Engineering and Physical Sciences Research Council

Interpretation of Crowdsourced Activities Using Provenance Network Analysis

Trung Dong Huynh[1], Mark Ebden[2], Matteo Venanzi[1], Sarvapali Ramchurn[1], Stephen Roberts[2], and Luc Moreau[1]

(1) University of Southampton - (2) University of Oxford

Overview

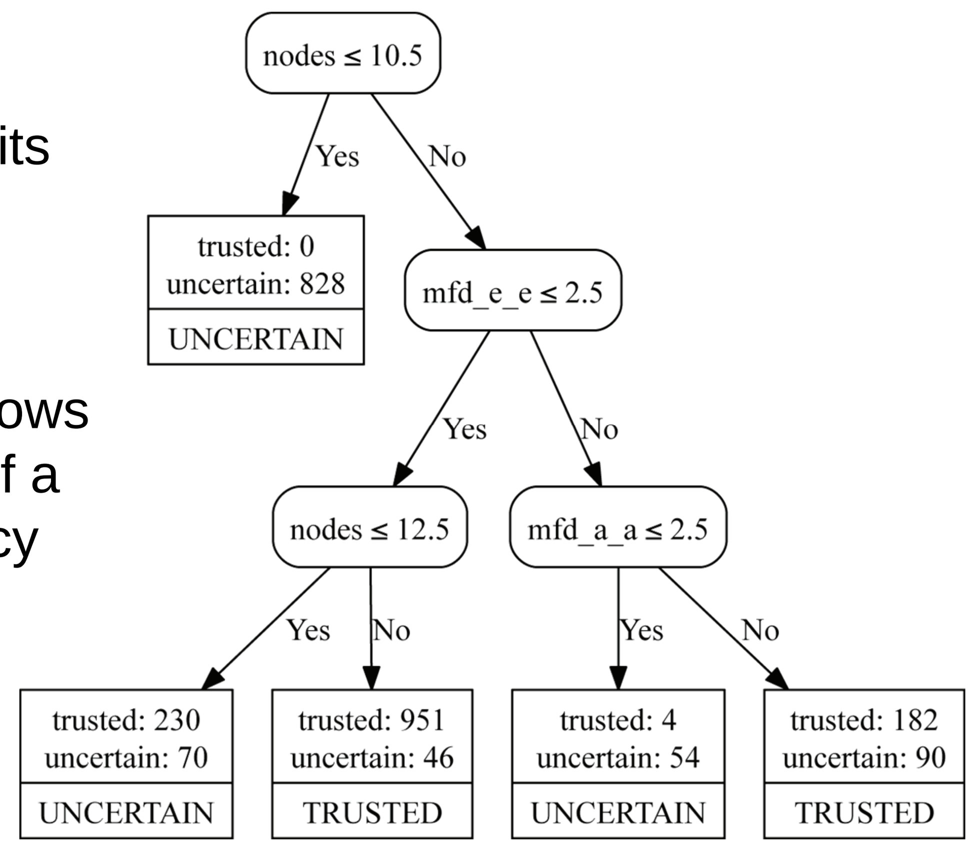
- Analytical study on various network measures of over 5,000 provenance graph from the crowdsourcing application CollabMap.
- Classifying the trustworthiness of crowdsourced data using the network metrics of their *dependency* provenance graphs.
- A novel methodology for analysing properties of crowd-generated data using provenance graphs.
- Results: over 95% accuracy in assigning trust categories to CollabMap's buildings, evacuation routes, and route sets.

Provenance Network Analytics

- The *dependency graph* $D_{G,a}$ of an entity a is a provenance graph containing only the nodes that were directly or indirectly influenced by it:

$$D_{G,a} = (V_{G,a}, E_{G,a}), \text{ where } V_{G,a} = \{v \in V : v \rightarrow^{*} a\} \text{ and}$$
$$E_{G,a} = \left(e \in E : \exists v_s, v_t \in V_{G,a} \cdot e = (v_s, v_t)\right)$$

- Correlate the network metrics of an entity with its properties, such as its quality, using machine learning techniques.
- A high correlation will allows predicting the property of a node from its dependency graph's network metrics.

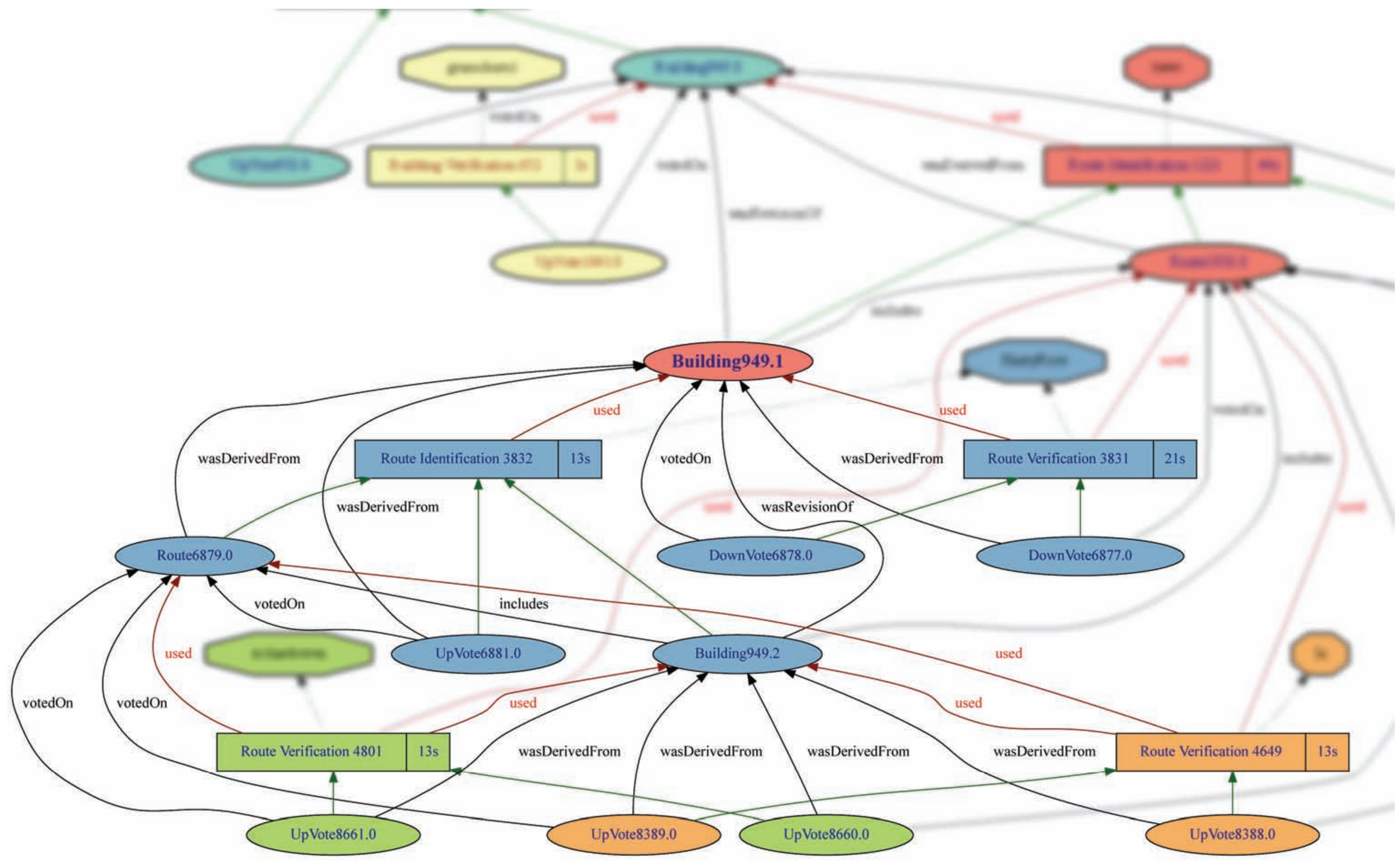


Example: The decision tree for predicting trust labels of routes (the depth of the tree was limited to 3)

CollabMap Provenance Graphs

Crowdsourcing the identification of buildings and evacuation routes

- City-wide mapping of buildings and evacuation routes for disaster recovery simulations
- Results cross-checked by human users and automatically verified using Ordnance Survey maps
- Provenance recorded for auditing data quality



The dependency graph of the node `Building949.1` (top of the graph). The blurred-out nodes and edges belong to the full provenance graph of a task, but are not included in the building's dependency graph.

CollabMap Trust Classification

- Method
  - Buildings, routes, and route sets generated in CollabMap were given trust labels *trusted* or *uncertain* as calculated from their user votes.
  - The data were randomly divided into training sets and test sets.
  - Decision tree classifiers were trained on test sets to predict the trust labels of the buildings, routes, and route sets, taking their dependency graphs' network metrics as input features.
  - The sensitivity, specificity, and accuracy of the classifiers were assessed on the relevant test sets.
- Results

Local Deployment
High classification accuracy over buildings routes, and route sets (over 95%).

	Sensitivity	Specificity	Accuracy
Building	96.61%	99.17%	97.00%
Route	94.78%	97.32%	95.28%
Route Set	97.23%	97.78%	97.77%

AMT Deployment
First table: The classifier trained with the Local Deployment tested against data generated by workers on Amazon Mechanical Turk platform.
Building classification suffered from the higher proportion of inaccurate buildings (21.5% vs 1.5%).

Second table: Performance of classifiers retrained with AMT data.

	Sensitivity	Specificity	Accuracy
Building	72.43%	50.19%	77.23%
Route	99.78%	93.08%	96.48%
Route Set	100%	90.53%	95.05%

	Sensitivity	Specificity	Accuracy
Building	96.61%	99.17%	97.00%
Route	94.78%	97.32%	95.28%
Route Set	97.23%	97.78%	97.77%

Future Work

- Crowd Behavioural Change

  Reflected in changes in the relevance of each network metric in predicting the trust labels of buildings, routes, and route sets.

Local Deployment

Network Metric	Building	Route	Route Set
Number of nodes	0.087	0.704	0.502
Number of edges	0.900	0.193	0.190
Graph diameter	0.012	0.025	0.308
MFD (entity → entity)	0.001	0.067	–
MFD (entity → activity)	–	0.006	–
MFD (activity → activity)	–	0.005	–

AMT Deployment

Network Metric	Building	Route	Route Set
Number of nodes	0.474	0.893	0.230
Number of edges	0.505	0.020	0.770
Graph diameter	0.021	0.046	–
MFD (entity → entity)	–	0.006	–
MFD (entity → activity)	–	0.035	–
MFD (activity → activity)	–	–	–

- Validating the method in new application domains.
- Extending the analytics to include provenance network metrics that characterise the evolution of provenance graphs.
- Incorporating generic node attributes (e.g. the value of votes).
- Applying graph analytics methods to identify key agents (e.g. users), activities, data in a task or a deployment.

BAE SYSTEMS · ACFR Australian Centre for Field Robotics · SECURE · University of Southampton