Reputation Management in Online Social Networks
A New Clustering-based Approach

Sana Hamdi[1,2], Alda Lopes Gancarski[1], Amel Bouzeghoub[2] and Sadok Ben Yahia[1,2]
[1]*University of Tunis El-Manar, Faculty of Sciences of Tunis, LIPAH-LR 11ES14, 2092 Tunis, Tunisia*
[2]*SAMOVAR, Telecom SudParis, CNRS, Université Paris-Saclay, 9 rue Charles Fourier, 91001 Evry Cedex, France*

Keywords: Social Networks, Reputation, Trust, Clustering.

Abstract: Trust and reputation management stands as a corner stone within the Online Social Networks (OSNs) since they ensure a healthy collaboration relationship among participants. Currently, most trust and reputation systems focus on evaluating the credibility of the users. The reputation systems in OSNs have as objective to help users to make difference between trustworthy and untrustworthy, and encourage honest users by rewarding them with high trust values. Computing reputation of one user within a network requires knowledge of trust degrees between the users. In this paper, we propose a new Clustering Reputation algorithm, called *RepC*, based on trusted network. This algorithm classifies the users of OSNs by their trust similarity such that most trustworthy users belong to the same cluster. We conduct extensive experiments on a real online social network dataset from Twitter. Experimental results show that our algorithm generates better results than do the pioneering approaches of the literature.

1 INTRODUCTION

Reputation-based trust management has been used as an effective solution to evaluate how much one user can trust others, to help users to make the difference between trustworthy and untrustworthy users and encourage honest users by rewarding them with high trust values.

Despite reputation is closely related to the concept of trust, it has not to be confused nor treated as trust. In fact, trust is often considered as a personal and subjective measure because it is computed primarily based on a set of personalized factors and can be derived from a combination of personal experience and relationships (Hamdi et al., 2012). However, reputation is often considered as a collective and objective measure of trustworthiness based on the transactional experiences and direct interactions of different users.

In this paper, we propose an algorithm called *RepC* for reputation management in Online Social Networks (OSNs). The proposed algorithm is based on direct and indirect trust values computed respectively in previous works (Hamdi et al., 2012) and (Hamdi et al., 2016).

RepC is truly unique since it is based on a clustering approach. In fact, it divides OSNs users into clusters (groups) such that trustworthy users belong to the same cluster.

The remainder of this paper is organized as follows. Section 2 recalls the key notions used throughout this paper. Section 3 reviews the related work dedicated to the reputation management in OSNs. Sections 4 and 5 introduce our approach that classifies the OSNs' users according to their reputation helping requesters to differentiate between benevolent and malicious ones. Section 6 describes the evaluation procedure as well as the results obtained from the real OSN Twitter. The final section sketches our contributions and points out avenues of future work.

2 BACKGROUND AND KEY NOTIONS

In OSNs, a trust network is critical and it is the basis for the reputation evaluation of users. In fact, it contains some important information as direct trust relations between users and social relations. Extracting the trust network between users becomes a fundamental and essential step before performing the reputation values of users and has important influences on their evaluation.

For example, an OSN's user A is looking for a

Hamdi, S., Gançarski, A., Bouzeghoub, A. and Yahia, S.
Reputation Management in Online Social Networks - A New Clustering-based Approach.
DOI: 10.5220/0006433104680473
In *Proceedings of the 14th International Joint Conference on e-Business and Telecommunications (ICETE 2017) - Volume 4: SECRYPT*, pages 468-473
ISBN: 978-989-758-259-2

Tennis coach and B is a Tennis coach and member on the same OSN. In such a situation, as indicated in the theory of Social Psychology (Christianson and Harbison, 1996) and Computer Science (Hamdi et al., 2016), A can evaluate the trustworthiness of B based on the trust social network.

We have defined on the basis of a previous work (Hamdi et al., 2012) how to compute direct trust values in social networks. Yet, social actors are often connected by more than one kind of relationship. Accordingly, our model titled **IRIS**, built direct trust relations (local reputation) by aggregating different ties in a multiplex network (the direct interactions between users, their existing relationship types and their interest similarity). In fact, multiplex networks exist when actors are connected through more than one type of socially relevant tie. The different ties reflect the diverse roles played by users in the network (Heaney, 2014). In an additional previous work (Hamdi et al., 2016), we have proposed an accurate model **TISoN** to infer trust in OSNs based on direct relationships between users.

In the following, we briefly sketch the key notions that will be of use in the remainder of this paper.

Definition 1.(TRUST) Trust is the subjective probability by which an individual, A, expects that another individual, B, performs a given action on which its welfare depends (Chen et al., 2011).

Definition 2.(DIRECT TRUST) In OSNs, a user A's trust in another user B is the subjective confidence, faith or expectation of the user A receiving positive outcomes through the transactions with the user B.

Definition 3.(INDIRECT TRUST) In OSNs, a user A's trust in another user B is the confidence, faith or expectation of the user A in the user B according to other users.

Definition 4.(REPUTATION) In OSNs, a user A's reputation is the global perception of its trustworthiness in the network. Furthermore, the trustworthiness can be evaluated from its past and current behaviours.

In this paper, we propose a new algorithm for reputation management in OSNs. The proposed algorithm is based on direct and indirect trust values. The different OSN users participate to help the requester to have an idea about the reputation of an OSN user (target). Some users, whom are indirectly connected to the target, are observers and propagators, they observe direct interactions and, based on their experiences, they propagate information about trust with the different users. Other users (assessors), directly connected to the target, are observers and evaluators since they evaluate directly the trust in the target. The requester can so scan the reputation of the target based on direct and indirect trust.

3 RELATED WORK

In the literature, several approaches are designed to describe how to identify the reputation of users. In the remainder of this section, we present and describe a set of some of the most representative reputation approaches for distributed networks.

3.1 SemanticWeb

SemanticWeb is a trust and reputation model specific for social networks presented in (Zhang et al., 2006). The trustworthiness between two users is computed by searching all the paths connecting them; then, for each path the ratings associated with each edge are multiplied; finally, all the scores are added (normalizing that aggregation).

Let n be the number of paths from agent A to agent B. D_i denotes the number of users between A and B on the i^{th} path. The set of B's friends or neighbours is called M, m_i denotes B's direct friend or neighbour on the i^{th} path. w_i denotes the weight of the i^{th} path. The weight of each path is computed as follows (giving a higher weight to shorter paths):

$$w_i = \frac{\frac{1}{D_i}}{\sum_{i=1}^{N} \frac{1}{D_i}}; \tag{1}$$

The reputation of B from A's point of view is computed as follows:

$$R_{A \to B} = \sum_{i=1}^{N} T_{m_i \to B} \times \prod_{i \to j} R_{i \to j} \times w_i; \tag{2}$$

Where the reliable factor $R_{i \to j}$ denotes to which degree i believes directly in j's opinions or behaviours.

In this work, the authors did not compute a global value reflecting the reputation of one user in the whole network, their model only computes the reputation of each user based on the opinion of other user. We can consider these computed scores, simply, as indirect trust between two users. In addition, these computed values are essentially based on a direct trust ($R_{i \to j}$). However, authors did not show or mention how to calculate them nowhere.

3.2 REMSA

Authors in (Lee and Oh, 2015) introduced a new model named REMSA for reputation computation in OSNs. The proposed model considers the information associated to users to model how reputation is spread within the social network. In REMSA, each

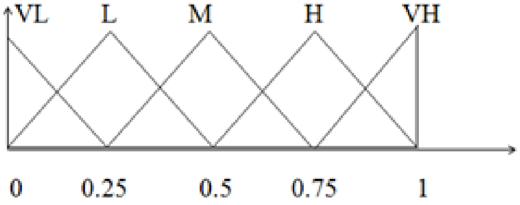

Figure 1: Stratification of reputation values.

user updates reputations values affected to his neighbours based on the history of interactions and by considering the frequency of interactions in recent history. In addition, ReMSA, uses a voting mechanism to aggregate neighbours' opinions when updating reputation values. The voting process is recursive and aims to reach to every user in the network.

With REMSA, the reputation is only based on the opinions of the neighbours. If one user has only one friend, does the trustworthiness of its opinion guaranteed?

In this work, we propose a new algorithm called *RepC* for reputation management in OSNs. *RepC* is truly unique since it is based on clustering algorithms. In fact, it aims to divide OSNs' users into clusters (groups) such that the most trustworthy users belong to the same cluster.

4 CHOOSING THE INITIAL CENTROIDS OF THE CLUSTERS

We associate a Triangular Fuzzy Number (TFN), as given in Figure 1, that enables us to specify a range for a given reputation level instead of giving it a particular discrete value. The meaning of the different linguistic values (fuzzy set) are defined as: Very Low (VL), Low (L), Medium (M), High (H), and Very High (VH), to range users from very untrustworthy to very trustworthy. The advantage of this stratification is that a reputation value denoted as "high" of one user is acknowledged by others as a high reputation value, which is also true for the other values. Thus, we avoid the problem of "what does a reputation value of 0.2, or 20%, mean? Is it high or low?", for example.

Next section describes our algorithm *RepC* based on a partitioning clustering algorithm. Indeed, the time and space complexities of the partitioning algorithms are typically lower than those of the hierarchical algorithms (Day, 1992). In fact, partitioning methods have advantages in applications requiring large datasets as OSNs, which is not always the case for hierarchical clustering for which the construction of a tree is computationally expensive. In addition, the problem of the choice of the number of desired output clusters accompanying the use of a partitioning algorithm is solved. In fact, each reputation strata shown above, presents a cluster, thus, the number of clusters (VL, L, M, H and VH) is 5.

5 REPC : A CLUSTERING ALGORITHM FOR REPUTATION MANAGEMENT IN OSNS

In this Section, we introduce the *RepC* algorithm for the clustering of OSNs' users based on their reputation. In *RepC*, the global reputation of each user u is weighted by aggregating the direct and indirect trust values assigned to user u by other ones. In Subsection 5.1, we discuss how to aggregate these normalized trust values in a sensible manner in order to obtain the corresponding reputation values. Then, in Subsection 5.2, we tackle the problem of classifying users into different clusters based on their reputation similarity such that most trustworthy users belong to the same cluster.

5.1 Aggregating Direct and Inferred Trust Values

We have defined, on the basis of our previous work (Hamdi et al., 2012), how to compute direct trust values in OSNs. Accordingly to our model titled *IRIS*, we build direct trust relations by aggregating different ties in a multiplex network (the direct interactions between users, their existing relationship types and their interest similarity). In addition, we use the accurate model TISoN we proposed in (Hamdi et al., 2016) to infer indirect trust in OSNs based on direct relationships between users. The direct and indirect computed values are normalized (all values are set between the unit interval) in the aim to lead to an elegant probabilistic interpretation.

A natural way to evaluate the reputation of an OSN user j is to aggregate the opinion of all users about that user, i.e., to consider all direct and indirect trust values assigned to him (c.f., Eq. 3).

$$r_j = \frac{\sum_{i=1}^{n} \alpha.t_{ij}/i \neq j}{n+n'} \tag{3}$$

Here r_j denotes the reputation of the OSN user j based on all users' opinions; t_{ij} is the trust value assigned to j by the user i; n is the number of OSN users and n' is the number of direct relations in the

Algorithm 1: The RepC Algorithm.

```
    Data: $R$: reputation vector with $r_j$ values,
            $j = 1 \dots n$.
          $C$: the set of initialized centroids $c_p$,
    $p = 1 \dots 5$.
          $\varepsilon$: error threshold, $\varepsilon \approx 0$.
    Result: $G$: the set of final clusters or groups
            $g_p$.
1   begin
2       $k \leftarrow 0$
3       $VL \leftarrow \emptyset; L \leftarrow \emptyset; M \leftarrow \emptyset; H \leftarrow \emptyset\ ; VH \leftarrow \emptyset$;
4       $G \leftarrow$
        $\{(VL, c_1); (L, c_2); (M, c_3); (H, c_4); (VH, c_5)\}$;
5       repeat
6           $k \leftarrow k + 1$;
7           Update clusters $\in G$ by assigning each
            user $j$ to one cluster $g_p$ such that
            $Min(|r_j - c_p| / p \in [1 \dots 5])$;
8           Recompute the vectors of centroids
            $C^k = [c_p]^k$ by using Eq. 4;
9       until $|C^k - C^{k-1}| < \varepsilon$;
```

OSN. The parameter α is defined as follows:

$$\alpha = \begin{cases} 1 & \text{if } t_{ij} \text{ is an inferred trust value} \\ 2 & \text{if } t_{ij} \text{ is a direct trust value} \end{cases}$$

We can write this in matrix notation: if we define T to be the square matrix $[t_{ij}]$, then R is the column vector with r_j values such as $j \in [1..n]$. This is a useful way to have each user gain a view of the OSN that is wider than his own experience.

5.2 Description of the RepC Algorithm

In *RepC*, we adopt the typical k-Means algorithm (Hartigan and Wong, 1979), which is the simplest and most used partitioning algorithm since it is easy to implement and its time complexity is about $O(n)$, where n is the number of objects. K-means starts with a random initial partition and keeps reassigning the object to clusters based on the similarity between the object and the cluster centroid until a convergence criterion is met. In our work, as mentioned in Section 4, the initial clusters are properly chosen and their number is equal to 5. Thus, we do not face the major drawback of k-means which is sensitive to the random selection of the initial partition.

The process, used by *RepC*, is sketched by Algorithm 1. First, in lines 3 to 4, *RepC* creates the set G of 5 empty clusters (VL, L, M, H,VH) with the initialized centroids such that $c_1 = 0; c_2 = 0.25; c_3 = 0.5; c_4 = 0.75$; and $c_5 = 1$. Second, in line 7, the algorithm assigns each user j to one cluster g_p such that his reputation r_j is closer to this cluster centroid c_p. Then, in line 8, *RepC* recomputes the centroid of each cluster as the mean of reputations of users belonging to the cluster. The process of updating and recomputing centroids of clusters as well as assigning users to the adequate clusters is repeated until the stability condition is reached (line 9).

$$c_p = \frac{\sum_{g_p} r_j}{l}, p \in [1 \dots 5], l \in [1 \dots n] \tag{4}$$

In Eq. 4, n is the number of the OSN users, l is the number of users j belonging to cluster g_p with the centroid c_p.

6 EXPERIMENTS

In this section, we describe the experiments we lead on the proposed algorithm *RepC*. In one hand, we use different types of criteria for clustering evaluation to validate the effectiveness of *RepC*. In the other hand, we aim to test our results' quality by comparing *RepC* with the existing algorithm for reputation management in OSNs REMSA.

Since trust is not randomly distributed, we conduct our experiments in the real OSN dataset Twitter [1].

6.1 Twitter Data Set

We use a data set [2] collected from the real social network Twitter. This data set containing more than 250000 users and 320000 relations, uses a social labelled graph. Each node of the graph presents a Twitter member and each edge denotes the number of retweets one user gives to another user.

6.2 Accuracy Metrics

6.2.1 Validation of the Clusters

The clusters' evaluation or the assessment of the quality of the obtained clusters presents an important topic related to clustering. Most of cluster validity measures evaluate the trade-off between cluster compactness and separability (Portmann, 2012). Other measures are used to evaluate how well a clustering approach performs on a dataset (Vendramin et al., 2010).

[1] https://twitter.com/

[2] https://snap.stanford.edu/data/higgs-twitter.html

In our experiments, we adapt the internal criteria indexes of Dunn (Dunn, 1974) as well as that of Davies and Bouldin (Davies and Bouldin, 1979). Both of these criteria consider a clustering algorithm as good and successful whenever it generates clusters with high intra-cluster homogeneity, good inter-cluster separation and high connectedness between neighbouring data objects.

The Dunn Index. The Dunn Index, I_D, identifies clusters which are well separated and compact. The goal is therefore to maximize the inter-cluster distance while minimizing the intra-cluster distance. As shown in Eq. 5, I_D is the ratio between the maximum distance separating two users classified together and the minimum distance between two users classified separately. For a good clustering, I_D should be as high as possible.

$$I_D = min_{1 \leq i \leq n}[min_{1 \leq j \leq n, i \neq j}(\frac{d(i,j)}{max_{1 \leq k \leq n} d'(k)})] \quad (5)$$

With:

- $d(i,j)$: the distance between clusters i and j
- $d'(k)$: the diameter of cluster k

The Davies and Bouldin Index. Davies and Bouldin Index, I_{DB}, identifies clusters which are far from each other. It is defined by the average of cluster evaluation measures for all the clusters as described in Eq. 6. For a good clustering, the I_{DB} should be as low as possible.

$$I_{DB} = \frac{1}{n} \sum_{i=1}^{n} max_{i \neq j}(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}) \quad (6)$$

With:

- n: the number of clusters.
- c_i: the centroid of i^{th} cluster
- σ_i: the average distance between objects of cluster i and the centroid c_i
- $d(c_i, c_j)$: the distance between centroids c_i and c_j

6.2.2 The F-score Metric

We adopt the commonly used metric in information retrieval, F-score metric, defined in Eq. 7, to test the accuracy of the proposed method. It is based on precision and recall metrics defined successively in Eq. 8 and Eq. 9. Parameters used to compute the accuracy are as follows:

X = the set of users whom are actually trustworthy (Reputation value greater than 0.5);

Y = the set of users that the algorithm suggests to be trustworthy.

Table 1: The I_D and I_{DB} Cluster validity values for the *RepC* algorithm.

#Users	I_D	I_{DB}
100	0.25	0.52
500	0.25	0.53
1000	0.32	0.48
5000	0.38	0.45
10000	0.31	0.45
50000	0.31	0.56
100000	0.24	0.92
200000	0.23	1.23

$$F-score = \frac{2 \times (Pecision \times Recall)}{(Pecision + Recall)} \quad (7)$$

$$Precision = \frac{X \bigcap Y}{Y} \quad (8)$$

$$Recall = \frac{X \bigcap Y}{X} \quad (9)$$

The higher the recall and precision are, the more desirable the measures are for a good algorithm performance. Thus, we make use of F-score to indicate our algorithms performances. Obviously, high F-score values are desirable.

6.3 Performance Study

In order to assess the performance of our algorithm, we conduct different experiments. Firstly, we simulate the Twitter dataset with *RepC* and we run the programs computing the I_D and I_{DB} indexes by varying the number of users. Then, we compute the F-score measure for our algorithm *RepC* to find to which degree *RepC* provides more relevant results, and we compare it versus those obtained using the F-score for the existing algorithm *REMSA*.

As shown in Table 1, after 5000 users, as far as the number of users increases, the I_D decreases and the I_{DB} increases. This finding is due to a decrease in the minimum distance inter-cluster and an increase in the maximum diameter intra-cluster. This is caused by the rise in the number of clusters leading to the cluster's expansion (resp. an increase in a cluster diameter), and thus a higher degree of clusters overlap (resp. a decrease in the distance between clusters).

To compute the F-score values, we define, in Eq.10, the importance degree notion I_{Di}, presenting the actual reputation of a user i in Twitter. In fact, the more a user is reputable and important in the network, the higher the number of his shared (retweeted) tweets is.

$$I_{Di} = \frac{Tw_i}{Tw_m} \quad (10)$$

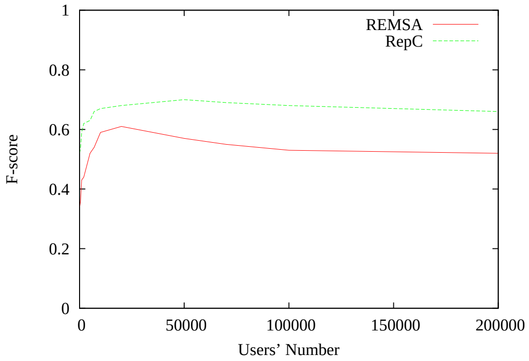

Figure 2: F-score results for *RepC* and *REMSA* with varying the number of users.

In Eq.10, Tw_i is the number of shared tweets of the user i and m is the user having the maximum number of shared tweets.

The simulation results, are shown in Figure 2.

The latter shows that *RepC* outperforms *REMSA* whatever the number of users.

The F-score values for *RepC* increase as far as the number of users increases. In fact, by increasing the number of users, direct and indirect trust values, considered to compute reputation values, increase leading to a rise in the authentication success trust rate. Then reputation values are more accurate.

However, the accuracy of *REMSA* reaches its best values for a number of users standing within the range [5000, 20000], decreasing again when the number of users exceeds 50000. In fact, considering only neighbours of a user to compute its reputation instead of the whole network, generates a decrease of precision and recall values and consequently a decline of the F-score. This is not the case of *RepC* which respects the objectivity property of reputation and keeps producing correct results with an increasing of the number of users.

7 CONCLUSION

In this paper, we have proposed a new reputation management algorithm *RepC* based on a trust network generated in previous works. Our algorithm classifies an OSN users into clusters by their trust similarity such that most trustworthy users belong to the same cluster. The evaluation results, based on the real social network Twitter, show that our algorithm can generate high quality results.

As forthcoming works, we plan to simulate a fuzzy version of *RepC* and compare between the hard and fuzzy methods.

REFERENCES

Chen, D., Chang, G., Sun, D., Li, J., Jia, J., and Wang, X. (2011). Trm-iot: A trust management model based on fuzzy reputation for internet of things. *Computer Science and Information Systems*, (20):1207–1228.

Christianson, B. and Harbison, W. S. (1996). Why isn't trust transitive? In *Proceedings of the Security Protocols Workshop, Cambridge, United Kingdom*, pages 171–176.

Davies, D. L. and Bouldin, D. W. (1979). A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1(2):224–227.

Day, W. H. E. (1992). *Complexity theory: An introduction for practitioners of classification*, chapter 6, pages 199–235. World Scientific Publishing.

Dunn, J. C. (1974). Well separated clusters and optimal fuzzy-partitions. *Journal of Cybernetics*, 4:95–104.

Hamdi, S., Gançarski, A. L., Bouzeghoub, A., and Yahia, S. B. (2016). Tison: Trust inference in trust-oriented social networks. *ACM Trans. Inf. Syst.*, 34(3):17.

Hamdi, S., Gançarski, A. L., Bouzeghoub, A., and BenYahia, S. (June 25-27, 2012). Iris: A novel method of direct trust computation for generating trusted social networks. In *Proceedings of the 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2012, Liverpool, United Kingdom*, pages 616–623.

Hartigan, J. A. and Wong, M. A. (1979). A K-means clustering algorithm. *Applied Statistics*, 28:100–108.

Heaney, M. T. (2014). Multiplex networks and interest group influence reputation: An exponential random graph model. *Social Networks*, 36:66–81.

Lee, J. and Oh, J. C. (2015). A node-centric reputation computation algorithm on online social networks. *Applications of Social Media & Social Network Analysis*, pages 1–22.

Portmann, E. K. (2012). The fora framework: A fuzzy grassroots ontology for online reputation management. *PhD thesis, Faculty of Sciences, University of Fribourg, Switzerland.*

Vendramin, L., Campello, R. J. G. B., and Hruschka, E. R. (2010). Relative clustering validity criteria: A comparative overview. *Stat. Anal. Data Min.*, 3(4):209–235.

Zhang, Y., Chen, H., and Wu, Z. (2006). A social network-based trust model for the semantic web. In *Proceedings of the Third International Conference on Autonomic and Trusted Computing*, ATC'06, pages 183–192, Berlin, Heidelberg. Springer-Verlag.