Visualization of Complex Relations in E-Government Knowledge Taxonomies

Per Myrseth, Jørgen Stang, David Skogan
Det Norske Veritas
{per.myrseth, jorgen.stang, david.skogan}@dnv.com

Abstract

The successful collaboration and interoperability between fully and partially related E-government subject domains requires well understood and high quality definitions of terms and a unified view of the relationships between the defined terms. The common terms and corresponding relation are defined in knowledge taxonomies (or even ontologies) and several good tools exist to create and maintain these models for the appropriate sub domains. The engineering process is carried out in a multi-user environment including remote workers editing the taxonomy. However, the sheer complexity and size of the full models dictates more powerful and dedicated visualization tools to graphically inspect, assess and diagnose the full taxonomies. This article describes a case where a social network analysis (SNA) tool is used as a part of a regime for the quality assurance of a knowledge taxonomy for e-government interoperability. In addition to the visual aids provided by the SNA tool, some comments are also made as to the applicability of SNA centrality metrics to knowledge taxonomies.

Keywords --- **Visualization, knowledge taxonomies, social network analysis, ontology, e-government, metadata, data quality**.

1. Introduction

In order to make governmental data collection and public interfaces more effective and less error prone, the Norwegian government has over the past few years made a substantial effort to harmonize and build common knowledge taxonomies across governmental departments. The ultimate goal of this initiative is manifold; (i) avoid duplication and inconsistencies when collecting data from the public by eliminating inherent data entry redundancy (asking for the same information in multiple data entry (web) forms), (ii) enable pre-filled forms only requiring user confirmation, (iii) reuse collected information for multiple regulations, and (iv) to ensure governmental rules are applied consistently within and across departments.

The quality of the resulting knowledge taxonomy will be determent to the users trust and the general usefulness of the common model. The syntactical data quality is maintained by continuously measuring well defined modeling rules and by defining responsibility matrices and feedback loops to the appropriate data modeler[1]. At the same time, there is a need for efficient tools and process to utilize existing models to identify and learn from both best practices, mal practices and inconsistencies in the existing terms and taxonomies. Hence, leveraging an educational process as the knowledge taxonomy is actively in progress. This in turn requires effective visualization of the taxonomies which goes well beyond what is provided by the OWL or UML modeling tools normally used for defining the models.

By considering each term as an actor (node) and the relations between them (the taxonomy) as inter actor communication (edges), visualization and analysis tools frequently employed for social network analysis can be applied to both birds eye views of large scale taxonomies as well as provide useful drilldowns for diagnosis and pattern matching at a finer granularity. Identified patterns could include overlaps, inconsistencies, dangling (unresolved) entities and wanted or unwanted clusters. Also, the layered model design approach used for the governmental knowledge taxonomies described here could well lend itself to 3D visualizations by orthogonally offsetting each layer and so emphasizing on the layer connectivity [2].

In addition to the pure visual and educationalist aspects of applying SNA tools to knowledge taxonomies, we also attempt to relate several centrality metrics [3] to the taxonomy and determine if these metrics can provide useful characteristics of the taxonomy, in particular we consider; (i) inbetweenes centrality, (ii) degree centrality and (iii) closeness centrality. These SNA metrics are tested and evaluated as diagnostic metrics for a set of quality patterns we describe in a multi-user taxonomy / ontology engineering environment.

2. E-governmental metadata framework

The Norwegian Semantic Repository of Electronic Services (SERES) e-governmental metadata framework is designed to provide an effective means of connecting data submitted by the public (paper based or web forms based) to the governing rules and regulations. Also, the framework will support interoperability between

departments to prevent data inconsistencies and duplication.

2.1. Architecture

The metadata framework architecture comprises three distinct entity layers; (1) *implementation*, (2) *structure* and (3) *semantics* [1]. Each layer contains entities with a set of properties and relations to other entities. The implementation layer defines the entities as they are entered by the user, the structure layer defines aggregated (related) types that can be reused by several implementation entities, and the semantics layer defines the terms that are being used by the departmental subject matter experts. At the moment, the collected metadata can be termed a knowledge taxonomy, however, as the model evolves and become more mature it is intended to be extended to an ontology which can be used for inference engines. Figure 1 illustrates how the model can be used to define a taxpayer in the current implementation.

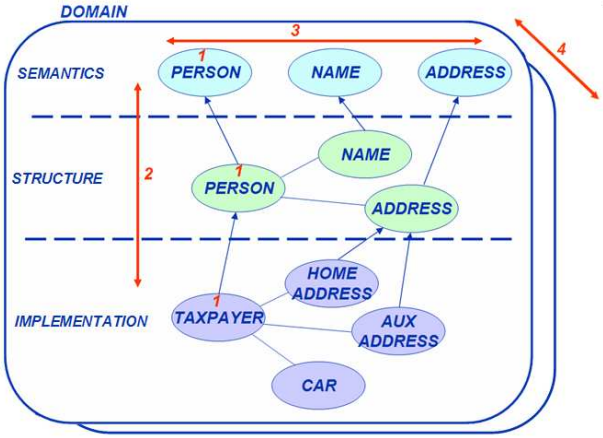

Figure 1 – Metadata example

The numbers 1-4 indicate levels of quality assurance, (1) by entity, (2) between entities, (3) across terms in the semantic layer and (4) between domains. The visualization activity described here mainly supports (2) and (3) and to some extent (4). Several syntactical rules have already been defined and is routinely used to verify (1) and (2) in figure 1[1]. However, manual inspections are necessary to combine both modeling and domain expertise to ensure optimal model consistence and integrity. The total number of entities in the tax domain (all three layers) is currently in excess of 80 000.

2.2. Modeling process

The current modeling process is largely based on a bottom-up process. Skilled data modelers are collecting implementation entities from web based input forms and relate these entities to both the structure layer and the semantics layer. If no suitable entity is found in the above layers the modeler either modify existing or create new entities. Generally both the implementation layer and the semantics layer will provide rich descriptions of the collected data and the legal terms respectively. On the other hand, the structure layer will be more generic and hence contain fewer entities than the other two layers; however, the structure layer will provide a large number of properties on each entity to facilitate efficient reuse by entities in the implementation and semantics layer. Possible reuse of governmental terms have previously been studied both on national and international levels [4].

To successfully develop a working taxonomy that will support both transactions between layers and domains, different skills and tools will be required. Subject matter experts proficient in the appropriate legal terms and definitions will develop the semantics layer whereas executive officers specialized in data modeling will contribute to both the implementation and the structure layer. This disjoint workflow will present challenges when it comes to how to eventually map and represent the governing terms efficiently and unambiguously; from the semantic layer to the interface presented to the public in the implementation layer. It is considered a particular high risk that subject matter experts could be reluctant to contribute if they perceive that the quality of underlying layers are low. To build their trust in the supporting structure an efficient vehicle for model discussions and communication must be established at a level that will efficiently span the user communities. The dedicated UML modeling tools and other table based repository browsers are generally useful to display subsets and verify specific hypothesizes. However, they do not provide good model overviews that can be used for collaboration at a general level, both to discover existing structures as well as to learn how to connect to or extend entities at a particular level. Also, patterns for best- or mal- practices that are not a part of any modeling guidelines or existing hypothesizes can be readily identified. This knowledge can subsequently be formalized in the guidelines and added to the routine syntactical verification. An efficient and powerful visualization of the e-governmental knowledge taxonomies is considered a substantial contribution to this discovery and collaboration process.

The rest of this article introduces Social Network Analysis (SNA) and describes how it can be used for both tentative taxonomy analysis as well as for providing good 2D layouts and visualizations. To further enhance the visualizations, the layered nature of the models are exploited to offset each layer in a 2.5D layout which can be viewed in 3D visualization tools such as provided for the extensible 3D markup language (X3D).

3. Social network analysis overview

Social network analysis (SNA) have been used for decades [5] to model the interactions between actors in a community. The area of application is wide and includes communication, transportation, sensor networks, knowledge discovery, chemistry, physics and anthropology [6]. Also, the suitability of SNA applied to ontology discovery has been described in [7] and the notion of network and data islands (cohesion,

connectivity) in the context of quality assurance of taxonomies and ontologies is used in [16].

3.1. 2D Layout algorithms

Frequently, the considered networks do not have an explicit geometric layout and several algorithms have been devised to distribute the nodes in 2D or 3D space. The resulting layouts will aim to optimize visualization by clustering nodes with high communication frequency and spreading out disjointed data islands. Most of the work showed here use variations of force direction algorithms [8] as provided by the GUESS [9][10] visualization tool. The force direction algorithms generally consider edges as forces (or springs), thus pulling highly connected nodes together until some sort of equilibrium is achieved. In addition, the resulting layouts are non-overlapping and largely symmetrical. Figure 2 shows a typical example of a force directed layout applied to the knowledge taxonomy found in a subset of the Tax Administration domain.

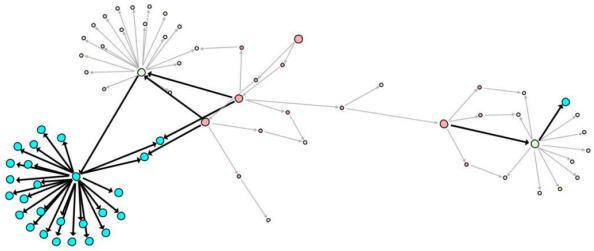

Figure 2 – Force directed layout of a subset of the Tax Administration metadata

Figure 2 is meant for illustrating the 2D layout only, however, by simple means such as color coding the layers and using different circle radius for entities (large) and properties (small), the usefulness for visualizing the knowledge taxonomies is evident.

3.2. Orthogonal 3D offsets

Several network layout algorithms offer 3D layouts and this was initially introduced to the e-governmental metadata. However, the resulting layouts offered little or no improvements on the 2D layouts as long as the third dimension was applied randomly and not as a contribution to clarify the inherent layered structure of the taxonomy. To alleviate this, the layout generation was performed in two separate steps; (1) a force directed algorithm was used to generate an initial 2D layout, optimizing the layout based on connectivity and aesthetics, and subsequently (2) each layer was offset orthogonally relative to each other to produce the model showed in figure 3.

As compared to the 2D layout, the layout in figure 3 offers a clear separation between individual layer connectivity and inter-layer connectivity.

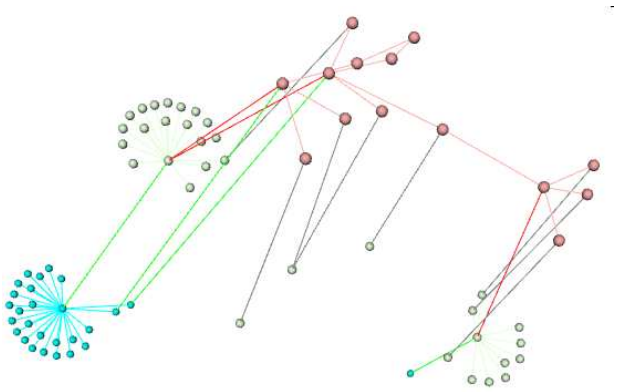

Figure 3 – Orthogonal offsets on force directed layout

Another example is given in figure 4 where a circular layout has been generated to visualize degree centrality for the same model.

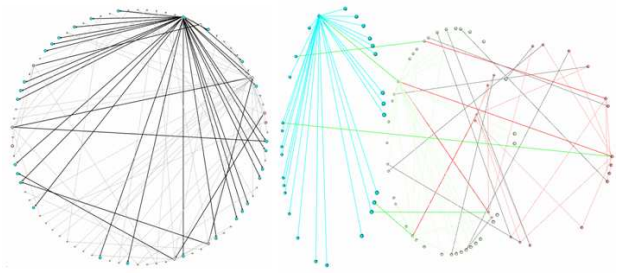

Figure 4 – (a) 2D and (b) 3D views of circular layouts

Both views (a) and (b) in figure 4 effectively highlights the degree centrality, however, the 3D models also illustrate the degree centrality per layer and connectivity between layers.

3.3. Centrality metrics

Several characteristic metrics have been developed to measure the performance of networks, where the majority is concerned with social aspects to assess how the individual nodes impact and interact with the overall network. In the work described here we focus on the centrality metrics. *Degree centrality* measures the number of direct connections for any node in the network. In the context of social networks this measures the individual's level of immediate connections. The *closeness centrality* is similar to degree except it also considers reach, meaning it will measure how well connected and how far an individual's connections can extend. *Inbetweeness centrality* measures how many nodes much pass through an individual's node to successfully communicate. Typically high inbetweeness indicates individuals with few direct connections; however, they are crucial by indirectly connecting other nodes. Removing nodes of high inbetweeness will typically result in disjointed clusters on either side of the removed node. In the case study presented in section 4, we argue for how the centrality metrics can be used to characterize the e-governmental knowledge taxonomies.

3.4. Model patterns

Our experience from multi-user taxonomy / ontology engineering during the last 10 years, we have learned that to achieve further quality improvements the engineers need tools to identify challenges, not related to single nodes, but related to patterns of nodes. The engineering quality patterns targeted in this article are listed below. We have used well established SNA centrality metrics, and visually inspecting the 2D and 3D models to identify the quality patterns in the taxonomy. The taxonomy used in our test bed is a subset from the Tax Administration. Section 4 describes the particular cases where the patterns were identified and possible relations to the SNA centrality metrics.

Overlap – Full and partial overlaps are considered here. In addition Soundex [11] (similar sound) and edit distance [12] (similar spelling) could be investigated. Full overlaps occur when two or more entities in the implementation layer refer to identical properties in the structure layer, and partial when they share a subset of properties.

Abundance – Entities in the semantics layer can be modeled standalone or with a rich set of relations to other nodes. The abundance pattern denotes rich semantics entities where the underlying entities fail to take advantage of the expressiveness and rather refer repeatedly to a single entity.

Incomplete – Many entities will have a good match in expressiveness across all three layers. Still, some matching properties might fail to be connected reducing the actual expressiveness as compared to the possible expressiveness. The incomplete pattern comprises entities which underutilize the potential connectivity offered by the above entities.

Inconsistency – Entities in the implementation layer can refer both to entities in the structure layer and entities in the semantics layer. To produce valid taxonomies the same implementation entity is not allowed refering to unrelated entities in the semantics layer. The inconsistency pattern hence denotes all constellations where an implementation entity both directly refers to a semantics entity and indirectly (via the structure layer) refers to another unrelated semantics entity.

Ambiguity pattern – The ambiguity pattern is a variation of the inconsistency pattern, however, the implementation entity does not misrepresent by inconsistent references. Rather, the entity properties refer to a different structure entity than the owning entity. Hence, one single implementation entity refers to two different structure entities.

All the above patterns are believed to adversely affect the quality of the model, both as a knowledge taxonomy and as an ontology. The list of patterns could easily been extended by e.g. dangling nodes. However, the assessment of the exact implications is outside the scope of this article and should be investigated in further work.

4. Case – Norwegian Tax Administration

To illustrate the described visualization, metrics and patterns we use production data from the Norwegian Tax Administration metadata repository. The metadata have been subjected to a rigorous syntactical data quality assessment and has been found to score close to 100% for compliance with the modeling guidelines. Hence the purpose of the visualization exercise is to add quality metrics to the already defined syntactical validations. This case limits itself to describe the discovery of additions to syntactical rules; however, future scenarios will also include the assessment of compliance to the real world and any inter-departmental issues.

4.1. Overview

The complete metadata for the Tax Administration office was extracted to produce figure 5. Several interesting characteristics can be noted at this level.

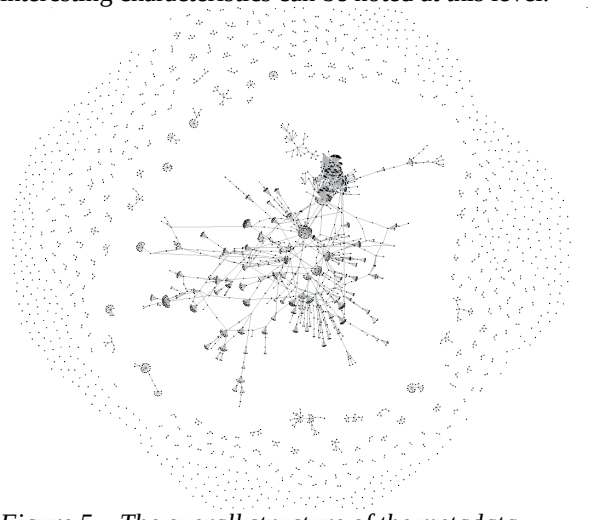

Figure 5 – The overall structure of the metadata

The main core is well connected and represents the model entities which are mature and have evolved over time. The disjointed clusters appearing at the fringe represents work in progress, where single nodes or groups of nodes have been defined but are not fully integrated in the domain. It is expected that a timeline animation would illustrate how entities travel from the outskirts of the model to the core as they evolve. Separate domains could be compared as a function of density and number of clusters to give a relative indicator of maturity.

4.2. Metrics

The impact of the SNA centrality metrics on the knowledge taxonomy is illustrated in figure 6. The figure depicts the metadata for one particular input form used by the Tax Administration. High degree centrality was found to denote well defined entities underutilized by other entities. For example the semantics entity *vehicle* could relate to a number of specific vehicles, however, none or few of the specializations were used. On the

other hand, high closeness centrality identified well defined *and* well used entities, and high inbetweeness typically identified key values or nodes unintentionally left dangling.

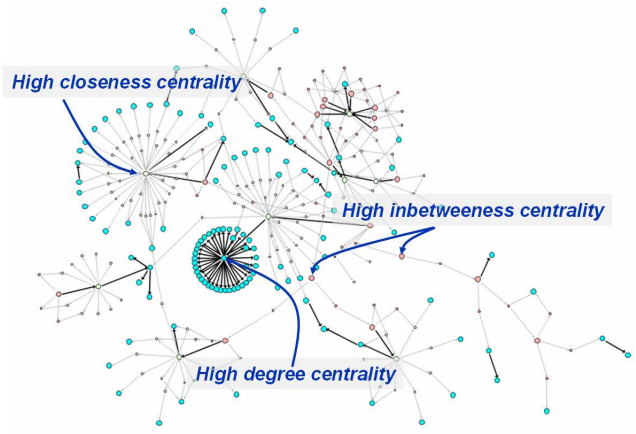

Figure 6 – SNA centrality for metadata

High inbetweeness for non-key nodes would typically increase the degrees of separation [15] between the implementation level and the semantics level, indicating that several of the patterns described in the next section could be expected to be found.

4.3. Patterns

By visually inspecting the networks, several recurring patterns could be identified. Similar techniques have previously been applied to identify access patterns on web pages [13].

The overlap pattern shown in figure 7 forms dense symmetrical clusters where all or subsets of the property connections (small circles) are identical.

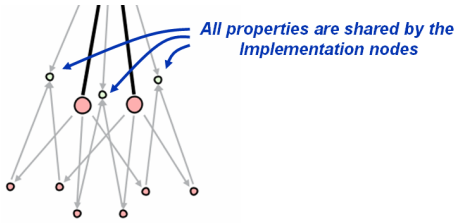

Figure 7 – Overlap pattern

The abundance pattern shown in figure 8 is described previously and is closely linked to degree centrality.

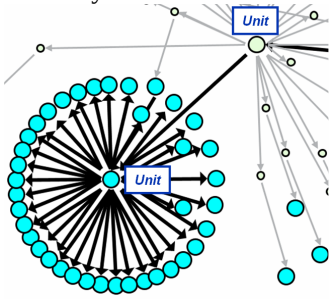

Figure 8 – Abundance pattern

Several entities were similarly defined across all three layers, however, the property connections would only run across the two bottom layers. This is labeled an incompleteness pattern and will result in *increased* inbetweeness centrality and *decreased* closeness centrality.

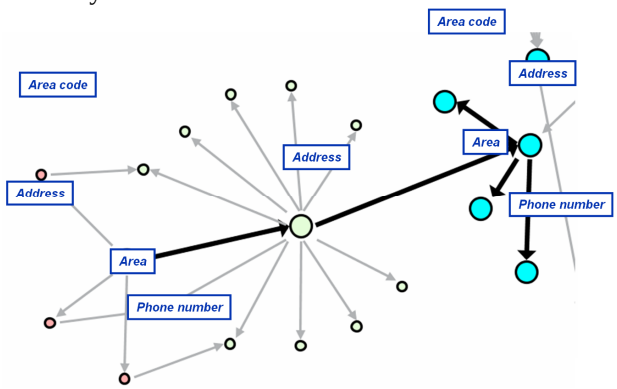

Figure 9 – Incompleteness pattern

Implementation level entities should only refer to one entity in the structure level, however, the ambiguity pattern shown in figure 10 shows how this is circumvented by referring to properties on disjointed structure entities. This will *increase* the degree centrality and again *decrease* the closeness centrality.

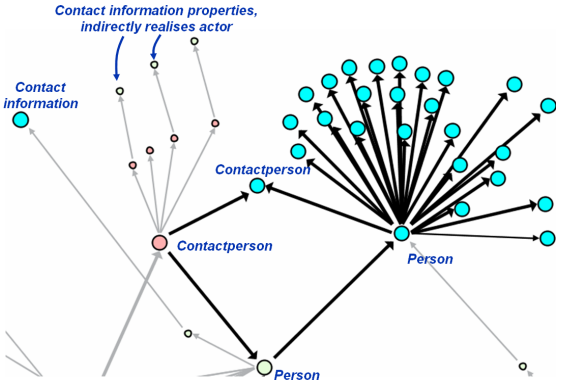

Figure 10 – Ambiguity pattern

In a similar fashion to the ambiguity pattern, implementation level entities should only connect to the semantics layer in closed loops (ie. the semantics entity should be identical or connected). Figure 11 shows an example of the inconsistency pattern. The *account* implementation entity refers both directly to *account* and indirectly to *account number* in the semantics level.

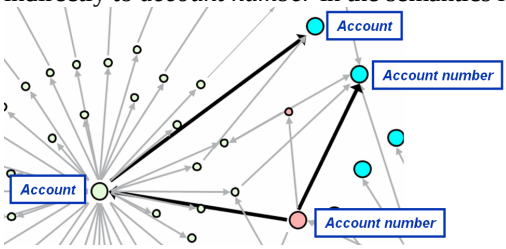

Figure 11 – Inconsistency pattern

The inconsistency pattern has largely the same effect on the centrality metrics as the ambiguity pattern; the

closeness centrality will *decrease* whereas the degree centrality will become more prominent.

5. Discussion

Social network analysis metrics and visualizations have been applied to aid the quality assurance, pattern discovery and communication of complex knowledge taxonomies for e-government metadata carried out in a multi-user engineering environment. Several patterns in the model were identified and provided useful input to best practices and validation rules. Full and partial overlap, inconsistencies and data islands (clusters) could easily be spotted and communicated to both domain experts and data modelers. The layered nature of the e-government metadata suggested a 2.5D visualization technique. The overall layout was calculated in 2D and each layer was subsequently offset orthogonally to aid the inspection the entities both individually and for interlayer integrity.

The social network analysis centrality metrics were found to have clear impacts on the metadata structure. The top nodes (key candidates) were found to have high *inbetweenness* centrality as all nodes should be reached from the top. Low *inbetweeness* for top nodes frequently indicated unwanted disjoints in the model. High *degree* centrality indicates well defined entities with low usage, whereas high *closeness* centrality indicates central nodes with rich definitions and high usage. Also, the majority of the common modeling patterns that were identified could be expressed directly as functions of the centrality metrics.

Social network analysis has proved a useful tool to diagnose and inspect complex knowledge taxonomies. Several issues could be identified which would be onerous to detect with more traditional means such as tree structures and table views. However, it did introduce some added complexity and some users could be deferred by the more elaborate navigation in a 2D/3D graphical world as opposed to classical table based interfaces. In addition to user adoption, more work is also required to further investigate both scalability and how to benefit further from existing social network methodologies. Animation could also be employed to show both how modeling trends change as a function of time and also how the usage of terms evolve (semantic drift [14]).

Most importantly, the visualization of the e-governmental metadata structures have shown substantial promise as a test bed for bridging the gap between subject experts and data modelers, offering a less specialized view than typically provided by the dedicated tools applied to the data collection (*implementation layer*) in one end and to the definition of the legal terms in the other end (*semantics layer*). The visualization of the knowledge taxonomies will also be important to improve the subject matter expert's trust in the model.

Often this trust is fragile and will be based on incidental perceptions, visualization will make the model more accessible and transparent and hence the perceptions can be solidly funded in how the model actually is implemented.

Acknowledgements

This work was carried out as part of the Semicolon project (project no. 183260/S10), funded by the Norwegian Research Council, Det Norske Veritas and other Semicolon participants.

References

[1] P. Myrseth, J. Stang and V. Dalberg, A Data Quality Framework Applied to E-Government Metadata, *The International Conference on E-Business and E-Government (ICEE2011)*, Shanghai, China, 2011

[2] W. Peng and L. SiKun, Social Network Visualization via Domain Ontology, International Conference on Information Engineering and Computer Science (ICIECS 2009), Wuhan, China, 2009

[3] R.A. Hanneman and M. Riddle, An Introduction to Social Networks, *url: http://faculty.ucr.edu/~hanneman/nettext/*, University of California, 2005

[4] S.M. Lui and C.C. Yu, Information Reuse Among Government Websites in Asian Countries, *International Conference on Information Information Reuse and Integration (IRI 2007)*, Las Vegas, USA, 2007

[5] J.A. Barnes, Class and Committees in a Norwegian Island Parish, Human Relations vol.7 no1, 1954

[6] U. Brandes, Social Network Analysis and Visualization, *IEEE Signal Processing Magazine*, November, 2008

[7] B. Hoser, A. Hotho, R. Jaschke, C. Schmitz and G. Stumme, Semantic Network Analysis of Ontologies, Proceedings of the 3rd European Semantic Web Conference, 2006

[8] Bas van Schaik, Force-directed Methods for Clustered Graph Drawing, *url: http://www.cs.uu.nl/docs/vakken/gd/bas2.pdf*, Universiteit Utrecht, 2005

[9] E. Adar, GUESS: A Language and Interface for Graph Exploration, *Conference on Human Factors in Computing Systems (CHI 2007)*, Montreal, Canada, 2007

[10] NWB Team, Network Workbench Tool. Indiana University, Northeastern University, and University of Michigan, http://nwb.slis.indiana.edu, 2006

[11] D. Knuth, The Art of Computer Programming, *Addison-Wesley*, 1968

[12] A. McCallum, String Edit Distance, *Computational Linguistics*, University of Massachusetts Amherst, 2006

[13] M. Kawamoto and T. Itoh, A Vizualisation Technique for Access Patterns and Link Structures of Web Sites, *International Conference Information Vizualisation (IV2010)*, London, 2010

[14] J. A. Gulla et. al., Semantic Drift in Ontologies, *International Conference on Web Information Systems (WEBIST 2010)*, Valencia, Spain, 2010

[15] P. Laddha, Degree of Separation in Social Networks, url: *http://arxiv.org/find/grp_cs/1/au:+Laddha_Prerana/0/1/0/all/0/1*, Cornell University Library

[16] S. Tartir &al, Metric Based Ontology Quality Analysis, *International Conference on Data Mining (ICDM 2005)*, Texas, USA, 2005