A fuzzy traceability vector model for requirements validation

Vibha Gaur and Anuja Soni*

Department of Computer Science,
University of Delhi,
Delhi, India
Email: 3.vibha@gmail.com
Email: 30.anuja@gmail.com
*Corresponding author

Abstract: Requirements Traceability (RT) is an indispensable activity to chronologically interrelate the uniquely identifiable requirements that help the developers to discover the origin of each requirement. The manual and traditional RT methods are prone to errors and overlook the concerns of analysts in retrieving the requirements. As the requirements are vague, uncertain and subjective in nature, this work combines the theory of fuzzy sets with the traditional Vector Model (VM) approach in modelling the vagueness, haziness and non-specificity associated with the requirements and hence facilitates tracing the requirements up to a desired degree of relevance. A number of performance measures, viz. recall, precision, F-measure, fall-outs and miss, are employed to examine the efficiency of the approach in retrieving the requirements. In addition, the proposed methodology also proposes validation metrics, viz. completeness, correctness and consistency, that result in a comprehensive, complete and consistent Software Requirements Specification (SRS). The proposed approach, in this work, is exemplified in the context of a Multi-Agent System.

Keywords: requirements traceability; user story card; agent card, multi-agent system; fuzzy theory; requirements validation; vector model; software requirements specification.

Reference to this paper should be made as follows: Gaur, V. and Soni, A. (2013) ‘A fuzzy traceability vector model for requirements validation’, *Int. J. Computer Applications in Technology*, Vol. 47, Nos. 2/3, pp.172–188.

Biographical notes: Vibha Gaur is Associate Professor in University of Delhi. She has 16 years of experience, and has published more than 30 papers in reputed journals and international conferences. Her areas of interests are software engineering, e-commerce, software quality, fuzzy logic, users’ trust.

Anuja Soni is Assistant Professor in University of Delhi. She has 14 years of experience, and has published more than ten papers in reputed journals and international conferences. Her areas of interests are requirements engineering, multi-agent system, fuzzy logic.

1 Introduction

The importance of the requirements phase in software development is widely acknowledged. Requirements Engineering (RE) is a process of discrete chronological activities viz. elicitation, definition, negotiation, prioritisation and validation concerned with delineating and articulating the requirements of a system with a rationale of reducing the project failure in the inception of the project development (Klaus, 2011). As the requirements gathering commences, clients or stakeholders usually consider it exigent to produce an absolute, consistent and comprehensive set of requirements.

Requirements Validation (RV) contributes extensively in obtaining a correct, complete and consistent list of requirements. RV is the process of determining the degree to which Software Requirements Specification (SRS) is the accurate representation of the users’ needs from the perspective of the intended use (Akbar, 2008; Sreenivas, 2006). The joint team involving users and developers mainly contributes in validating the requirements. The developers necessitate requirements tracing in order to ensure the origin and consistency of expanded and analysed requirements in users’ needs and expectations while users ensure the completeness of their needs in SRS. This checking of completeness, correctness and consistency necessitates an effective mechanism to trace the users’ needs in SRS and vice versa.

The Requirements Traceability (RT) is the ability to get across the life of a requirement, in both a forward and backward direction, i.e. from its origins to its specification

(Maté, 2011). A number of RT techniques viz. Improving RT (Huffman, 2003), RT in Agent Oriented Development (Castro and Pinto, 2003), Goal Centric Traceability (Hayes et al., 2003), an Augmented Vector Space (Udagawa, 2011), a Trace Meta Model (Maté, 2011) and RT for Object Oriented Systems (Ali, 2011) are reported in literature.

Most of these approaches employ the VM (Vector Model) approach in retrieving the requirements relevant to the index terms in the query. In VM, the requirements are decomposed in terms of a number of weighted index terms. These approaches compute the weights of index terms in a query with respect to its occurrences in other requirements. But, these methods do not deal with the fuzziness and vagueness associated with the requirements.

This work proposes a Fuzzy Traceability Vector Model for RV that combines the theory of fuzzy sets (Klir and Yuan, 1995; Cross,1994) with the traditional Vector Model (VM) (Hayes et al., 2003) approach in modelling the vagueness, haziness and non-specificity associated with the requirements and hence facilitates tracing the requirements up to a desired degree of relevance. The use of the FT in RT has a number of advantages over classical methods. The fuzzy relevance relations and fuzzy thesauri employed are more expressive than their crisp counterparts and their interpretation is more pragmatic. The concept of α-cuts supported by FT is an efficient means to establish the relevance of the retrieved requirements up to a desired degree and thus imparts directives to the users and developers to regulate the scrutinisation of requirements.

A number of performance measures viz. recall, precision, F-measure, fall-outs and miss are employed to examine the efficiency of the approach in retrieving the requirements. The proposed methodology for tracing the requirements improves the retrieval of relevant requirements from users' needs by enhancing the values of various parameters viz. recall, precision, F-measure and reducing the values of parameters miss and fall-outs respectively.

In addition, this work also proposes RV metrics viz. completeness, correctness and consistency that ensure a comprehensive, precise and a consistent SRS. RV is a process of determining the degree to which SRS is the accurate, clear, succinct and unambiguous representation of the users' needs from the perspective of the intended use. Several RV techniques have been reported in literature namely Requirements Inspections, Requirements Prototyping, Requirements Testing, View Point Oriented RV and Use Cases in combination with Scenarios (Stephane, 2005; Sreenivas, 2006; Yousuf, 2008; Palyagar, 2006). Many of these validation techniques do not incorporate real requirements of the users due to lack of users' involvement and may result in an invalid SRS. The proposed approach employs the concept of USCs (User Story Cards) that is a simple user oriented approach to enrich users' involvement in requirements elicitation (Gaur et al., 2010).

The contribution of the proposed work is twofold: (a) to exemplify the characteristics of RT using Fuzzy Theory (FT) augmented with VM by comparing it with the contemporary RT methods. (b) to facilitate RV by providing validation metrics viz. correctness, completeness and consistency.

The application of the proposed methodology is illustrated in the context of Agent Oriented RE (AORE). The AORE models the requirements of a system in terms of autonomous interactive components agents. Software agents are computer programs that act autonomously on behalf of their users across open and distributed environments. The projected approach employs User Story Cards (USCs) to elicit the requirements and Agent Cards (ACs) to map the users' needs to various agents introduced in our previous work (Gaur and Soni, 2010). AC works as a repository of information of an agent in terms of goals, tasks and various estimations viz. schedule, cost etc. that facilitates the developers to comprehend complete set of requirements.

The organisation of the paper is as follows: Section 2 provides a brief overview of role of FT and VM in IR. Section 3 proposes a Fuzzy Traceability Vector Model for Requirements Validation. Section 4 presents experimental details. Section 5 presents a comparative analysis of RT techniques. Finally section 6 concludes the paper.

2 The role of Fuzzy Theory (FT) and Vector Model (VM) in retrieving information

The problem of Information Retrieval (IR) is to match the words or other symbols of the inquiry with those characterising the individual documents and make the appropriate selections.

The Fuzzy Information Retrieval (FIR) refers to the methods of IR based upon the theory of fuzzy sets (Klir and Yuan, 1995; Cross, 1994). The problem of IR involves finite crisp sets $X=\{x_1,x_2,\ldots,x_n\}$ and $Y=\{y_1,y_2,\ldots,y_m\}$, where X is a set of n index terms and Y is the set of m documents. In FIR, the relevance of index terms is expressed by a fuzzy relation R, where

$$R: X x Y \rightarrow [0,1]$$

such that the membership value $R(x_i,y_j)$ specifies for $\forall x_i \in X$ and $\forall y_j \in Y, 1 \le i \le n, 1 \le j \le m$ the grade of relevance of index term x_i to document y_j. A few of the fuzzy relations employed in FIR are defined below (Klir and Yuan, 1995).

Fuzzy Relations

Definition i: The fuzzy thesaurus T is a reflexive fuzzy relation defined on X^2. For each pair of index terms $(x_i, x_k) \in X^2, T(x_i, x_k)$ expresses the degree of association of x_i with x_k; that is, the degree to which the meaning of index term x_k is compatible with the meaning of the given index term x_i, where $1 \le i, k \le n$.

Definition ii: Given a fuzzy relation R:XxY and any number $\alpha \in [0,1]$, the α-cuts symbolised by ${}^{\alpha}R$, are the crisp sets that have membership degrees no less than α i.e.

$$^{\alpha}R = \left\{ y_j \left| \mu_R \left(x_i, y_j \right) \geq \alpha, x_i \in X, y_j \in Y \right. \right\}$$

where $1 \leq i \leq n$, $1 \leq j \leq m$

FIR is employed in various problems such as Retrieval of Research Archival Information (Smith, 1998), Interpretation of Fuzzy-Fingerprints for Text-Based IR (Stein, 2005) and in Detecting similar HTML documents (Yerra, 2005) etc.

VM is an algebraic model for representing text documents in IR as vectors of index terms and its weights (Hayes et al., 2003). The vector model approach can be described by the following definitions.

Vector Model

Definition i: Given $Y=\{y_1,y_2,\ldots y_m\}$, a set of m textual documents, each document y_i ($1<=i<=m$) can be represented as a vector $y_i=(w_1,w_2,\ldots,w_N)$, where N is the number of terms and w_j ($1<=j<=N$) is the weight of the root index term in y_i.

Definition ii: The weight of index term in a document is driven by its frequency in the current document and its total frequency in other documents. The total frequency Tf_{td} of term t in document collections d can be described as:

$$Tf_{td} = \sum_{x \in d} f_t(x) \text{ where } f_t(x) = \begin{cases} 1 & \text{if } t \text{ is available} \\ 0 & \text{otherwise} \end{cases}$$

The Vector Model is employed in various applications viz. information filtering, information retrieval and Requirements Traceability etc. (Hayes et al., 2003; Shuda et al., 2009).

3 A fuzzy traceability vector model for requirements validation

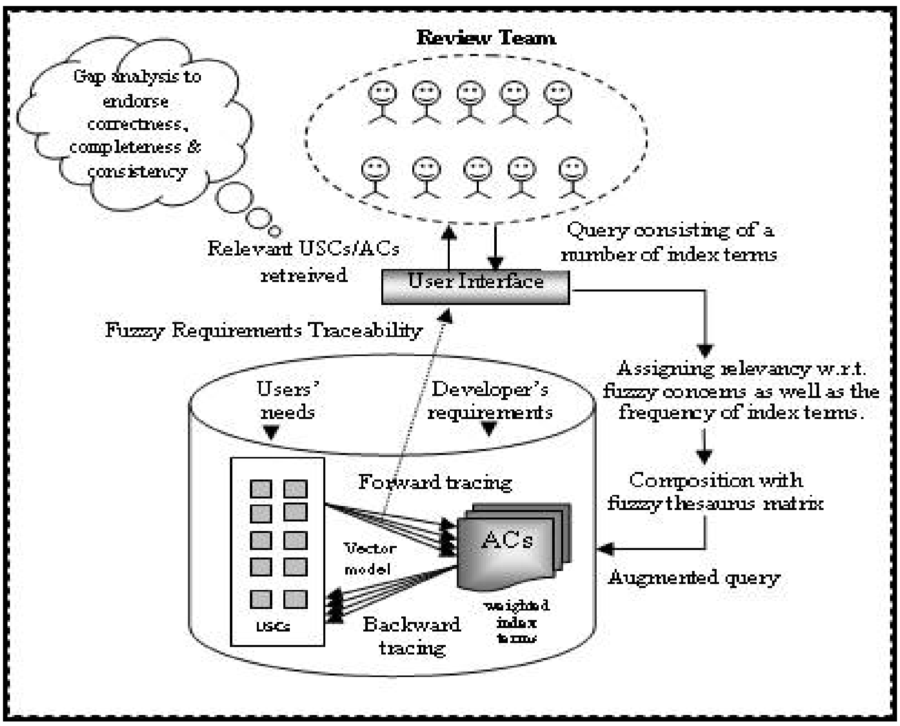
RV is a process of determining the degree to which SRS is the accurate, clear, succinct and unambiguous representation of the users' needs from the perspective of the intended use (Akbar, 2008; Sreenivas, 2006) and RT is the ability to get across the life of a requirement, in both a forward and backward direction, i.e. from its origins to its specification to provide a support for validating the requirements. The manual and traditional RT methods are prone to errors and ignore the fuzziness involved in requirements' retrieval.

This work proposes a Fuzzy Traceability Vector Model that augments the theory of fuzzy sets to the traditional Vector Model (VM) for modelling the vagueness, haziness and non-specificity associated with the requirements and hence facilitates tracing the requirements up to a desired degree of relevance. The functionality of proposed model is illustrated in Figure 1.

The requirements stored in USCs and ACs are represented in the form of weighted index terms using VM approach. A user interface is introduced to facilitate the joint team to input a query which is represented in terms of a number of index terms and its weights. The weights of index terms in the query are driven by its frequency and fuzzy concerns of analysts. A fuzzy thesaurus is employed to take into account the presence of the keywords synonymous or otherwise related to the index terms of the query. The proposed model facilitates tracing of requirements from USCs to ACs and vice versa that are further endorsed for its completeness, correctness and consistency.

The details of the projected approach are elaborated in the following sub-sections.

Figure 1 Fuzzy traceability vector model for requirements validation



3.1 User Story Cards (USCs)

The User Story Card is a simple user oriented method to acquire the requirements directly from users that prevents chances of misunderstanding among users and developers. The concept of USC is introduced in our previous work (Gaur and Soni, 2010). User story (Smith, 2002) is a simple user oriented approach to enrich users' involvement in requirements elicitation. The template for the user story is illustrated as below.

As a <user type>, I want to <purpose> So That <reason> (1)

USC consists of a purpose representing requirements of the user and many other parameters like USC_NO, DATE_OF_CREATION etc. as shown in Figure 2. USCs once validated form the artefact of SRS at the later stage of RE.

3.2 Agent Card (AC)

The Agent Card (AC), a novel structure acts as a repository of the information for the requirements of various agents in Multi-Agent System (MAS) and is characterised by their structural, functional and social behaviour as illustrated in Figure 3. The structure of AC has been introduced in our previous work (Gaur et al., 2010). ACs are expanded form of USCs that include comprehensive details of agents' requirements viz. agent_name, goals etc. The requirements obtained from various USCs are defined in terms of goals, tasks that are assigned to various agents in MAS. These ACs facilitate the developers to obtain the comprehensive set of requirements of a system in terms of various agents and their goals.

Figure 2 Example of USC (see online version for colours)

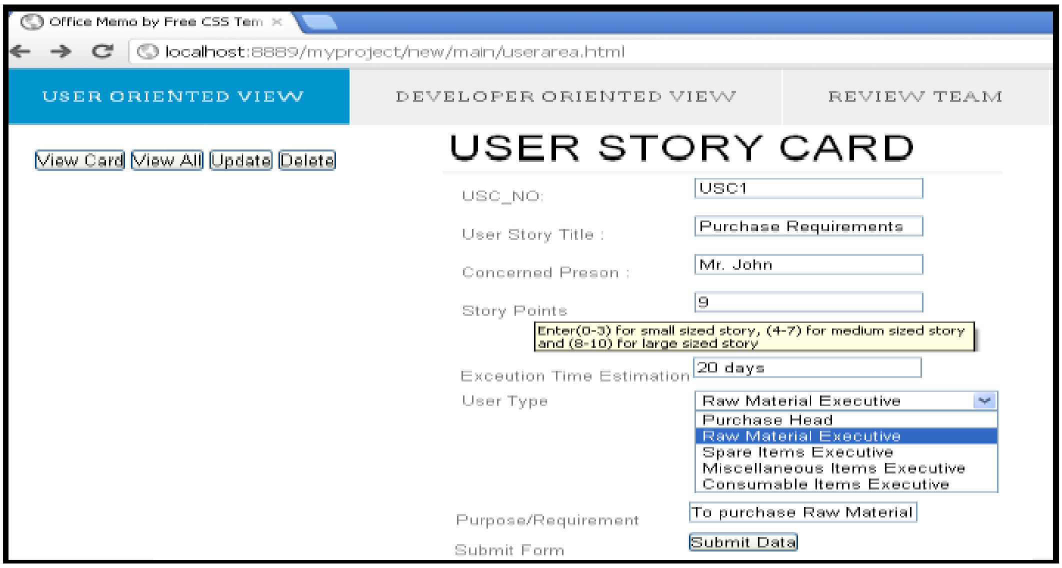
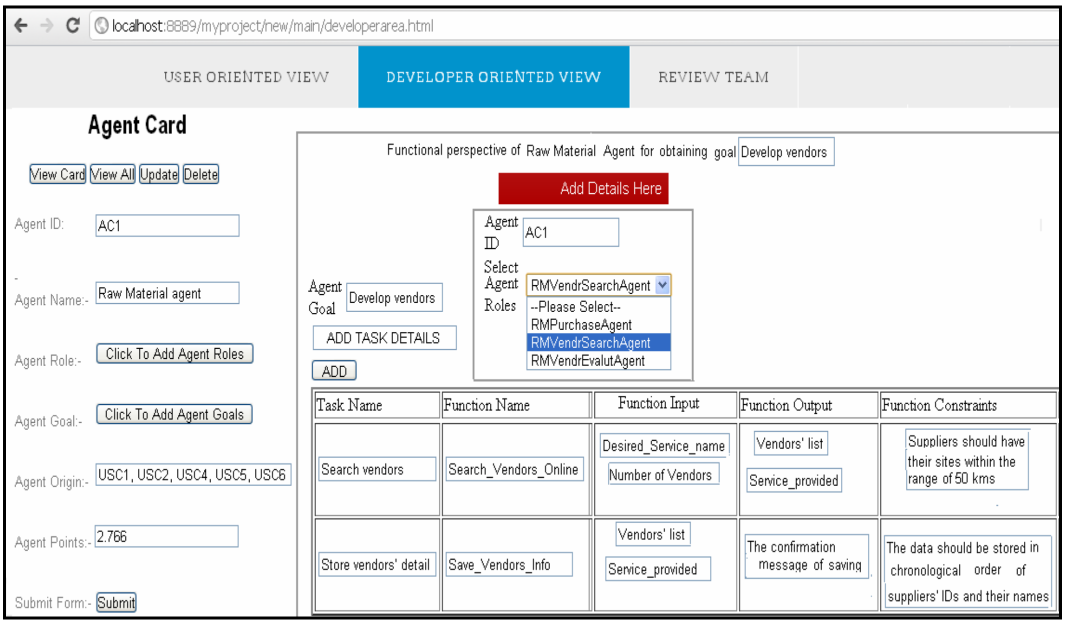


Figure 3 Example of agent card (see online version for colours)



ACs once validated form the artefact of SRS at the later stage of RE.

3.3 *Validating requirements using fuzzy traceability vector model*

The Requirements Traceability (RT) is the complementary activity of RV to get across the life of a requirement, in both a forward and backward direction (Maté, 2011) in order to ensure a complete, precise and comprehensive SRS. The projected approach incorporates the potentials of FT and VM to trace the requirements from USCs to ACs & vice versa and henceforth facilitates to endorse completeness, correctness and consistency of requirements.

The proposed approach makes up the following speculation about several variables employed in the context.

Suppose

- *i*=(1..N), where N is number of USCs
- *j*=(1..M), where M is number of ACs
- *e*=(1..h or 1..y), where h is total number of index terms in USC_i and y is total number of index terms in query q.

Following steps are involved for backward traceability from ACs to USCs. The same process can be extended for forward traceability from USCs to ACs as well.

1 A list of words appearing in the requirements represented by USCs that have no indexing value viz. 'and', 'or' etc. are eliminated.

2 The remaining words are converted to their 'root form'. The collection of these root words forms sets of index terms i.e. Tu_e (1<=e<=h) for the various requirements of USCs and are entitled as USCIT (User Story Card Index Terms), where $USCIT_i=\{Tu_1,Tu_2\ldots Tu_e\ldots Tu_h\}$.

3 The weight entitled as '$w_{term}(Tu_e, USC_i)$' of term Tu_e in USC_i is calculated as (Hayes et al., 2003):

$$w_{term}\left(Tu_e, USC_i\right) = FTu_{e_{USC_i}} * IFTu_{e_{USCs}} \tag{2}$$

where $FTu_{e_{USC_i}}$ defines the frequency of term Tu_e in USC_i and $IFTu_{e_{USCs}}$ is the inverse frequency of the term Tu_e in the USCs' collection. The Inverse frequency $IFTu_{e_{USCs}}$ is computed as:

$$IFTu_{e_{UCSs}} = \log_2 \frac{N}{N_{USC_{Tu_e}}} \tag{3}$$

where $N_{USC_{Tu_e}}$ signifies the number of USCs containing term Tu_e and N denotes total number of USCs.

The value of $w_{term}(Tu_e,USC_i)$ is normalised in [0, 1] by dividing its value by the maximum value of w_{term}.

4 Once the weight of each term in the user story is computed, the USCs are represented as a set of index terms and their weights as below.

$$USC_i = \left(\left(Tu_1, w_{i1}\right)\ldots\ldots\left(Tu_e, w_{ie}\right)\ldots\left(Tu_h, w_{ih}\right)\right)$$

Where $w_{ie} = w_{term}(Tu_e,USC_i)$ in [0,1] and 1<=i <=N.

5 A query q reflecting the goals of AC_j is also converted into a similar vector $q=((Tq_1,w_{q1}),\ldots\ \ldots\ (Tq_e,w_{qe})\ldots(Tq_y,w_{qy}))$ of index terms and their weights, where Tq_e (1<=e<=y) refers to the e_{th} index term and the weight w_{qe} is computed as the summation of weighted factors F_{qe}^1 and F_{qe}^2 using the following equation.

$$w_{qe} = \gamma_1 F_{qe}^1 + \gamma_2 F_{qe}^2 \tag{4}$$

where the value of factor F_{qe}^1 is determined by the frequency of an index term in the query while the value of factor F_{qe}^2 is enforced by the fuzzy concerns of analysts. The values of parameters $\left(0 \le \gamma_1, \gamma_2 < 1 \text{such that}, \gamma_1 + \gamma_2 = 1\right)$ monitor the weights assigned to the factors F_{qe}^1 and F_{qe}^2 If $\gamma_1 > \gamma_2$, then the frequency of the index term in q is weighted higher than the analyst's expectation, As the relevance of the index term in a query can be determined better by an analyst, therefore generally $\gamma_2 > \gamma_1$ is undertaken.

The details of obtaining the aforementioned factors are given below.

5.1 The value of factor F_{qe}^1 is obtained by the following equation.

$$F_{qe}^1 = \left(\frac{FTq_{e_q}}{y}\right) \tag{5}$$

where FTq_{e_q} refers to the frequency of index term Tq_e in q. The rationale behind using the equation (5) is that higher the frequency of Tq_e in q, more its weight irrespective of its occurrence in rest of other requirements.

5.2 The value of factor F_{qe}^2 refers to the fuzzy concerns of analysts and is obtained in linguistic terms viz. Important (I), Highly Important (HI) etc. as per the relevance in the context. The linguistic weights are mapped to Triangular Fuzzy Numbers (TFNs) as illustrated in Figure 4.

- The TFNs (l, m, u) with l as smallest, m as most promising and u as the largest value is converted into corresponding crisp interval $^{\alpha}A$ by α-cut operation using the equation given below (Gaur and Soni, 2010).

$$^{\alpha}A = \left[\left(m-1\right)\alpha + 1, -\left(u-m\right)\alpha + u\right] = \left[I_{\alpha 1}, I_{\alpha u}\right] \tag{6}$$

where $I_{\alpha l}$ =lower bound of crisp interval, $I_{\alpha u}$=upper bound of crisp interval. Crisp values C_{α}^{μ} against the crisp intervals $^{\alpha}A$ can be computed as:

$$C_{\alpha}^{\mu} = \mu I_{\alpha u} + \left(1-\mu\right) I_{\alpha 1}, \textit{ where } \mu \in [0,1] \tag{7}$$

where μ is called as the index of optimism representing optimistic level of analyst as optimistic, moderate or pessimistic. Higher value of μ represents the higher degree of optimism. The value of α represents the confidence level of analyst. Value of α is taken as 0.5 to show the moderate level of confidence in most promising crisp value.

- The crisp weight C_α^μ obtained in the previous step, can be employed to obtain the final value of the factor F_{qe}^2 in [0,1] using the following equation.

$$F_{qe}^2 = \frac{C_\alpha^\mu}{Upper\ Limit\ Of\ Scale\ Of\ TFN} \tag{8}$$

where $Upper\ Limit\ Of\ Scale\ Of\ TFN$ signifies the upper limit of scale of TFNs.

6 A fuzzy set T consisting of linguistic terms viz. Partially Similar (PS), Highly Similar (HS) etc. as shown in Figure 5, can be comprised on the thesaurus so as to take into account the presence of the keywords synonymous or otherwise related to the index terms of the query.

7 By composing query q with the thesaurus T, an augmented query A_q can be obtained as (Klir and Yuan, 1995):

$$A_q = qoT \tag{9}$$

$$where\ {}^{A}_{q}(x_i) = \max_{k \in X} \min_{1 \le i \le y} \left[q(x_i), T(x_i, x_k) \right] \tag{10}$$

where X and y refer to the total number of distinct terms in T and query q respectively.

8 The weights of the all the index terms ∈X from various USCs are extracted and formulated in the form of a two dimensional fuzzy relation R.

9 The retrieved USCs, expressed by fuzzy set D are obtained by composing the augmented query, expressed by A_q with the fuzzy relation R (Klir and Yuan, 1995) i.e.

$$A_q o\ R = D \tag{10}$$

10 All the relevant USCs are retrieved by extracting α-cuts against various values of α (from definition ii). The distinct values of D are treated as various values of α.

11 The paramount value of α producing the most relevant USCs can be determined by measuring its effectiveness in terms of recall, precision, fall-outs and miss.

Figure 4 Linguistic weights mapped to TFNs

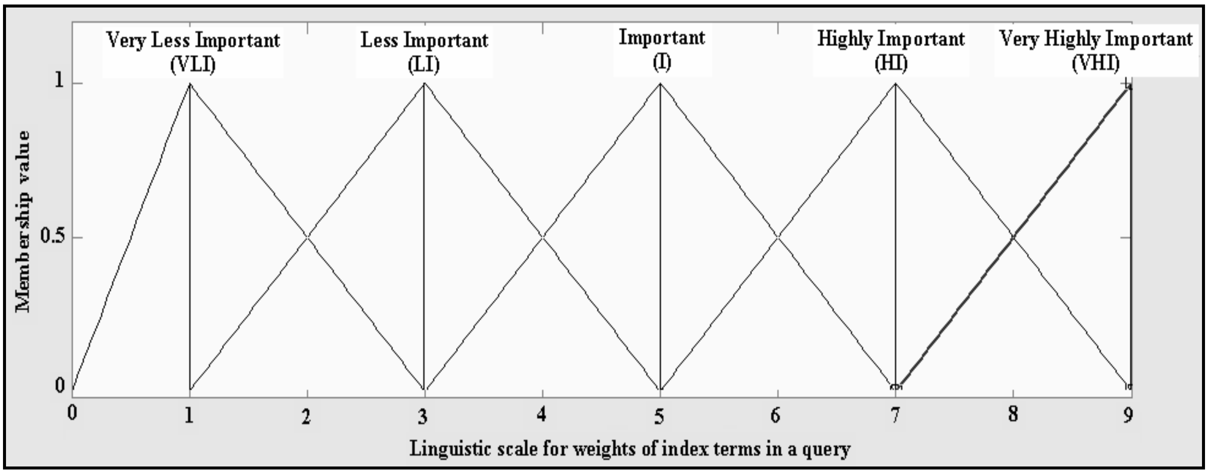
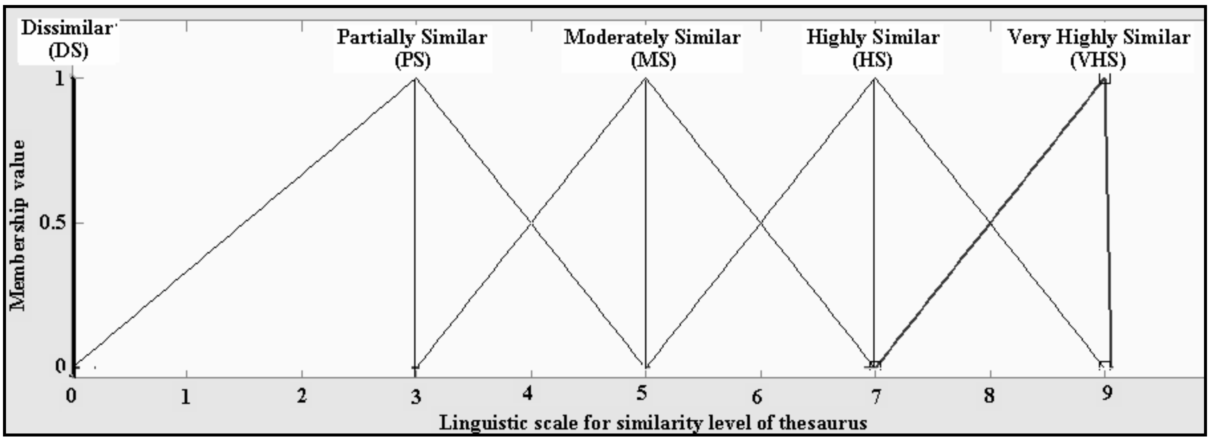


Figure 5 Linguistic scale for similarity level of thesaurus



The retrieval of the relevant USCs using the aforementioned steps assists to ensure the correctness, completeness and consistency of a specific AC in the following manner.

- *Correctness*

Correctness (Sreenivas, 2006) signifies whether the system reflects the requirements of the users accurately. Correctness can be associated with finding out the several goals of AC_j (1<=j<=M) in various USCs. The goals that do not map to any of USCs require to be approved by the users in some workshop. Once endorsed would ensure the correctness of AC_j and hence all the ACs in the system.

Following equation is used to ensure the correctness of AC_j.

$$CORR_{AC_j} = \frac{MapG_{AC_j}}{TOTG_{AC_j}} \tag{11}$$

where $TOTG_{AC_j}$ refers to total number of goals in AC_j and $MapG_{AC_j}$ to number of goals mapped to any USCs. The value of $CORR_{AC_j}$ would be in [0,1]. $CORR_{AC_j} = 0$, would indicate that none of the goal is originated from any of USCs, hence AC_j is considered to be erroneous to meet users' requirements. The value $CORR_{AC_j} = 1$ would endorse an AC to meet the users' expectations appropriately. If $CORR_{AC_j} \neq 1$, then developer cannot proceed further and is required to conduct a workshop with users to validate unmapped goals.

- *Completeness*

The completeness of requirements is an ability that refers to the incorporation of 'user intentions' accurately in the SRS (Sreenivas, 2006). To ensure the coverage of all relevant user stories in AC_j, equation (12) is utilised.

$$COM_{AC_j} = \frac{Map\ Relvnt\ USCs_{AC_j}}{TOT\ Relvn\ USC_{AC_j}} \tag{12}$$

where $TOT\ Relvn\ USC_{AC_j}$ refers to total number of USCs relevant to AC_j and $Map\ Relvnt\ USCs_{AC_j}$ refers to relevant USCs mapped to ACj. The value of COM_{AC_j} would always be in [0,1]. The value $COM_{AC_j} = 1$ would endorse the mapping of all relevant USCs in some of the goals of AC_j.

If $COM_{AC_j} \neq 1$, then developer cannot proceed further and is required to conduct a workshop with users to validate unmapped USCs.

- *Consistency*

The requirements consistency is concerned with the identification of conflicts between two or more related requirements (Sreenivas, 2006). The level of consistency is measured by the following equation.

$$Lvl\ Consistency_{AC_j} = \frac{MapG_{AC_j} - NDG_{AC_j}}{MapG_{AC_j}} \tag{13}$$

where NDG_{AC_j} is the number of non deterministic goals in AC_j. The goals originated from the contradictory requirements are delineated as the non deterministic goals. The value of $Lvl\ Consistency_{AC_j}$ as 1 would ensure all the goals of AC_j consistent otherwise a non zero value of $Lvl_{Consistency}$ would direct the developer to conduct a workshop with the users to clarify the inconsistent requirements.

4 Experimental details

To see the application of the proposed work, an experiment was carried out to validate the requirements of Materials Procurement MAS using the proposed approach. Materials Procurement MAS is composed of Purchase Head Agent, Raw Material Agents, Spares Agents, Packaging Agents, Consumable Agents and Miscellaneous Items Agents etc. involved with procuring various items viz. raw materials, spares, packaging, consumable and miscellaneous items for various projects distributed over various locations geographically.

The experiment was carried out in JSP with Tomcat server 4.1 having SQLyog in front end and MySQL 5.2 in backend on Intel CPU having 1.99 GB of RAM and speed 1.86 GHz. This involved 225 candidate requirements obtained from various users involved in several procurement activities. Requirements pertaining to materials procurement environment were obtained using user story template in the following manner.

"As a Purchase had I want Raw Material Incharge to develop vendors for supplying raw materials efficiently and effectively" (14)

Similarly the various requirements obtained were placed in USCs as illustrated in Figure 6. Each of the USCs after having a dialogue with users was analysed into a number of requirements and ultimately allocated to various agents in terms of various goals and tasks. A total of 15 ACs attributed to a number of procurement activities viz. asset management, developing vendors, purchasing raw materials, packaging items etc. were comprised and accredited to various agents.

Figure 6 Users' requirements stored in various USCs

USC_1: To develop and evaluate vendors for supplying raw materials efficiently and effectively.

USC_2: To display settlement status with suppliers for input materials.

USC_3: To display settlement status with vendors for packaging items.

USC_4: To prepare the purchase order for raw materials.

USC_5: Go for in-house sourcing for procuring raw materials.

USC_6: To prepare the report of raw materials weekly, monthly and yearly.

USC_7: To establish vendors for supplying packaging items.

USC_8: To prepare the purchase order for packaging items.

USC_9: To purchase packaging items.

USC_{10}: To prepare the report of packaging items weekly, monthly and yearly.

Out of these activities, the raw material goals were delegated to Raw Material Agent and packaging goals to Packaging Agent respectively. The agent cards AC_1 and AC_2 having their origin in relevant USCs are described below.

ACs	*Agent role*	*Origin*
AC_1:	Raw Material Agent	USC_1, USC_2, USC_4, USC_5, USC_6
AC_2:	Packaging Agent	USC_3, USC_7, USC_8, USC_9, USC_{10}

The raw material requirements obtained from concerned USCs were refined into a number of goals $G_{1_1}, G_{1_2}, G_{1_3}, G_{1_4}$, $G_{1_5}, G_{1_6}, G_{1_7}$ etc. and accredited to AC_1 as illustrated in Figure 7

Figure 7 Goals associated with AC_1

G_{1_1} : Search vendors for supplying raw materials efficiently
G_{1_2} : Evaluate vendors against their performance scores
G_{1_3} : Maintain vendors' files
G_{1_4} : Display vendors' evaluation
G_{1_5} : Insert raw material
G_{1_6} : Delete raw materials.
G_{1_7} : Prepare the purchase order for raw materials
G_{1_8} : Procure raw materials by in-house sourcing
G_{1_9} : Prepare the report of raw materials weekly, monthly and yearly
$G_{1_{10}}$: Display purchase requisition
$G_{1_{11}}$: Display vendors' quotation
$G_{1_{12}}$: Display vendors' scheduling agreement

This study is aimed to focus on the exemplification of the proposed approach in forward and backward traversal from ACs to USCs to validate the requirements. The details of the experiment for the traversal from ACs to USCs are mentioned in the following steps.

The unnecessary words viz. 'to', 'for', 'the', 'and', 'of', 'go' were eliminated from the requirements of various USCs. Afterwards USCs were converted to their root words as shown in Figure 8.

Figure 8 USCs converted to their root words

USCs	Root words
USC_1	Develop, vendors, evaluate, supply, raw materials, efficiently, effectively
USC_2	Display, settlement, status, suppliers, input materials
USC_3	Display, settlement, status, vendors, packaging items
USC_4	Prepare, purchase, order, raw materials
USC_5	In-house, sourcing, Procure, raw materials
USC_6	Maintain, report, raw materials, weekly, monthly, yearly
USC_7	Establish, vendors, supply, packaging items
USC_8	Prepare, purchase, order, packaging items
USC_9	Purchase, packaging items
USC_{10}	Maintain, report, packaging items, weekly, monthly, yearly

Figure 9 depicts the interface in MySQL with JSP in front end for the computation of weights in various USCs and the weights of index terms computed for USC_1 are illustrated in Figure 10.

Figure 9 Screen shot depicting the interface to compute the weights of index terms in various USCs (see online version for colours)

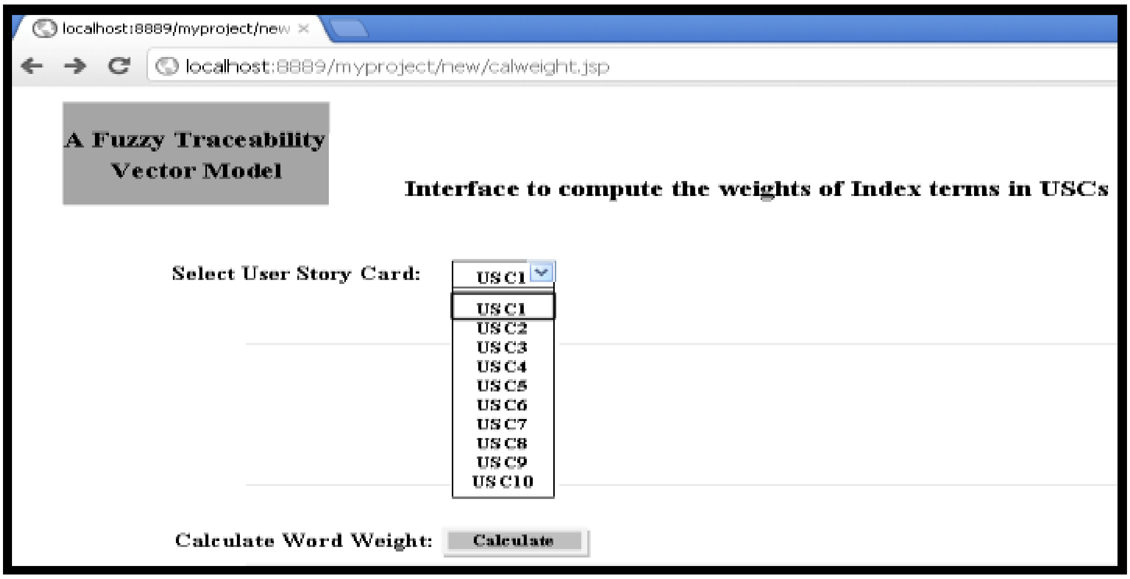
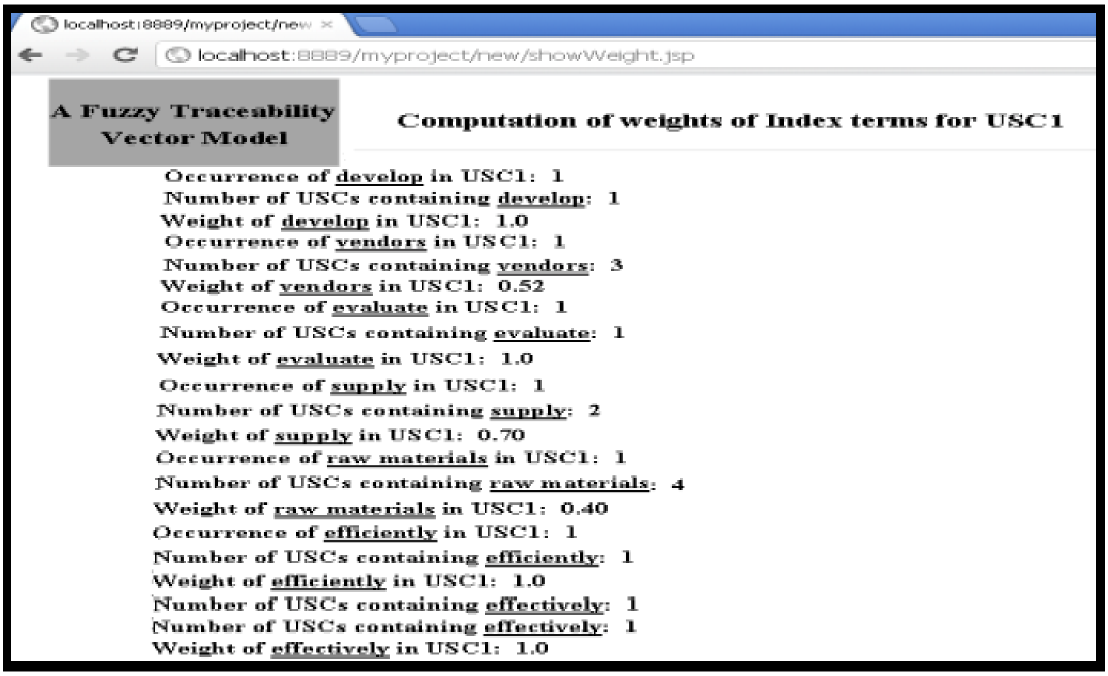


Figure 10 Computation of weights of Index terms for USC_1 (see online version for colours)



In a similar manner, various USCs were speculated in terms of their index terms and normalised weights using equations (2) and (3) as illustrated in Figure 11. To trace the goals of AC_1 in various USCs, a query q was comprised.

Figure 11 USCs represented in the form of index terms and their weights

USC_1={((Develop, 1), (Vendors, 0.52), (Evaluate, 1), (Supply, 0.70), (Raw Materials, 0.40), (Efficiently, 1), (Effectively, 1)}
USC_2={((Display, 0.70), (Settlement, 0.70), (Status, 0.70), (Suppliers, 1), (Input Materials, 1)}
USC_3={((Display, 0.70), (Settlement, 0.70), (Status, 0.70), (Vendors, 0.52), (Packaging items, 0.30)}
USC_4={(Prepare, 0.40), (Purchase, 0.40),(Order, 0.70), (Raw materials, 0.40)}
USC_5={(In-house, 1), (Sourcing, 1), (Procure, 1) (Raw materials, 0.40)}
USC_6={(Prepare, 0.40), (Report, 0.70), (Raw materials, 0.40), (Weekly, 0.70), (Monthly, 0.70), (Yearly, 0.70)}
USC_7={(Establish, 1), (Vendors, 0.52), (Supply, 0.70), (Packaging items, 0.30)}
USC_8={(Prepare, 0.40), (Purchase, 0.40),(order, 0.70), (Packaging items, 0.30)}
USC_9={(Purchase, 0.40),(Packaging items, 0.30)}
USC_{10}={(Prepare, 0.40), (Report, 0.70), (Packaging items, 0.30), (Weekly, 0.70), (Monthly, 0.70), (Yearly, 0.70)}

q='Obtain USCs related to requisition, vendors, their quotation but in the context of raw materials efficiently'.

The query q was expressed by the following crucial index terms relevant to the goals of AC_1.

x_1 = raw materials

x_2 = vendors

x_3 = efficiently

x_4 = requisition

x_5 = quotation

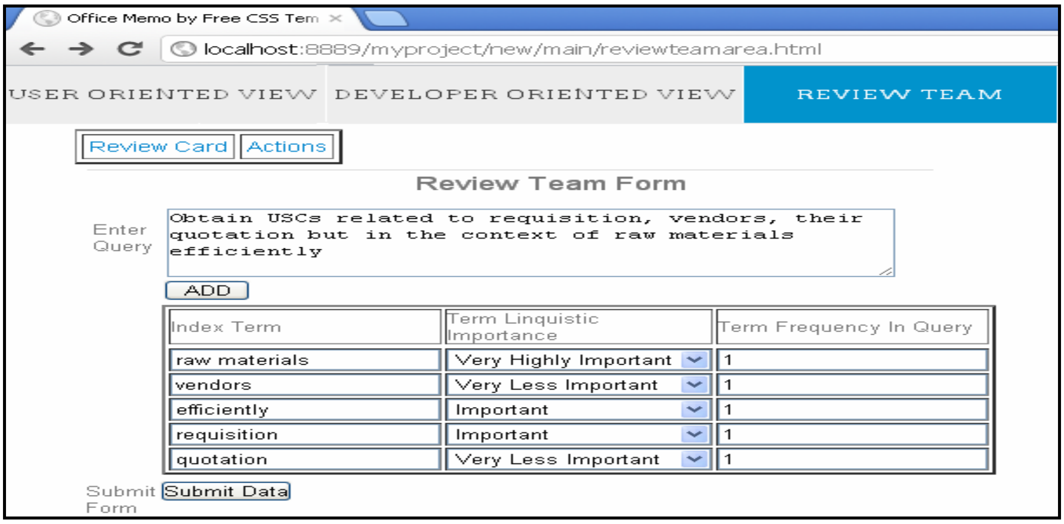
The q was represented in the form of a set of weights $\{W_{qi}\}$, where weights $w_{qi}, 1 \le i \le 5$ of various index terms were computed with respect to its frequency in query q and its relevance determined by the review team. The values of parameters $\gamma_1 and\ \gamma_2$ were assumed to ¼ and ¾ to give higher weightage to the concerns of review team.

As the frequency of each index term in q was observed as one, therefore using equation (5) the value of $F_{qe}^1\ \forall$ $index\ term\ 1 \le e \le 5$ was computed as 0.2, while the factor F_{qe}^2 was enforced by the fuzzy concerns of review team using linguistic terms viz. VHI, HI etc. Suppose the review team was highly concerned for the index term x_1, significantly concerned for the index terms x_3,x_4 and much less concerned for rest of other index terms. Therefore to compute the values of factor $F_{qe}^2\ \forall 1 \le e \le 5$, the linguistic weight 'VHI' was assigned to highly required index term x_1, 'I' to x_3 and x_4 and 'VLI' to rest of other index terms x_2 and x_5 as illustrated in Figure 12. These linguistic terms were replaced by TFNs and ultimately were converted to their normalised values viz. $F_{q1}^2 = 1, F_{q2}^2 = 0.14,$ $F_{q3}^2 = 0.55, F_{q4}^2 =$ using the equations (6), (7) and (8).

Finally using the equation (4), the query q was expressed in terms of weights w_{qe} driven by the factors $F_{qe}^1\ and\ F_{qe}^2$ as illustrated in the following matrix.

$$q = \begin{bmatrix} X_1 & X_2 & X_3 & X_4 & X_5 \\ 0.8 & 0.16 & 0.46 & 0.46 & 0.16 \end{bmatrix}$$

Figure 12 Interface for review team for inputting query and its index terms (see online version for colours)



The index terms in the query q were found to be relevant to the following terms as well.

X_6 = effectively

X_7 = Input materials

X_8= Suppliers

The similarity of index terms x_1,x_2,x_3,x_4,x_5 with the terms x_6,x_7, and x_8 was expressed in linguistic terms using the linguistic scale illustrated in Figure 5. The interface for inputting thesaurus terms is illustrated in Figure 13. The fuzzy thesaurus matrix was expressed in the following manner.

$$T = \begin{array}{c} \\ X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \end{array} \begin{array}{c} \begin{array}{cccccccc} X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 & X_8 \end{array} \\ \begin{bmatrix} VHS & DS & DS & DS & DS & DS & VHS & DS \\ DS & VHS & DS & DS & DS & DS & DS & VHS \\ DS & DS & VHS & DS & DS & HS & DS & DS \\ DS & DS & DS & VHS & DS & DS & DS & DS \\ DS & DS & DS & DS & VHS & DS & DS & DS \end{bmatrix} \end{array}$$

The thesaurus matrix T consisting of crisp values using equations (6), (7) and (8) was represented as:

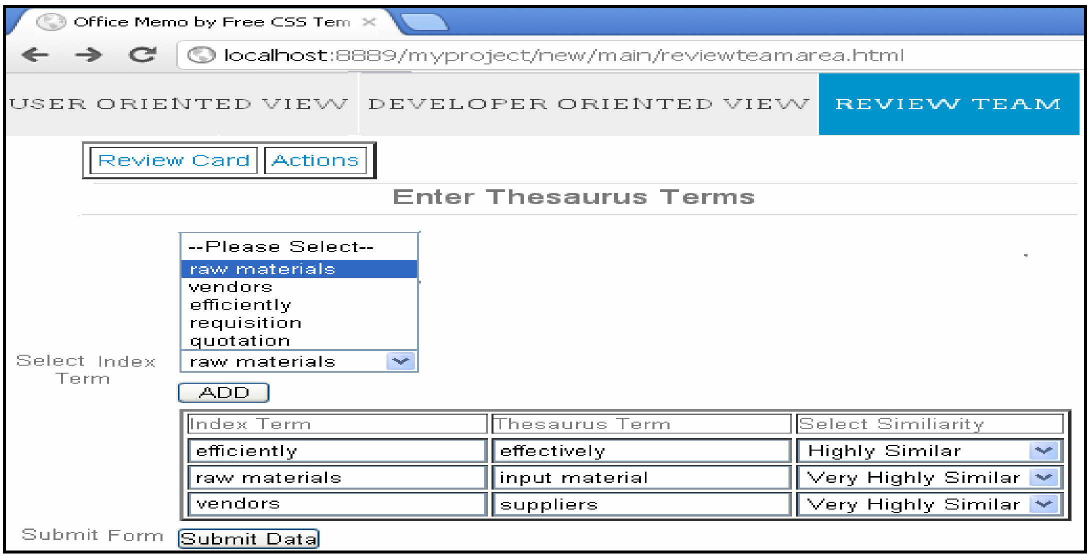
$$T = \begin{array}{c} \\ X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \end{array} \begin{array}{c} \begin{array}{cccccccc} X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 & X_8 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0.77 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Using equation (9), the augmented query was obtained in the form of matrix A_q.

$$A_q = \begin{array}{c} \begin{array}{cccccccc} X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 & X_8 \end{array} \\ \begin{bmatrix} 0.8 & 0.16 & 0.46 & 0.46 & 0.16 & 0.77 & 0.8 & 0.16 \end{bmatrix} \end{array}$$

Fuzzy relation R was comprised by specifying the relevance of grade of $\forall\, index\ term\ x_i\ (1 \le i \le 8)$ in $\forall\, USC_j$, $(1 \le j \le 10)$.

Figure 13 Interface for review team for inputting Thesaurus terms (see online version for colours)



Thereafter the retrieved set of USCs entitled as D was obtained by composing augmented query A_q with relation R using equation (10). Thus obtained relations R and D are illustrated in the following matrices.

$$R = \begin{bmatrix} & USC_1 & USC_2 & USC_3 & USC_4 & USC_5 & USC_6 & USC_7 & USC_8 & USC_9 & USC_{10} \\ X_1 & 0.40 & 0 & 0 & 0.40 & 0.40 & 0.40 & 0 & 0 & 0 & 0 \\ X_2 & 0.52 & 0 & 0.52 & 0 & 0 & 0 & 0.52 & 0 & 0 & 0 \\ X_3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ X_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ X_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ X_6 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ X_7 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ X_8 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} USC_1 & USC_2 & USC_3 & USC_4 & USC_5 & USC_6 & USC_7 & USC_8 & USC_9 & USC_{10} \\ 0.46 & 0.8 & 0.16 & 0.40 & 0.40 & 0.40 & 0.16 & 0 & 0 & 0 \end{bmatrix}$$

The joint team involving users and developers was involved to validate the goals of AC_1 against users' needs accommodated in various USCs.

Various layers of USCs as per their relevance expressed by several α-cuts (from definition ii) were obtained from matrix D as illustrated in Table 1. The distinct values of D were treated as various values of α. For illustration α-cuts corresponding to the value of α =0.40 involves USCs associated with the values>=0.40.

Table 1 USCs retrieved against various values of α

Various values of α	*α –cuts (Sets of USCs)*
0.8	USC_2
0.46	USC_1, USC_2
0.40	USC_1, USC_2, USC_4, USC_5, USC_6
0.16	USC_1,USC_2, USC_3,USC_4, USC_5, USC_6, USC_7
0	USC_1, USC_2, USC_3, USC_4, USC_5, USC_6, USC_7, USC_8, USC_9, USC_{10}

To obtain the paramount value of α, a result analysis was done that is illustrated in the subsequent sub-section.

4.1 Result analysis

For evaluating the pre-eminent value of α in retrieving the most relevant requirements, a number of performance measures were computed that are described in the following equations.

The recall (R) (Ali, 2011) can be defined as the proportion of the relevant requirements retrieved out of the total relevant requirements. In the proposed approach with backward traceability from ACs to USCs, with the fuzziness associated with the relevancy of USCs the recall can be obtained by the following equation.

$$R_{USC} = \frac{\#\{\{Highly\ relevant\ USC\} \cap \{retrieved\ USCs\}\}}{\#\{relevant\ USCs\}} \quad (15)$$

In forward tracing from USCs to ACs, the recall can be defined as the proportion of highly relevant ACs retrieved out of the total relevant ACs.

$$R_{AC} = \frac{\#\{\{Highly\ relevant\ ACs\} \cap \{retrieved\ ACs\}\}}{\#\{relevant\ ACs\}} \quad (16)$$

In backward tracing from ACs to USCs, the precision (P) (Ali, 2011) can be defined as the proportion of highly relevant USCs retrieved out of the total retrieved USCs.

$$P_{USC} = \frac{\#\{\{Highly\ relevant\ USCs\} \cap \{retrieved\ USCs\}\}}{\#\{retrieved\ USCs\}} \quad (17)$$

In a similar fashion in forward tracing from USCs to ACs, P can be defined by the following equation.

$$P_{AC} = \frac{\#\{\{Highly\ relevant\ ACs\} \cap \{retrieved\ ACs\}\}}{\#\{retrieved\ ACs\}} \quad (18)$$

Another measurement for measuring the effectiveness of retrieval of USCs/ACs is known as F-measure (Ali, 2011) that is evaluated by the following equation.

$$F_\beta = \frac{(1+\beta^2)\cdot P \cdot R}{\beta^2 \cdot P + R} \quad (19)$$

where $(0 \le \beta \le \infty)$ is a parameter that monitors the weight assigned to P and R. IF β is 1, then R and P are equally weighted and F_1 becomes the harmonic mean of R and P. If β>1, then R is weighted higher than P else if β<1, then F_β incorporates higher weightage of P. At β=0 value of F_β becomes equal to P. Value of F_β lies in [0,1]. The value of F_β reaches its best value at 1 and worst score at 0.

Another performance measure is the fall-out (FO) that is delineated as the trivial requirements retrieved, out of all trivial requirements available. The same can be obtained in the context of USCs and ACs using equations (20) and (21).

$$FO_{USC} = \frac{\#\{\{trivial\ USCs\} \cap \{retrieved\ USCs\}\}}{\#\{trivial\ USCs\}} \quad (20)$$

$$FO_{AC} = \frac{\#\{\{trivial\ ACs\} \cap \{retrieved\ ACs\}\}}{\#\{trivial\ ACs\}} \quad (21)$$

The value of FO as 0 is considered as the superlative case and 1 as the worst case in tracing the requirements. The USCs retrieved having no matching with any index terms or the index terms with 'VLI' were considered to be trivial USCs.

The miss (M) is another means to measure the effectiveness of the proposed approach. It refers to the fraction of highly relevant requirements missed to be retrieved and is computed using the equations (22) and (23) in the context of USCs and ACs.

$$M_{USC} = \frac{\#\{Highly\ relevant\ USCs\ not\ retrieved\}}{\#\{Highly\ relevant\ USCs\}} \quad (22)$$

$$Miss_{AC} = \frac{\#\{Highly\ relevant\ ACs\ not\ retrieved\}}{\#\{Highly\ relevant\ ACs\}} \quad (23)$$

The result analysis of the retrieval of various USCs against several values of α is presented in Table 2.

From Table 2, the value of α=0.40 was found to be the paramount value to retrieve the most relevant USCs with no miss, no fallout, highest recall, precision and F-measure. A comparative analysis of performance measures for various values of α is illustrated in Figure 14. Screen shot using MySQL illustrating the retrieval of relevant USCs against various levels of α is shown in Figure 15.

Table 2 Resultant analysis for retrieval of USCs against various values of α

α	*R (%)*	*P (%)*	*F-measureβ=1 (%)*	*M (%)*	*FO (%)*
0.80	14.20	100	24.99	80	0
0.46	28.57	100	44.44	60	0
0.40	71.42	100	83.32	0	0
0.16	71.42	71.4	71.42	0	40
0	71.42	50	100	0	100

Figure 14 A comparative analysis of performance measures for various values of α (see online version for colours)

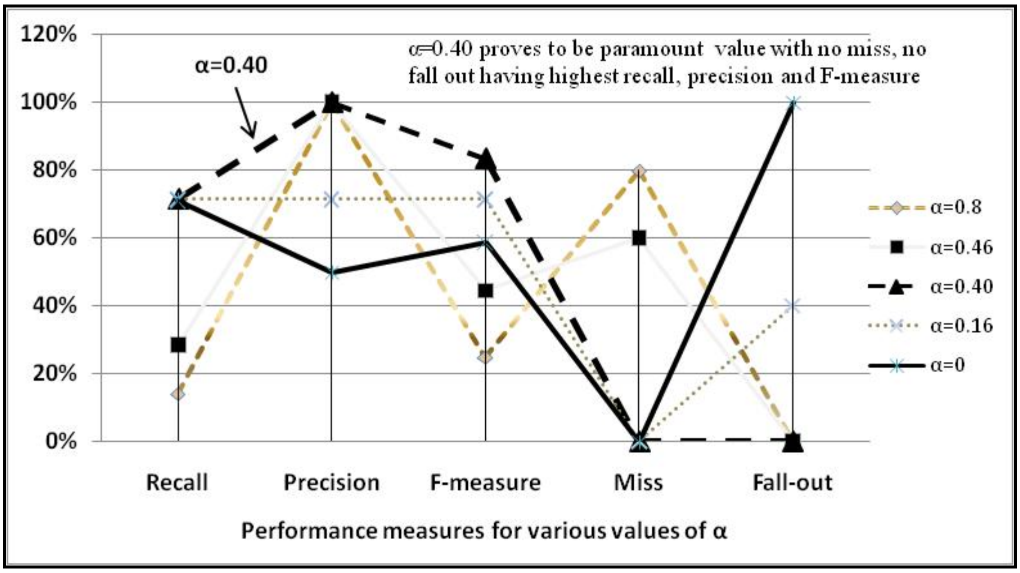
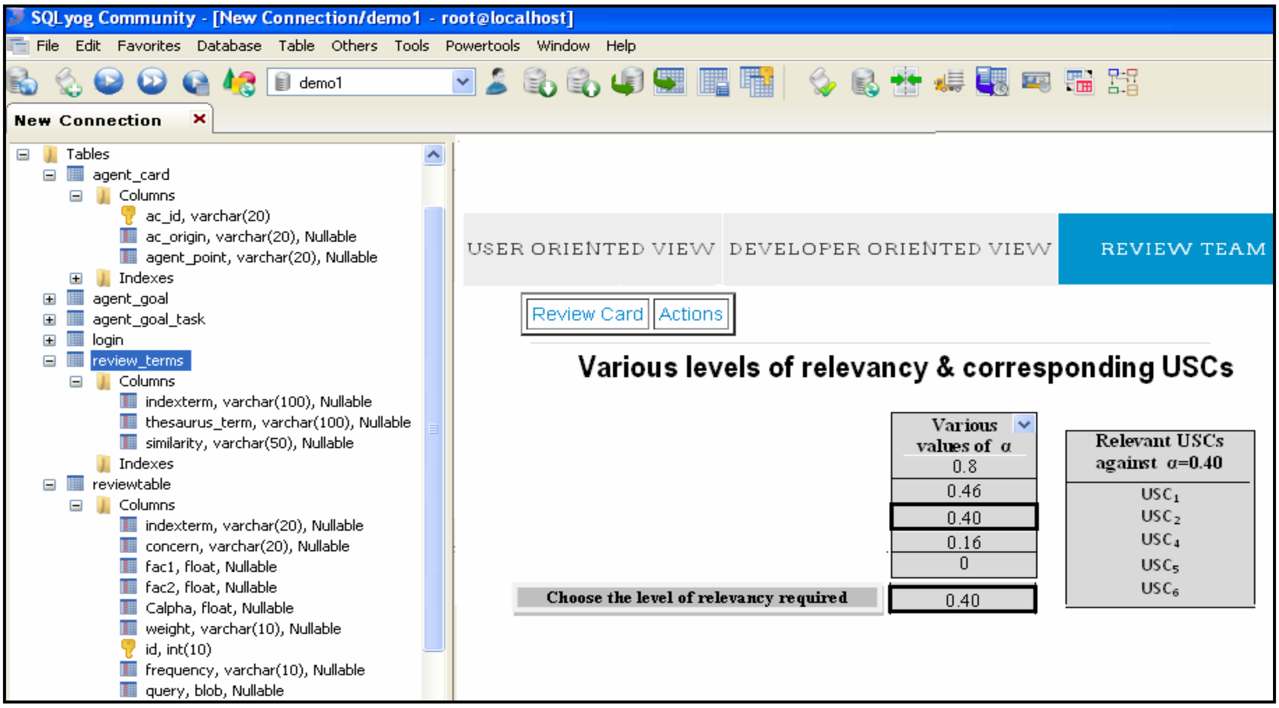


Figure 15 MySQL based Screen shot depicting relevant USCs against value of α=0.40 (see online version for colours)



4.2 Gap analysis to endorse completeness, correctness & consistency

Gap analysis is delineated as the difference between what is needed and what is available. Gap analysis is a comparison process of two systems, and is undertaken as a means of bridging the gap between them (Amaral and Faria, 2010). The goal of the gap analysis is to identify the discrepancies in the defined requirements and users' needs.

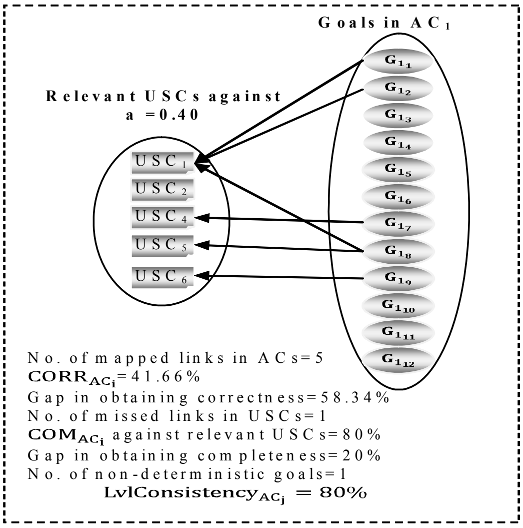
From Table 2, α=0.40 was found to be the paramount value to trace out the relevant USCs with respect to AC_1. A gap analysis was performed on the requirements retrieved from USC_1, USC_2, USC_4, USC_5, USC_6 and the goals in AC_1. The results are illustrated in Figure 16.

The goals $G_{1_3}, G_{1_4}, G_{1_5}, G_{1_{16}}, G_{1_{10}}, G_{1_{11}}, G_{1_{12}}$ were found to be unmapped to any of the USCs leading to a gap of 58.34% in obtaining the correctness. Thereafter observation was made to find out the unmapped USCs. It was observed that USC_2 was left out to be mapped to any of the goals of AC_1, hence obtained a gap of 20% in the completeness. The goal G_{1_8} 'Procuring raw materials' mapped to the USC_1 and USC_5 was found to be non deterministic as 'Procuring raw materials in-house' contradicts to 'Vendor development for raw materials'.

As the goals in AC_1 were found to be incomplete, inconsistent and not as per the expectations of users, hence a workshop was required to be conducted with the users so as to validate the requirements. The process was iteratively executed till all users as well as developers, both were satisfied and the values of completeness, correctness and consistency metrics i.e. $COM_{AC_1}, CORR_{AC_1}, Lvl\ Consistency$ were found to be 1.

The same process was extended for rest of all ACs also. This way a complete, correct and consistent set of USCs and ACs worked as a SRS for the system.

Figure 16 Gap analysis to validate goals of AC_j against various USCs



5 A comparative analysis of requirements traceability techniques

A number of RT techniques viz. Improving RT (IRT) (Hayes et al., 2003), RT in Agent Oriented Development (RTAOD) (Castro and Pinto, 2003), Goal Centric Traceability (GCT) (Hayes et al., 2003), An Augmented Vector Space (AVS) (Yoshihisa, 2011), A Trace Meta Model (TMM) (Alejandro, 2011) and RT for Object Oriented Systems (RTOOS) (Ali, 2011) are reported in literature so as to facilitate RV. A brief description of these would be in order.

An evaluation of all the prescribed RT methods including the proposed approach was carried out. The weaknesses and strengths of these methods along with a number of parameters viz. Year, publication details, traceability technique, thesaurus etc. are presented in Table 3.

Table 3 A comparative analysis of various requirements traceability methods

S.No.	*Author name*	*Year & Publication*	*Ease of use*	*Concepts*	*Traceability technique*	*Support for*			*Weaknesses*	*Strengths*
						Performance analysis	*Thesaurus*	*Fuzzy concerns*		
1	Huffman	2003, Proceedings, IEEE	expressive, logic based	RT	VM, similarity measure	recall, precision, miss	in [0,1], informal	no support	As weights of index terms are not enforced by the fuzzy concerns of analysts, hence their expectations are not reflected truly in the final retrieval	Incorporation of thesaurus amplifies the spectrum of relevancy in the final retrieval
2	Jaelson	2003, LNCS, Springer	expressive, diagrams based	actors, goals, tasks	strategic dependency & rationale diagrams	impact analysis	no support	no support	It is purely diagram based and does not incorporate any algorithm to be automated	Enhances the comprehension of Tropos methodology by exploring a number of traceable links using a diagram based reference model

Table 3 A comparative analysis of various requirements traceability methods (continued)

S.No.	*Author name*	*Year & Publication*	*Ease of use*	*Concepts*	*Traceability technique*	*Support for*			*Weaknesses*	*Strengths*
						Performance analysis	*Thesaurus*	*Fuzzy concerns*		
3	Jane Cleland	2005, Proceedings, ACM	expressive, logic based	GCT, NFR	probabilistic network algorithm, event based traceability	impact analysis	no support	no support	It appears to be in its formative stages for imparting enough operational details	It outlines a framework for improving the change management and impact analysis of NFR by establishing dynamically executable traces between NFR goals and architectural assessment models
4	Yoshihisa	2011, Proceedings, ACM	expressive, logic based	SysMl, document analyzer, traceability generator	VM, key terms, main terms, family terms	recall, precision	no support	no support	Problem of scalability with amplified main terms and family terms in large sized projects. As weights of family terms are not enforced by the fuzzy concerns of analysts, their expectations are not reflected truly in the final retrieval.	Enables traceability from requirements to designs and vice versa by augmenting VM using family terms and attributed terms. It facilitates a better comprehension of the relationships among various artefacts. At every stage it breaks down the document and contributes in improving recall and precision in requirements' trace.
5	Alejandro	2011, LNCS, Springer	expressive, logic based	DW, validation	QVT rules, traceability models, CIM, PIM	no support	no support	no support	Problem of scalability and long term maintenance with large heterogeneous data	Capable of tracing the large amount of requirements from the conceptual models and various heterogeneous data sources. It enables to easily assess the impact of changes and regenerate the affected parts

Table 3 A comparative analysis of various requirements traceability methods (continued)

S.No.	*Author name*	*Year & Publication*	*Ease of use*	*Concepts*	*Traceability technique*	*Support for*			*Weaknesses*	*Strengths*
						Performance analysis	*Thesaurus*	*Fuzzy concerns*		
6	Nasir Ali	2011, Proceedings, IEEE	expressive, logic based	requirements, source code	VM, Code partitioning and Voting	recall, precision, F-measure	no support	no support	Problem of scalability with large sized source code. As weights of index terms in query and the source elements are not enforced by the fuzzy concerns of analysts, hence their expectations are not reflected truly in the final retrieval	Enables the traceability relations between requirements and source code. It improves the accuracy in final retrieval by reducing the false positive links using the source code partitioning approach in terms of classes, methods, variables and comments
7	Proposed approach		expressive, logic based	USC, AC, agents, goals, users, RT, RV	VM, FT	recall, precision, miss, fall-out, F-measure	in [0,1], formal, using linguistic terms & TFNs	weights of index terms in query are enforced by the fuzzy concerns of analysts	Manual efforts are required in finding out unmapped links in gap analysis	Augments FT to VM in modelling the vagueness, haziness and non-specificity associated with the requirements. It facilitates tracing the requirements up to a desired degree of relevance and hence contributes in improving recall, precision, F-measure and reducing the miss & fall-outs. The weights of index terms in the query are more pragmatic as these are driven by its occurrence as well as the fuzzy concerns of analysts. Incorporation of fuzzy thesaurus is more realistic and amplifies the spectrum of relevancy in the final retrieval.

It was observed that the methods IRT, AVS and RTOOS employ the VM approach in retrieving the requirements relevant to the index terms in the query. In these approaches the weights of index terms in a query are computed with respect to its occurrences in other requirements. The term occurring in the query more number of times would be credited more weight but may not be highly relevant to the analyst, hence these methods lack in meeting the vague and fuzzy concerns of analyst in retrieving the requirements. RTAOD is a diagram based approach and expressive to comprehend, while TMM is capable of tracing the large amount of requirements from the conceptual models and various heterogeneous data sources, however in these approaches, the vagueness inherent in the concerns of the analysts are not contemplated. Out of these approaches the concept of thesaurus is augmented in IRT, the same is amplified in the proposed approach using the concept of TFNs. Though all the prescribed methods endow with a good precision by tracing highly relevant requirements but the expectations of analyst are not truly reflected in the final retrieval.

Therefore the propensity to miss the relevant requirements and incorporate the trivial requirements formulated in the form of fall outs is high, but the proposed approach cuts down the fall outs and amplifies the precision by retrieving the requirements up to the desired degree of relevance of the analyst.

The proposed approach provides a number of following advantages over other prescribed methods.

Firstly it can deal with the subjectivity and imprecise data associated with the queries and thesaurus through the use of linguistic terms and brings the retrieval of requirements closer to the expectation of analysts. Secondly, the weights of index terms in the query are more pragmatic as these are driven by their occurrence as well as the fuzzy concerns of analysts. Thirdly, the concept of α-cut allows the partial matching and thereby retrieval of the requirements that approximate the query, hence ensures the retrieval of requirements with high recall, precision, F-measure with no miss and fall outs. Additionally it provides the joint team with the greater flexibility in tracing the requirements up to a desired degree of relevancy. If the total number of retrieved requirements is too large, value of α facilitates to ignore some requirements according to their degree of irrelevancy. Fourthly, the fuzzy relevance relations and fuzzy thesauri engaged in the proposed approach are more expressive than their crisp counterparts and their comprehension is highly rational. Fifthly, as the proposed approach can be automated, therefore it is less prone to errors. Finally it incorporates the gap analysis to ensure the completeness, correctness and consistency in retrieved set of requirements. In turn, these benefits help lowering the validation costs of the system.

6 Conclusions

The errors overlooked in the requirements phase outlay tremendously in the subsequent phases and hence may result in the collapse of the system. Therefore the significance of the RV cannot be repudiated. RT is the ability to get across the life of a requirement, in both a forward and backward direction, i.e. from its origins to its specification. The manual and traditional RT methods are prone to errors and ignore the fuzziness involved in requirements' retrieval. The proposed approach employs the potential benefits of FT and VM for tracing the requirements from USCs to ACs and vice versa. The concept of α-cuts is a stupendous means to model the vagueness, fuzziness and non-specificity associated with the requirements. It establishes the relevance of the retrieved requirements up to a desired degree and thus imparts directives to the review team to regulate the scrutinisation of requirements. A number of performance measures, viz. recall, precision, F-measure, fall out and miss, are employed in evaluating the paramount value of α to ensure the utmost relevancy and fewest drop outs. The projected approach assists in ensuring the correctness, completeness and consistency of requirements by avoiding superfluous efforts and consequently, contributes in trimming down the validation costs of the system.

References

Akbar, U. (2009) 'Empirical studies of RV techniques', *IEEE*, pp.1–9.

Ali, N. (2011) 'RT for object oriented systems by partitioning source code', *18th Working Conference on Reverse Engineering*, pp.45–54.

Amaral, L.M.G. and Faria, J.P. (2010) 'A gap analysis methodology for the team software process', *IEEE*, pp.424–429.

Castro, J. and Pinto, R. (2003) 'RT in agent oriented development', *LNCS, Springer*, Vol. 2603, pp.169–191.

Cleland-Huang, J. (2005) 'Toward improved traceability of NFRs', *Proceedings on Traceability, ACM*, pp.14–19.

Cross, V. (1994) 'Fuzzy information retrievals', *Journal of Intelligent Information Systems*, Vol. 3, pp.29–56.

Gaur, A. and Soni, A. (2010) 'An integrated approach to prioritize requirements using fuzzy decision making', *IACSIT International Journal of Engineering and Technology*, Vol. 2, No. 4, pp.320–328.

Gaur, V., Soni, A. and Bedi, P. (2010) *An Agent-Oriented Approach to Requirements Engineering*, IEEE Press and IEEE Xplore, USA, pp.449–454.

Hayes, J.H., Dekhtyar, A. and Osborne, J. (2003) 'Improving requirements tracing via IR', *IEEE Proceedings on RE*, pp.138–147.

Klaus, P. (2011) *RE- Fundamentals, Principles and Techniques*, Springer.

Klir, G.J. and Yuan, B. (1995) *Fuzzy Sets and Fuzzy Logic*, PHI publications.

Maté, A. and Trujillo, J. (2011) 'A trace meta model proposal based on the model driven architecture framework for the traceability of user requirements in data warehouses', *Proceedings of the 23rd international conference on Advanced information systems engineering*, *Springer LNCS*, pp.123–137.

Palyagar, B. (2006) 'Validating RE process improvement-a case study', *Proceedings of the first international workshop on Requirements Engineering Visualization*, p.9.

Shuda, W., Jiangping, L. and Riu, W. (2009) 'Research of information filtering based on vector space model', *IEEE*, pp.42–46.

Smith, M.H. (1998) 'Fuzzy data mining for querying and retrieval of research archival information', *Proceedings on Fuzzy Information Processing*, pp.140–145.

Sreenivas, A. (2006) 'A rigorous approach to RV', *IEEE*, pp.236–245.

Stein, B. (2005) 'Fuzzy-fingerprints for text-based information retrieval', *Knowledge Management*, pp.572–579.

Stephane, S. (2005) 'Use cases based RV with scenarios', *IEEE*, pp.465–466.

Udagawa, Y. (2011) 'An augmented vector space IR for recovering RT', *Proceedings of IEEE*, pp.771–778.

Yerra, R. (2005) 'Detecting similar HTML documents using a fuzzy set information retrieval approach', *IEEE*, Vol. 2, pp.693–699.

Yousuf, F. (2008) 'RV techniques in GSD', *IEEE*, pp.553–557.