TrustVis: A Trust Visualisation Service for Online Communities

Sanat Kumar Bista, Payam Aghaei Pour, Nathalie Colineau, Surya Nepal, and Cecile Paris

Information Engineering Laboratory
CSIRO ICT Centre, Australia
Firstname.lastname@csiro.au

Abstract. Visualisation of social behaviour of members in online communities is a challenging issue. It provides holistic information on the behaviour of the community to the administrators/moderators and helps individual members in the community to monitor and analyse their own behaviour. This paper presents the design and implementation of a social trust visualisation service, called *TrustVis*, where the social trust is derived from the social behaviour of members in the community. One of the unique features of *TrustVis* is that it supports the faceted browsing and monitoring of members' social behaviour based on activities, contexts, time and roles. *TrustVis* is implemented and deployed in an online community we are currently trialling in collaboration with a government department to deliver support services to welfare recipients during their transition back to work. We describe the look and feel and the working of *TrustVis* in our production environment.

1 Introduction

Recently, the Web has created a space for people to conduct social activities like meeting each other, exchanging ideas, dating potential partners or sharing experiences. The emergence and popularity of many social network sites such as Facebook and LinkedIn have shown a phenomenal success of the Web in creating a social space for everyone. The term *Social Web* was coined by Rheingold to describe websites that support social interactions through web technologies [1]. Social networks can be classified into different categories based on the context of their application and use [2]. With respect to members' participation, social networks can be public (open) or private (by invitation only); with respect to applications, they can be either generic (no specific objective) or be targeted to realise some specific objectives. We have developed a specific member-only online community for a government agency which aims to deliver support services to welfare recipients during their transition back to work [3,4].

Social networks are quite complex. It is thus difficult to have a holistic view of such networks as well as monitor and analyse them to extract meaningful information. Visualisation tools have been widely promoted to monitor social networks [5-8]. The concept of network visualisation is not new. It has been used to monitor and diagnose

A. Ghose et al. (Eds.): ICSOC 2012, LNCS 7759, pp. 191–202, 2013.

computer networks [9-11]. In recent time, similar concepts have been used to analyse and monitor social networks. Examples of such tools include Social Network Visualiser (*SocNetV*)[1], *NetVis*[2] and *Last Forward*[3].

The design of social network visualisation tools can be categorised into two broad groups: analysis-centric design and application-centric design. Analysis-centric designs focus more on identifying degree and density of nodes in the network and are more or less generic in nature (or at least their theoretical models are generic). *Pajek* [12], *Ucinet* [13] and *Tnet* [14] are examples of analysis-centric visualisation tools. These tools offer a way of measuring network properties, such as cohesiveness, degree centrality and friendships. Application-centric designs are driven by the specific needs of the application. The underlying analysis could still be borrowed from standard network analysis theory, but the visualisation is modelled by considering the requirements of the problem scenario. Examples of application-centric designs include Communication and Activity Visualisation for the Enterprise (CAVALIER) [5], student interaction visualisation in online communications [8] and editing behaviour visualisation of Wikipedia editors [15]. From these observations, we come to the following conclusions: visualisation tools are designed to analyse specific properties of the networks. Hence, they are either unsuitable or cannot be used directly to analyse a new property that is not supported (such as social trust); and visualisation tools are designed with a specific application in mind and are not generic enough to be migrated to new applications, as each application brings its own challenge and features (such as an online community for welfare recipients).

We have implemented and deployed an online community to support parents receiving welfare payments so that they can meet parents in similar situation. The purpose of the online community is to provide both emotional and informational support to them during the period of their transition from welfare to work. One of the key research questions we are considering is how to build *social capital* in the community through *social trust*. In our research, the social capital is the density of interactions that benefit the community, and social trust represents the positive interactions between members in the community. Since our application domain has a specific property (i.e., social trust), it has different visualisation needs than those encountered in other online communities. For this reason, we cannot simply adapt an existing visualisation tool and deploy it. Furthermore, none of the existing social network visualisation tools support the visualisation of contextualised social behaviour and social trust. Hence, there is a need to build a visualisation tool.

Visualisation of social trust in our community is complex and multi-dimensional. In addition to aesthetic and layout aspects, the visualisation tool must support a number of different functionalities. First, it has to allow one to visualise the network's interactions in different contexts (e.g., Discussion Forum and Livechat). We define a context as the environment in which an interaction takes place. Second, it has to allow one to visualise the network based on different user activities such as rating,

[1] `http://socnetv.sourceforge.net/`

[2] `http://www.netvis.org/index.php`

[3] `http://lastforward.sourceforge.net/`

commenting or inviting others to be friends. Third, it has to support temporal filtering so that one can visualise the network at different time intervals. Fourth, it has to support the exploration of network so that one can find information about nodes and edges including social trust of specific individuals by navigating though the network without much difficulty. Finally, the visualisation should incorporate role-based access to information to preserve privacy and offer different views to people with different roles so that members and administrators can use the same, but have different views of the network. In order to incorporate these requirements, we have developed and implemented a novel social trust visualisation tool, called *TrustVis*. It supports faceted browsing and monitoring of social behaviours of the members in the community and has been implemented as a service.

The rest of the article is organised as follows: Section 2 presents the context of the work in terms of the application and network property in relation to social trust. Application specific requirements are explained in Section 3. Section 4 illustrates the design and implementation of *TrustVis*. Section 5 presents a brief review of related work. The final section presents the concluding remarks and some possible future work.

2 Context

We define the context of our work on *TrustVis* along two dimentions. First, we outline our application scenario, and then, we provide specific information on social trust, a new network property that needs to be supported by *TrustVis*.

2.1 Application

We have built and deployed an online community to deliver government services to citizens as a trial for 12 months [3,4]. In this community, membership is by invitation only, i.e., specific individuals (individuals receiving a specific type of welfare payments and required to look for work) are invited to join the community. This group is in a transition phase, being asked (by legislation) to move from one type of welfare payment to another. The identity of the community members is kept anonymous (i.e., members present themselves with an avatar and a name of their choice). Members have also an individual profile, through which they can choose to disclose what they want to others.

The aim of the community is several fold. First, it is a place for the government to target its information and services when dealing with a specific target group of welfare recipients. Second, it is to bring people with the same concerns together, hoping that they will share experiences, ideas and tips, thus providing social, emotional and moral support to each other. Although they are all strangers to each other, they all share the same situation and concerns. Third, it is a space in which we invite individuals to go on a reflection journey, in order to better prepare them for the transition and their return to work.

2.2 STrust: Social Trust Model

TrustVis is designed to help explore the social behaviour of the community members. It relies on an underlying social trust model, *STrust*[16], that computes trust based on interactions between community members. *STrust* has three unique features. First, it distinguishes three trust types: 1) the popularity trust (*PopTrust*) that captures the trust that an individual member has received from other members in the community; 2) the engagement trust (*EngTrust*), which reflects the trust that an individual member has about other members in the community; and, 3) the social trust (*STrust*), that combines the popularity and engagement trust. The second feature of the *STrust* model is that it considers both active and passive behaviour of members. Active behaviour refers to actions that generate: (a) content for other members in the community to consume (e.g., Contributions to Forum, Livechat, etc.), and (b) actions that require other members to act (e.g., invite somebody to be a friend). Passive behaviour refers to actions that do not generate any content or actions for other members (e.g., visiting the community, reading posts, reading Livechat content, etc.). The third feature of the model is that it considers online communities as two mode social networks, where nodes in the networks can be classified into two types: *active* and *passive*. Active nodes are those that can engage in the community (typically, people), while passive nodes are those that cannot engage in the community (i.e., they do not have engagement trust), such as articles and posts. Our current implementation of *TrustVis* considers only active nodes. Interactions with passive nodes are treated like active node interactions by removing the intermediate passive node and linking directly the members interacting with the passive node (the activity is then grouped as *Same Interest*).

We now describe the computational aspect of *STrust*. Let M be the total number of members in the online community. Let m_i and m_j represent the members of the community. If a member m_j has a positive interaction with a member m_i, the interaction is represented as "+". Similarly, the negative interaction is represented as "-". The popularity trust (*PopTrust*) of a member m_i is then defined as:

$$PopTrust(m_i) = \frac{\sum_{j=1, j\neq i}^{M} \frac{|PT_{ij}^{kd+}|+1}{|PT_{ij}^{kd+}|+|PT_{ij}^{kd-}|+2}}{M-1}$$

Where PT_{ij}^{kd+} and PT_{ij}^{kd-} represents the positive and negative popularity interaction a member m_i has with a member m_j with respect to an activity d in the context k. Similarly, the engagement trust (*EngTrust*) of a member m_i is defined as:

$$EngTrust(m_i) = \frac{\sum_{j=1, j\neq i}^{M} \frac{|ET_{ij}^{kd+}|+1}{|ET_{ij}^{kd+}|+|ET_{ij}^{kd-}|+2}}{M-1}$$

with ET_{ij}^{kd+} and ET_{ij}^{kd-} representing the positive and negative engagement interactions between members m_i and m_j respectively.

A member in the community may be involved in a number of activities related to a single context. For example, a member may comment, rate and/or view a post in the forum. Here, the forum represents the *context*, and commenting, rating and viewing are considered as *activities*. It is possible for each activity and context to have different weights. w_d represents the weight for activity *d* and $\sum w_d = 1$. w_k represents the weight for context *k* and $\sum w_k = 1$. We need to consider this while evaluating positive and negative engagement and popularity interactions. Thus, the weights for positive and negative popularity and engagement interactions are defined as follows:

$$For\,|\,PT_{ij}^{kd+}\,|\,and\,|\,ET_{ij}^{kd+}\,| \qquad\qquad |\,For\,|\,PT_{ij}^{kd-}\,|\,and\,|\,ET_{ij}^{kd-}\,|$$

$$\sum_{k=1}^{K} w_k\left(\sum_{d=1}^{D} w_d\left(\sum_{x=1}^{X} +1\right)\right) \qquad\qquad \sum_{k=1}^{K} w_k\left(\sum_{d=1}^{D} w_d\left(\sum_{x=1}^{X} -1\right)\right)$$

Here, K represents the number of contexts, D the number of activities in each context, and X represents the number of interactions related to the activity and context. The social trust (*STrust*) of an individual member m_i in the community is then given by:

$$SocialTrust(m_i) = \alpha.PopTrust(m_i) + (1-\alpha).EngTrust(m_i)$$

Where α represents the value of a weight in the range of 0 to 1. If alpha is 1, the social trust of an individual indicates how much other members in the community trust him or her. For further details on trust model, we refer the readers to our earlier work in [17,16].

3 Social Trust Visualisation Requirements

In the introduction, we have listed several unique requirements of our application. In the following, those requirements are explained with a few examples to provide a better understanding of the requirements for *TrustVis*.

Context Filter: A requirement for *TrustVis* is to be able to offer a contextual filter for the community members' social behaviour. A context in our model refers to a setting where specific interactions take place. We have defined a range of contexts. This includes the forum (where members post, rate and comment), the resource section (where members can read and rate information), the activity pages (where members work on some guided tasks such as identifying skills, writing resume, identifying barriers to work and studies, etc), the buddy program (where members socialise with each other sending and accepting invitations), the media page (where members can watch videos and listen to audio resources) and the Livechat room (where members can have live discussions with experts).

Activity Filter: Each of the contexts outlined above can have multiple activities associated to them, such as rating, commenting, viewing, etc. Therefore it is important for *TrustVis* to be able to filter the social behaviours of the members on the basis of such activities.

Temporal Filter: Online communities such as ours evolve over time. Thus, being able to filter social behaviour on the basis of time offers an interesting analysis of the network. By including a temporal filter, *TrustVis* presents the visualisation of selected activities in particular contexts for specific dates or periods of time.

Information: *TrustVis* displays the computations from the social trust model *STrust*. Since *STrust* as well as the context and activity filters are unique to our application, there is a requirement for the visual representation to be able to capture right information and present them at right places. This includes displaying of trust and all other interactions of members at appropriate locations.

Aesthetic and Layouts: An important aspect of any visualisation is aesthetic of the representation of different information. *TrustVis* supports multiple layouts and is easy to use. For example, *TrustVis* offers avatar-based view of members with drag-able nodes supporting multiple layouts. Related information is provided at designated space as well as on mouse over.

Roles: *TrustVis* is required to support role-based views for both community administrators/moderators and individual members. We provide a holistic view of the community (called the *System View*) to the administrators/moderators (referred later as systems users) whereas individual members are only able to access information about their social behaviour (called the *User View*).

Based on these requirements, we designed and implemented the trust visualisation service, *TrustVis*, the details of which are outlined in the following section.

4 Service Design and Implementation

Fig. 1 shows a high level architecture focusing on *TrustVis* components.

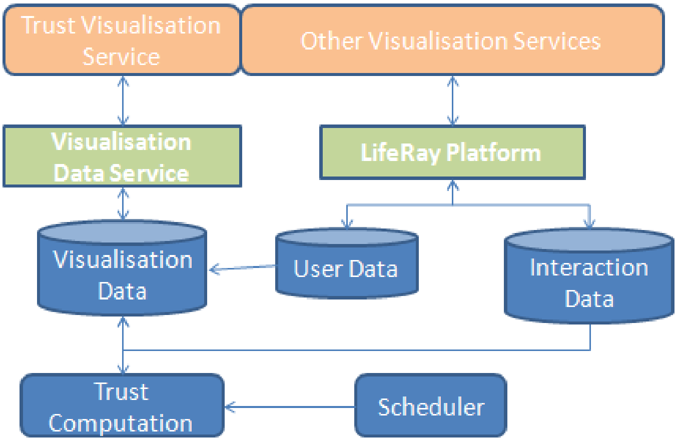
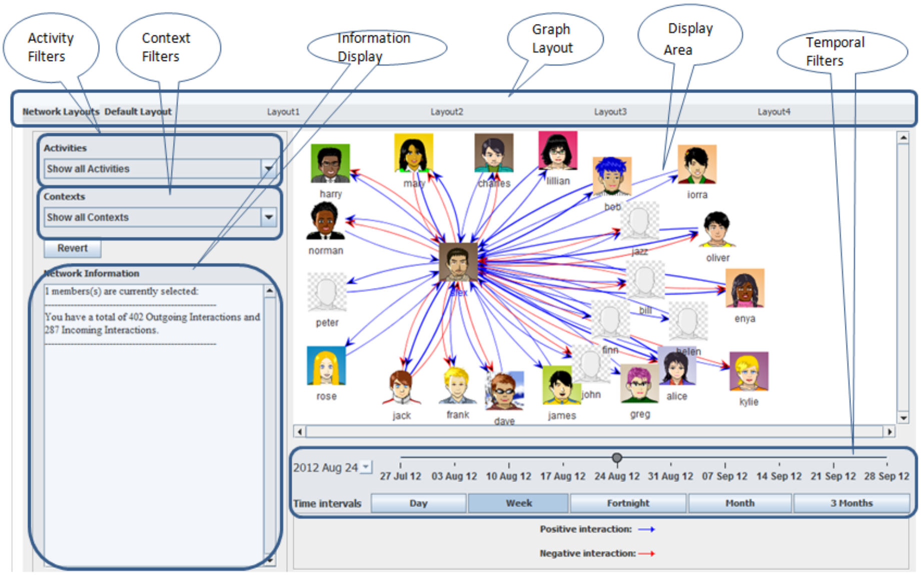


Fig. 1. A High-Level System Architecture Showing the Trust Visualisation Components

User data consist of the data relevant to users (that includes both community members and system users) such as their screen name and profile. This includes user roles. We currently distinguish two user roles: a system user or a community member. As outlined in the requirements, this distinction is necessary as each of these types of users need to see different aspects of the visualisation and can take part in different activities and contexts. Interaction data consist of all active and passive interactions of users. This includes all possible activities in all possible contexts. We extract relevant data from interaction data and user data and create a visualisation data. The trust computation component implements the STrust model described earlier and computes the different types of trust values. The purpose of the scheduler is to schedule the trust computation. In our current implementation, the scheduler computes trust values once a day. This constraint was put to increase the efficiency of the visualisation service. Trust computation is time and resource demanding and this aspects should not limit the visualisation aspect.We plan to improve this aspect in future by optimising the trust algorithms.

TrustVis has been deployed as an applet in the community, and GraphML has been used as a file format for graphs. Users of the system interact with this service to get the visual output to query the system for information. The *Visualisation Data Service,* as an intermediate layer, captures user requests to query required information from the underlying visualisation data (refer Figure 1). In addition to serving the visualisation data, *user data* and *interaction data* also provide necessary information to other elements of the online community, such as the online community user interface. The online community has been developed using the Liferay[4] platform, and TrustVis is developed using JUNG[5].

Fig. 2 presents the user interface of TrustVis for community members. In this view, the display area presents the network representing the member's social behaviour.



1 members(s) are currently selected:

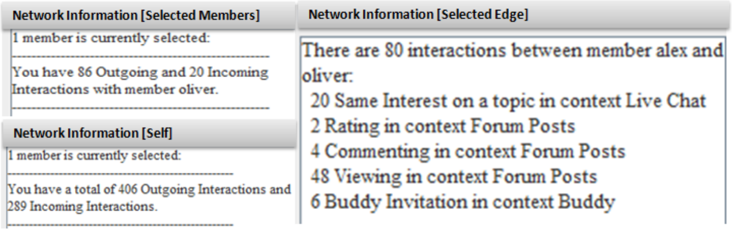
You have a total of 402 Outgoing Interactions and 287 Incoming Interactions.

Fig. 2. *TrustVis* User Interface

[4] http://www.liferay.com/

[5] http://jung.sourceforge.net/

There are limitations in the amount of information available in this view. Specific trust values are not available to community members. Members are presented with high level information such as their total number of incoming and outgoing interactions with other community members in different contexts. Fig. 3 presents samples of member and link information available to community members. Activity, Context and Temporal filters are enabled in this view.


Network Information [Selected Members]

1 member is currently selected:

You have 86 Outgoing and 20 Incoming Interactions with member oliver.

Network Information [Self]

1 member is currently selected:

You have a total of 406 Outgoing Interactions and 289 Incoming Interactions.

Network Information [Selected Edge]

There are 80 interactions between member alex and oliver:
20 Same Interest on a topic in context Live Chat
2 Rating in context Forum Posts
4 Commenting in context Forum Posts
48 Viewing in context Forum Posts
6 Buddy Invitation in context Buddy

Fig. 3. Network Information in User View

Fig. 4 shows an expanded view of the activity and context filters (there are a total of 8 activities and 8 contexts, and not all activities are applicable to all contexts). Users can select any of the members or the links between them to see the details of the interactions. Members in the user view are represented by their avatars and the network can be resized by dragging the avatars. Negative interactions such as negative ratings and cancellations of invitations appear as red links while all other positive interactions appear as blue. The thickness of the links is proportional to their values. The greater the value the thicker the link appears in the graph.

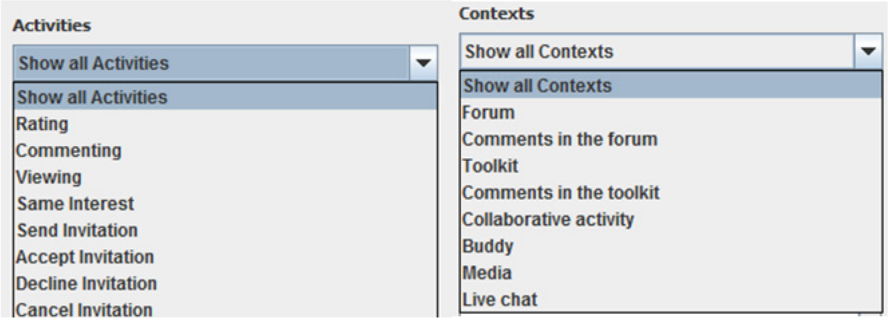
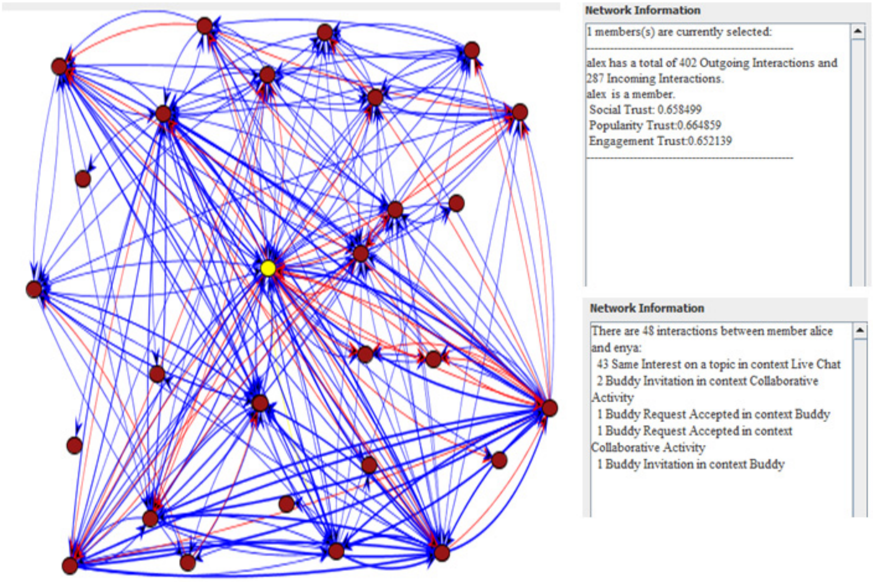


Fig. 4. Expanded view of the Activity and Context Filter

In addition to what is available in the user view, the system view presents more details about members. This enables administrators/moderators to monitor the community more effectively. This view consists of the whole network with the capability to click on individual members and check their exclusive network. Fig. 5 shows a sample network display and the information available on members and their links, including trust values calculated using the STrust model.

System users are also able to select nodes and remove them from the visualisation to see the effect of an individual member on the whole community. In addition, they



```
Network Information
1 members(s) are currently selected:
-----------------------------------------------
alex has a total of 402 Outgoing Interactions and
287 Incoming Interactions.
alex  is a member.
 Social Trust: 0.658499
 Popularity Trust:0.664859
 Engagement Trust:0.652139
-----------------------------------------------
```

```
Network Information
There are 48 interactions between member alice
and enya:
  43 Same Interest on a topic in context Live Chat
  2 Buddy Invitation in context Collaborative
Activity
  1 Buddy Request Accepted in context Buddy
  1 Buddy Request Accepted in context
Collaborative Activity
  1 Buddy Invitation in context Buddy
```

Fig. 5. System View of the Network and Related Information

can check the nodes that have not had any interaction with other members in the community yet.

5 Related Work

Visualisation and measurement as central factors to the growth of social network analysis offer investigators new insights of network structure [6]. Node-link (points and lines) and matrix based representations are the two distinct form of display used in visualisation of social networks since the early 1930s [7] [6]. To offer better visualisation and analysis capabilities, hybrid approaches have also been used in recent time. For example, NodeTrix [18] has been developed to use node-link diagrams to show the global network structure, and adjacency matrix to support the analysis of local communities [19]. Analysis of social networks can provide useful information about its actors, roles and positions of actors in the community to name a few [20] [18]. Pajek [12], Ucinet [13] and Tnet [14] support node-link based and adjacency matrix based social network analysis.

Recently, online social networks have gained tremendous popularity leading to an explosive growth in the degree and density of interactions over social networks. While the underlying theory of analysis could still be supported by established principles of network analysis, there is clearly a need to identify better visualisation techniques as required by the context of application. Vizster [7] has been developed over the node-link layout, but with several customisations to better support visual interconnectivity between graphs and with features to automatically identify and visualise community structures. The resulting tool was used to visualise 1.5 million Friendster crawl. Xiong and Donath [21] argue that traditional social network analysis based visualisation, being fairly complex, are more useful to analysts than for an average user who just intends to get a sense of connectedness in the community. They have proposed an interesting visualisation, called *data portrait*, that uses a flower metaphor for individual portraits and garden metaphor for combining the portraits to represent an online interaction environment. Viegas and Donath [22] propose an alternative approach for visualisation of social networks representing the frequency of connections over time (referred to as PostHistory). In contrast to graph visualisations that demonstrate the strength of connections between members in a social network, this visualisation depicts the frequency of connections between the members. Their work underlines that current depictions of social networks have shortcomings and that there is a need to find alternate ways to visualise online social networks.

Visualisation of trust over online communications is also an interesting area of research. O'Donovan et al [23,24] propose a model that extracts negative information from the feedback comments on eBay, computes personalised and feature-based trust and presents this information graphically. The graph shows the trust value and the trust strength calculated based on the number of transactions/comments between two users. In [25], the authors propose a trust-based visualisation of cooperation context between members. Bimrah et al. [26] propose a visualisation language for trust related requirements elicitation.

In line with this body of work, we have also used the traditional node-link representation in TrustVis to visualise the social behaviour of the members in our online community. However, other requirements of TrustVis have originated from the application's needs and the social trust model. These requirements provide us with an opportunity to invent a unique way of visualising social trust and behaviours based on three filters: context, activities and time.

6 Conclusion and Future Works

In this paper, we described the design and implementation of a trust visualisation service, called TrustVis. We deployed TrustVis in an online community that we developed in collaboration with a government agency with the aim of providing informational and emotional support services to welfare recipients transitioning back to work. Due to the inherent requirements of our online community, existing methods of visualisation were either too complex or not sufficient enough to capture the social trust and behaviour requirements we needed. This motivated the development of TrustVis. The key feature of TrustVis is that it enables users to visualise social trust and behaviours using three types of filters: context, activities and time. These three features together offer a unique experience to administrators/moderators and to individual community members. In the future, we intend to extend TrustVis in the following directions: (a) we plan to include both active and passive nodes in TrustVis by providing a functionality to convert a three-mode network to a two-mode network; (b) we plan to provide the provenance of trust by providing the functionality to drill down from the interaction level to the level of activities and contexts; (c) we plan to extend TrustVis to a full-fledged faceted monitoring service including other activities and contexts such as individual activities like login; and (d) we plan to conduct a usability evaluation of TrustVis.

Acknowledgements. This research has been funded under the Human Services Delivery Research Alliance (HSDRA) between the CSIRO and the Australian Government Department of Human Services. We would like to thank Payam Aghaei Pour, Hon Hwang, Brian Jin, Alex Sun and Bo Yan for their contribution to the implementation of this work.

References

1. Rheingold, H.: The Virtual Community: Homesteading on the Electronic Frontier (Revised Edition). MIT Press, USA (2000)
2. Ellison, N.B., Steinfield, C., Lampe, C.: The Benefits of Facebook "Friends:" Social Capital and College Students' Use of Online Social Network Sites. Journal of Computer-Mediated Communication 12(4), 1143–1168 (2007), doi:10.1111/j.1083-6101.2007.00367.x
3. Colineau, N., Paris, C., Dennett, A.: Exploring the Use of an Online Community in Welfare Transition Programs. In: 25th BCS Conference on Human-Computer Interaction, Newcastle-upon-Tyne, United Kingdom, pp. 455–460. British Computer Society (2011)
4. Colineau, N., Paris, C., Dennett, A.: Capitalising on the Potential of Online Communities to Help Welfare Recipients. International Reports on Socio-Informatics (IRSI) 2, 59–65 (2011)
5. Dekker, A.: Visualisation of social networks using CAVALIER. In: Proceedings of the 2001 Asia-Pacific Symposium on Information Visualisation, Sydney, Australia, vol. 9, pp. 49–55. Australian Computer Society, Inc. (2001)
6. Freeman, L.C.: Visualizing Social Networks. Journal of Social Structure 1(1) (2000)
7. Heer, J., Boyd, D.: Vizster: visualizing online social networks. In: IEEE Symposium on Information Visualization, INFOVIS 2005, USA, October 23-25, pp. 32–39 (2005), doi:10.1109/infvis.2005.1532126
8. Jyothi, S., McAvinia, C., Keating, J.: A visualisation tool to aid exploration of students' interactions in asynchronous online communication. Computers & Education 58(1), 30–42 (2012), doi:10.1016/j.compedu.2011.08.026
9. Girardin, L.: An eye on network intruder-administrator shootouts. In: Proceedings of the 1st Workshop on Intrusion Detection and Network Monitoring, Santa Clara, California, vol. 1. USENIX Association (1999)
10. Ball, R., Fink, G.A., North, C.: Home-centric visualization of network traffic for security administration. In: Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security, Washington DC, USA, pp. 55–64. ACM (2004), doi:10.1145/1029208.1029217
11. Erbacher, R.F.: Intrusion behavior detection through visualization. In: IEEE Systems, Man and Cybernetics Conference, Cristal City, Virginia, USA, October 5-8, pp. 2507–2513. IEEE (2003), doi:10.1109/icsmc.2003.1244260
12. de Nooy, W., Mrvar, A., Batagelj, V.: Exploratory Social Network Analysis with Pajek. Cambridge University Press, USA (2011)

13. Borgatti, S.P.: Ucinet for Windows: Software for Social Network Analysis. Analytic Technologies, Harvard (2002)
14. Opsahl, T.: Structure and Evolution of Weighted Networks. University of London (Queen Mary College), London (2009)
15. Iba, T., Nemoto, K., Peters, B., Gloor, P.A.: Analyzing the Creative Editing Behavior of Wikipedia Editors: Through Dynamic Social Network Analysis. Procedings of Social and Behavioral Sciences 2(4), 6441–6456 (2010), doi:10.1016/j.sbspro.2010.04.054
16. Nepal, S., Sherchan, W., Paris, C.: Building Trust Communities Using Social Trust. In: Ardissono, L., Kuflik, T. (eds.) UMAP 2011 Workshops. LNCS, vol. 7138, pp. 243–255. Springer, Heidelberg (2012)
17. Nepal, S., Paris, C., Bista, S.K., Sherchan, W.: A Trust Model Based Analysis of Social Networks. International Journal of Trust Management in Computing and Communications (2012) (forthcoming)
18. Henry, N., Fekete, J.D., McGuffin, M.J.: NodeTrix: a Hybrid Visualization of Social Networks. IEEE Transactions on Visualization and Computer Graphics 13(6), 1302–1309 (2007), doi:10.1109/tvcg.2007.70582
19. Henry, N., Fekete, J.D.: MatrixExplorer: a Dual-Representation System to Explore Social Networks. IEEE Transactions on Visualization and Computer Graphics 12(5), 677–684 (2006), doi:10.1109/tvcg.2006.160
20. Scott, J.: Social Network Analysis: A Handbook. Sage Publications, London (1987)
21. Xiong, R., Donath, J.: PeopleGarden: creating data portraits for users. In: Proceedings of the 12th Annual ACM Symposium on User Interface Software and Technology, Asheville, North Carolina, USA, pp. 37–44. ACM (1999), doi:10.1145/320719.322581
22. Viégas, F.B., Donath, J.: Social network visualization: can we go beyond the graph. In: Workshop on Social Networks for Design and Analysis: Using Network Information, Chicago, USA (2004)
23. O'Donovan, J.: Capturing Trust in Social Web Applications Computing with Social Trust. In: Golbeck, J. (ed.) Human–Computer Interaction Series, pp. 213–257. Springer, London (2009), doi:10.1007/978-1-84800-356-9_9
24. O'Donovan, J., Smyth, B., Evrim, V., McLeod, D.: Extracting and visualizing trust relationships from online auction feedback comments. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence, Hyderabad, India, pp. 2826–2831. Morgan Kaufmann Publishers Inc. (2007)
25. Guerriero, A., Kubicki, S., Halin, G.: Trust-Oriented Multi-visualization of Cooperation Context. In: Proceedings of the Second International Conference in Visualisation, pp. 96–101. IEEE Computer Society, Washington, DC (2009), doi:10.1109/viz.2009.48
26. Bimrah, K.K., Mouratidis, H., Preston, D.: Modelling Trust Requirements by Means of a Visualization Language. In: Proceedings of the 2008 Requirements Engineering Visualization, Barcelona, Spain, pp. 26–30. IEEE Computer Society (2008), doi:10.1109/rev.2008.3