IJRIT International Journal of Research in Information Technology, Volume 1, Issue 5, May 2013, Pg. 212-219


International Journal of Research in Information Technology (IJRIT)

www.ijrit.com ISSN 2001-5569

# Identifying Non-stakeholders using NSIA algorithm

[1] P.Dugalya, [2] R.Reenedevi

[1] PG Scholar, Department of CSE, Sona College of Technology, Salem, TN, India

[2] Assistant Professor, Department of CSE, Sona College of Technology, Salem, TN, India

[1] dugalya.89@gmail.com , [2] Reena-poun@yahoo.co.in

**Abstract**

Requirement elicitation is the software engineering activity in which it requires involvement with client domain experts and end users in order to establish a client's needs. For large scale projects several methods are used to identify and prioritize the requirement using techniques. Various social network measures are used for prioritization process. Stakeholders are asked to rate an initial set of requirements; they can also suggest other requirements. The requirements are then prioritized using their ratings weighted by their influence in project. The method which is used for prioritization assumes that stakeholder provide recommendation honestly. But malicious stakeholder may provide responses for their personal benefits such as recommending non stakeholders exclude some stakeholder's. This in turn affects the quality of stakeholders and requirements. Thus to reduce the malicious stake holder's requirement ranking, based on the changing behavior of their requirement specification rating an algorithm is developed to identify the non-stakeholders. NSIA-Non stakeholder identification algorithm is an algorithm which identifies the Non stakeholders.

**Index terms-** *Requirements/Specification, Elicitation methods, Requirement prioritization, Recommender system, Social network analysis, Stakeholder*

## 1. Introduction

Nowadays growth in software system is increasing. Large-scale projects contain more number of stakeholders and they all are diverse in their needs. Requirement elicitation is the practice of collecting the requirements from users, customers and other stakeholders [16]. It includes practices such as interviews, questionnaires, user observation, Workshops and role playing. It consists of process such as identifying stakeholders, requirements from these stakeholders and prioritizing their requirements. Requirement prioritization is used for determining which candidate requirements of a software product should be included in a certain release [15] [4] [5].

Stake Rare is the method which is used for identifying and prioritizing the requirements. It can be done using social networks and collaborative filtering. Stake Rare stands for-Stakeholder Recommender assisted method for requirements elicitation. Due to large numbers they were unable to meet, because they are diverse by their locations and time constraint [9].

Stake rare uses social network to identify and prioritize stakeholders and their roles in the project. Then, it asks the stakeholders to rate an initial set of requirements using their ratings weighted by their project influence derived from their position on the social network. The process involves construction of social network with stakeholders as nodes and their recommendations as links. A social network measure [7] [8] [10] such as Betweenness centrality is used for prioritization process. Stakeholders are asked to rate an initial set of requirements; they can also suggest other requirements.

The requirements are then prioritized using their ratings weighted by their influence in the project. Collaborative filtering is the process of filtering the information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources [11] [12] [13]. This is used for prediction of user information. The prediction can be done by using variety of algorithms such as K-Nearest Neighbor (KNN).K-Nearest Neighbor algorithm [11] is used to find like-minded users.

Social network measures [16] [7] [8] [10] such as Betweeness centrality, Load centrality, Degree centrality, In-degree centrality, and Out-degree centrality were used to analyze collaboration behavior. Stake Rare assumes that stakeholder provide recommendation honestly. But malicious stakeholder may provide responses for their personal benefits such as recommending non stakeholders exclude some stakeholders [9].

The system is resilient against stakeholders who misuse their signature power, since their only obligation is not to sign conflicting blocks, the only way they could double spend is if they first sign one block so it achieves a majority, then sign a different one so that it achieves a greater majority. Thus to reduce the malicious stakeholder's requirement ranking, based on the changing behavior of their requirement specification rating an algorithm is developed to identify the non-stakeholders. The Malicious Stakeholder can be identified based on number of clicks they used for rating the requirements.

## 2. Related work

2.1 Requirement-driven collaboration is the collaboration during the management and development of requirements.

- Constructs a requirement-centric social network which represents the membership and relationships among members working on a requirement and its associated downstream artifacts.
- Outlines a number of social network analysis techniques to study collaboration aspects such as communication, awareness, and the alignment of technical dependencies driven by development of requirements and social interactions.

**2.2**.Requirement elicitation is the software engineering activity in which stakeholder needs are understood. It involves identifying and prioritizing requirements—a process difficult to scale to large software projects with many stakeholders. This paper proposes Stake Rare, a novel method that uses social networks and collaborative filtering to identify and prioritize requirements in large software projects [9].

**2.3**. Adequate, timely and effective consultation of relevant stakeholders is of paramount importance in the requirements engineering process. The literature suggests examples of stakeholders, and categories of stakeholder, but does not provide help in identifying stakeholders for a specific system. In this paper, we discuss current work in stakeholder identification, propose an approach to identifying relevant stakeholders for a specific system, and propose future directions for the work [18].

2.4. The primary measure of success of a software system is the degree to which it meets the purpose for which it was intended. Software systems requirements engineering (RE) is the process of identifying stakeholders and their needs, and documenting these in a form that is amenable to analysis, communication, and subsequent implementation. There are a number of inherent difficulties in this process. Stakeholders (including paying customers, users and developers) may be numerous and distributed. Their goals may vary and conflict, depending on their perspectives of the environment in which they work and the tasks they wish to accomplish [19].

## 3. Proposed work

The NSIA algorithm is proposed to identify the Non–Stakeholders. The architecture for the proposed system is illustrated in Figure1.The architecture describes the steps for identifying, prioritizing stakeholders and their requirements.

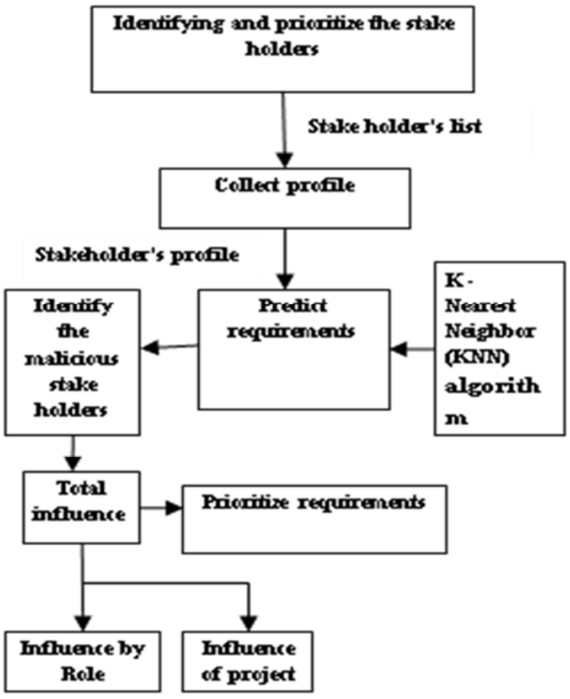



Figure 1: An architecture for Non-stakeholder Identification

### 3.1 Identify and Prioritize Stakeholder

Stakeholders have to be prioritized. As their level of influence in the project affects the priority of the requirements. Stake net is a stakeholder analysis method used to identify initial Stakeholders. It then asks to recommend other Stakeholders [9].It uses various social network measures to prioritize the Stakeholders.

### 3.2 Collect Profile from stake holders

Elicitation methods such as interviews can be used to identify initial set of requirements. Stakeholders identified from previous step can provide their preferences on the Requirements which are in the form of [9]:

<Stakeholder, Requirement, Rating>.

Table 1: Concepts

| Concept | Description |
|---|---|
| Stakeholder | An Individual or a group |

| | |
|---|---|
| | who can influence or influenced by the project success or failure of a project[19]. |
| Stakeholder role | The Stakeholder's position in the Project[18] |
| Requirement | The goals, funtions and constraints on software systems.[13] |
| Rating | Numerical Importance given in terms of values Range from 1-5[11] |

### 3.3 Predict Requirements

In this step requirements are predicted based on the collected Stakeholder's profile. Collaborative filtering systems use KNN algorithm to find Stakeholder's and measure the similarity between their profiles; it then generates predicted level of interest. Finally rating can be given to the predicted requirements. [9]

### 3.4 Finding malicious stake holders

To reduce the malicious stake holder's requirement ranking, based on the changing behavior of their requirement specification rating an algorithm was proposed. Using the algorithm, the no of clicks made for rating the requirement can be calculated. If the clicks made by stakeholders for rating the requirements exceeds more than threshold value, then resultant is a non-stakeholder.

The algorithm for identifying Non-Stakeholder is as follows,

Input data:
Where(i =1 to N)
/*N=Total number of stakeholder
Compute $C_i$
/*$C_i$=Clicks made by user
/*TR=Total number of Requirements
If $C_i \leq$(TR ± α),1<α<3
then
$C_i$ ξ Stakeholders
else
$C_i$ ξ Non-Stakeholders

**Table 2: NSIA**

### 3.5 Prioritize Requirements

The requirements are then prioritized based on the influence of Stakeholder role in the project, influence of Stakeholder role. The importance can be calculated based on the values of Influence of the stakeholder's role in the project and Influence of the stakeholders in their roles is determined using the formulas:

- The influence of stakeholder i's role in the project is calculated as,

$$Influence_{role(i)}=\frac{RRmax+1-rank(role(i))}{\sum_{j=1}^{n}(RRmax+1-rank(role(j)))}$$

- The influence of stakeholder i in the role is calculated as,

$$Influence_i=\frac{RSmax+1-rank(i)}{\sum_{k=1}^{n} RSmax+1-rank(k))}$$

- The influence of stakeholder i in the project is calculated as,

$$ProjectInfluence_{i=}Influence_{role(i)}\times Influence_i$$

- The importance of a requirement is calculated as,

$$Importance_{R=}\sum_{i=1}^{n} ProjectInfluence\times r_i$$

Finally based on importance their requirements are prioritized [9].

## 4. Evaluation

The Requirements identified by Stake rare was compared in terms of precision and recall. These are the results or data obtained by using NSIA.

TABLE 3-Data

| **Factors** | **Numbers** |
|---|---|
| Number of requirements | 20 |
| Number of Stakeholders provided rating | 35 |
| Number of stakeholders | 24 |
| Number of non-Stakeholders | 11 |

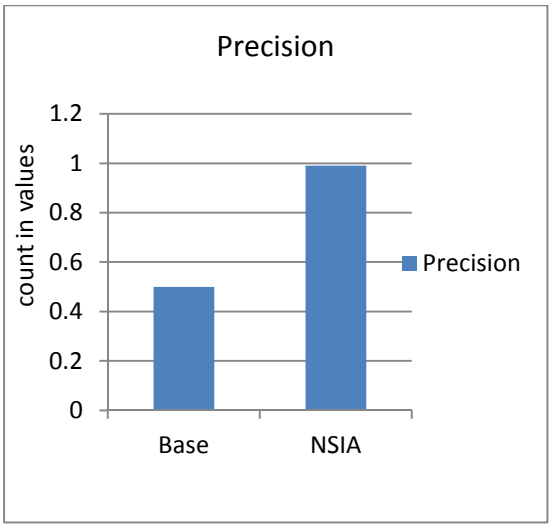
From these data's the important requirements are calculated by considering only stakeholders rating.

### 4.1 Precision

Precision is the probability that a (randomly selected) retrieved document is relevant. The requirement identified is the number of actual requirements in the set of identified requirements divided by the total number of identified requirements.

$$Precision = \frac{|\{\times\} \cap \{GroundTruth\}|}{|\{GroundTruth\}|}$$

Here X is the set of requirements identified by NSIA or existing method



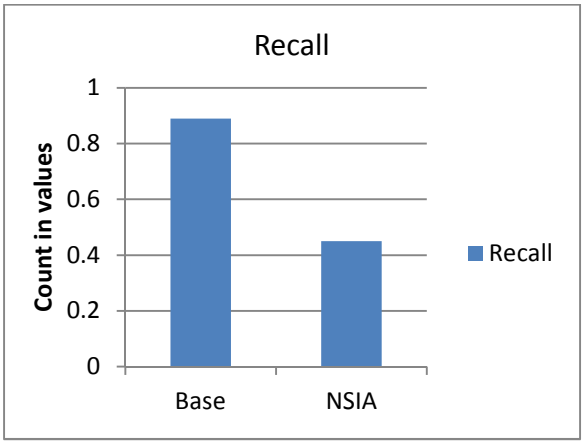

Graph 1

**4.2 Recall**

Recall is the probability that a (randomly selected) relevant document is retrieved in a search. The Requirements identified in the set of identified requirements is the number of actual requirements divided by the total number of actual requirement with X the same as for precision. Both precision and recall range from 0 to 1.

$$Recall = \frac{|\{\times\} \cap \{GroundTruth\}|}{|\{GroundTruth\}|}$$

A precision of 1 means all the identified requirements are actual requirements. A recall of 1 means that all the actual requirements are identified.




Graph 2

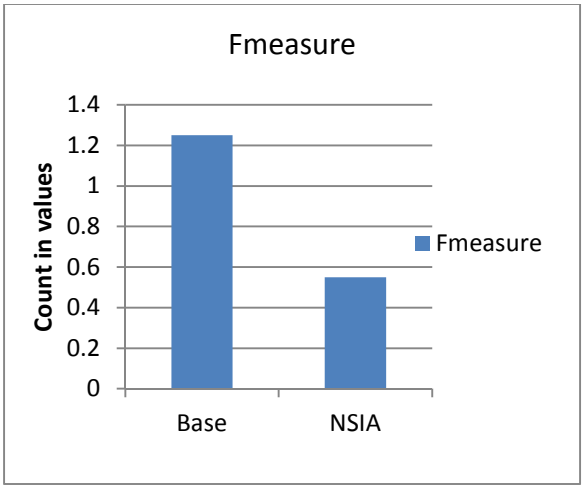
### 4.3 F-measure

A measure that combines precision and recall is the harmonic mean of precision and recall, the traditional F-measure or balanced F-score:

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

This is also known as $F_1$ measure, because recall and precision are evenly weighted.

Two other commonly used $F$ measures are the $F_2$ measure, which weights recall higher than precision, and the $F_{0.5}$ measure, which puts more emphasis on precision than recall.




Graph 3

## 5. Conclusion

Large-scale projects include more number of stakeholders and they all are diverse in their needs. Requirement elicitation is the practice of collecting the requirements from users, customers and other stakeholders.

Requirements rated by malicious stakeholders affect the quality of product. In this, NSIA algorithm is used to identify the Non-Stakeholders based on no of clicks made by them for rating the requirements. It also helps to find out all the important requirements of the projects.

## 6. References

[1] L. Lehtola, M. Kauppinen, and S. Kujala, “Requirements Prioritization Challenges in Practice,” Proc. Int’l Conf. Product Focused

[2] S.L. Lim, “Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation,” PhD thesis, Univ. of New South Wales,//www.cs.ucl.ac.uk/staff/S.Lim/phd/thesis_soolinglim.pdf, 2010.

[3]A.vanLamsweerde, “Goal-Oriented Requirements Engineering: A Guided Tour,” Proc. Fifth IEEE Int’l Symp. Requirements Eng., pp. 249-262, 2001.

[4] J. Karlsson and K. Ryan, “A Cost-Value Approach for Prioritizing Requirements,” IEEE Software, vol. 14, no. 5, pp. 67-74, Sept. /Oct. 1997.

[5] J. Azar, R.K. Smith, and D. Cordes, “Value-Oriented Requirements Prioritization in a Small Development Organization,” IEEE Software, vol. 24, no. 1, pp. 32-37, Jan. /Feb. 2007.

[6] N.R. Mead, Requirements Prioritization Introduction. Software Eng. Inst., Carnegie Mellon Univ., 2006.

[7] S. Wasserman and K. Faust, Social Network Analysis: Methods and Applications. Cambridge Univ. Press, 1994.

[8] J. Scott, Social Network Analysis: A Handbook. Sage, 2000.

[9] S.L. Lim, and A. Finkelstein“StakeRare: Using Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation”, IEEE transactions on software engineering, vol. 38, no. 3, may/June 2012

[10] R.A. Hanneman and M. Riddle, Introduction to Social Network Methods. Univ. of California, Riverside, 2005.

[11] J. Schafer, D. Frankowski, J. Herlocker, and S. Sen., “Collaborative Filtering Recommender Systems,” The Adaptive Web: Methods and Strategies of Web Personalization, pp. 291-324, Springer, 2007.

[12] J.L. Herlocker, J.A. Konstan, A. Borchers, and J. Riedl, “An Algorithmic Framework for Performing Collaborative Filtering,” Proc. 22nd Ann. Int’l ACM SIGIR Conf. Research and development in Information Retrieval, pp. 230-237, 1999.

[13] D. Damian, I. Kwan, and S. Marczak, “Requirements-Driven Collaboration: leveraging the Invisible Relationships between Requirements and People,” Collaborative Software Engineering, Springer, 2010.

[14] N. Lathia, “Computing Recommendations with Collaborative Filtering,” Collaborative and Social Information Retrieval and Access: Techniques for Improved User Modeling, Information Science Reference, 2008.

[15] A. Herrmann and M.Daneva, “Requirements Prioritization Based on Benefit and Cost Prediction: An Agenda for Future Research, ”Proc. 16th IEEE Int’l Conf. Requirements Eng., pp. 125-134, 2008.

[16] S.L. Lim, D. Damian, and A. Finkelstein, “Stake Source: Harnessing the Power of Crowd sourcing and Social Networks in Stakeholder Analysis,” Proc. 32nd ACM/IEEE Int’l Conf. Software Eng. vol. 2, pp. 239-242, 2010.

[17] S.L. Lim, D. Damian, and A. Finkelstein, “StakeSource2.0: Using Social Networks of Stakeholders to Identify and Priorities Requirements,” Proc. 33rd ACM/IEEE Int’l Conf. Software Eng., 2011.

[18] H. Sharp, G.H. Galal, and A. Finkelstein, “Stakeholder Identification in the Requirements Engineering Process,” Proc. Database and Expert System Applications Workshop, pp. 387-391, 1999.

[19] B. Nuseibeh and S. Easterbrook, “Requirements Engineering: ARoadmap,” Proc. Conf. Future of Software Eng., pp. 35-46, 2000.