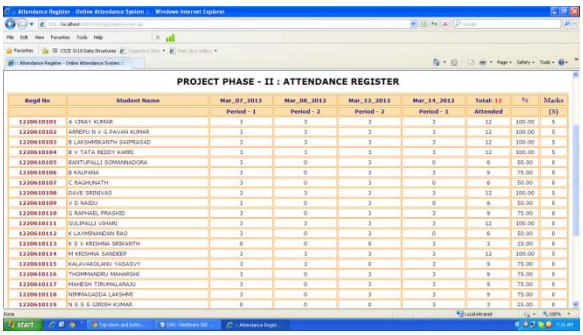



*Fig. 7 :* Attendance Register View


*Fig. 8 :* HOD Module – Selection Interface for Monthly Report

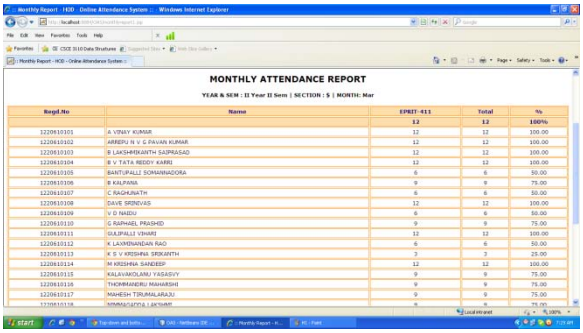



*Fig. 9 :* Cumulative Report for selected Month with all subjects

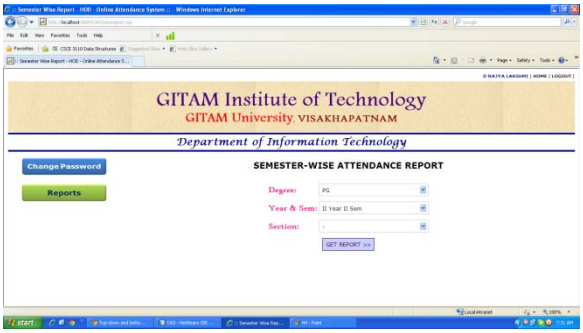



*Fig. 10 :* HOD Module – Selection Interface for Semester-Wise Attendance Report

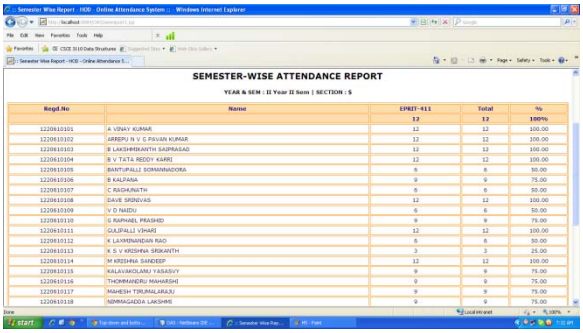



*Fig. 11 :* Cumulative Semester Wise Report with all subjects

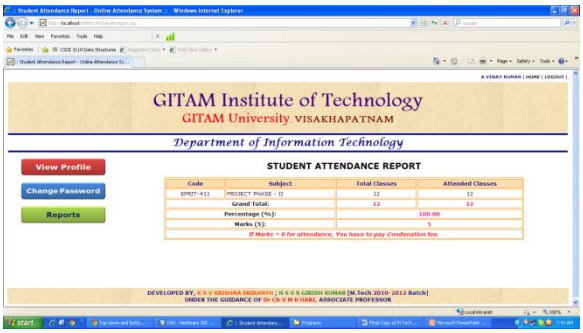



*Fig. 12 :* Attendance Report Viewed by Student

## V. Conclusion

This paper mainly elaborates basic principles of System Design and enumerates reusability is best practice for deliver product facility. The Online Attendance System that is developed meets the design objectives of the system design for which it has been developed. The users associated with the system understand its advantage and easily navigates with the user interface. It was intended to solve as requirement specification. The current system can be a good reference when implementing a similar system in other institutions as the system is proved to be workable and effective.

## References Références Referencias

1. Software Engineering: A Practitioner's Approach – Roger S Pressman
2. An Integrated Approach for Software Engineering – Pankaj Jalote
3. Data Coupling Design Principle: http://it. toolbox. com/blogs/enterprise-solutions/ design-principles-coupling-data-and-otherwise-160 61
4. Functional Cohesion Design Principle: http:// it. toolbox.com/blogs/enterprise-solutions/ design-principles-cohesion-16069
5. Nabil.M.A.M and A.Govardhan, A Comparison Between Five Models of Software Engineering, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 5, pp: 94-102,September 2010.
6. Sanjana Taya and Shaveta Gupta, Comparative Analysis of Software Development Life Cycle Models, IJCST Vol. 2, Issue 4, pp:536-539, Oct . - Dec. 2011.
7. Imran Baig, Measuring Cohesion and Coupling of Object-Oriented Systems, Master Thesis Software Engineering, Thesis no: MSE-2004:29, Month: August Year: 2004
8. H.C. Ting and T.O. Ting, An Online Attendance Record System, ICEED 2009, Month: December Year: 2009

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# Requirement Implementation and Defect Removal across Component Versions: A Simulation Based Approach

By P K Suri & Sandeep Kumar
*Galaxy Global Group of Institutions, Dinarpur, Ambala, Haryana, India*

*Abstract -* Competition in component market and short time to market the software components forces the organization to develop and launch the components in an iterative manner. Components are launched in various versions. All gathered requirements cannot be implemented in initial version. So requirements need to be prioritized and implemented in subsequent versions. Similarly defects in one version are taken care of in subsequent versions. In the present work we have proposed a simulation model that can be used to study the operational characteristics of the requirements implementation process and defect removal process in a Component Based Software.

*Keywords :* *Component Based Software, COTS, Simulation, Requirements Implementation, Defects Removal, Exponential Distribution, Normal Distribution.*

*GJCST-C Classification:* *D.2.1*


*Strictly as per the compliance and regulations of:*

# Requirement Implementation and Defect Removal across Component Versions: A Simulation Based Approach

P K Suri [α] & Sandeep Kumar [σ]

*Abstract -* Competition in component market and short time to market the software components forces the organization to develop and launch the components in an iterative manner. Components are launched in various versions. All gathered requirements cannot be implemented in initial version. So requirements need to be prioritized and implemented in subsequent versions. Similarly defects in one version are taken care of in subsequent versions. In the present work we have proposed a simulation model that can be used to study the operational characteristics of the requirements implementation process and defect removal process in a Component Based Software.

*Keywords : Component Based Software, COTS, Simulation, Requirements Implementation, Defects Removal, Exponential Distribution, Normal Distribution.*

## I. Introduction

Traditionally, development of software would focus on developing for a particular kind of application only. But Component Based Development is a market driven technology. Here Components are not developed for a specific application. Rather they are developed to be reused in many different kinds of applications. Different organizations in the common marketplace offer different components for the same functionality. Though, competition in the market is not that much high at present, but looking at the growth trends of the component based technology, the day doesn't seem to be far away when there will be a cut throat competition in the COTS market. In such type of scenario, it becomes very important for various market players and stakeholders to develop quality software components in least available time and release them in market. In this market driven environment, Requirement Engineering is getting more and more attention **[1],[2],[3]**

It's not only that quality components are to be released in the market as soon as possible; equally important is that the quality of the components is improved gradually. Various organizations throughout geographically dispersed locations improve the components continuously and release their independent versions in the market after the improvements **[4], [5], [6].** If we follow Evolutionary development paradigm **[14]** then an organization must deliver first operational version of the software or component as fast as the system architecture is defined. This first version should incorporate a minimum set of requirements in such a way so that the end user can start working with it. That's why this initial version should be called an operational model. One of the reasons for the early release of the first operational version and subsequent versions of a software component is short time to market for components. Once first version of the component has been released in the market and used by the users, remaining system requirements (that were not incorporated in the earlier version) and some new requirements can be added to the component in its future version releases. Not only this, users of the component will come up with certain defects in the earlier version of the component. These defects can also be removed in such a way that they are not present in all future versions. Though it is possible that some new defects may creep-up in the current version release, and they can always be taken care of in the next version.

Although it is never possible to freeze the requirements in any software development paradigm, still efforts should be made to gather as much requirements as possible, before the release of first version of that component. Out of these most important features can be implemented in the first version of the component and rest of the features can be implemented in subsequent component versions, along with newly generated requirements between release of any two versions, and defects identified in previous component version removed.

For the efficient development of software component, requirements should be properly elicited, analysed and documented at the beginning of a project. Also important is the correct implementation and management of these requirements in the later stages of the component development and integration. This becomes all the more important because all other component development activities are based on how efficiently requirements have been managed. In an ideal software component development environment it is just sufficient to elaborate the requirements into working

*Author α : Dean, Research and Development; Chairman CSE/IT/MCA, HCTM Technical Campus, Kaithal, Haryana, India.-136034.*
*Author σ : Assistant Professor and Head, Faculty of Computer Applications, Galaxy Global Group of Institutions, Dinarpur, Ambala, Haryana-133207, India. E-mail : sandeepnain77@gmail.com*

software component designs, code, and tests. But software development in general and component development in particular is not that much a straight forward thing. In practical software development process (component development processes in that sense), the requirements keep changing, new requirements keep coming and sometimes old requirements need to be removed also. So the process of management and implementation of requirements is a complex task. Need to develop and release initial component version as quickly as possible makes the process more complex. Due to this sometimes problems are encountered in maintaining consistency among the various releases of the component versions. These problems generally come into picture when components are integrated in a component based system.

The process improvement proposals in an organization can be analysed by carrying out pilot studies or controlled experiments in that organization [7]. But this method is very time consuming and resource crunching. Alternatively, simulations can be used to study the behaviour of such a system [8], [9]. Simulation approach has been applied in many areas of engineering and is suitable for application in evaluation of software development processes also. After analysing the new processes using simulation, they can also be analysed in experiments and case studies to establish the fairness of the results obtained using simulation. In this way simulations can be a natural part of technology transfer [10] and evaluation. If simulation is applied for the evaluation of new technologies and processes then it becomes easy to identify the changes and evaluate them in experiments and pilot projects. Sometimes there are certain changes that do not result in process improvement. Application of simulation reduces the risks associated with such changes. Lot of human resources are often involved in experiments and pilot-studies in an organization, introduction and evaluation of wrong changes can lead to a lot of problems. This can potentially damage the continued process improvement work in the organisation for a long time. Hence simulation is needed in the evaluation of new software process technologies.

In the present work, we have proposed application of discrete event simulation [11] using queuing network model [12]. Objective of the study is to find ways for effective management of the human resources of an organization for requirement management and implementation and defect removal while releasing various versions of a software component one after the other. The motivation for the proposed model has come from REPEAT (Requirement Engineering Process At Telelogic)[15].

Basic idea is to have a database of all the requirements to be implemented in a software component. But it is not possible to implement all the requirements in first (or few subsequent versions for that matter) version of the component due to many factors. Most influential of these factors being the competition from other market players and very short time to market. Due to this reason requirements need to be prioritized on some basis and implemented according to their priorities. The steps of the REPEAT lifecycle model for requirement implementation are given as follow [13]:

### New

This state represents the initial state of a requirement, and every requirement is defined as new immediately after it has been issued and given an initial priority.

### Assigned

A requirement is elevated to the assigned state when an expert team has been assigned to investigate the requirement and determine the value of a number of attributes.

### Classified

When reaching this state, an expert team has assigned values to attributes representing a rough estimate of cost and architectural impact. Comments and implementation ideas may also be stated.

### Selected

All requirements in this state are selected for implementation for the coming release. They are sorted in priority order on two list: a must-list for mandatory requirements and a wish-list for “nice-to-have” requirements. They also have attributes assigned concerning detailed cost and impact estimations. There is also a more detailed textual specification of the requirement. A selected requirement may be deselected, due to changed circumstances, and then re-enters the classification state or gets rejected.

### Applied

This is an end-state indicating that the requirement has been implemented and verified. The requirement is now incorporated in a component release that can be marketed to customers.

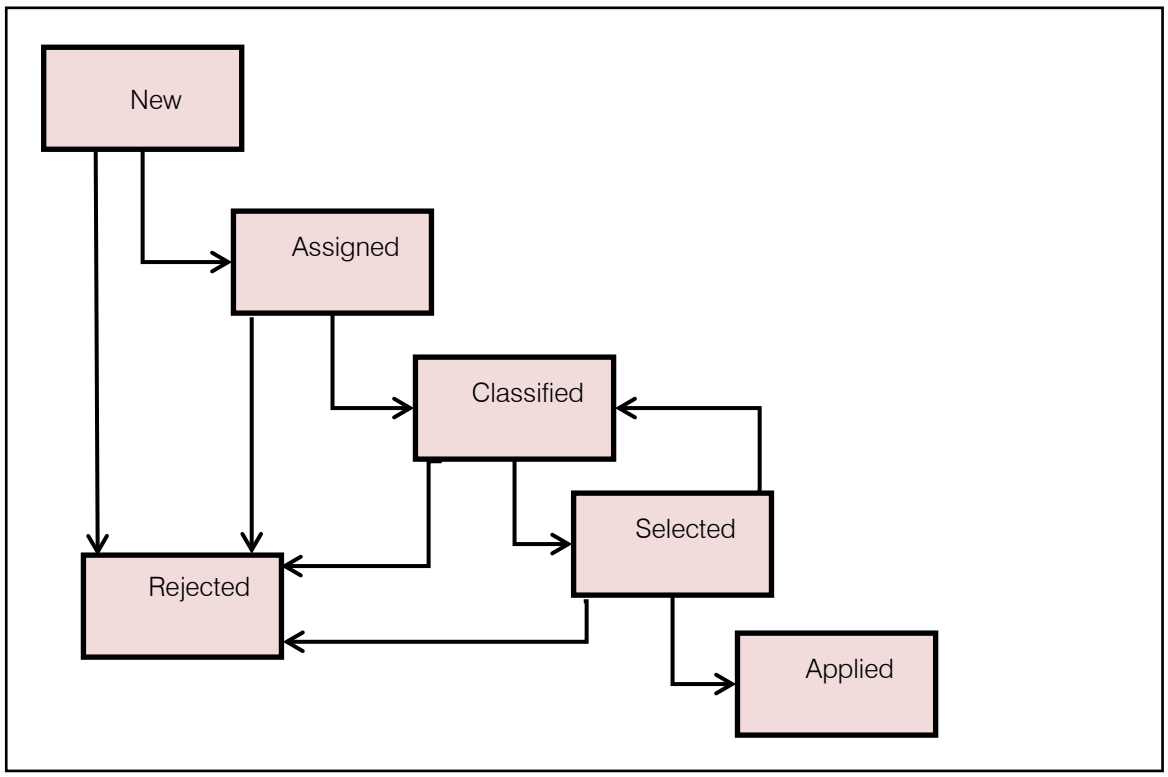



*Fig.1 :* Repeat Requirement Lifecycle Model [CAR 2000]

**Rejected**

This is an end-state indicating that the requirement has been rejected, e.g. because it is a duplicate, already implemented, or it does not comply with the long-term product strategy.

## II. Proposed Model

For the proposed model, we assume that first operational version of the component has been released in the market. Model can be implemented from second version onwards. Once, the first component version (with bare minimum requirements) becomes operational, process for the release of second and subsequent versions start. More requirements may be added to the requirement database between the releases of any two versions. So this set of requirements to be implemented in a version of software component form a queue. These requirements are to be implemented by a team of developers. Once a component has been released in the market, it is used by various end users in their applications and feedback from the users is received. Certain defects may also be reported by the users. These defects may have crept in due to implementation errors or discrepancies. These defects need to be removed so that they are not part of any future version of the component. So these defects form another set of inputs to the system. We assume that requirements to be implemented in the future component version and defects reported from the previous component version form a common queue. System is modelled as "two parallel servers" queuing system. It is the job of Software Component project manager, modelled as team T, to decide which of the inputs are new requirements, and which of the inputs are defects. Depending upon the nature of input, it is assigned to a different team. Requirement implementation is performed by team TR and defect removal process is performed by team TD.

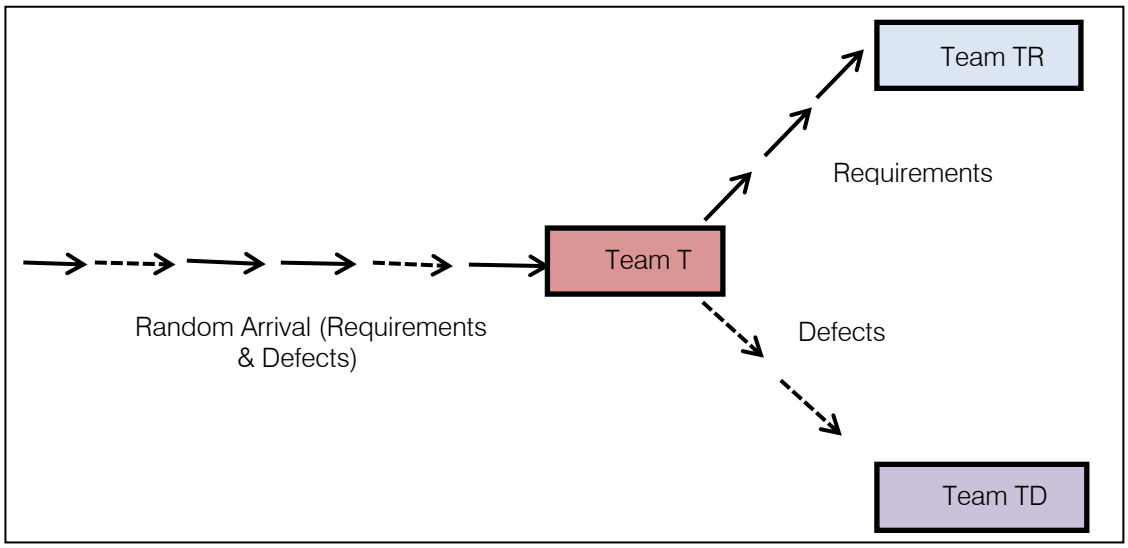



*Fig. 2 :* Requirements and Defects Arrival and Service

## III. Terms and Notations

Following terms and notations have been used for the proposed queuing model of the system as far as arrival patterns and service patterns are concerned:

| | | |
|---|---|---|
| λ | : | Average requirement/defect inter arrival time at team T. |
| $\mu$ | : | Requirements/Defects arrival rate at team T. |
| mR | : | Mean service time of team TR. |
| sdR | : | Standard Deviation of service times at team TR. |
| mD | : | Mean service time at team TD. |
| sdD | : | Standard Deviation of service times at team TD. |
| N | : | Total no. of arrivals at team T (requirements+ defects). |
| R | : | No. of requirements implemented by team TR. |
| D | : | No. of defects removed by team TD. |
| qR | : | Queue length of requirements at team TR. |
| qD | : | Queue length of defects at team TD. |
| sR | : | Service terminating at team TR (Requirements). |
| sD | : | Service terminating at team TD (Defects). |
| IAT | : | Inter arrival time between any two consecutive requirements/ defects. |
| NAT | : | Next arrival time of requirement/ defect. |
| wtR | : | Time a requirement waits in queue before it is implemented. |
| wtD | : | Time a defect waits in the queue before it is removed. |
| itR | : | Idle time of team TR. |
| itD | : | Idle time of team TD. |
| btR | : | Busy time of team TR |
| btD | : | Busy time of team TD |
| stR | : | Team TR service time. |
| stD | : | Team TD service time. |
| SRUNS | : | No. of simulation Runs. |
| $r_i$ | : | Random number. |
| maxqR | : | Maximum requirements in queue at team TR at any time. |
| maxqD | : | Maximum defects in queue at team TD at any time. |

## IV. Algorithm

Formally, algorithm for the model is described as follows:

1. Read Input Data.
2. Initialize SRUNS. Set clock:=0, N:=0, R:=0, D:=0, qR:=0, qD:=0, sR:=0, sD:=0, wtR:=0, wtD:=0, itR:=0, itD:=0.
3. Generate random numbers $r_i$'s.
4. Compute inter arrival times of requirements/defects (IAT's) at team T using exponential distribution with arrival rate $\mu$.
5. (At team T, categorise arrival as a requirement or defect.)

If ($r_i$< .8),

Designate the arrival as a requirement, increment qR.

Else

Designate the arrival as a defect, increment qD.

6. (Check present status of team TR)

a. If (clock >=sR), then do

Update wtR.

If qR is positive, then do

i. Decrement qR by 1.

ii. Generate stR's using normal distribution with mean mR and standard deviation sdR.

iii. sR:= clock+stR.

iv. Increment R by 1.

Else do

Update itR (idle time of team TR).

b. If (clock <sR) then do

Update waiting time, wtR of requirement at team TR.

7. (Check present status of team TD)

a. If (clock >=sD), then do
Update wtD.
If qR is positive, then do

i. Decrement qD by 1.

ii. Generate stD's using normal distribution with mean mD and standard deviation sdD.

iii. sD:= clock+stD.

iv. Increment D by 1.
Else do
Update itD (idle time of team TD).

b. If (clock <sD), then do
Update waiting time, wtD of requirement at team TD.

8. Compute total Busy and Idle times of team TR and TD
9. Compute average waiting times of requirements and defects.
10. Print Required Data.
11. Stop.

## V. Results and Discussion

Simulator was executed for various values of SRUNS. If we assume that on an average 1 requirement or defect arrives at team T every 7 time units, with exponential distribution, requirements are implemented by team TR at a service rate that is normally distributed with value of mR=6.0 and sdR=2.0; and defects are removed by team TD at a service rate that is again normally distributed with value of mD=12.0 and sdD=6.0, then results for various values of SRUNS are shown in table 1.

Graph in figure 3 shows that values of various operational characteristics have larger variation if simulator is run less than 10000 times. Values of btR, itR, btD and itD tend to stabilize after 10000 simulation runs. Same is true for the results depicted in figure 4. Hence it can be said that 10000 simulation runs are sufficient to get the accurate results.
Relationships between number of simulation runs v/s N, R and D is shown in figure 5.

Table 2 shows the results obtained from 50000 simulation runs where value of λ varies from 5 to 10 in steps of 1. Table contains values of idle times of teams TR (itR) and TD (iTD), waiting times of teams TR (wtR) and team TD (wtD) and maximum queue lengths at TR and TD for various values of λ. It is clear from figure 4 that idle times of team TR (itR) and team TD (itD) increase with the increase in the value of λ. waiting times of the requirements and defects decrease with increase in the values of λ. There is variation in maximum queue lengths initially, but as the value of λ increases, maximum queue lengths for both the teams tend to get stabilize.

| SRUNS | btR (%) | btD (%) | itR (%) | itD (%) | wtR (avg.) | wtD (avg.) | Ma qR | Max qD | N | R | D |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1000 | 74.43 | 27.7 | 25.57 | 72.3 | 6.43 | 10.38 | 6 | 3 | 149 | 127 | 22 |
| 2000 | 76.79 | 35.04 | 23.21 | 64.96 | 8.53 | 5.38 | 7 | 3 | 308 | 257 | 51 |
| 10000 | 69.14 | 34.13 | 30.86 | 65.87 | 7.55 | 3.93 | 11 | 3 | 1429 | 1151 | 278 |
| 20000 | 67.44 | 34.72 | 32.56 | 65.28 | 6.89 | 3.52 | 11 | 3 | 2805 | 2232 | 572 |
| 30000 | 68.35 | 35.2 | 32.65 | 64.8 | 7.32 | 3.87 | 11 | 4 | 4257 | 3396 | 861 |
| 40000 | 68.63 | 35.47 | 31.37 | 64.53 | 7.02 | 3.62 | 11 | 4 | 5711 | 4555 | 1156 |
| 50000 | 68.73 | 35.64 | 31.27 | 64.36 | 6.85 | 3.72 | 11 | 4 | 4146 | 5698 | 1441 |

*Table 1 :*

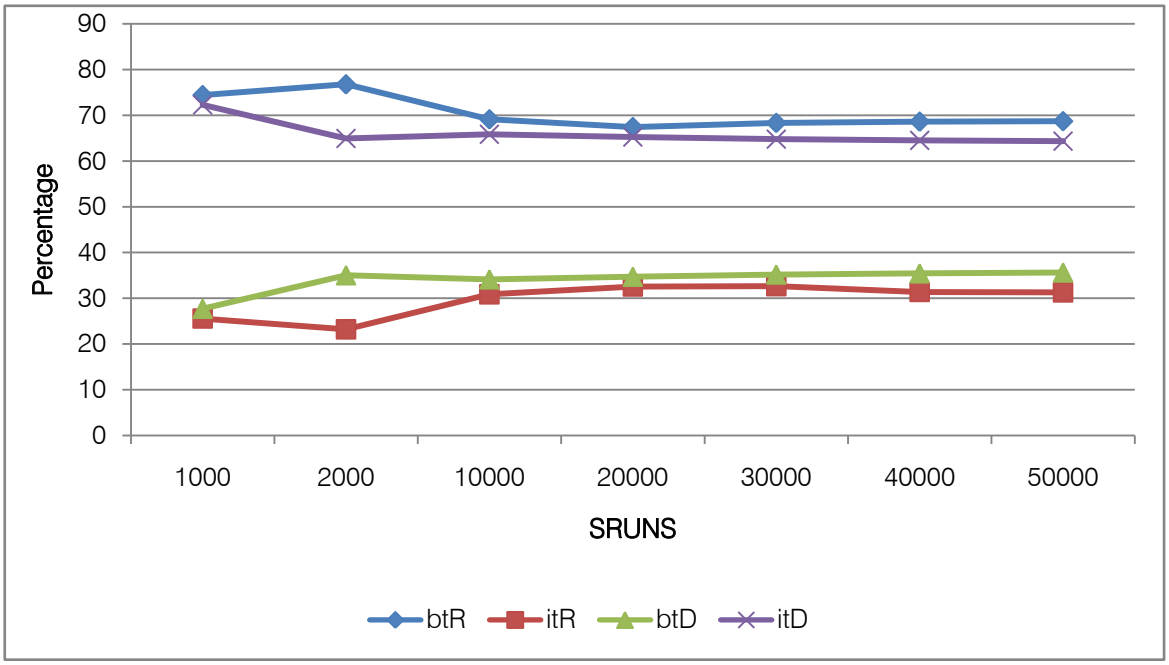



*Fig. 3 :*

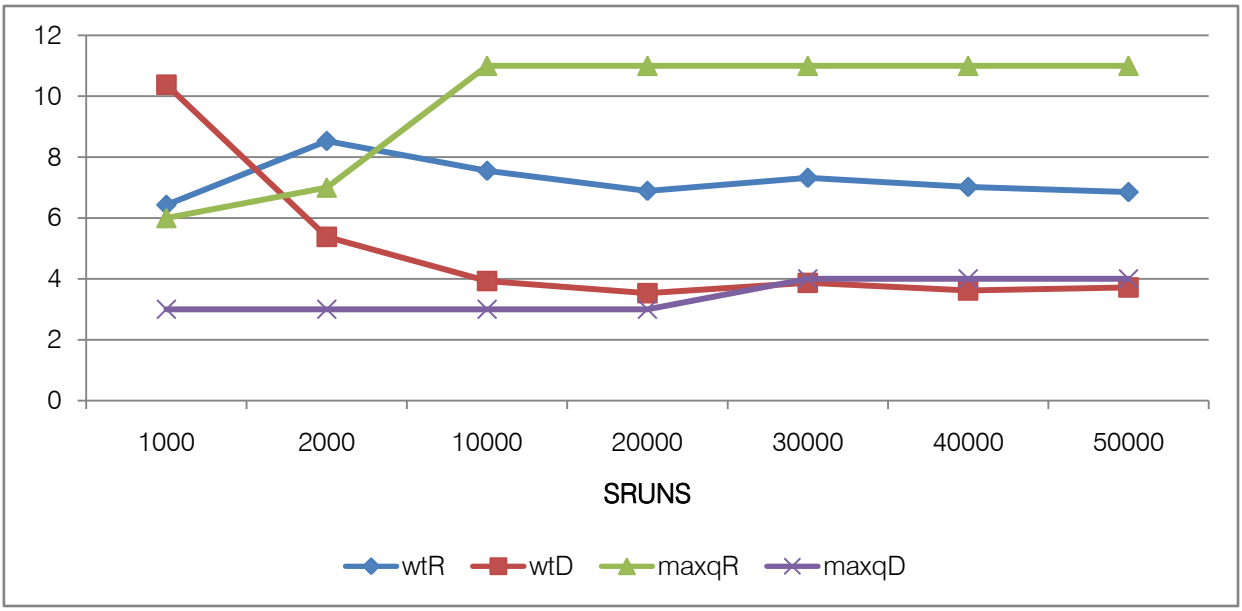



*Fig. 4 :*

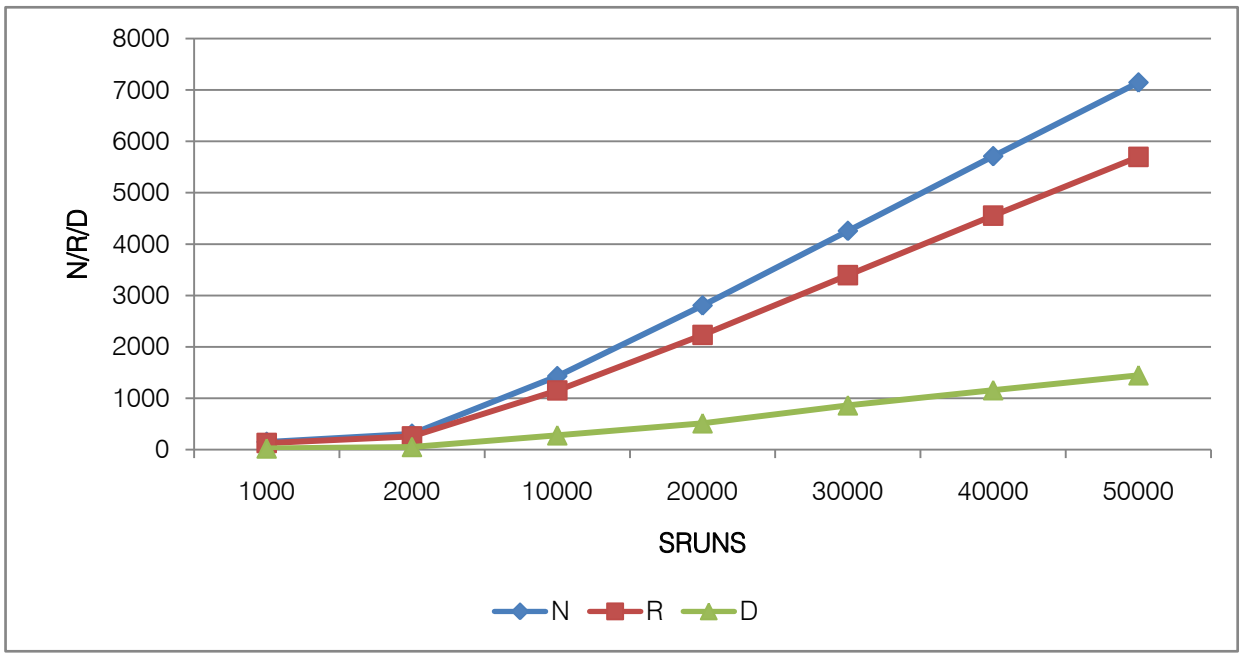



*Fig. 5 :*

| $\lambda$ | $\mu$ | itR | itD | wtR | wtD | maxqR | maxqD |
|---|---|---|---|---|---|---|---|
| 5 | 0.2 | 4.35 | 52.65 | 49.13 | 7.07 | 30 | 6 |
| 6 | 0.1667 | 19.21 | 59.24 | 13.9 | 5.73 | 23 | 7 |
| 7 | 0.1429 | 31.16 | 68.8 | 7.42 | 4.21 | 7 | 3 |
| 8 | 0.125 | 38.76 | 70.04 | 5.06 | 3.26 | 8 | 4 |
| 9 | 0.1111 | 45.6 | 73.2 | 4.02 | 2.73 | 8 | 4 |
| 10 | 0.1 | 57.32 | 75.56 | 3.03 | 2.43 | 6 | 3 |

*Table 2 :*

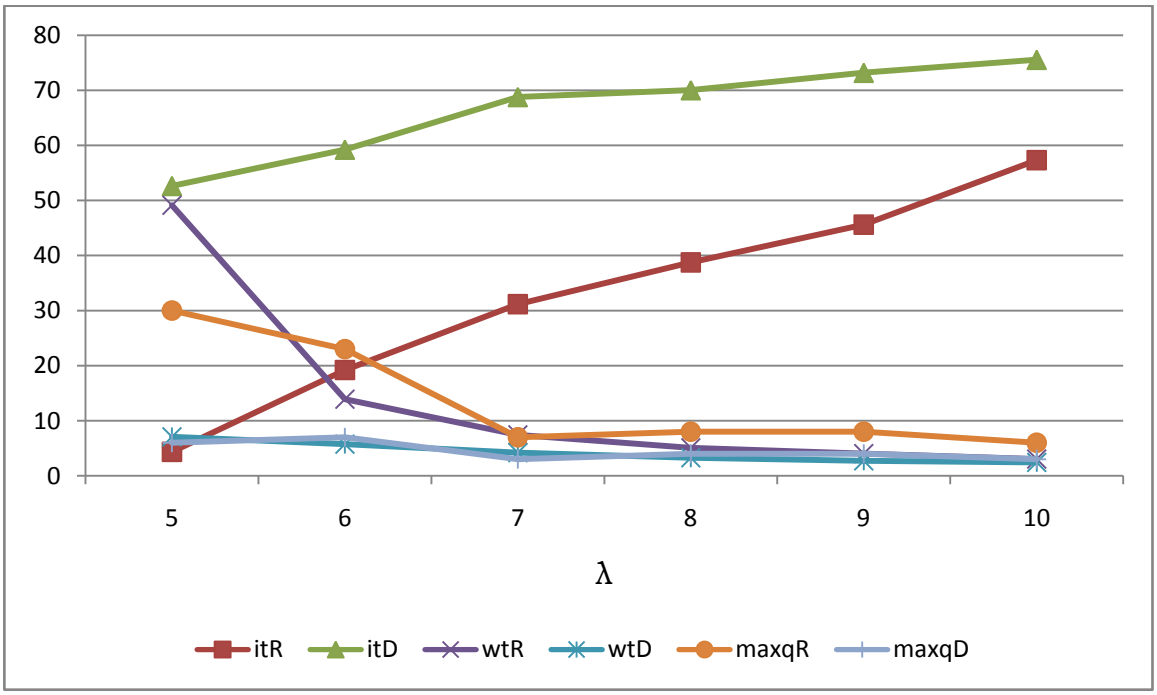



*Fig. 6 :*

## VI. Conclusion

In the presented work, a simulator has been proposed that can be helpful in implementation of user requirements and removal of defects across various versions of a software component in such a way so that size of the requirements implementation team and defects removal team can be optimized. Simulator has been modelled as a two parallel server queuing model, where requirements and defects initially form a common queue and then they are categorized as requirements or defects depending upon their characteristics. Requirements and defects are then handled by different teams. Busy and idle times of both the teams can be studied and depending upon that team size can be decided. Simulator can also be used to study other operational characteristics like the time a requirement or defect has to spend waiting before it is implemented/ removed and maximum length of the queues formed by requirements and defects at each team.

## References Références Referencias

1. Lubars M., Potts C. and Richter C. (1993), A Review of the State of the Practice in Requirements Modeling, Proceedings of First IEEE International Symposium on Requirements Engineering (RE'93), San Diego USA, IEEE Computer Society Press.
2. Potts C.(1995), Invented Requirements and Imagined Customers: Requirements Engineering for Off-the Shelf Software, Proceedings of Second IEEE International Symposium on Requirements Engineering (RE'95), York UK, IEEE Computer Society Press.
3. Yeh A.(1992), Requirements Engineering Support Technique (REQUEST) – A Market Driven Requirements Management Process, Proceedings of Second Symposium of Quality Software Development Tools, New Orleans USA, IEEE Computer Society Press, 211-223.
4. Szyperski C.(1998), Component Software—Beyond Object-Oriented Programming. Addison-Wesley/ACM Press: Boston, MA, 1998.
5. Heineman G.T. and Councill W.T. (2001.), Component-Based Software Engineering: Putting the Pieces Together. Addison-Wesley: Reading.
6. Crnkovic I. et al.(2001), Proceedings of the Fourth ICSE Workshop on Component-Based Software Engineering: Component Certification and System Predication. Software Engineering Institute: Pittsburgh.
7. Wohlin C., Runeson P., Höst M., Ohlsson M.C., Regnell B. and Wesslén A. (2000), Experimentation in Software Engineering -An Introduction , Kluwer Academic Publishers.
8. Kellner M.I., Madachy R.J. and Raffo, D.M.(1999), Software Process Simulation Modeling: Why? What? How? Journal of Systems and Software, Vol. 46, No. 2-3, 91-105.
9. Pfahl D. and Lebsanft K.(2000), Using Simulation to Analyse the Impact of Software Requirement Volatility on Project Performance, Proceedings of the combined 11th European Software Control and Metrics Conference and the 3rd SCOPE conference on Software Product Quality , Munich, Germany,267-275.
10. Linkman S. and Rombach, H.D.(1997), Experimentation as a Vehicle for Software

Technology Transfer - A Family of Software Reading Techniques, Information and Software Technology, Vol. 39, No. 11, pp. 777-780.
11. Banks J., Carson J. S. and Nelson, B. L.(1996), Discrete-Event System Simulation, 2nd Ed., Prentice Hall.
12. King P. J. B.(1990), Computer and Communication Systems Performance Modelling, Prentice Hall.
13. Carlshamre P. and Regnell B.(2000), "Requirements Lifecycle Management and Release Planning in Market-Driven Requirements Engineering Processes", Published by IEEE CS press in the Proceedings of International Workshop on the Requirements Engineering Process: Innovative Techniques, Models, and Tools to support the RE Process, Greenwich UK.
14. Sommersville I.(1996), Software Process Models", ACM Computing Surveys (CSUR) Volume 28, Issue 1, 269-271.
15. Regnel B., Beremark P. and Eklundh, O(1998). "A Market-Driven Requirements Engineering Process - Results from an Industrial Process Improvement Programme", Journal of Requirements Engineering, Vol. 3, No. 2, 21-29.

This page is intentionally left blank

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# Impact of Mediated relations as Confounding Factor on Cohesion and Coupling Metrics: For Measuring Fault Proneness in Oo Software Quality Assessment

By Amjan.Shaik, Dr.C.R.K.Reddy & Dr.A.Damodaram
*JNTUH, Hyderabad, Andhra Pradesh, India*

*Abstract -* Mediated class relations and method calls as a confounding factor on coupling and cohesion metrics to assess the fault proneness of object oriented software is evaluated and proposed new cohesion and coupling metrics labeled as mediated cohesion (MCH) and mediated coupling (MCO) proposed. These measures differ from the majority of established metrics in two respects: they reflect the degree to which entities are coupled or resemble each other, and they take account of mediated relations in couplings or similarities. An empirical comparison of the new measures with eight established metrics is described. The new measures are shown to be consistently superior at measure the fault proneness.

*GJCST-C Classification:* *D.2.8*


*Strictly as per the compliance and regulations of:*

# Impact of Mediated relations as Confounding Factor on Cohesion and Coupling Metrics: For Measuring Fault Proneness in Oo Software Quality Assessment

Amjan.Shaik[α], Dr.C.R.K.Reddy[σ] & Dr.A.Damodaram[ρ]

*Abstract* - Mediated class relations and method calls as a confounding factor on coupling and cohesion metrics to assess the fault proneness of object oriented software is evaluated and proposed new cohesion and coupling metrics labeled as mediated cohesion (MCH) and mediated coupling (MCO) proposed. These measures differ from the majority of established metrics in two respects: they reflect the degree to which entities are coupled or resemble each other, and they take account of mediated relations in couplings or similarities. An empirical comparison of the new measures with eight established metrics is described. The new measures are shown to be consistently superior at measure the fault proneness.

## I. Introduction

Object Oriented (OO) design and code, for instance, [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]. These metrics offer ways to evaluate the excellence of software and their use in former phases of software development can help organizations in evaluating large software development quickly, at a low cost [3]. But how do we know which metrics are functional in capturing important quality attributes such as Degree of Fault prone, effort, efficiency or amount of maintenance adaptations. Experiential studies of real systems can provide relevant answers. There have been few empirical studies evaluating the effect of object-oriented metrics on software quality and constructing models that utilize them in predicting quality attributes in the system, such as [16, 17, 18, 19, 5, 20, 21, 22, 23, 8, 12, 24]. More data based by empirical studies, which are capable of being verified by observation or experiment are needed. The evidence gathered through these empirical studies is today considered to be the most powerful support possible for testing a given hypothesis.

A well designed component, in which the functionality has been appropriately distributed to its various subcomponents, is more likely to be fault free and will be easier to adapt. Appropriate distribution of function underlies two key concepts of object-oriented design: coupling and cohesion. Coupling is the extent to which the various subcomponents interact. If they are highly interdependent then changes to one are likely to have significant effects on the behavior of others. Hence loose coupling between its subcomponents is a desirable characteristic of a component. Cohesion is the extent to which the functions performed by a subsystem are related. If a subcomponent is responsible for a number of unrelated functions then the functionality has been poorly distributed to subcomponents. Hence high cohesion is a characteristic of a well designed subcomponent.

Many metrics have been proposed to measure the coupling and cohesion to predict the fault-prone and maintainability of software. However, few studies had been done using coupling and cohesion to assess the quality of components.

In this context we therefore analyzed the mediated relations of the classes and method calls as a confounding factor for coupling and cohesion metrics and proposing two new metrics called Mediated coupling and Mediated cohesion to measure the fault proneness to assess the quality of the software.

The rest of the paper organized as, in section II the traditional cohesion and coupling metrics revealed, which followed by section III that explores transitivity as a confounding factor.

## II. The Coupling and Cohesion in OO Programming

### a) Measuring Coupling

The term coupling is usually used in a derogatory manner in design review meetings. Even so, it's not possible to design aefficient OO application without coupling. At any time if one object interacts with another object, then it is coupling. In reality, what you need to try to minimize is coupling factors. Strong coupling means that one object is strongly coupled with the implementation details of another object. Strong coupling is discouraged because it results in less

*Author α : Research Scholar, Department of CSE, JNTUH, Hyderabad, Andhra Pradesh, India. E-mail : amjan_shahi@yahoo.com*
*Author σ : Professor and HOD of CSE CBIT, Gandipet, Hyderabad, Andhra Pradesh, India. E-mail : crkreddy@gmail.com*
*Author ρ : Professor of CSE and Director, Academic Audit Cell, JNTUH, Kukatpally, Hyderabad, Andhra Pradesh, India.*
*E-mail : damodarama@rediffmail.com*

flexible, less scalable application software. However, coupling can be used so that it enables objects to talk to each other while also preserving the scalability and flexibility.

Though this seems like a difficult task, OO metrics can help you to measure the right level of coupling.

Coupling between Objects (CBO): CBO is defined as the number of non-inherited classes associated with the target class. It is counted as the number of types that are used in attributes, parameters, return types, throws clauses, etc. Primitive types and system types (e.g. Java.lang.*) is not counted.

Data Abstraction Coupling (DAC): DAC is defined as the total number of referring types in attribute declarations. Primitive types, system types, and types inherited from the super classes are not counted.

Method Invocation Coupling (MIC): MIC is defined as the relative number of classes that receive messages from a particular class.

$$MIC = \frac{nMIC}{N-1}$$

Where $N$ isthe total number of classes defined within the project.

$\mathrm{nMIC}$ is the total number of classes that receive a message from the target class.

Demeter's Law: Ian Holland first proposed the Law of Demeter. The class form of Demeter's Law has two versions: a strict version and a minimized version. The strict form of the law states that every supplier class of a method must be a preferred supplier. The minimization form is more permissive than the first version and requires only minimizing the number of acquaintance classes of each method.

Definition 1 (Client): Method M is a client of method $f$ attached to class C, if in $M$ message $f$ is sent to an object of class $C$, or to $C$. If $f$ is specialized in one or more subclasses, then $M$ is only a client of $f$ attached to the highest class in the hierarchy. Method $M$ is a client of some method attached to $C$.

Definition 2 (Supplier): If $M$ is a client of class $C$ then $C$ is a supplier to M. In other words, a supplier class to a method is a class whose methods is called in the method. In Listing 1, the Product class is a supplier class to the client class Order.

Definition 3 (associate Class): A class $C1$ is an acquaintance class of method $M$ attached to class $C2$, if $C1$ is a supplier to $M$ and $C1$ is not one of the following:

The same as $C2$;

A class used in the declaration of an argument of $M$

A class used in the declaration of an instance variable of $C2$

Definition 4 (Preferred-supplier class): Class $B$ is called a preferred-supplier to method $M$ (attached to the class $C$) if $B$ is a supplier to $M$ and one of the following conditions holds:

$B$ is used in the declaration of an instance variable of $C$

$B$ is used in the declaration of an argument of $M$, including $C$ and its super classes.

$B$ is a preferred acquaintance class of $M$.

*b) Measuring Cohesion*

In OO methodology, classes contain certain data and exhibit certain behaviors. This concept may seem fairly obvious, but in practice, creating well-defined and cohesive classes can be tricky. Cohesive means that a certain class performs a set of closely related actions. A lack of cohesion, on the other hand, means that a class is performing several unrelated tasks. Though lack of cohesion may never have an impact on the overall functionality of a particular class—or of the application itself—the application software will eventually become unmanageable as more and more behaviors become scattered and end up in the wrong places.

Thus, one of the main goals of OO design is to come up with classes that are highly cohesive. Luckily, there's a metric to help to verify that the designed class is cohesive.

The LCOM Metric: Lack of Cohesion in Methods

The Lack of Cohesion in Methods metric is available in the following three formats:

LCOM1: Take each pair of methods in the class and determine the set of fields they each access. If they have disjointed sets of field accesses, the count P increases by one. If they share at least one field access, Q increases by one. After considering each pair of methods:

$$\mathrm{RESULT} = (\mathrm{P} > \mathrm{Q})?\ (\mathrm{P} - \mathrm{Q}) : 0$$

A low value indicates high coupling between methods. This also indicates the potentially high reliability and good class design. Chidamber and Kemerer provided the definition of this metric in 1993.

LCOM2: This is an improved version of LCOM1. Say you define the following items in a class:

$m$ : number of methods in a class

$a$ : number of attributes in a class.

$mA$ : number of methods that access the attribute a.

$\text{sum}(mA)$ : sum of all mA over all the attributes in the class.

$$LCOM2 = 1 - \text{sum}(mA)/(m*a)$$

If the number of methods or variables in a class is zero (0), LCOM2 is undefined as displayed as zero (0).

LCOM3: This is another improvement on LCOM1 and LCOM2 and is proposed by Henderson-Sellers. It is defined as follows:

$$LCOM3 = (m - \text{sum}(mA)/a) / (m-1)$$

where $m$, $a$, $mA$, $\text{sum}(mA)$ are as defined in LCOM2.

The following points should be noted about LCOM3:

The LCOM3 value varies between 0 and 2. LCOM3>1 indicates the shortage of cohesion and is considered a kind of alarm.

If there is only one method in a class, LCOM 3 is undefined and also if there are no attributes in a class LCOM3 is also undefined and displayed as zero (0).

Each of these different measures of LCOM has a unique way to calculate the value of LCOM.

An extreme lack of cohesion such as LCOM3>1 indicates that the particular class should be split into two or more classes.

If all the member attributes of a class are only accessed outside of the class and never accessed within the class, LCOM3 will show a high-value.

A slightly higher value of LCOM means that you can improve the design by either splitting the classes or re-arranging certain methods within a set of classes.

## III. Mediated Relations of Classes and Method Calls as Confounding Factor

### a) Confounding Factor

The term confounding refers to a situation in which an association between an independent variable and a dependent variable is thought to be the result of the influence of a third variable[17]. The suggestion is that an apparent association between the independent and dependent variables may be partly or completely accounted for by a third variable. By the same token, the absence of an apparent association between independent and dependent variables may be the result of a failure to account for the effects of a third variable. The third variable that distorts the true association between the independent and dependent variables is usually called a confounding variable. The distortion that results from perplexing may lead to overestimation or underestimation of an association, depending on the direction and magnitude of the relations that the confounding variable has with the independent and dependent variables [18].

To quantitatively analyze the confounding factor, a number of confounding factor analysis models using various modeling techniques, such as linear, logistic, and probity regression, have been developed [16], [17], [19], [20], [21], [22]. Among these models, the confounding factor analysis model based on linear regression techniques has been widely used in health sciences and epidemiological research [16], [19], [20]. Compared to models based on other modeling techniques, the linear-regression-based model has two main advantages: 1) A number of statistical methods have been developed for this model to test for a confounding variable [16], [19] and 2) it is easy to determine whether a confounding variable leads to overestimation or underestimation of the true association between the independent and dependent variables [16], [20].

### b) Mediated relation as dependent variable

The objective of this study is to empirically investigate to identify the cohesion and coupling metrics under consideration of mediated class relations and method calls as confounding factors and assessing the association between these cohesion and coupling metrics and degree of fault-proneDegree of Fault prone is an important external quality attribute and identifying faults-prone classes is very useful because: 1) It enables software developers to take focused preventive actions that can reduce maintenance costs and improve quality and 2) it helps software managers to allocate resources more effectively. In this study, Degree of Fault prone denotes the extent of class responsibility in component failure. We need to select the depth of the transitivity in class relations and method calls as the dependent variable for our study.

## IV. Mediated Coupling Between Objects[MCBO]

We begin by regarding any object-oriented software system as a directed graph, in which the vertices are the classes comprising the system. Suppose such a system comprises a set of classes $C \equiv (C_i \in C \mid \{i = 1..m\})$. Let $m(C_j) \equiv \{m(C_j)_i \in m(C_j) \mid (i = 1..n)\}$ be the methods of the class $C_j$, and $mI_{(C_j \to C_i)}$ the set of methods and instance variables in class $C_i$ invoked by class $C_j$ for $j \neq i$. An edge from $C_j$ to $C_i$ exists if and only if the $mI_{(C_j \to C_i)} > 0$, which can be used to generate the weight of that directed edge. The graph is directed since $mI_{(C_j \to C_i)}$ is not necessarily equal to $mI_{(C_i \to C_j)}$. Let consider that $mI_{(C_j \to C)}$ is the set of all methods and instance variables in other classes of $C$ that are invoked by class $C_j$. '$mI_{(C_j \to C)}$' can be represented as follows:

$$mI_{(C_j \to C)} = \bigcup_{i=1}^{m} mI_{(C_j \to C_i)}$$

*a) Finding a Degree of Directed Coupling (DDC)*

The directed edge weight $cw_{(C_j \to C_i)}$ between classes $C_j$ and $C_i$ can be represented as

$cw_{(C_j \to C_i)} = \frac{mI_{(C_j \to C_i)}}{mI_{(C_j \to C)}}$, the directed edge weight also can refer as degree of direct coupling (DDC) between two classes $cw$ is always between 0 and 1.

*b) Finding a degree of mediated coupling (DMC)*

Based on this degree of direct coupling between two classes, we can generalize the process of detecting the degree of mediated coupling $mcw$ between any two classes $C_j$ and $C_k$ exists such that $mI_{(C_j \to C_k)} \cong 0$, which follows:

$$mcw_{(C_j \to C_k, p)} = 1 - \frac{1}{\left( \sum_{i=1}^{|e_{j \to k}|} cw_i \right)} \; (iff \; \mathrm{mI}_{(C_j \to C_k)} \cong 0)$$

In above equation

$e_{j \to k}$ is the set of DDCs of edges, which are building path $p$ between class $C_j$ and $C_k$

$cw_i$ is DDC of an edge $i$ that belongs to $e_{j \to k}$.

$p$ is one of the path out of set of paths $P$ between $C_j$ and $C_k$

*c) Applying Confounding factor*

The confounding factor of path $p$ is $cf_{(p)}$, that assessed as follows:

$$cf_{(C_j \to C_k, p)} = \frac{|e(p)| - 1}{|e(p)|}$$

Here in the above equation

$e_{(p)}$ is set edges that belongs to the path $p$.

Then the generalized degree of mediated coupling between class $C_j$ and $C_k$ $mcw_{(C_j \to C_k)}$ can be found as follows

$$mcw_{(C_j \to C_k)} = 1 - \frac{1}{\left( \sum_{i=1}^{|P|} (mcw_{(C_j \to C_k, i)} + cf_{(C_j \to C_k, i)}) \right)}$$

The following hypothesis is a convention from the empirical study conducted on applications that are confirmed as fault prone:

If '$mcw$'is the degree of mediated coupling between two objects $O_1$ and $O_2$ then $(mcw \times 100)\%$ is the percentage of $O_1$ and $O_2$ objects in application's fault proneness.

## V. Mediated Cohesion (Mch)

The proposed cohesion metric is based on transitive function calls. The Hypothesis of the proposed cohesion metric can be defined as:

If a method A invoking a method B and method B is invoking method C, then the connection between A and C can be considerable and their cohesiveness is transitive if and only if A,B and C belongs to a same class or classes in an inheritance hierarchy .

We build a graph based on the function calls between the functions of the same class.

The edge between any two functions represents thetotal number of similar properties used similar functions invoked in both functions.

Finding Degree of Direct Cohesion(DDCH)

The Degree of Direct Cohesion Between two functions that represents the edge weight can be generalized as follows:

Let $pM_{(i)}$ is set of properties used in a method $M_i$ of the class $C$ such that $pM_i \in pC$ and $M_i \in mC$ , here $pC$ is set of properties declared in the class $C$ and $mC$ is a set of methods belongs to the class $C$. Let $mM_{(i)}$ is set of methods invoked in method $M_i$ of the class $C$ such that $mM_i \in mC$.

Let $pM_{(j)}$ is set of properties used in the method $M_j$ of the class $C$ such that $pM_j \in pC$ and $M_j \in mC$ , here $pC$ is set of properties declared in the class $C$ and $mC$ is a set of methods belongs to the class $C$. Let $mM_{(j)}$ is set of methods invoked in method $M_j$ of the class $C$ such that $mM_j \in mC$.

If $(M_j \in mM_i \parallel M_i \in mM_j)$ then there an edge exists between these two methods. The graph is not a directed graph, since edge weight is not changing under any direction of direct connection between the two functions. The DDCH that referred as edge weight can be measured as follows:

$$chw_{(M_i \oplus M_j)} = \frac{\left(\left(\frac{|pM_i \cap pM_j|}{|pM_i \cup pM_j|}\right)+\left(\frac{|mM_i \cap mM_j|}{|mM_i \cup mM_j|}\right)\right)-1}{\left(\left(\frac{|pM_i \cap pM_j|}{|pM_i \cup pM_j|}\right)+\left(\frac{|mM_i \cap mM_j|}{|mM_i \cup mM_j|}\right)\right)}$$

,here $1 \geq chw_{(M_i \oplus M_j)} \geq 0$

*a) Finding Degree of Mediated Cohesion (DMCH)*

Based on this degree of direct Cohesion between two methods, we can generalize the process of detecting the degree of mediated cohesion $mchw$ between any two methods $M_j$ and $M_k$ of same class exists such that $chw_{(M_j \oplus M_k)} \cong 0$ , which follows:

$$mchw_{(M_j \oplus M_k, p)} = 1 - \frac{1}{\left(\sum_{i=1}^{|e_{j \to k}|} chw_i\right)}$$

$$(iff\ \text{chw}_{(M_j \oplus M_k)} \cong 0)$$

In above equation

$e_{j \to k}$ is the set of DDCHs of edges, which are building a path $p$ between methods $M_j$ and $M_k$

$chw_i$ is DDCH of an edge $i$ that belongs to $e_{j \to k}$ .

$p$ is one of the path out of set of paths $P$ between $M_j$ and $M_k$

*b) Applying Confounding factor*

The confounding factor of the path $p$ is $cf_{(p)}$ , that assessed as follows:

$$cf_{(M_j \oplus M_k, p)} = \frac{|e(p)|-1}{|e(p)|}$$

Here in the above equation

$e_{(p)}$ is set edges that belongs to the path $p$ .

Then the generalized degree of mediated cohesion between methods $M_j$ and $M_k$ $mchw_{(M_j \oplus M_k)}$ can be found as follows

$$mchw_{(M_j \oplus M_k)} = 1 - \frac{1}{\left(\sum_{i=1}^{|P|} (mchw_{(M_j \oplus M_k, i)} + cf_{(M_j \oplus M_k, i)})\right)}$$

Since the class level cohesiveness is significant to predict the fault proneness than the method level cohesiveness.

The class level confounding factor of a class $C$ measures as follows:

$$ccf_{(C)} = 1 - \frac{1}{\left(\frac{|P'|}{|mC|}\right)}$$

Since the the ration between number paths build in the graph and number of methods exists indicates the cohesiveness, if the majority of paths between same classes can be considered as a confounding factor. Hence the above equation justifies the measurement of the class level confounding factor.

Here in this equation $|P'|$ represents the total number of paths build between the methods of a class $C$, $|mC|$ total number of methods in class $C$.

Then the class level mediated cohesiveness can be measured as follows:

$$mchw(C) = 1 - \frac{1}{\left(\sum_{i=1}^{|P'|} mchw(p_i)\right) + ccf(C)}$$

Here in the above equation

$mchw(p_i)$ is the degree of mediated cohesion between methods that build path $p_i$

The following hypothesis is a convention from the empirical study conducted on applications that are confirmed as fault prone:

If '$mchw$'is the degree of mediated cohesionof class $C$ then $(mchw \times 100)\%$ is class $C$ fault proneness in application's fault proneness.

## VI. Results Analysis

We conducted experiments on applications build under SDLC standards. We make sure the heterogeneity in number of classes of the applications considered for experiments. We measured the Fault proneness prediction accuracy of the MCBO and MCH as fallow:

$$\text{S(MCBO)} = \frac{\text{Classes correctly predicted as fault prone}}{\text{Classes actually fault prone}}$$

$$\text{S(MCH)} = \frac{\text{Classes correctly predicted as fault prone}}{\text{Classes actually fault prone}}$$

$$\text{S(MCBO} \oplus \text{MCH)} = \frac{\text{(Correctly predicted as fault prone by MCBO and MCH}}{\text{Actually fault prone}}$$

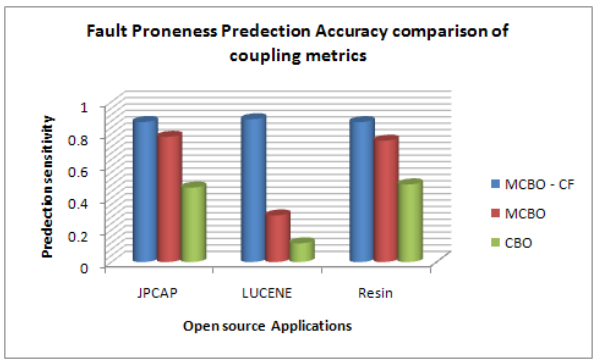



*Fig. 3 :* Fault proneness prediction sensitivity of Mediated Coupling Between Objects(MCBO), MCBO with confounding factor(MCBO - CF) and CBO

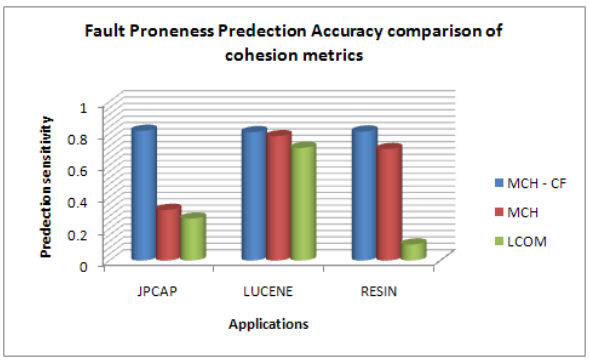



*Fig. 4 :* Fault proneness prediction sensitivity of Mediated cohesion (MCH), MCH with confounding factor(MCH - CF) and LCOM

Here in fig 3 we can observe the performance of the MCBO with the confounding factor in predicting the sensitivity of fault proneness, which stands with approximately 90% and miles ahead when compared to CBO. If path lengths are not considered as confounding factors then the sensitivity of MCBO is as low as CBO. This we can observe in the case of Lucene. Since the Lucene is having considerable variations in path lengths between any two classes that are connected in a transitive manner. In other two applications JPCAP and RASIN are having a minimal number of paths between two classes and also the variation between any two paths is negligible. The similar kind of performance can be observed for MCH with confounding factor. Fig 4 indicating the advantage of MCH with the number of paths as confounding factors over LCOM. The significance of the number of paths as confounding factor can be observed in the case of JPCAP. In majority classes the number of paths builds between same methods of the class. Hence the performance of the MCH without confounding factor is as low as LCOM(see fig 4).

## VII. Conclusion

These results clearly demonstrate that the proposed metrics MCBO and MCH for coupling and coherence are very good predictors for fault proneness. It is clearly identified that

1. Mediated coupling between two objects is having an impact of the number of connections and path length variation as confounding factors.
2. Mediated Cohesion between the methods of a class is having an impact of the number of paths build between any two methods of a class as confounding factors.

These two metrics MCBO and MCH are measuring as numeric values rather in binary quantity.

The performance of mediated coupling between objects and mediated cohesion of class is miles ahead over CBO and LCOM, The number of paths and length of the paths concluded as confounding factors that influence the performance of the MCBO and MCH.

## References Références Referencias


1. K.K.Aggarwal, Yogesh Singh, ArvinderKaur, RuchikaMalhotra, "Analysis of Object-Oriented Metrics", International Workshop on Software Measurement (IWSM), Montréal, Canada , 2005.
2. L.Briand, J.Daly and J. Wust, "A Unified Framework for Cohesion Measurement in Object-Oriented Systems", Empirical Software Engineering, 3, 65-117, 1998.
3. L.Briand ,J.Daly and J. Wust, "A Unified Framework for Coupling Measurement in Object-Oriented Systems", IEEE Transactions on software Engineering, vol. 25, 91-121, 1999.
4. J.Bieman, B.Kang, "Cohesion and Reuse in an Object-Oriented System", Proc. CM Symp. Software Reusability (SSR'94), 259-262, 1995.
5. M.Cartwright, M.Shepperd, "An Empirical Investigation of an Object- Oriented Software System", IEEE Transactions of Software Engineering. vol.26, Issue 8, 786 – 796, Aug. 2000.
6. S.Chidamber and C.Kemerer, "A metrics Suite for Object-Oriented Design", IEEE Trans. Software Engineering, vol. SE-20, no.6, 476-493, 1994.
7. S.Chidamber, C. Kemerer, "Towards a Metrics Suite for Object Oriented design", Proc. Conference on Object-Oriented Programming: Systems, Languages and Applications (OOPSLA'91), Published in SIGPLAN Notices, vol 26 no. 11, 197-211, 1991.
8. R.Harrison, S.J.Counsell, R.V.Nithi, "An Evaluation of MOOD set of Object- Oriented Software Metrics", IEEE Trans. Software Engineering, vol. SE-24, no.6, 491-496, 1998.
9. B.Henderson-sellers, "Object-Oriented Metrics, Measures of Complexity", Prentice Hall, 1996.
10. M.Hitz, B. Montazeri, "Measuring Coupling and Cohesion in Object-Oriented Systems", Proc. Int. Symposium on Applied Corporate Computing, Monterrey, Mexico, 1995.
11. A.Lake, C.Cook, "Use of factor analysis to develop OOP software complexity metrics", Proc. 6th Annual Oregon Workshop on Software Metrics, Silver Falls, Oregon, 1994.
12. W.Li, S.Henry, "Object-Oriented Metrics that Predict Maintainability', Journal of Systems and Software, vol. 23, no.2, 111-122, 1993.
13. Y.Lee, B.Liang, S.Wu, F.Wang, "Measuring the Coupling and Cohesion of an Object-Oriented program based on Information flow", International Conference on Software Quality, Maribor, Slovenia 1995.
14. M.Lorenz, J.Kidd, "Object-Oriented Software Metrics", Prentice-Hall, 1994.
15. D.Tegarden, S. Sheetz, D.Monarchi, "A Software Complexity Model of Object- Oriented Systems", Decision Support Systems, vol. 13 no.3-4, 241-262, 1995.
16. V.Basili, L.Briand, W.Melo, "A Validation of Object-Oriented Design Metrics as Quality Indicators", IEEE Transactions on Software Engineering, vol. 22 no.10, 751-761, 1996.
17. A.Binkley and S.Schach, "Validation of the Coupling Dependency Metric as a risk Predictor", International Conference on Software Engineering (ICSE), 452- 455, 1998.
18. L.Briand ,J.Daly, V.Porter, J. Wust, "Exploring the relationships between design measures and software quality", Journal of Systems and Software, vol. 5, 245- 273, 2000.
19. L. Briand, J. Wüst, H. Lounis, "Replicated Case Studies for Investigating Quality Factors in Object-Oriented Designs, Empirical Software Engineering: An International Journal, vol 6, no 1, 11-58, 2001.
20. S.Chidamber, D. Darcy, C. Kemerer, "Managerial use of Metrics for Object- Oriented Software: An Exploratory Analysis", IEEE Transactions on Software Engineering, vol.24, no.8, 629-639, 1998.
21. K.ElEmam, S. Benlarbi, N.Goel , S. Rai, "A Validation of Object-Oriented Metrics", Technical Report ERB-1063, National Research Council of Canada (NRC), 1999.
22. K.ElEmam, W. Melo, J. Machado, "The Prediction of Faulty Classes Using Object-Oriented Design Metrics", Journal of Systems and Software, vol. 56, 63- 75, 2001.
23. T.Gyimothy, R. Ferenc I. Siket, "Empirical validation of object-oriented metrics on open source software for fault prediction", IEEE Trans. Software Engineering, vol. 31, Issue 10, 897 – 910, Oct. 2005.
24. Yu Ping, Ma Xiaoxing, LuJian "Predicting Degree of Fault prone using OO Metrics: An Industrial Case Study, CSMR 2002, Budapest, Hungary, 99-107.

This page is intentionally left blank

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# An Approach to Email Classification Using Bayesian Theorem

By Denil Vira, Pradeep Raja & Shidharth Gada
*K.J.Somaiya College of Engineering, Mumbai, India*

*Abstract -* Email Classifiers based on Bayesian theorem have been very effective in Spam filtering due to their strong categorization ability and high precision. This paper proposes an algorithm for email classification based on Bayesian theorem. The purpose is to automatically classify mails into predefined categories. The algorithm assigns an incoming mail to its appropriate category by checking its textual contents. The experimental results depict that the proposed algorithm is reasonable and effective method for email classification.

*Keywords :* *Bayesian, Email classification, tokens, text, probability, keywords.*

*GJCST-C Classification:* *E.5*


*Strictly as per the compliance and regulations of:*

# An Approach to Email Classification Using Bayesian Theorem

Denil Vira [α], Pradeep Raja [σ] & Shidharth Gada [ρ]

*Abstract* - Email Classifiers based on Bayesian theorem have been very effective in Spam filtering due to their strong categorization ability and high precision. This paper proposes an algorithm for email classification based on Bayesian theorem. The purpose is to automatically classify mails into predefined categories. The algorithm assigns an incoming mail to its appropriate category by checking its textual contents. The experimental results depict that the proposed algorithm is reasonable and effective method for email classification.

*Keywords : Bayesian, Email classification, tokens, text, probability, keywords.*

## I. Introduction

Internet e-mail is an essential communication method for most computer users and has been treated as a powerful tool intended to idea and information exchange, as well as for users' commercial and social lives. Globalization has resulted in an exponential increase in the volume of e- mails. Nowadays, a typical user receives about 40-50 email messages every day. For some people hundreds of messages are usual. Thus, users spend a significant part of their working time on processing email. As the popularity of this mean of communication is growing, the time spent on reading and answering emails will only increase. At the same time, a large part of email traffic consists of non-personal, non time critical information that should be filtered. Irrelevant emails greatly affect the efficiency and accuracy of the aimed processing work. As a result, there has recently been a growing interest in creating automated systems to help users manage an extensive email flow.

Consider the following scenario –

You have just returned from a relaxing two week vacation. There has been no phone, no email for two wonderful weeks, and now you are back. You open your inbox and ... Wow! There are 347 new messages! How could you manage to read all of them? Probably, you will spend the whole day trying to sort out all this mail. Having done this burdensome work, you feel like you need a vacation again. What is worse is that most of those messages are out of your interest or out of date.

Here comes the need for automatic email classification system that would sort the important and the unimportant mails, thus saving a much precious time of the users.

The rest of the paper is organized as follows. The next section describes the generic Bayesian filtering logic. Section 3 introduces the proposed algorithm for email classification. Section 4 presents experiments and results. The final section consists of the conclusion.

## II. Bayesian Filter

Bayesian filter has been used widely in building spam filters. The Naïve Bayes classifier is based on the Baye's rule of conditional probability. It makes use of all the attributes contained in the data, and analyses them individually as though they are equally important and independent of each other. The rule for conditional probability is as follows

$$P(H \mid E) = P(E \mid H)P(H) / P(E) \qquad \text{.. Eq.} \quad (1)$$

Where $P(H/E)$ is the conditional probability that hypothesis H is true given an evidence E; $P(E/H)$ the conditional probability of E given H, $P(H)$ the prior probability of H, $P(E)$ the prior probability of E. In the case of spam classification, hypothesis H can be defined as spam or legitimate given an email E.

In applying this theorem, an email needs to be tokenized, and the extracted *n* tokens (i.e. words or phrases) are then used as the evidences E{e1, e2,…,en}, and their probability of spam or non-spam is calculated from previous emails that have been classified. Using past classified emails to estimate the probabilities of the tokens belonging to spam or no spam is a learning process and then they are used to predict the spam probability of a new incoming email. Assume that n key-words are extracted from the content of an email as evidences, then the probability that the email is spam can be calculated by:

$$P(S|E(e1,e2..en)) = P(E(e1,e2,..en)|S)*P(S) / P(E(e1,e2..en) \qquad \text{… Eq.} \quad (2)$$

Where, $P(S/E)$ is the probability that an email E is spam S. In practice, an assumption is commonly made naïvely for simplifying calculation, that is, all the evidences are independent from each other. For example, if an email contains four evidences: e1, e2, e3 and e4, then the joint and conditional probabilities given S can be easily calculated by,

*Author α, σ, ρ : Department of Computer Engineering, K.J.Somaiya College of Engineering, Mumbai, India.*
*E-mail α : denilvira@gmail.com*
*E-mail σ : pradeepraja13@gmail.com*
*E-mail ρ : siddgada@gmail.com*

$P(e1,e2,e3,e4) = P(e1)P(e2)P(e3)P(e,),$
$P(e1,e2,e3,e4|S) = P(e1|S)P(e2|S)P(e3|S)P(e4|S)$

Thus the probability that a given email is considered spam when evidences (e*1, e2, e3* and *e4*) appear in the given email is calculated by Equation (2), and a decision can be made if it is higher than a pre-set decision threshold, 0.5, usually.

## III. Proposed Algorithm for Email Classification

The entire email classification process is divided into two phases.

*a) First phase is the Training phase/Learning phase*

During this phase, the classifier will be trained to recognize attributes for each category. So that later on when a new mail arrives, it compares the attributes of the mail with attributes of each category and the mail is classified into the category having most similar attributes as that of the mail. To build the attribute list (also referred as keywords database) for each category, the emails will be classified manually in to different categories by the user. For better understanding of the working, we create two categories for emails say, *work* and *personal*.

The user manually specifies whether each mail belongs to *work* or *personal* category based on its contents including the subject and the body.

For each email classified manually by the user, the algorithm extracts keywords from the mail and stores them into the keyword database for that particular category along with count of the number of occurrences. For eg. If the user classifies the mail as *work,* the keywords extracted from that mail will be stored in *work_database.* Every category will have its own set of keywords database.

Also, before the mail is processed for keywords, it is first filtered. Mail is divided into set of tokens separated by blank space or any other punctuation marks of English language. Proverbs, articles, html tags, noise words and other unnecessary contents are removed and then the keywords are extracted. Thus the training phase is responsible for building the keyword database for each predefined category.

*b) The second is Classification phase*

Once the learning is sufficiently done, the algorithm is ready to move to next phase. The new mails that arrive should to be automatically assigned their categories. Basically, here we compare the contents of the mail with those of all category keyword databases using Bayesian theorem and look for best matching category for the mail. The new incoming mail (also referred as unclassified mail) is broken into tokens and filtered. The tokens are then compared with keyword databases of each category. The probability that the mail belongs to a category is found out for every category. The category for which the probability outcome is highest, it is then compared with the threshold of same category and classified into that category if following condition holds true else the mail stays unclassified.

$P_n = 1$ , if P(category | E) > threshold
$P_n = 0$ , if P(category | E) < threshold
Where E is the newly arrived mail.

## IV. Algorithm

*a) Training /Learning*

1. *For each mail, specify its category manually.*
2. *Divide the mail into tokens (both subject and body)*
3. *Filter out stop words such as html tags, articles, proverbs, noise words.*
4. *Extract keywords and store them along with frequency count in to keywords database of the selected category.*

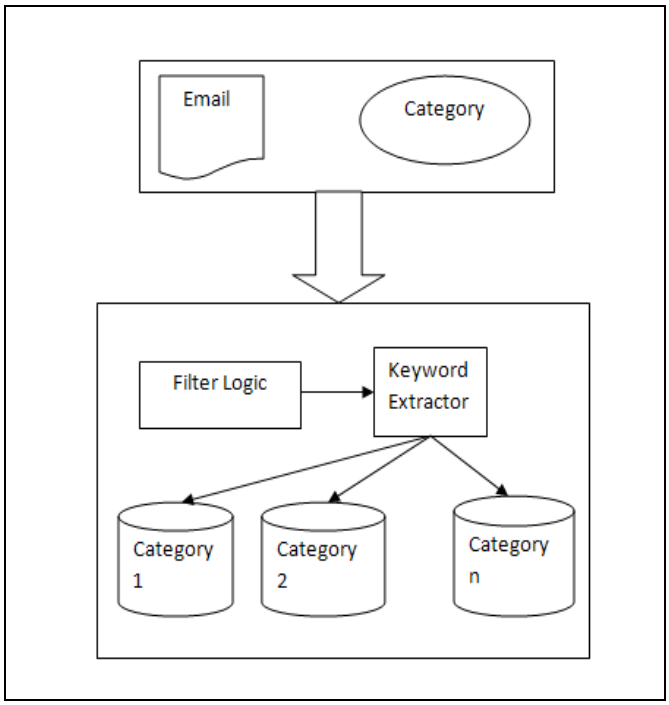



*Figure 1 :* Training process

*b) Classification*

1. *For each newly arrived mail, divide the mail into set of tokens. (Consider both, the subject and the body)*
2. *Filter out stop words such as html tags, articles, proverbs, noise words and extract the keywords, say E {e1,e2, e3…en} is the list of extracted keywords.*
3. *Find P(category| E) = P(E | category)*P(category)/ P(E) for all categories where , P (E | category) = P(e1|category)*P(e2|category)*…..*P(en | category)*
4. *Find the category for which the value of P(category | E) is highest.*

5. *Compare the value with threshold value of that category.*

*If P(category | E ) > threshold , the mail is classified into that category.*

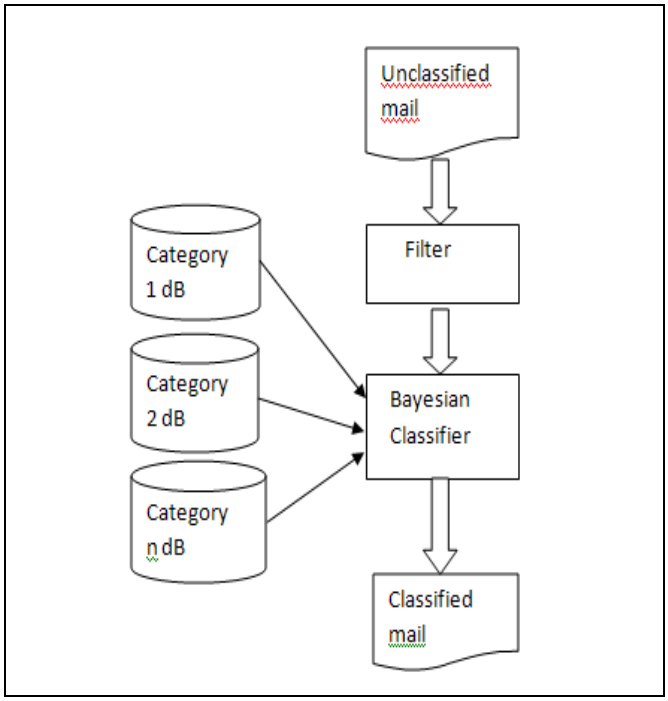



*Figure 2 :* Classification process

## V. Implementation and Results

The implementation of the algorithm was done in Java. Authors' working email accounts were used for fetching the mails. A total of 5175 e-mails are used for the purpose of experiment.

In classification task, the performance of algorithm is measured in terms of accuracy.

Accuracy = $N_c / N$

$N_c$ : Number of correctly classified mails

N : Number of total mails classified

The accuracy of classification is dependent on several parameters such as number of mails considered during training phase, number of categories defined, threshold value of each category , size of mail etc.

We initially created two categories, *work* & *personal* and then gradually added three more categories. The test was repeated for different values of parameters. Through repeated tests, we choose the best-performing configuration for our algorithm. 10-fold cross-validation was used in our test: the corpus was partitioned randomly into ten parts, and the experiment was repeated ten times, each time reserving a different part for testing and using the remaining nine parts for training. All the results were then averaged over the ten iterations.

*Table 1 :* Performance results of Email classifier based on proposed algorithm

| Size of training data | No. of categories | Accuracy |
|---|---|---|
| 100 | 2 | 0.841 |
| 200 | 2 | 0.925 |
| 600 | 2 | 0.992 |
| 1000 | 2 | 0.991 |
| 200 | 3 | 0.936 |
| 500 | 3 | 0.973 |
| 1000 | 3 | 0.995 |
| 200 | 4 | 0.762 |
| 500 | 4 | 0.836 |
| 1000 | 4 | 0.861 |
| 200 | 5 | 0.846 |
| 500 | 5 | 0.921 |
| 1000 | 5 | 0.967 |

The size of training data is the number of mails manually classified during learning phase.

Note that the accuracy is also dependent on number of overlapping words among different categories. We observed that when two different categories have many keywords in common, the classifier accuracy is low. With distinct keywords for each category, with minimum overlapping, the classifier accuracy was above 0.9. The results also show that larger the size of training data better is the accuracy.

## VI. Conclusion

Considering the requirements of improving efficiency in processing emails, this paper introduced an approach to classify mails based on Bayesian theorem. The algorithm was implemented and the experimental results were studied. The results show that our approach to classify mails is a reasonable and effective one. However, there is lot of work to be done in future. The future research includes improving accuracy of the classifier ,working with languages other than English, making the keyword extraction more efficient, classifying based on attachments, understanding the semantics of email text and so on.

## VII. Acknowledgements

The authors would like to acknowledge the contribution of Mr.Dhruven Vora and Mr.Jenish Jain for helping us in coding of the algorithm and testing of results. We would also like to acknowledge Ms.Prof. Kavita Kelkar for valuable suggestions and guidance. Lastly, we would like to thank the entire Department of Computer Engineering, K.J.Somaiya College of Engineering, Mumbai for providing us with all the resources needed for the research.

## References Références Referencias


1. M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A Bayesian approach to filtering junk e-mail," Proceedings of AAAI-98 Workshop on Learning for Text Categorization, pp. 55-62, 1998.
2. I.Androutsopoulos, G. Paliouras, E. Michelakis, "Learning to Filter Unsolicited Commercial E-Mail," Technical Report of National Centre for Sciential Research "Demokritos", 2004.
3. I. Androutsopoulos, J. Koutsias, K.V. Chandrinos, G. Paliouras and C.D. Spyropoulos, "An Evaluation of Naïve Bayesian Anti-Spam Filtering," Proc. of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning (ECML 2000), Barcelona, Spain, 2000, pp. 9-17.
4. S. Youn and D. McLeod, "Efficient Spam e-mail Filtering using Adaptive Ontology," International Conference on Information Technology (ITNG'07), pp.249-254, 2007.
5. S. Hershkop, and J. Stolfo, "Combining e-mail models for false positive reduction," Proc of KDD'05 of ACM. Chicago : [s.n.], pp. 98—107, 2005.
6. Andrew McCallum, Kamal Nigam, "A Comparison of Event Models for Naive Bayes Text Classification," AAAI-98 Workshop on Learning for Text Categorization, Technical Report WS-98-05, 1998, pp. 41-48.
7. Wang, W. et al (2000): Diversity between neural networks and decision trees for building multiple classifier systems. *Multiple Classifier Systems*: pp 240-249.
8. R. Beckermann, A. McCallum, and G. Huang. Automatic categorization of email into folders: benchmark experiments on Enron and SRI corpora. Technical report IR-418, University of Massachusetts Amherst, 2004.
9. F. Peng, D. Schuurmans, and S. Wang. Augmenting naive bayes classifiers with statistical language models. Information Retrieval, 7:317–345, 2004.
10. David D. Lewis and William A. Gale. A sequential algorithm for training text classfiers. In Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval, 1994.

Global Journal of Computer Science and Technology
Software & Data Engineering
Volume 12 Issue 13 Version 1.0 Year 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# Bayesian Classifiers Programmed in SQL Using PCA

By K.Venkat Nagarjuna & P.V Subba Reddy
*QIS College of Engg & Technology Ongole, Andhrapradesh, India*

*Abstract -* The Bayesian classifier is a fundamental classification technique. We also consider different concepts regarding Dimensionality Reduction techniques for retrieving lossless data. In this paper, we proposed a new architecture for pre-processing the data. Here we improved our Bayesian classifier to produce more accurate models with skewed distributions, data sets with missing information, and subsets of points having significant overlap with each other, which are known issues for clustering algorithms. so, we are interested in combining Dimensionality Reduction technique like PCA with Bayesian Classifiers to accelerate computations and evaluate complex mathematical equations. The proposed architecture in this project contains the following stages: pre-processing of input data, Naïve Bayesian classifier, Bayesian classifier, Principal component analysis, and database. Principal Component Analysis(PCA) is the process of reducing components by calculating Eigen values and Eigen Vectors. We consider two algorithms in this paper: Bayesian Classifier based on KMeans( BKM) and Naïve Bayesian Classifier Algorithm(NB).

*Keywords :* *Dimensionality Reduction, PCA, Classifiers, K-means.*

*GJCST-C Classification:* *H.2.3*


*Strictly as per the compliance and regulations of:*

# Bayesian Classifiers Programmed in SQL Using PCA

K.Venkat Nagarjuna[α] & P.V Subba Reddy[σ]

*Abstract* - The Bayesian classifier is a fundamental classification technique. We also consider different concepts regarding Dimensionality Reduction techniques for retrieving lossless data. In this paper, we proposed a new architecture for pre-processing the data. Here we improved our Bayesian classifier to produce more accurate models with skewed distributions, data sets with missing information, and subsets of points having significant overlap with each other, which are known issues for clustering algorithms. so, we are interested in combining Dimensionality Reduction technique like PCA with Bayesian Classifiers to accelerate computations and evaluate complex mathematical equations. The proposed architecture in this project contains the following stages: pre-processing of input data, Naïve Bayesian classifier, Bayesian classifier, Principal component analysis, and database. Principal Component Analysis(PCA) is the process of reducing components by calculating Eigen values and Eigen Vectors. We consider two algorithms in this paper: Bayesian Classifier based on KMeans( BKM) and Naïve Bayesian Classifier Algorithm(NB).

*Keywords : Dimensionality Reduction, PCA, Classifiers, K-means.*

## I. Introduction

In this paper, we focus on programming Bayesian classifiers in SQL using Principal Component Analysis(PCA). PCA allows us to compute a linear transformation that maps data from a high dimensional space to a lower dimensional space.PCA "combines" the essence of attributes by creating an alternative, smaller set of variables. In this paper, We studied two complementary aspects: increasing accuracy and generating efficient SQL code. We introduce two classifiers: Naive Bayes and a classifier based on class decomposition using K-means clustering. We consider two complementary tasks: model computation and scoring a data set. We study several layouts for tables and several indexing alternatives. We analyse how to transform equations into efficient SQL queries and introduce several query optimizations.

Our contributions are the following: We present two efficient SQL implementations of Naı¨ve Bayes for numeric and discrete attributes. We introduce a classification algorithm that builds one clustering model per class, which is a generalization of K-means [1], [4]. Our main contribution is a Bayesian classifier programmed in SQL, extending Naı¨ve Bayes, which uses K-means to decompose each class into clusters. We generalize queries for clustering adding a new problem dimension. That is, our novel queries combine three dimensions: attribute, cluster, and class subscripts. We identify Euclidean distance as the most time-consuming computation. Thus, we introduce several schemes to efficiently compute distance considering different storage layouts for the data.

*Author α : M.tech student, Dept of CSE, QIS College of Engg & Technology Ongole, Andhrapradesh, India.*
*Author σ : Associate Professor, Dept of CSE, QIS College of Engg & Technology, Ongole, Andhrapradesh, India.*

## II. Definitions

We focus on computing classification models on a data set X={x1 . . . ; xn} with d attributes X1. . . ..Xd, one discrete attribute G(class or target), and n records (points). We assume G has m=2values. Data set X represents a d x n matrix, where xi represents a column vector. We study two complementary models: 1) each class is approximated by a normal distribution or histogram and 2) fitting a mixture model with k clusters on each class with K-means. We use subscripts I, j, h, g as follows: i=1 . . . n; j= 1 . . . k; h=1 . . . d; g=1 . . .m. The T superscript indicates matrix transposition.

Throughout the paper, we will use a small running example, where d=4; k=3 (for K-means) and m=2 (binary).

## III. Bayesian Classifiers Programmed in SQL using Principal Component Analysis

### a) Classification

Classification and prediction are two forms of data analysis that can be used to extract models describing important data classes or to predict future data trends. Such analysis can help provide us with a better understanding of the data at large. Whereas classification predicts categorical (discrete, unordered) labels, prediction models continuous valued functions.

### b) Bayesian Classification

The Bayesian Classification represents a supervised learning method as well as a statistical method for classification. It can solve diagnostic and predictive problems.

This Classification is named after Thomas Bayes (1702-1761), who proposed the Bayes Theorem. Bayesian classification provides practical learning algorithms and prior knowledge and observed data can be combined. Bayesian Classification provides a useful

perspective for understanding and evaluating many learning algorithms. It calculates explicit probabilities for hypothesis and it is robust to noise in input data. Bayesian classification is based on Bayes' theorem. A simple Bayesian classifier is known as the naïve Bayesian classifier. Bayesian classifiers have exhibited high accuracy and speed when applied to large databases.

*c) Naïve Bayesian Classification*

It is based on the Bayesian theorem It is particularly suited when the dimensionality of the inputs is high. Parameter estimation for naive Bayes models uses the method of maximum likelihood. In spite over-simplified assumptions, it often performs better in many complex real world situations

The main advantage of Naïve Bayesian classification is it requires a small amount of training data to estimate the parameters.

*d) Data Reduction*

Data reduction techniques can be applied to obtain a reduced representation of the data set that is much smaller in volume, yet closely maintains the integrity of the original data.

Strategies for data reduction include the following:

Data cube aggregation
Attribute subset selection
Dimensionality reduction
Numerosity reduction
Discretization and concept hierarchy generation

*e) Dimensionality reduction*

Data encoding or transformations are applied so as to obtain a reduced or "compressed" representation of the original data. If the original data can be reconstructed from the compressed data without any loss of information, the data reduction is called lossless. If, instead, we can reconstruct only an approximation of the original data, then the data reduction is called lossy. Although they are typically lossless, they allow only limited manipulation of the data. In this section, we instead focus on two popular and effective methods of lossy dimensionality reduction: wavelet transforms and principal components analysis.

*f) Principal Components Analysis*

PCA allows us to compute a linear transformation that maps data from a high dimensional space to a lower dimensional space. Suppose that the data to be reduced consist of tuples or data vectors described by n attributes or dimensions. Principal components analysis, or PCA (also called the Karhunen-Loeve, or K-L, method), searches for k n dimensional orthogonal vectors that can best be used to represent the data, where k _ n. The original data are thus projected onto a much smaller space, resulting in dimensionality reduction. Unlike attribute subset selection, which reduces the attribute set size by retaining a subset of the initial set of attributes, PCA "combines" the essence of attributes by creating an alternative, smaller set of variables. The initial data can then be projected onto this smaller set.

*g) Methodology*

Suppose $x_1, x_2, ..., x_M$ are Nx1 vectors

Step 1: $x = \frac{1}{M}\sum_{i=1}^{M} xi$

Step 2: subtract the mean: $F_i = x_i - x$

Step 3: form the matrix $A = [F_1\ F_2\ \ldots\ F_M]$ (NxM matrix), then compute:

$$C = = AA^T \qquad \frac{1}{M}\sum_{i=n}^{M} FnFnT$$

(Sample covariance matrix, NxN, characterizes the scatter of the data)

Step 4: compute the Eigen values of C: $\lambda_1 > \lambda_2 > \ldots > \lambda_N$

Step 5: compute the eigenvectors of C: $u_1, u_2, \ldots, u_N$

Since C is symmetric, $u_1, u_2, \ldots, u_N$ form a basis, (i.e., any vector x or actually $(x - \bar{x})$, can be written as a linear combination of the eigenvectors):

$$x - \bar{x} = b_1u_1 + b_2u_2 + \ldots + b_N u_N = \sum_{i=1}^{N} biui$$

Step 6: (dimensionality reduction step) keep only the terms corresponding to the K largest Eigen values: $\hat{x} - \bar{x} =$

$$\sum_{i=1}^{k} biui$$

where K << N.

These are the steps we should follow to perform principal component analysis (PCA) to reduce dimensionality of the high dimensional data.

*h) Naı¨ve Bayes*

We consider two versions of NB: one for numeric attributes and another for discrete attributes. Numeric NB will be improved with class decomposition. NB assumes attributes are independent, and thus, the joint class conditional probability can be estimated as the product of probabilities of each attribute [2]. We now discuss NB based on a multivariate Gaussian. NB has no input parameters. Each class is modelled as a single normal distribution with mean vector Cg and a diagonal variance matrix Rg. Scoring assumes a model is available and there exists a data set with the same attributes in order to predict class G. Variance computation based on sufficient statistics [5] in one pass can be numerically unstable when the variance is much smaller compared to large attribute values or when the data set has a mix of very large and very small numbers. For ill conditioned data sets, the computed variance can be significantly different from the actual

variance or even become negative. Therefore, the model is computed in two passes: a first pass to get the mean per class and a second one to compute the variance per class. The mean per class is given by Cg = Pxi2Yg xi=Ng, where Yg _ X are the records in class g. Equation Rg = 1=NgPn xi2Ygðxi _CgÞðxi _ CgÞT gives a diagonal variance matrix Rg, which is numerically stable, but requires two passes over the data set.

The SQL implementation for numeric NB follows the mean and variance equations introduced above. We compute three aggregations grouping by g with two queries. The first query computes the mean Cg of class g with a sum ∂XhÞ=count ∂_Þ aggregation and class priors _g with a count() aggregation. The second query computes Rg with sum(∂Xh _ _hÞ2Þ. Note the joint probability computation is not done in this phase. Scoring uses the Gaussian parameters as input to classify an input point to the most probable class, with one query in one pass over X. Each class probability is evaluated as a Gaussian. To avoid numerical issues when a variance is zero, the probability is set to 1 and the joint probability is computed with a sum of probability logarithms instead of a product of probabilities. A CASE statement pivots probabilities and avoids a max() aggregation. A final query determines the predicted class, being the one with maximum probability, obtained with a CASE statement. We now discuss NB for discrete attributes. For numeric NB, we used Gaussians because they work well for large data sets and because they are easy to manipulate mathematically. That is, NB does not assume any specific probability density function (pdf). Assume X1; . . .;Xd can be discrete or numeric. If an attribute Xh is discrete (categorical) NB simply computes its histogram: probabilities are derived with counts per value divided by the corresponding number of points in each class. Otherwise, if the attribute Xh is numeric then binning is required. Binning requires two passes over the data set, pretty much like numeric NB. In the first pass, bin boundaries are determined. On the second pass, one dimensional frequency histograms are computed on each attribute.

The bin boundaries (interval ranges) may impact the accuracy of NB due to skewed distributions or extreme values. Thus, we consider two techniques to bin attributes: 1) creating k uniform intervals between min and max and 2) taking intervals around the mean based on multiples of the standard deviation, thus getting more evenly populated bins. We do not study other binning schemes such as quantiles (i.e., equidepth binning).The implementation in SQL of discrete NB is Straight forward. For discrete attributes, no pre-processing is required. For numeric attributes, the minimum, maximum, and mean can be determined in one pass in a single query. The variance for all numeric attributes is computed on a second pass to avoid numerical issues. Then, each attribute is discretized finding the interval for each value. Once we have a binned version of X, then we compute histograms on each attribute with SQL aggregations. Probabilities are obtained dividing by the number of records in each class. Scoring requires determining the interval for each attribute value and retrieving its probability. Each class probability is also computed by adding logarithms. NB has an advantage over other classifiers: it can handle a data set with mixed attribute types (i.e.,discrete and numerical).be in single-column format and must be centered.

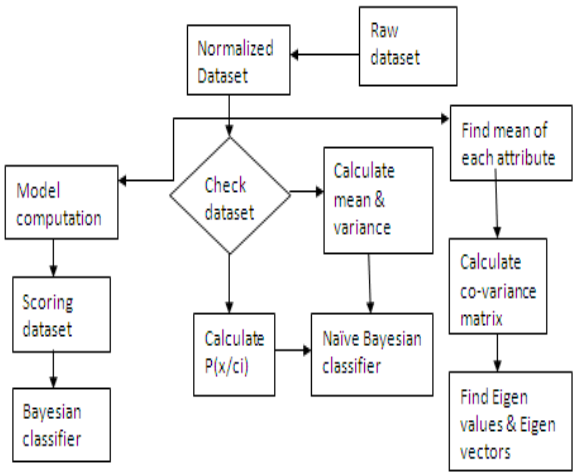



*Figure 3 :* Architecture of the proposed method

## IV. Algorithms

We consider two algorithms in this paper: Bayesian Classifier Based on K-Means(BKM) algorithm and Naïve Bayesian classifier algorithm(NB).

*a) Naïve Bayesian classifier algorithm(NB)*

NB has no input parameters. Each class is modeled as a single normal distribution with mean vector (Cg) and a diagonal variance matrix (Rg). Therefore, the model is computed in two passes:

1. A first pass to get the mean per class and
2. Second one to compute the variance per class.

*b) Bayesian Classifier Based on K-Means*

We now present BKM, a Bayesian classifier based on class decomposition obtained with the K-means algorithm. BKM is a generalization of NB, where NB has one cluster per class and the Bayesian classifier has k > 1 clusters per class. L, the linear sum of points; L,Q, the Gaussian parameters.

We generalize K-means to compute m models, fitting a mixture model to each class. K-means is initialized, and then, it iterates until it converges on all classes.

The algorithm is as given below.
Initialization:

1. Get global N; L;Q and mean and standard deviation
2. Get k random points per class to initialize C.

While not all m models converge:

1. E step: get k distances j per g; find nearest cluster j per g;update N; L;Q per class.
2. M step: update W;C;R from N; L;Q per class; compute model quality per g; monitor convergence.

## V. Experimental Evaluation

We analyze three major aspects: 1) classification accuracy, 2) query optimization, and 3) time complexity and speed. We compare the accuracy of NB, BKM, and decision trees (DTs).

*a) Setup*

We used the Teradata DBMS running on a server with a 3.2 GHz CPU, 2 GB of RAM, and a 750 GB disk. Parameters were set as follows: We set €= 0:001 for K-means. The number of clusters per class was k=4 (setting experimentally justified). All query optimizations were turned on by default (they do not affect model accuracy).experiments with DTs were performed using a data mining tool. We used real data sets to test classification accuracy (from the UCI repository) and synthetic data sets to analyze speed (varying d; n). Real data sets include pima (d = 6; n = 768), spam (d = 7; n = 4.601), bscale (d = 4; n = 625), and wbcancer (d =7; n = 569). Categorical attributes (≥3 values) were transformed into binary attributes.

*b) Model Accuracy*

In this section, we measure the accuracy of predictions when using Bayesian classification models. We used 5-fold cross validation for each run, the data set was partitioned into a training set and a test set. The training set was used to compute the model, whereas the test set was used to independently measure accuracy. The training set size was 80 percent and the test set was 20 percent.

Comparing Accuracy: NB, BKM, and DT

| Dataset | Algorithm | Global | Class-0 | Class-1 |
|---|---|---|---|---|
| pima | NB | 76% | 80% | 68% |
| | BKM | 76% | 87% | 53% |
| | DT | 68% | 76% | 53% |
| spam | NB | 70% | 87% | 45% |
| | BKM | 73% | 91% | 43% |
| | DT | 80% | 85% | 72% |
| bscale | NB | 50% | 51% | 30% |
| | BKM | 59% | 59% | 60% |
| | DT | 89% | 96% | 0% |
| wbcancer | NB | 93% | 91% | 95% |
| | BKM | 93% | 84% | 97% |
| | DT | 95% | 94% | 96% |

BKM ran until K-means converged on all classes. Decision trees used the CN5.0 algorithm splitting nodes until they reached a minimum percentage of purity or became too small. Pruning was applied to reduce over fit. The number of clusters for BKM by default was k = 4.

*c) Query Optimization*

Our best distance strategy is two orders of magnitude faster than the worst strategy and it is one order of magnitude faster than its closest rival. The explanation is that I/O is minimized to n operations and computations happen in main memory for each row through SQL arithmetic expressions. Note a standard aggregation on the pivoted version of X (XV) is faster than the horizontal nested query variant .Table 6 compares SQL with UDFs to score the data set(computing distance and finding nearest cluster per class). We exploit scalar UDFs [5]. Since finding the nearest cluster is straight forward in the UDF, this comparison considers both computations as one. This experiment favors the UDF since SQL requires accessing large tables in separate queries. As we can see, SQL (with arithmetic expressions) turned out be faster than the UDF. This was interesting because both approaches used the same table as input and performed the same I/O reading a large table. We expected SQL to be slower because it required a join and XH and XD were accessed. The explanation was that the UDF has overhead to pass each point and model as parameters in each call.

Query Optimization: Distance Computation $n = 100$k (Seconds)

| Distance scheme | $d = 8$ $k = 8$ | $d = 8$ $k = 16$ | $d = 16$ $k = 8$ | $d = 16$ $k = 16$ |
|---|---|---|---|---|
| Horizontal | 2 | 7 | 2 | 7 |
| Horizontal temp | 80 | 211 | 99 | 225 |
| Horizontal nested q. | 201 | 449 | 311 | 544 |
| Hybrid Vertical | 84 | 155 | 146 | 155 |

Query Optimization: SQL versus UDF (Scoring, $n = 1$M) (Seconds)

| $k$ | $d = 4$ SQL | $d = 4$ UDF | $d = 8$ SQL | $d = 8$ UDF |
|---|---|---|---|---|
| 2 | 23 | 36 | 34 | 61 |
| 4 | 31 | 47 | 42 | 71 |
| 6 | 40 | 55 | 61 | 85 |
| 8 | 56 | 62 | 66 | 111 |

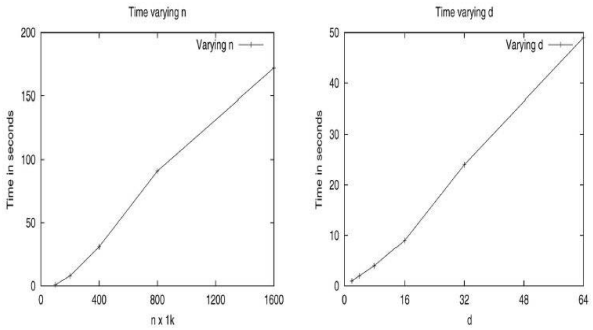



*Fig. 1 :* BKM Classifier: Time Complexity; (default d=4, k=4, n=100k)

*d) Speed and Time Complexity*

We compare SQL and C++ running on the same computer. We also comp are the time to export with ODBC. C++ worked on flat files exported from the DBMS. We used binary files in order to get maximum performance in C++. Also, we shut down the DBMS when C++ was running. In short, we conducted a fair comparison. Table 7 compares SQL, C++, and ODBC varying n. Clearly, ODBC is a bottleneck. Overall, both languages scale linearly. We can see SQL performs better as n grows because DBMS overhead becomes less important. However, C++ is about four times faster. Fig. 1 shows BKM time complexity varying n; d with large datasets. Time is measured for one iteration. BKM is linear in n and d , highlighting its scalability.

## VI. Related Work

The most widely used approach to integrate data mining Algorithms into a DBMS is to modify the internal source code. Scalable K-means (SKM) [1] and O-cluster [3] are two examples of clustering algorithms internally integrated with a DBMS. A discrete Naive Bayes classifier has been internally integrated with the SQL Server DBMS. On the other hand, the two main mechanisms to integrate data mining algorithms without modifying the DBMS source code are SQL queries and UDFs. A discrete Naı¨ve Bayes classifier programmed in SQL is introduced in [6]. We summarize differences with ours. The data set is assumed to have discrete attributes: binning numeric attributes is not considered. Ituses an inefficient large pivoted intermediate table, whereas our discrete NB model can directly work on a horizontal layout. Our proposal extends clustering algorithms in SQL [4] to perform classification, generalizing Naı¨ve Bayes [2]. K-means clustering was programmed with SQL queries introducing three variants [4]: standard, optimized, and incremental. We generalized the optimized variant. Note that classification represents a significantly harder problem than clustering. User-Defined Functions are identified as an important extensibility mechanism to integrate data mining algorithms [5], [8]. Atlas [8] extends SQL syntax with object-oriented constructs to define aggregate and table functions (with initialize, iterate, and terminate clauses), providing a user friendly interface to the SQL standard. We point out several differences with our work. First, we propose to generate SQL code from a host language, thus achieving Turing-completeness in SQL (similar to embedded SQL). We showed the Bayesian classifier can be solved more efficiently with SQL queries than with UDFs. Even further, SQL code provides better portability and Atlas requires modifying the DBMS source code. Class decomposition with clustering is shown to improve NB accuracy [7]. The classifier can adapt to skewed distributions and overlapping subsets of points by building better local models. In [7], EM was the algorithm to fit a mixture per class. Instead, we decided to use K-means because it is faster, simpler, better understood in database research and has less numeric issues.

## VII. Conclusions

We presented two Bayesian classifiers programmed in SQL: the Naı¨ve Bayes classifier (with discrete and numeric Versions) and a generalization of Naı¨ve Bayes (BKM), based on decomposing classes with K-means clustering. We studied two complementary aspects: increasing accuracy and generating efficient SQL code. We introduced query optimizations to generate fast SQL code. The best physical storage layout and primary index for large tables is based on the point subscript. Sufficient statistics are stored on denormalized tables. The Euclidean distance computation uses a flattened (horizontal) version of the cluster centroids matrix, which enables arithmetic expressions. The nearest cluster per class, required by Kmeans, is efficiently determined avoiding joins and aggregations. Experiments with real data sets compared NB, BKM, and decision trees. The numeric and discrete versions of NB had similar accuracy.BKM was more accurate than NB and was similar to decision trees in global accuracy. However, BKM was more accurate when computing a breakdown of accuracy per class. A low number of clusters

produced good results in most cases. We compared Equivalent implementations of NB in SQL and C++ with large data sets: SQL was four times slower. SQL queries were faster than UDFs to score, highlighting the importance of our optimizations. NB and BKM exhibited linear scalability in data set size and dimensionality. There are many opportunities for future work. We want to derive incremental versions or sample-based methods to accelerate the Bayesian classifier. We want to improve our Bayesian classifier to produce more accurate models with skewed distributions, data sets with missing information, and subsets of points having significant overlap with each other, which are known issues for clustering algorithms. We are interested in combining dimensionality reduction techniques like PCA or factor analysis with Bayesian classifiers. UDFs need further study to accelerate computations and evaluate complex mathematical equations.

## References Références Referencias


1. P. Bradley, U. Fayyad, and C. Reina, "Scaling Clustering Algorithms to Large Databases," Proc. ACM Knowledge Discovery and Data Mining (KDD) Conf., pp. 9-15, 1998.
2. T. Hastie, R. Tibshirani, and J.H. Friedman, The Elements of Statistical Learning, first ed. Springer, 2001.
3. B.L. Milenova and M.M. Campos, "O-Cluster: Scalable Clustering of Large High Dimensional Data Sets," Proc. IEEE Int'l Conf. Data Mining (ICDM), pp. 290-297, 2002.
4. C. Ordonez, "Integrating K-Means Clustering with a Relational DBMS Using SQL," IEEE Trans. Knowledge and Data Eng., vol. 18, no. 2, pp. 188-201, Feb. 2006.
5. C. Ordonez, "Building Statistical Models and Scoring with UDFs," Proc. ACM SIGMOD, pp. 1005-1016, 2007.
6. S. Thomas and M.M. Campos, SQL-Based Naive Bayes Model Building and Scoring, US Patent 7,051,037, US Patent and Trade Office, 2006.
7. R. Vilalta and I. Rish, "A Decomposition of Classes via Clustering to Explain and Improve Naive Bayes," Proc. European Conf. Machine Learning (ECML), pp. 444-455, 2003.
8. H. Wang, C. Zaniolo, and C.R. Luo, "ATLaS: A Small but Complete SQLExtension for Data Mining and Data Streams "

# Global Journals Inc. (US) Guidelines Handbook 2012

www.GlobalJournals.org

## FELLOW OF ASSOCIATION OF RESEARCH SOCIETY IN COMPUTING (FARSC)

- 'FARSC' title will be awarded to the person after approval of Editor-in-Chief and Editorial Board. The title 'FARSC" can be added to name in the following manner. eg. **Dr. John E. Hall, Ph.D., FARSC or William Walldroff Ph. D., M.S., FARSC**
- Being FARSC is a respectful honor. It authenticates your research activities. After becoming FARSC, you can use 'FARSC' title as you use your degree in suffix of your name. This will definitely will enhance and add up your name. You can use it on your Career Counseling Materials/CV/Resume/Visiting Card/Name Plate etc.
- 60% Discount will be provided to FARSC members for publishing research papers in Global Journals Inc., if our Editorial Board and Peer Reviewers accept the paper. For the life time, if you are author/co-author of any paper bill sent to you will automatically be discounted one by 60%
- FARSC will be given a renowned, secure, free professional email address with 100 GB of space eg.johnhall@globaljournals.org. You will be facilitated with Webmail, Spam Assassin, Email Forwarders, Auto-Responders, Email Delivery Route tracing, etc.
- FARSC member is eligible to become paid peer reviewer at Global Journals Inc. to earn up to 15% of realized author charges taken from author of respective paper. After reviewing 5 or more papers you can request to transfer the amount to your bank account or to your PayPal account.
- Eg. If we had taken 420 USD from author, we can send 63 USD to your account.
- FARSC member can apply for free approval, grading and certification of some of their Educational and Institutional Degrees from Global Journals Inc. (US) and Open Association of Research, Society U.S.A.
- After you are FARSC. You can send us scanned copy of all of your documents. We will verify, grade and certify them within a month. It will be based on your academic records, quality of research papers published by you, and 50 more criteria. This is beneficial for your job interviews as recruiting organization need not just rely on you for authenticity and your unknown qualities, you would have authentic ranks of all of your documents. Our scale is unique worldwide.
- FARSC member can proceed to get benefits of free research podcasting in Global Research Radio with their research documents, slides and online movies.
- After your publication anywhere in the world, you can upload you research paper with your recorded voice or you can use our professional RJs to record your paper their voice. We can also stream your conference videos and display your slides online.
- FARSC will be eligible for free application of Standardization of their Researches by Open Scientific Standards. Standardization is next step and level after publishing in a journal. A team of research and professional will work with you to take your research to its next level, which is worldwide open standardization.

- FARSC is eligible to earn from their researches: While publishing his paper with Global Journals Inc. (US), FARSC can decide whether he/she would like to publish his/her research in closed manner. When readers will buy that individual research paper for reading, 80% of its earning by Global Journals Inc. (US) will be transferred to FARSC member's bank account after certain threshold balance. There is no time limit for collection. FARSC member can decide its price and we can help in decision.

## MEMBER OF ASSOCIATION OF RESEARCH SOCIETY IN COMPUTING (MARSC)

- 'MARSC' title will be awarded to the person after approval of Editor-in-Chief and Editorial Board. The title 'MARSC" can be added to name in the following manner. eg. Dr. John E. Hall, Ph.D., MARSC or William Walldroff Ph. D., M.S., MARSC
- Being MARSC is a respectful honor. It authenticates your research activities. After becoming MARSC, you can use 'MARSC' title as you use your degree in suffix of your name. This will definitely will enhance and add up your name. You can use it on your Career Counseling Materials/CV/Resume/Visiting Card/Name Plate etc.
- 40% Discount will be provided to MARSC members for publishing research papers in Global Journals Inc., if our Editorial Board and Peer Reviewers accept the paper. For the life time, if you are author/co-author of any paper bill sent to you will automatically be discounted one by 60%
- MARSC will be given a renowned, secure, free professional email address with 30 GB of space eg.johnhall@globaljournals.org. You will be facilitated with Webmail, Spam Assassin, Email Forwarders, Auto-Responders, Email Delivery Route tracing, etc.
- MARSC member is eligible to become paid peer reviewer at Global Journals Inc. to earn up to 10% of realized author charges taken from author of respective paper. After reviewing 5 or more papers you can request to transfer the amount to your bank account or to your PayPal account.
- MARSC member can apply for free approval, grading and certification of some of their Educational and Institutional Degrees from Global Journals Inc. (US) and Open Association of Research, Society U.S.A.
- MARSC is eligible to earn from their researches: While publishing his paper with Global Journals Inc. (US), MARSC can decide whether he/she would like to publish his/her research in closed manner. When readers will buy that individual research paper for reading, 40% of its earning by Global Journals Inc. (US) will be transferred to MARSC member's bank account after certain threshold balance. There is no time limit for collection. MARSC member can decide its price and we can help in decision.

# AUXILIARY MEMBERSHIPS

## ANNUAL MEMBER

- Annual Member will be authorized to receive e-Journal GJCST for one year (subscription for one year).
- The member will be allotted free 1 GB Web-space along with subDomain to contribute and participate in our activities.
- A professional email address will be allotted free 500 MB email space.

## PAPER PUBLICATION

- The members can publish paper once. The paper will be sent to two-peer reviewer. The paper will be published after the acceptance of peer reviewers and Editorial Board.

# Process of submission of Research Paper

The Area or field of specialization may or may not be of any category as mentioned in 'Scope of Journal' menu of the GlobalJournals.org website. There are 37 Research Journal categorized with Six parental Journals GJCST, GJMR, GJRE, GJMBR, GJSFR, GJHSS. For Authors should prefer the mentioned categories. There are three widely used systems UDC, DDC and LCC. The details are available as 'Knowledge Abstract' at Home page. The major advantage of this coding is that, the research work will be exposed to and shared with all over the world as we are being abstracted and indexed worldwide.

The paper should be in proper format. The format can be downloaded from first page of 'Author Guideline' Menu. The Author is expected to follow the general rules as mentioned in this menu. The paper should be written in MS-Word Format (*.DOC,*.DOCX).

The Author can submit the paper either online or offline. The authors should prefer online submission.Online Submission: There are three ways to submit your paper:

**(A) (I) First, register yourself using top right corner of Home page then Login. If you are already registered, then login using your username and password.**

**(II) Choose corresponding Journal.**

**(III) Click 'Submit Manuscript'. Fill required information and Upload the paper.**

**(B) If you are using Internet Explorer, then Direct Submission through Homepage is also available.**

**(C) If these two are not convenient, and then email the paper directly to dean@globaljournals.org.**

Offline Submission: Author can send the typed form of paper by Post. However, online submission should be preferred.

# Preferred Author Guidelines

**MANUSCRIPT STYLE INSTRUCTION (Must be strictly followed)**

Page Size: 8.27" X 11'"

- Left Margin: 0.65
- Right Margin: 0.65
- Top Margin: 0.75
- Bottom Margin: 0.75
- Font type of all text should be Swis 721 Lt BT.
- Paper Title should be of Font Size 24 with one Column section.
- Author Name in Font Size of 11 with one column as of Title.
- Abstract Font size of 9 Bold, "Abstract" word in Italic Bold.
- Main Text: Font size 10 with justified two columns section
- Two Column with Equal Column with of 3.38 and Gaping of .2
- First Character must be three lines Drop capped.
- Paragraph before Spacing of 1 pt and After of 0 pt.
- Line Spacing of 1 pt
- Large Images must be in One Column
- Numbering of First Main Headings (Heading 1) must be in Roman Letters, Capital Letter, and Font Size of 10.
- Numbering of Second Main Headings (Heading 2) must be in Alphabets, Italic, and Font Size of 10.

**You can use your own standard format also.**
**Author Guidelines:**

1. General,

2. Ethical Guidelines,

3. Submission of Manuscripts,

4. Manuscript's Category,

5. Structure and Format of Manuscript,

6. After Acceptance.

**1. GENERAL**

Before submitting your research paper, one is advised to go through the details as mentioned in following heads. It will be beneficial, while peer reviewer justify your paper for publication.

**Scope**

The Global Journals Inc. (US) welcome the submission of original paper, review paper, survey article relevant to the all the streams of Philosophy and knowledge. The Global Journals Inc. (US) is parental platform for Global Journal of Computer Science and Technology, Researches in Engineering, Medical Research, Science Frontier Research, Human Social Science, Management, and Business organization. The choice of specific field can be done otherwise as following in Abstracting and Indexing Page on this Website. As the all Global

Journals Inc. (US) are being abstracted and indexed (in process) by most of the reputed organizations. Topics of only narrow interest will not be accepted unless they have wider potential or consequences.

**2. ETHICAL GUIDELINES**

Authors should follow the ethical guidelines as mentioned below for publication of research paper and research activities.

Papers are accepted on strict understanding that the material in whole or in part has not been, nor is being, considered for publication elsewhere. If the paper once accepted by Global Journals Inc. (US) and Editorial Board, will become the copyright of the Global Journals Inc. (US).

**Authorship: The authors and coauthors should have active contribution to conception design, analysis and interpretation of findings. They should critically review the contents and drafting of the paper. All should approve the final version of the paper before submission**

The Global Journals Inc. (US) follows the definition of authorship set up by the Global Academy of Research and Development. According to the Global Academy of R&D authorship, criteria must be based on:

1) Substantial contributions to conception and acquisition of data, analysis and interpretation of the findings.

2) Drafting the paper and revising it critically regarding important academic content.

3) Final approval of the version of the paper to be published.

All authors should have been credited according to their appropriate contribution in research activity and preparing paper. Contributors who do not match the criteria as authors may be mentioned under Acknowledgement.

Acknowledgements: Contributors to the research other than authors credited should be mentioned under acknowledgement. The specifications of the source of funding for the research if appropriate can be included. Suppliers of resources may be mentioned along with address.

**Appeal of Decision: The Editorial Board's decision on publication of the paper is final and cannot be appealed elsewhere.**

**Permissions: It is the author's responsibility to have prior permission if all or parts of earlier published illustrations are used in this paper.**

Please mention proper reference and appropriate acknowledgements wherever expected.

If all or parts of previously published illustrations are used, permission must be taken from the copyright holder concerned. It is the author's responsibility to take these in writing.

Approval for reproduction/modification of any information (including figures and tables) published elsewhere must be obtained by the authors/copyright holders before submission of the manuscript. Contributors (Authors) are responsible for any copyright fee involved.

**3. SUBMISSION OF MANUSCRIPTS**

Manuscripts should be uploaded via this online submission page. The online submission is most efficient method for submission of papers, as it enables rapid distribution of manuscripts and consequently speeds up the review procedure. It also enables authors to know the status of their own manuscripts by emailing us. Complete instructions for submitting a paper is available below.

Manuscript submission is a systematic procedure and little preparation is required beyond having all parts of your manuscript in a given format and a computer with an Internet connection and a Web browser. Full help and instructions are provided on-screen. As an author, you will be prompted for login and manuscript details as Field of Paper and then to upload your manuscript file(s) according to the instructions.

To avoid postal delays, all transaction is preferred by e-mail. A finished manuscript submission is confirmed by e-mail immediately and your paper enters the editorial process with no postal delays. When a conclusion is made about the publication of your paper by our Editorial Board, revisions can be submitted online with the same procedure, with an occasion to view and respond to all comments.

Complete support for both authors and co-author is provided.

**4. MANUSCRIPT'S CATEGORY**

Based on potential and nature, the manuscript can be categorized under the following heads:

Original research paper: Such papers are reports of high-level significant original research work.

Review papers: These are concise, significant but helpful and decisive topics for young researchers.

Research articles: These are handled with small investigation and applications

Research letters: The letters are small and concise comments on previously published matters.

**5.STRUCTURE AND FORMAT OF MANUSCRIPT**

The recommended size of original research paper is less than seven thousand words, review papers fewer than seven thousands words also.Preparation of research paper or how to write research paper, are major hurdle, while writing manuscript. The research articles and research letters should be fewer than three thousand words, the structure original research paper; sometime review paper should be as follows:

**Papers**: These are reports of significant research (typically less than 7000 words equivalent, including tables, figures, references), and comprise:

(a)Title should be relevant and commensurate with the theme of the paper.

(b) A brief Summary, "Abstract" (less than 150 words) containing the major results and conclusions.

(c) Up to ten keywords, that precisely identifies the paper's subject, purpose, and focus.

(d) An Introduction, giving necessary background excluding subheadings; objectives must be clearly declared.

(e) Resources and techniques with sufficient complete experimental details (wherever possible by reference) to permit repetition; sources of information must be given and numerical methods must be specified by reference, unless non-standard.

(f) Results should be presented concisely, by well-designed tables and/or figures; the same data may not be used in both; suitable statistical data should be given. All data must be obtained with attention to numerical detail in the planning stage. As reproduced design has been recognized to be important to experiments for a considerable time, the Editor has decided that any paper that appears not to have adequate numerical treatments of the data will be returned un-refereed;

(g) Discussion should cover the implications and consequences, not just recapitulating the results; conclusions should be summarizing.

(h) Brief Acknowledgements.

(i) References in the proper form.

Authors should very cautiously consider the preparation of papers to ensure that they communicate efficiently. Papers are much more likely to be accepted, if they are cautiously designed and laid out, contain few or no errors, are summarizing, and be conventional to the approach and instructions. They will in addition, be published with much less delays than those that require much technical and editorial correction.

The Editorial Board reserves the right to make literary corrections and to make suggestions to improve briefness.

It is vital, that authors take care in submitting a manuscript that is written in simple language and adheres to published guidelines.

**Format**

*Language: The language of publication is UK English. Authors, for whom English is a second language, must have their manuscript efficiently edited by an English-speaking person before submission to make sure that, the English is of high excellence. It is preferable, that manuscripts should be professionally edited.*

Standard Usage, Abbreviations, and Units: Spelling and hyphenation should be conventional to The Concise Oxford English Dictionary. Statistics and measurements should at all times be given in figures, e.g. 16 min, except for when the number begins a sentence. When the number does not refer to a unit of measurement it should be spelt in full unless, it is 160 or greater.

Abbreviations supposed to be used carefully. The abbreviated name or expression is supposed to be cited in full at first usage, followed by the conventional abbreviation in parentheses.

Metric SI units are supposed to generally be used excluding where they conflict with current practice or are confusing. For illustration, 1.4 l rather than 1.4 × 10-3 m3, or 4 mm somewhat than 4 × 10-3 m. Chemical formula and solutions must identify the form used, e.g. anhydrous or hydrated, and the concentration must be in clearly defined units. Common species names should be followed by underlines at the first mention. For following use the generic name should be constricted to a single letter, if it is clear.

**Structure**

All manuscripts submitted to Global Journals Inc. (US), ought to include:

Title: The title page must carry an instructive title that reflects the content, a running title (less than 45 characters together with spaces), names of the authors and co-authors, and the place(s) wherever the work was carried out. The full postal address in addition with the e-mail address of related author must be given. Up to eleven keywords or very brief phrases have to be given to help data retrieval, mining and indexing.

*Abstract, used in Original Papers and Reviews:*

Optimizing Abstract for Search Engines

Many researchers searching for information online will use search engines such as Google, Yahoo or similar. By optimizing your paper for search engines, you will amplify the chance of someone finding it. This in turn will make it more likely to be viewed and/or cited in a further work. Global Journals Inc. (US) have compiled these guidelines to facilitate you to maximize the web-friendliness of the most public part of your paper.

Key Words

A major linchpin in research work for the writing research paper is the keyword search, which one will employ to find both library and Internet resources.

One must be persistent and creative in using keywords. An effective keyword search requires a strategy and planning a list of possible keywords and phrases to try.

Search engines for most searches, use Boolean searching, which is somewhat different from Internet searches. The Boolean search uses "operators," words (and, or, not, and near) that enable you to expand or narrow your affords. Tips for research paper while preparing research paper are very helpful guideline of research paper.

Choice of key words is first tool of tips to write research paper. Research paper writing is an art.A few tips for deciding as strategically as possible about keyword search:

- One should start brainstorming lists of possible keywords before even begin searching. Think about the most important concepts related to research work. Ask, "What words would a source have to include to be truly valuable in research paper?" Then consider synonyms for the important words.
- It may take the discovery of only one relevant paper to let steer in the right keyword direction because in most databases, the keywords under which a research paper is abstracted are listed with the paper.
- One should avoid outdated words.

Keywords are the key that opens a door to research work sources. Keyword searching is an art in which researcher's skills are bound to improve with experience and time.

Numerical Methods: Numerical methods used should be clear and, where appropriate, supported by references.

*Acknowledgements: Please make these as concise as possible.*

References

References follow the Harvard scheme of referencing. References in the text should cite the authors' names followed by the time of their publication, unless there are three or more authors when simply the first author's name is quoted followed by et al. unpublished work has to only be cited where necessary, and only in the text. Copies of references in press in other journals have to be supplied with submitted typescripts. It is necessary that all citations and references be carefully checked before submission, as mistakes or omissions will cause delays.

References to information on the World Wide Web can be given, but only if the information is available without charge to readers on an official site. Wikipedia and Similar websites are not allowed where anyone can change the information. Authors will be asked to make available electronic copies of the cited information for inclusion on the Global Journals Inc. (US) homepage at the judgment of the Editorial Board.

The Editorial Board and Global Journals Inc. (US) recommend that, citation of online-published papers and other material should be done via a DOI (digital object identifier). If an author cites anything, which does not have a DOI, they run the risk of the cited material not being noticeable.

The Editorial Board and Global Journals Inc. (US) recommend the use of a tool such as Reference Manager for reference management and formatting.

Tables, Figures and Figure Legends

*Tables: Tables should be few in number, cautiously designed, uncrowned, and include only essential data. Each must have an Arabic number, e.g. Table 4, a self-explanatory caption and be on a separate sheet. Vertical lines should not be used.*

*Figures: Figures are supposed to be submitted as separate files. Always take in a citation in the text for each figure using Arabic numbers, e.g. Fig. 4. Artwork must be submitted online in electronic form by e-mailing them.*

Preparation of Electronic Figures for Publication

Even though low quality images are sufficient for review purposes, print publication requires high quality images to prevent the final product being blurred or fuzzy. Submit (or e-mail) EPS (line art) or TIFF (halftone/photographs) files only. MS PowerPoint and Word Graphics are unsuitable for printed pictures. Do not use pixel-oriented software. Scans (TIFF only) should have a resolution of at least 350 dpi (halftone) or 700 to 1100 dpi (line drawings) in relation to the imitation size. Please give the data for figures in black and white or submit a Color Work Agreement Form. EPS files must be saved with fonts embedded (and with a TIFF preview, if possible).

For scanned images, the scanning resolution (at final image size) ought to be as follows to ensure good reproduction: line art: >650 dpi; halftones (including gel photographs) : >350 dpi; figures containing both halftone and line images: >650 dpi.



Color Charges: It is the rule of the Global Journals Inc. (US) for authors to pay the full cost for the reproduction of their color artwork. Hence, please note that, if there is color artwork in your manuscript when it is accepted for publication, we would require you to complete and return a color work agreement form before your paper can be published.

*Figure Legends: Self-explanatory legends of all figures should be incorporated separately under the heading 'Legends to Figures'. In the full-text online edition of the journal, figure legends may possibly be truncated in abbreviated links to the full screen version. Therefore, the first 100 characters of any legend should notify the reader, about the key aspects of the figure.*

**6. AFTER ACCEPTANCE**

Upon approval of a paper for publication, the manuscript will be forwarded to the dean, who is responsible for the publication of the Global Journals Inc. (US).

**6.1 Proof Corrections**

The corresponding author will receive an e-mail alert containing a link to a website or will be attached. A working e-mail address must therefore be provided for the related author.

Acrobat Reader will be required in order to read this file. This software can be downloaded

(Free of charge) from the following website:

www.adobe.com/products/acrobat/readstep2.html. This will facilitate the file to be opened, read on screen, and printed out in order for any corrections to be added. Further instructions will be sent with the proof.

Proofs must be returned to the dean at dean@globaljournals.org within three days of receipt.

As changes to proofs are costly, we inquire that you only correct typesetting errors. All illustrations are retained by the publisher. Please note that the authors are responsible for all statements made in their work, including changes made by the copy editor.

**6.2 Early View of Global Journals Inc. (US) (Publication Prior to Print)**

The Global Journals Inc. (US) are enclosed by our publishing's Early View service. Early View articles are complete full-text articles sent in advance of their publication. Early View articles are absolute and final. They have been completely reviewed, revised and edited for publication, and the authors' final corrections have been incorporated. Because they are in final form, no changes can be made after sending them. The nature of Early View articles means that they do not yet have volume, issue or page numbers, so Early View articles cannot be cited in the conventional way.

**6.3 Author Services**

Online production tracking is available for your article through Author Services. Author Services enables authors to track their article - once it has been accepted - through the production process to publication online and in print. Authors can check the status of their articles online and choose to receive automated e-mails at key stages of production. The authors will receive an e-mail with a unique link that enables them to register and have their article automatically added to the system. Please ensure that a complete e-mail address is provided when submitting the manuscript.

**6.4 Author Material Archive Policy**

Please note that if not specifically requested, publisher will dispose off hardcopy & electronic information submitted, after the two months of publication. If you require the return of any information submitted, please inform the Editorial Board or dean as soon as possible.

**6.5 Offprint and Extra Copies**

A PDF offprint of the online-published article will be provided free of charge to the related author, and may be distributed according to the Publisher's terms and conditions. Additional paper offprint may be ordered by emailing us at: editor@globaljournals.org .

the search? Will I be able to find all information in this field area? If the answer of these types of questions will be "Yes" then you can choose that topic. In most of the cases, you may have to conduct the surveys and have to visit several places because this field is related to Computer Science and Information Technology. Also, you may have to do a lot of work to find all rise and falls regarding the various data of that subject. Sometimes, detailed information plays a vital role, instead of short information.

**2. Evaluators are human:** First thing to remember that evaluators are also human being. They are not only meant for rejecting a paper. They are here to evaluate your paper. So, present your Best.

**3. Think Like Evaluators:** If you are in a confusion or getting demotivated that your paper will be accepted by evaluators or not, then think and try to evaluate your paper like an Evaluator. Try to understand that what an evaluator wants in your research paper and automatically you will have your answer.

**4. Make blueprints of paper:** The outline is the plan or framework that will help you to arrange your thoughts. It will make your paper logical. But remember that all points of your outline must be related to the topic you have chosen.

**5. Ask your Guides:** If you are having any difficulty in your research, then do not hesitate to share your difficulty to your guide (if you have any). They will surely help you out and resolve your doubts. If you can't clarify what exactly you require for your work then ask the supervisor to help you with the alternative. He might also provide you the list of essential readings.

**6. Use of computer is recommended:** As you are doing research in the field of Computer Science, then this point is quite obvious.

**7. Use right software:** Always use good quality software packages. If you are not capable to judge good software then you can lose quality of your paper unknowingly. There are various software programs available to help you, which you can get through Internet.

**8. Use the Internet for help:** An excellent start for your paper can be by using the Google. It is an excellent search engine, where you can have your doubts resolved. You may also read some answers for the frequent question how to write my research paper or find model research paper. From the internet library you can download books. If you have all required books make important reading selecting and analyzing the specified information. Then put together research paper sketch out.

**9. Use and get big pictures:** Always use encyclopedias, Wikipedia to get pictures so that you can go into the depth.

**10. Bookmarks are useful:** When you read any book or magazine, you generally use bookmarks, right! It is a good habit, which helps to not to lose your continuity. You should always use bookmarks while searching on Internet also, which will make your search easier.

**11. Revise what you wrote:** When you write anything, always read it, summarize it and then finalize it.

**12. Make all efforts:** Make all efforts to mention what you are going to write in your paper. That means always have a good start. Try to mention everything in introduction, that what is the need of a particular research paper. Polish your work by good skill of writing and always give an evaluator, what he wants.

**13. Have backups:** When you are going to do any important thing like making research paper, you should always have backup copies of it either in your computer or in paper. This will help you to not to lose any of your important.

**14. Produce good diagrams of your own:** Always try to include good charts or diagrams in your paper to improve quality. Using several and unnecessary diagrams will degrade the quality of your paper by creating "hotchpotch." So always, try to make and include those diagrams, which are made by your own to improve readability and understandability of your paper.

**15. Use of direct quotes:** When you do research relevant to literature, history or current affairs then use of quotes become essential but if study is relevant to science then use of quotes is not preferable.

**16. Use proper verb tense:** Use proper verb tenses in your paper. Use past tense, to present those events that happened. Use present tense to indicate events that are going on. Use future tense to indicate future happening events. Use of improper and wrong tenses will confuse the evaluator. Avoid the sentences that are incomplete.

**17. Never use online paper:** If you are getting any paper on Internet, then never use it as your research paper because it might be possible that evaluator has already seen it or maybe it is outdated version.

18. **Pick a good study spot:** To do your research studies always try to pick a spot, which is quiet. Every spot is not for studies. Spot that suits you choose it and proceed further.

**19. Know what you know:** Always try to know, what you know by making objectives. Else, you will be confused and cannot achieve your target.

**20. Use good quality grammar:** Always use a good quality grammar and use words that will throw positive impact on evaluator. Use of good quality grammar does not mean to use tough words, that for each word the evaluator has to go through dictionary. Do not start sentence with a conjunction. Do not fragment sentences. Eliminate one-word sentences. Ignore passive voice. Do not ever use a big word when a diminutive one would suffice. Verbs have to be in agreement with their subjects. Prepositions are not expressions to finish sentences with. It is incorrect to ever divide an infinitive. Avoid clichés like the disease. Also, always shun irritating alliteration. Use language that is simple and straight forward. put together a neat summary.

**21. Arrangement of information:** Each section of the main body should start with an opening sentence and there should be a changeover at the end of the section. Give only valid and powerful arguments to your topic. You may also maintain your arguments with records.

**22. Never start in last minute:** Always start at right time and give enough time to research work. Leaving everything to the last minute will degrade your paper and spoil your work.

**23. Multitasking in research is not good:** Doing several things at the same time proves bad habit in case of research activity. Research is an area, where everything has a particular time slot. Divide your research work in parts and do particular part in particular time slot.

**24. Never copy others' work:** Never copy others' work and give it your name because if evaluator has seen it anywhere you will be in trouble.

**25. Take proper rest and food:** No matter how many hours you spend for your research activity, if you are not taking care of your health then all your efforts will be in vain. For a quality research, study is must, and this can be done by taking proper rest and food.

**26. Go for seminars:** Attend seminars if the topic is relevant to your research area. Utilize all your resources.

**27. Refresh your mind after intervals:** Try to give rest to your mind by listening to soft music or by sleeping in intervals. This will also improve your memory.

**28. Make colleagues:** Always try to make colleagues. No matter how sharper or intelligent you are, if you make colleagues you can have several ideas, which will be helpful for your research.

29. **Think technically:** Always think technically. If anything happens, then search its reasons, its benefits, and demerits.

**30. Think and then print:** When you will go to print your paper, notice that tables are not be split, headings are not detached from their descriptions, and page sequence is maintained.

**31. Adding unnecessary information:** Do not add unnecessary information, like, I have used MS Excel to draw graph. Do not add irrelevant and inappropriate material. These all will create superfluous. Foreign terminology and phrases are not apropos. One should NEVER take a broad view. Analogy in script is like feathers on a snake. Not at all use a large word when a very small one would be

sufficient. Use words properly, regardless of how others use them. Remove quotations. Puns are for kids, not grunt readers. Amplification is a billion times of inferior quality than sarcasm.

**32. Never oversimplify everything:** To add material in your research paper, never go for oversimplification. This will definitely irritate the evaluator. Be more or less specific. Also too, by no means, ever use rhythmic redundancies. Contractions aren't essential and shouldn't be there used. Comparisons are as terrible as clichés. Give up ampersands and abbreviations, and so on. Remove commas, that are, not necessary. Parenthetical words however should be together with this in commas. Understatement is all the time the complete best way to put onward earth-shaking thoughts. Give a detailed literary review.

**33. Report concluded results:** Use concluded results. From raw data, filter the results and then conclude your studies based on measurements and observations taken. Significant figures and appropriate number of decimal places should be used. Parenthetical remarks are prohibitive. Proofread carefully at final stage. In the end give outline to your arguments. Spot out perspectives of further study of this subject. Justify your conclusion by at the bottom of them with sufficient justifications and examples.

**34. After conclusion:** Once you have concluded your research, the next most important step is to present your findings. Presentation is extremely important as it is the definite medium though which your research is going to be in print to the rest of the crowd. Care should be taken to categorize your thoughts well and present them in a logical and neat manner. A good quality research paper format is essential because it serves to highlight your research paper and bring to light all necessary aspects in your research.

## Informal Guidelines of Research Paper Writing

**Key points to remember:**

- Submit all work in its final form.
- Write your paper in the form, which is presented in the guidelines using the template.
- Please note the criterion for grading the final paper by peer-reviewers.

**Final Points:**

A purpose of organizing a research paper is to let people to interpret your effort selectively. The journal requires the following sections, submitted in the order listed, each section to start on a new page.

The introduction will be compiled from reference matter and will reflect the design processes or outline of basis that direct you to make study. As you will carry out the process of study, the method and process section will be constructed as like that. The result segment will show related statistics in nearly sequential order and will direct the reviewers next to the similar intellectual paths throughout the data that you took to carry out your study. The discussion section will provide understanding of the data and projections as to the implication of the results. The use of good quality references all through the paper will give the effort trustworthiness by representing an alertness of prior workings.

Writing a research paper is not an easy job no matter how trouble-free the actual research or concept. Practice, excellent preparation, and controlled record keeping are the only means to make straightforward the progression.

**General style:**

Specific editorial column necessities for compliance of a manuscript will always take over from directions in these general guidelines.

To make a paper clear

· Adhere to recommended page limits

Mistakes to evade

- Insertion a title at the foot of a page with the subsequent text on the next page

- Separating a table/chart or figure - impound each figure/table to a single page
- Submitting a manuscript with pages out of sequence

In every sections of your document

· Use standard writing style including articles ("a", "the," etc.)

· Keep on paying attention on the research topic of the paper

· Use paragraphs to split each significant point (excluding for the abstract)

· Align the primary line of each section

· Present your points in sound order

· Use present tense to report well accepted

· Use past tense to describe specific results

· Shun familiar wording, don't address the reviewer directly, and don't use slang, slang language, or superlatives

· Shun use of extra pictures - include only those figures essential to presenting results

**Title Page:**

Choose a revealing title. It should be short. It should not have non-standard acronyms or abbreviations. It should not exceed two printed lines. It should include the name(s) and address (es) of all authors.

**Abstract:**

The summary should be two hundred words or less. It should briefly and clearly explain the key findings reported in the manuscript--must have precise statistics. It should not have abnormal acronyms or abbreviations. It should be logical in itself. Shun citing references at this point.

An abstract is a brief distinct paragraph summary of finished work or work in development. In a minute or less a reviewer can be taught the foundation behind the study, common approach to the problem, relevant results, and significant conclusions or new questions.

Write your summary when your paper is completed because how can you write the summary of anything which is not yet written? Wealth of terminology is very essential in abstract. Yet, use comprehensive sentences and do not let go readability for briefness. You can maintain it succinct by phrasing sentences so that they provide more than lone rationale. The author can at this moment go straight to

shortening the outcome. Sum up the study, with the subsequent elements in any summary. Try to maintain the initial two items to no more than one ruling each.

- Reason of the study - theory, overall issue, purpose
- Fundamental goal
- To the point depiction of the research
- Consequences, including definite statistics - if the consequences are quantitative in nature, account quantitative data; results of any numerical analysis should be reported
- Significant conclusions or questions that track from the research(es)

Approach:

- Single section, and succinct
- As a outline of job done, it is always written in past tense
- A conceptual should situate on its own, and not submit to any other part of the paper such as a form or table
- Center on shortening results - bound background information to a verdict or two, if completely necessary
- What you account in an conceptual must be regular with what you reported in the manuscript
- Exact spelling, clearness of sentences and phrases, and appropriate reporting of quantities (proper units, important statistics) are just as significant in an abstract as they are anywhere else

**Introduction:**

The **Introduction** should "introduce" the manuscript. The reviewer should be presented with sufficient background information to be capable to comprehend and calculate the purpose of your study without having to submit to other works. The basis for the study should be offered. Give most important references but shun difficult to make a comprehensive appraisal of the topic. In the introduction, describe the problem visibly. If the problem is not acknowledged in a logical, reasonable way, the reviewer will have no attention in your result. Speak in common terms about techniques used to explain the problem, if needed, but do not present any particulars about the protocols here. Following approach can create a valuable beginning:

- Explain the value (significance) of the study
- Shield the model - why did you employ this particular system or method? What is its compensation? You strength remark on its appropriateness from a abstract point of vision as well as point out sensible reasons for using it.
- Present a justification. Status your particular theory (es) or aim(s), and describe the logic that led you to choose them.
- Very for a short time explain the tentative propose and how it skilled the declared objectives.

Approach:

- Use past tense except for when referring to recognized facts. After all, the manuscript will be submitted after the entire job is done.
- Sort out your thoughts; manufacture one key point with every section. If you make the four points listed above, you will need a least of four paragraphs.
- Present surroundings information only as desirable in order hold up a situation. The reviewer does not desire to read the whole thing you know about a topic.
- Shape the theory/purpose specifically - do not take a broad view.
- As always, give awareness to spelling, simplicity and correctness of sentences and phrases.

**Procedures (Methods and Materials):**

This part is supposed to be the easiest to carve if you have good skills. A sound written Procedures segment allows a capable scientist to replacement your results. Present precise information about your supplies. The suppliers and clarity of reagents can be helpful bits of information. Present methods in sequential order but linked methodologies can be grouped as a segment. Be concise when relating the protocols. Attempt for the least amount of information that would permit another capable scientist to spare your outcome but be cautious that vital information is integrated. The use of subheadings is suggested and ought to be synchronized with the results section. When a technique is used that has been well described in another object, mention the specific item describing a way but draw the basic



principle while stating the situation. The purpose is to text all particular resources and broad procedures, so that another person may use some or all of the methods in one more study or referee the scientific value of your work. It is not to be a step by step report of the whole thing you did, nor is a methods section a set of orders.

Materials:

- Explain materials individually only if the study is so complex that it saves liberty this way.
- Embrace particular materials, and any tools or provisions that are not frequently found in laboratories.
- Do not take in frequently found.
- If use of a definite type of tools.
- Materials may be reported in a part section or else they may be recognized along with your measures.

Methods:

- Report the method (not particulars of each process that engaged the same methodology)
- Describe the method entirely
- To be succinct, present methods under headings dedicated to specific dealings or groups of measures
- Simplify - details how procedures were completed not how they were exclusively performed on a particular day.
- If well known procedures were used, account the procedure by name, possibly with reference, and that's all.

Approach:

- It is embarrassed or not possible to use vigorous voice when documenting methods with no using first person, which would focus the reviewer's interest on the researcher rather than the job. As a result when script up the methods most authors use third person passive voice.
- Use standard style in this and in every other part of the paper - avoid familiar lists, and use full sentences.

What to keep away from

- Resources and methods are not a set of information.
- Skip all descriptive information and surroundings - save it for the argument.
- Leave out information that is immaterial to a third party.

**Results:**

The principle of a results segment is to present and demonstrate your conclusion. Create this part a entirely objective details of the outcome, and save all understanding for the discussion.

The page length of this segment is set by the sum and types of data to be reported. Carry on to be to the point, by means of statistics and tables, if suitable, to present consequences most efficiently.You must obviously differentiate material that would usually be incorporated in a study editorial from any unprocessed data or additional appendix matter that would not be available. In fact, such matter should not be submitted at all except requested by the instructor.

Content

- Sum up your conclusion in text and demonstrate them, if suitable, with figures and tables.
- In manuscript, explain each of your consequences, point the reader to remarks that are most appropriate.
- Present a background, such as by describing the question that was addressed by creation an exacting study.
- Explain results of control experiments and comprise remarks that are not accessible in a prescribed figure or table, if appropriate.
- Examine your data, then prepare the analyzed (transformed) data in the form of a figure (graph), table, or in manuscript form.

What to stay away from

- Do not discuss or infer your outcome, report surroundings information, or try to explain anything.
- Not at all, take in raw data or intermediate calculations in a research manuscript.

- Do not present the similar data more than once.
- Manuscript should complement any figures or tables, not duplicate the identical information.
- Never confuse figures with tables - there is a difference.

Approach

- As forever, use past tense when you submit to your results, and put the whole thing in a reasonable order.
- Put figures and tables, appropriately numbered, in order at the end of the report
- If you desire, you may place your figures and tables properly within the text of your results part.

Figures and tables

- If you put figures and tables at the end of the details, make certain that they are visibly distinguished from any attach appendix materials, such as raw facts
- Despite of position, each figure must be numbered one after the other and complete with subtitle
- In spite of position, each table must be titled, numbered one after the other and complete with heading
- All figure and table must be adequately complete that it could situate on its own, divide from text

**Discussion:**

The Discussion is expected the trickiest segment to write and describe. A lot of papers submitted for journal are discarded based on problems with the Discussion. There is no head of state for how long a argument should be. Position your understanding of the outcome visibly to lead the reviewer through your conclusions, and then finish the paper with a summing up of the implication of the study. The purpose here is to offer an understanding of your results and hold up for all of your conclusions, using facts from your research and generally accepted information, if suitable. The implication of result should be visibly described. Infer your data in the conversation in suitable depth. This means that when you clarify an observable fact you must explain mechanisms that may account for the observation. If your results vary from your prospect, make clear why that may have happened. If your results agree, then explain the theory that the proof supported. It is never suitable to just state that the data approved with prospect, and let it drop at that.

- Make a decision if each premise is supported, discarded, or if you cannot make a conclusion with assurance. Do not just dismiss a study or part of a study as "uncertain."
- Research papers are not acknowledged if the work is imperfect. Draw what conclusions you can based upon the results that you have, and take care of the study as a finished work
- You may propose future guidelines, such as how the experiment might be personalized to accomplish a new idea.
- Give details all of your remarks as much as possible, focus on mechanisms.
- Make a decision if the tentative design sufficiently addressed the theory, and whether or not it was correctly restricted.
- Try to present substitute explanations if sensible alternatives be present.
- One research will not counter an overall question, so maintain the large picture in mind, where do you go next? The best studies unlock new avenues of study. What questions remain?
- Recommendations for detailed papers will offer supplementary suggestions.

Approach:

- When you refer to information, differentiate data generated by your own studies from available information
- Submit to work done by specific persons (including you) in past tense.
- Submit to generally acknowledged facts and main beliefs in present tense.

## Administration Rules Listed Before Submitting Your Research Paper to Global Journals Inc. (US)

Please carefully note down following rules and regulation before submitting your Research Paper to Global Journals Inc. (US):

**Segment Draft and Final Research Paper:** You have to strictly follow the template of research paper. If it is not done your paper may get rejected.

- The **major constraint** is that you must independently make all content, tables, graphs, and facts that are offered in the paper. You must write each part of the paper wholly on your own. The Peer-reviewers need to identify your own perceptive of the concepts in your own terms. NEVER extract straight from any foundation, and never rephrase someone else's analysis.

- Do not give permission to anyone else to "PROOFREAD" your manuscript.

- Methods to avoid Plagiarism is applied by us on every paper, if found guilty, you will be blacklisted by all of our collaborated research groups, your institution will be informed for this and strict legal actions will be taken immediately.)
- To guard yourself and others from possible illegal use please do not permit anyone right to use to your paper and files.

## Criterion for Grading a Research Paper (Compilation) by Global Journals Inc. (US)

**Please note that following table is only a Grading of "Paper Compilation" and not on "Performed/Stated Research" whose grading solely depends on Individual Assigned Peer Reviewer and Editorial Board Member. These can be available only on request and after decision of Paper. This report will be the property of Global Journals Inc. (US).**

| Topics | Grades | | |
|---|---|---|---|
| | **A-B** | **C-D** | **E-F** |
| ***Abstract*** | Clear and concise with appropriate content, Correct format. 200 words or below | Unclear summary and no specific data, Incorrect form<br><br>Above 200 words | No specific data with ambiguous information<br><br>Above 250 words |
| ***Introduction*** | Containing all background details with clear goal and appropriate details, flow specification, no grammar and spelling mistake, well organized sentence and paragraph, reference cited | Unclear and confusing data, appropriate format, grammar and spelling errors with unorganized matter | Out of place depth and content, hazy format |
| ***Methods and Procedures*** | Clear and to the point with well arranged paragraph, precision and accuracy of facts and figures, well organized subheads | Difficult to comprehend with embarrassed text, too much explanation but completed | Incorrect and unorganized structure with hazy meaning |
| ***Result*** | Well organized, Clear and specific, Correct units with precision, correct data, well structuring of paragraph, no grammar and spelling mistake | Complete and embarrassed text, difficult to comprehend | Irregular format with wrong facts and figures |
| ***Discussion*** | Well organized, meaningful specification, sound conclusion, logical and concise explanation, highly structured paragraph reference cited | Wordy, unclear conclusion, spurious | Conclusion is not cited, unorganized, difficult to comprehend |
| ***References*** | Complete and correct format, well organized | Beside the point, Incomplete | Wrong format and structuring |

# Index

## L



## M



## O



## P



## Q



## R



## S



## T



## U



## V



## W

save our planet

# Global Journal of Computer Science and Technology

Visit us on the Web at www.GlobalJournals.org | www.ComputerResearch.org
or email us at helpdesk@globaljournals.org

ISSN 9754350