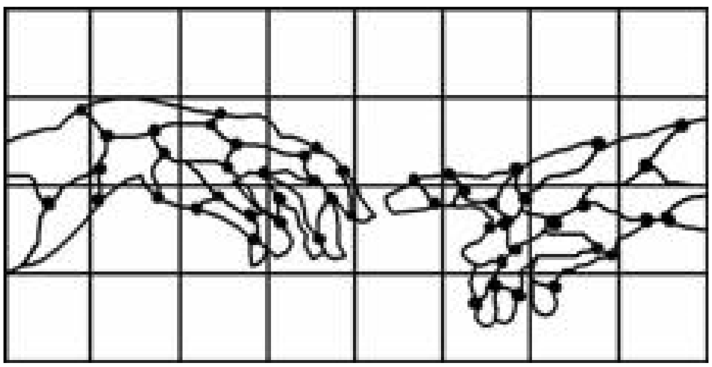
FIRST IJCAI WORKSHOP ON GRAPH STRUCTURES FOR KNOWLEDGE REPRESENTATION AND REASONING



Madalina Croitoru

Christophe Gonzales

Jerome Lang

Boris Motik

Marie – Laure Mugnier

Hierarchy Analysis of Knowledge Networks

Cliff Joslyn
Chief Scientist for Knowledge Sciences
Pacific Northwest National Laboratory
cjoslyn@pnl.gov

Abstract

Knowledge systems technologies are dominated by graphical structures such as ontologies, semantic graph databases, and concept lattices. A critical but typically overlooked aspect of all of these structures is their admission to analyses in terms of formal hierarchical relations. Transitivities of network links necessarily result in hierarchical levels, whether explicitly within directed acyclic graphs (DAGs) or implicitly through the identification of cycles. And whether from transitive link types in semantic graphs, or the explicit lattice structures of Formal Concept Analysis, the partial order representations of whatever hierarchy is present within a knowledge structure afford opportunities to exploit these hierarchical constraints to facilitate a variety of tasks, including ontology analysis and alignment, visual layout, and anomaly detection. In this short survey paper we introduce the basic concepts involved and address the impact of a hierarchical (order-theoretical) analysis on directed acyclic graphs in knowledge systems tasks.

1 Introduction

Knowledge systems technologies are dominated by graphical structures, including:

- **Semantic graph databases** [19] take the form of labeled directed graphs implemented in RDF[1]. Their OWL[2] ontological typing systems are also labeled directed graphs, frequently dominated by directed acyclic graph (DAG) and other hierarchical structures. Fig. 1 shows a toy example, where the ontology of classes on the left forms the typing system for the semantic graph of node and link instances on the right.
- **Concept lattices** [10; 11; 17] are hierarchical lattice structures derived from identifying the maximal connections among groups of objects and properties (rows and columns) of an attribute matrix, called a formal context. Fig. 2 shows a simple example from [11], indicating semantic generality of the attributes in terms of the number of their shared objects, and *vice versa.*

[1] http://www.w3.org/RDF
[2] http://www.w3.org/TR/owl-features

While other examples of graph-based knowledge structures abound, what characterizes these structures in particular is their hierarchical nature. There is an increasing emphasis on hierarchical structure in network science [4; 5], but these methods partition the set of nodes of an underlying simple (undirected) graph to produce a hierarchical decomposition. We are interested rather in the *intrinsic* hierarchical (level-based) nature of an underlying *directed* graph.

A good example is our concept lattice in Fig. 2, which is an explicit hierarchy in its entirety, as are semantic taxonomies such as the Gene Ontology [2] (GO[3]). But where OWL ontologies include hierarchical class structures, other portions can be non-hierarchical. And more general knowledge structures like semantic graphs are not explicitly or necessarily hierarchical, but may contain large hierarchical components.

In practice, ontologies are dominated by their "hierarchical cores", specifically their class hierarchies connected by `is-a` subsumptive and `has-part` compositional links. And many of the most common links in RDF graphs are transitive, including `causes`, `implies`, and `precedes`. We will show in Sec. 3 below that any transitive link yields a hierarchical structure in terms of the connectivity of its strongly connected components, and is thus amenable to a hierarchical analysis.

Whether from transitive link types in semantic graphs, or the explicit lattice structures of concept lattices, the partial order representation of whatever hierarchy is present within a knowledge structure affords opportunities to exploit these hierarchical constraints for a variety of tasks, including

Clustering and Classification: Including characterizing a portion of a hierarchy (e.g. groups of ontology nodes) to identify common characteristics [15; 23],

Alignment: Casting ontology matching [8][4] as mappings between hierarchical structures [13; 14].

Induction from Source Data: For example using concept lattices to induce ontologies from textual relations [17].

Visualization: Including exploiting the level structure of hierarchies to achieve a satisfactory layout [16].

In general, such a hierarchical analysis, when available, promises complexity reduction, improved user interaction

[3] http://www.geneontology.org
[4] http://www.ontologymatching.org

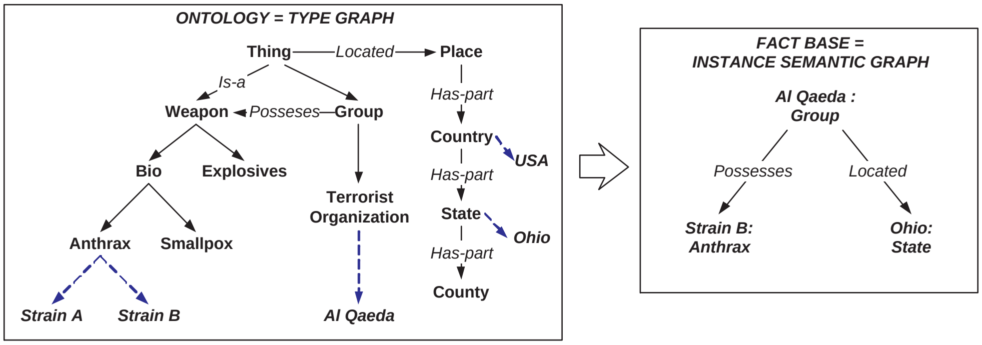


Figure 1: Toy model of a semantic graph database. (Left) Ontological typing system as a labeled, directed graph of classes (sample instances shown below dashed links). (Right) Conforming instance sub-graph.

with the knowledge base, and improved layout and visual analytics. In the remainder of this short survey paper we explicate the basic concepts referred to here and draw connections among these application areas.

2 DAGs and Partial Orders

Mathematically, hierarchies are represented as partially ordered sets (posets), which are reflexive, anti-symmetric, and transitive binary relations $\mathcal{P} = \langle P, \leq \rangle$ on an underlying finite set of nodes P [7]. While we typically think of hierarchies as tree structures, more general kinds of hierarchies have "multiple inheritance", where nodes can have more than one parent. These include lattice structures like the concept lattice in Fig. 2, where pairs of nodes have unique least common subsumers (and unique greatest lower bounds as well); partial orders where pairs of nodes can have an indefinite number of least common subsumers and greatest lower bounds; and finally general DAGs can also include "transitive links" which form shortcuts across paths.

Consider simple DAG in the top of Fig. 3. The two transitive links $1 \rightarrow H, 1 \rightarrow E$ connect the two paths $1 \rightarrow K \rightarrow H$ and $1 \rightarrow C \rightarrow I \rightarrow E$ respectively. Given a DAG $\mathcal{D}$, the DAG $\mathcal{P}(\mathcal{D})$ produced by including all possible transitive links consistent with its paths is its **transitive closure**, and determines an ordered set $\mathcal{P}(\mathcal{D}) = \langle P, \leq \rangle$ where $a \leq b \subseteq \mathcal{P}$ if there is a directed path from a to b in $\mathcal{D}$. The graph $\mathcal{V}(\mathcal{D})$ produced from a DAG $\mathcal{D}$ by removing all its transitive links (its transitive reduction [1]) determines a **cover relation** or **Hasse diagram**. Thus each cover relation $\mathcal{V}$ determines a unique poset $\mathcal{P}(\mathcal{V})$, and *vice versa* a poset $\mathcal{P}$ determines a unique cover $\mathcal{V}(\mathcal{P})$; each DAG $\mathcal{D}$ determines a unique poset $\mathcal{P}(\mathcal{D})$ and cover $\mathcal{V}(\mathcal{D})$; and each unique poset-cover pair determines a class of DAGs equivalent by transitive links.

For a DAG $\mathcal{D}$ we can measure its **degree of transitivity** as

$$TR(\mathcal{D}) := \frac{|\mathcal{D} \setminus \mathcal{V}(\mathcal{D})|}{|\mathcal{P}(\mathcal{D}) \setminus \mathcal{V}(\mathcal{D})|},$$

where $\setminus$ is set subtraction, we interpret each structure as the binary relation on P^2 of its incidence matrix, and $|\cdot|$ is cardinality, so that $|\cdot|$ is the number of links in $\cdot$, seen as a graph. $TR(\mathcal{D})$ measures the number $|\mathcal{D} \setminus \mathcal{V}(\mathcal{D})|$ of transitive links in $\mathcal{D}$ relative to the total possible number $|\mathcal{P}(\mathcal{D}) \setminus \mathcal{V}(\mathcal{D})|$ in its transitive closure $\mathcal{P}(\mathcal{D})$. In Fig. 3 we have $TR(\mathcal{D}) = \frac{2}{11}$, indicating a relatively low degree of transitivity.

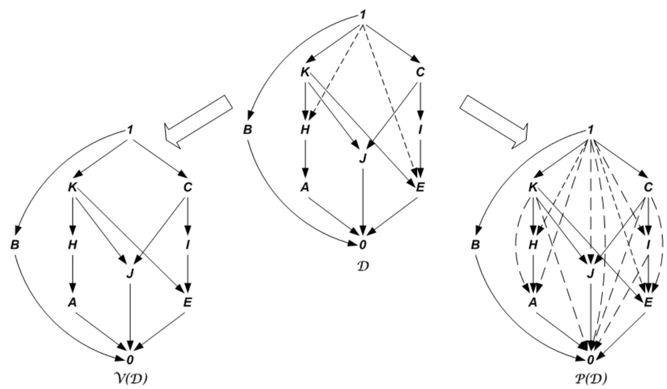
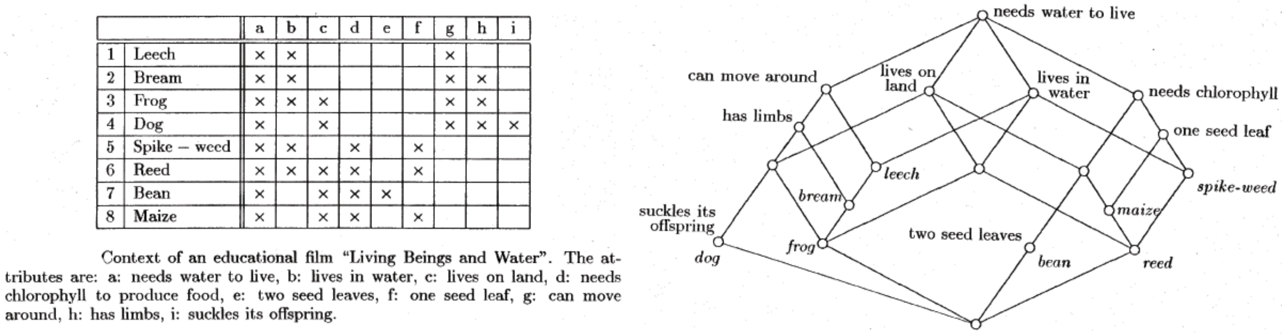


Figure 3: (Top) A DAG $\mathcal{D}$. (Left) Transitive reduction $\mathcal{V}(\mathcal{D})$. (Right) Transitive closure $\mathcal{P}(\mathcal{D})$.

In knowledge systems such as ontologies, our interpretation of the presence or absence of transitive links in DAGs is significant. If the link-type in question is anti-transitive, so that transitive links are disallowed, then clearly the presence of transitive links is in error. If, on the other hand, the link-type in question is atransitive, so that transitive links are allowed, but not required, then the $TR(\mathcal{D})$ measures this extent. But finally, if, as is the case with our subsumption and composition types, the link type represents a fully transitive property, then the presence of transitive links are irrelevant or erroneous. Effectively, such link types live in the trasitively equivalent class of DAGs, that is, in the partial order $\mathcal{P}(\mathcal{D})$, and $TR(\mathcal{D})$ can be used as an aid to the user or engineer to identify issues with the underlying ontology.

3 The Hierarchical Cores of Directed Graphs

So central to the consideration of hierarchy in a knowledge network is the question of the presence and prevalence of DAGs in general directed graphs which can possibly have cycles. Consider the network shown in Fig. 4 using a standard network layout with a primarily radial link distribution around centralized nodes. We can analyze this network as:



		a	b	c	d	e	f	g	h	i
1	Leech	×	×					×		
2	Bream	×	×					×	×	
3	Frog	×	×	×				×	×	
4	Dog	×		×				×	×	×
5	Spike – weed	×	×		×		×			
6	Reed	×	×	×	×		×			
7	Bean	×		×	×	×				
8	Maize	×		×	×		×			

Context of an educational film "Living Beings and Water". The attributes are: a: needs water to live, b: lives in water, c: lives on land, d: needs chlorophyll to produce food, e: two seed leaves, f: one seed leaf, g: can move around, h: has limbs, i: suckles its offspring.

Figure 2: (Left) A formal context of objects and their properties. (Right) Its concept lattice. From [11].

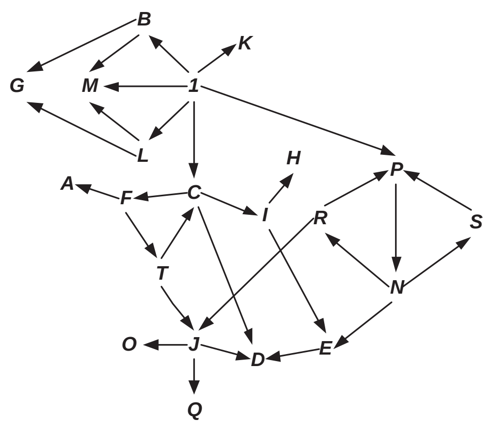


Figure 4: A network.

- Identify the leaves as those nodes $\{A, D, G, H, K, M, O, Q\}$ with no children.
- Identify the roots as those nodes $\{1\}$ with no parents.
- Identify the strongly connected components (SCCs [21]) as those sets of nodes which are directed cliques, where a directed path exists between all pairs on nodes. Each SCC is either a single directed cycle or a union of directed cycles, and are necessarily disjoint from each other: if two SCCs intersect, they'd form a single SCC together. In the example, there are two SCCs, $X = \{C, F, T\}$ (a single 3-cycle) and $Y = \{P, R, S, N\}$ (the union of the two 3-cycles $\{N, R, P\}$ and $\{N, S, P\}$).
- Identify the transitive links, these are the two links $1 \to M$ shortcutting the 3-paths $1 \to B \to M$ and $1 \to L \to M$, and the link $C \to D$ shortcutting the 4-path $C \to I \to E \to D$.

Fig. 5 shows our network with these components identified.

We proceed by contracting each SCC to a new node in a higher-order space, combining any multiple links between SCCs into one. The resulting structure is necessarily a DAG $\mathcal{D}$. Finally, we derive the transitive reduction of the cyclic decomposition to eliminate the transitive links, measuring

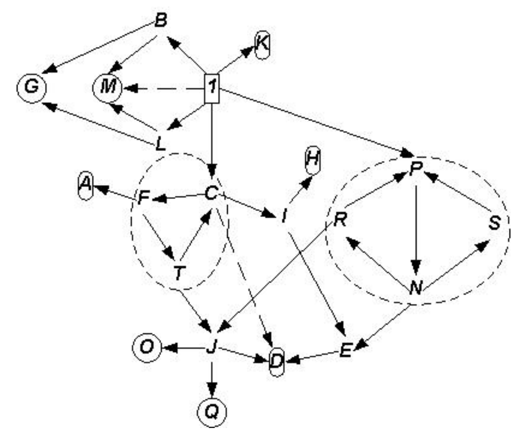


Figure 5: Components of the network's cyclic decomposition: roots are squares, leaves are oblongs, transitive links are dashed arrows, and strongly connected components are shown in dashed circles.

$TR(\mathcal{D})$. The resulting hierarchy is shown in Fig. 6, where the two SCCs are replaced with new nodes X and Y respectively. While this structure is substantially similar, of course these two new nodes will be identified as being new meta-nodes, and available for "double-clicking" to open up the SCCs revealing the original structure below.

While such an approach to a "cyclic decomposition" is a known standard for directed graphs [6], it is not usually applied for hierarchical anlaysis. Note that the layout is adjusted to bring the root to the top and the leaves to the bottom, and to emphasize the stratified nature of the hierarchy. This concept of "level" or rank is central to our approach, and will be dealt with in Sec. 4.1 below.

In addition to deriving the base cyclic decomposition, we are also interested in some numerical quantities such as the number of roots, leaves, and SCCs, along with the spectrum of sizes of the SCCs. Depending on the results of these measurements, it may or may not be appropriate to proceed with a hierarchical analysis of the network.

In particular, for a DAG $\mathcal{D}$ produced from an underlying directed graph $\mathcal{G}$, we can measure the degree of cyclicity as

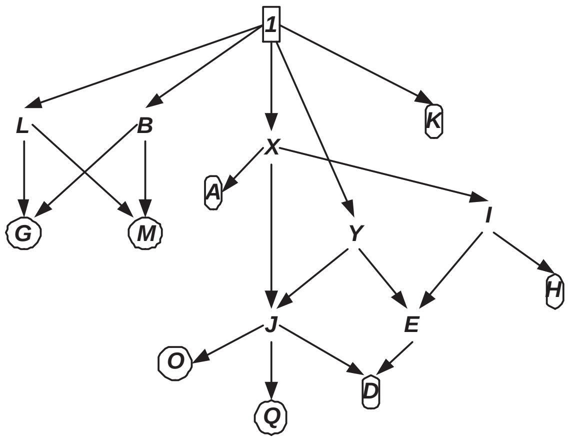


Figure 6: The network's cyclic decomposition displayed as a hierarchy.

$C(\mathcal{G}) := |\mathcal{D}|/|\mathcal{G}|$. As two extremes, $C(\mathcal{G}) = 1/|\mathcal{G}|$, so that the network consists of a single, large SCC; or, if $C(\mathcal{G}) = 1$, then the network was already a DAG at the outset, and no SCCs are observed. In either event, continuing with a hierarchical analysis would not be fruitful. But if there are a number of moderately sized SCCs, then the resulting hierarchical structure will provide greater simplification and a stratified view of the underlying complex structure.

4 Measures on Hierarchical Graphs

Given a hierarchical structure as a DAG represented by its transitive closure poset $\mathcal{P}$, perhaps derived as the cyclic decomposition of a network, or provided natively as in a taxonomic ontology, we now have a number of tools available to measure this hierarchical structure. Here we discusses **interval-valued rank** measuring the vertical level of nodes, and **order metrics** measuring the distances between nodes. See [12; 16; 18] for more details.

4.1 Interval-Valued Rank

Given a hierarchy e.g. in Fig. 6, we are concerned with the proper representation of the vertical level of each node, as represented by its positioning in a layout. We note that all children of the root are the same "distance" from the root, but if these are *also* leaves then they should be positioned further down. In other words, we need to exploit the vertical distance from *both* the top *and* a global bottom, in this case a virtual node $0 \in P$ we can insert and place below all the leaves.

For $a, b \in P$, let $h^*(a, b)$ be the length of the maximum path from a to b. Then the distance of a node $a \in P$ from the root $1 \in P$ is the **top rank** $r^t(a) := h^*(a, 1)$. Dually we define the **bottom rank** $r^b(a) := h^*(0, 1) - h^*(0, b)$, where $h^*(0, 1)$ is the overall **height** of the structure. Then the **interval rank** $\bar{R}(a) := [r^t(a), r^b(a)]$ becomes available as an interval-valued measure of the vertical levels over which a can range, while the **rank width** $W(a) := r^b(a) - r^t(a)$ is a measure of that range [16; 18].

We can exploit this vertical rank in terms of hierarchical layout and visualization, as shown for our example now in Fig. 7. Each node which sits on a complete chain (a path from 1 down to 0) of maximal size is placed horizontally at the center of the page. Nodes are laid out horizontally according to the size of their largest chains maximal chains. The result it to place maximal complete chains along a central axis, and short complete chains towards the outer edges. Nodes are placed vertically according to the mathematical quantity of the midpoint of their interval rank, but can be free to move between top rank $r^t(a)$ and bottom rank $r^b(a)$.

The result is that while nodes on maximal complete chains (all those intersecting the chain $0 \to D \to E \to I \to X \to 1$ in the example) exist at a single level, some (for example K) do not. While Fig. 7 shows a 2D layout, we have also deployed this concept in a 3D layout [16].

4.2 Order Metrics

Given the need to perform operations like clustering or alignment on ontologies represented as ordered sets $\mathcal{P} = \langle P, \leq \rangle$, it is essential to have a general sense of distance $d(a, b)$ between two nodes $a, b \in P$. The knowledge systems literature has focused on **semantic similarities** to perform a similar function, which are available when $\mathcal{P}$ is equipped with a probability distribution, derived, for example, from the frequency with which terms appear in documents (for the Wordnet[5] [9] thesaurus), or genes are annotated to GO nodes.

So assume a poset $\langle P, \leq \rangle$ with a base probability distribution $p \colon P \to [0, 1]$, $\sum_{a \in P} p(a) = 1$, and a "cumulative" function $\beta(a) := \sum_{b \leq a} p(a)$. We then generalize the join (least upper bound) and meet (greatest lower bound) operations in lattices as follows. Let $\uparrow a := \{b \geq a\}$ and $\downarrow a := \{b \leq a\}$ are the up-set (filter) and down-set (ideal) respectively of a node $a \in P$. Then for two nodes $a, b \in P$, let $a \nabla b := \uparrow a \cap \uparrow b$ and $a \Delta b := \downarrow a \cap \downarrow b$ be the set of nodes above or below respectively both of them. Then the generalized join $a \vee b$ is the set of minimal (lowest) nodes of $a \nabla b$, and the generalized meet $a \wedge b$ is the set of maximal (highest) nodes of $a \Delta b$. When $\mathcal{P}$ is a lattice, then $|a \vee b| = |a \wedge b| = 1$, recovering traditional join and meet.

Traditional choices for the semantic similarity $S(a, b)$ between two nodes then include the measures of Resnik, Lin, and Jiang and Conrath [3]:

$$S(a, b) = \max_{c \in a \vee b} [-\log_2(\beta(c))]$$

$$S(a, b) = \frac{2 \max_{c \in a \vee b} [\log_2(\beta(c))]}{\log_2(\beta(a)) + \log_2(\beta(b))}$$

$$S(a, b) = 2 \max_{c \in a \vee b} [\log_2(\beta(c))] - \log_2(\beta(a)) - \log_2(\beta(b))$$

respectively. But most of these are not metrics (not satisfying the triangle inequality), and all of these lack a general mathematical grounding and require a probabilistic weighting.

Our approach uses ordered set metrics [20; 22] which can use, but do not require, a quantitative weighting such as β, and always yield a metric. They are based on valuation functions $v \colon P \to \mathbb{R}^+$ which are, first, either isotone ($a \leq b \to v(a) \leq v(b)$) or antitone ($a \leq b \to v(a) \geq v(b)$); and then semimodular, in that

$$v(a) + v(b) \sim v^{\nabla}(a, b) + v_{\Delta}(a, b),$$

[5] http://wordnet.princeton.edu

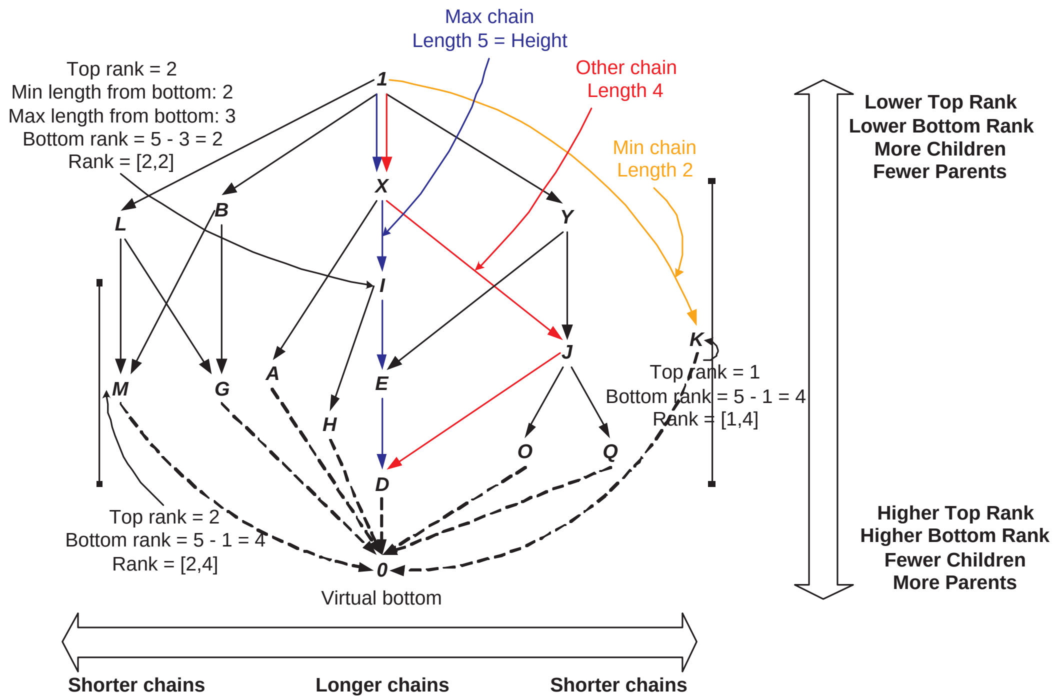


Figure 7: Chain layout of the cyclic decomposition of the network in Fig. 4.

where $\sim \in \{\leq, \geq, =\}$, yielding super-modular, sub-modular, and modular valuations respectively; and

$$v^{\nabla}(a,b) := \min_{c \in a \nabla b} v(c), \quad v_{\Delta}(a,b) := \max_{c \in a \Delta b} v(c).$$

Whether a valuation v is antitone or isotone, and then sub- or super-modular, determines which of four distance functions is generated, e.g. the antitone, supermodular case yields $d(a,b) = v(a) + v(b) - 2v^{\nabla}(a,b)$. When $\mathcal{P}$ is a lattice, then this simplifies to $d(a,b) = v(a) + v(b) - 2v(a \vee b)$.

Typical valuations v include the cardinality of up-sets and down-sets: $v(a) = |\uparrow a|, v(a) = |\downarrow a|$, and the cumulative probabilities used in semantic similarities $v(a) = \beta(a)$. In this way, poset metrics generalize semantic similarities and provide a strong basis for various analytical tasks.

5 Order Metrics in Ontology Alignment

A good example of the utility of this order theoretical technology in knowledge systems tasks is in ontology alignment [13; 14]. An ontology **alignment** is a mapping $f : \mathcal{P} \to \mathcal{P}'$ taking anchors $a \in P$ in one semantic hierarchy $\mathcal{P} = \langle P, \leq \rangle$ into anchors $a' \in P'$ in another $\mathcal{P}' = \langle P', \leq' \rangle$. In seeking a measure of the structural properties of the mapping f, our primary criterion is that f should not distort the metric relations of concepts, taking nodes that are close together and making them farther apart, or *vice versa.*

It should be noted that a "smooth" maping f is neither necessary nor sufficient to be a good alignment: one the one hand, a good structural mapping may be available between structures from different domains; and on the other, differences in semantic intent between the two structures may be irreconcilable. Nonetheless, other things being equal, it is preferable to have a more smooth mapping than not.

So, for two ontology nodes $a, b \in \mathcal{P}$, consider the **lower cardinality distance** $d_l(a,b) := |\downarrow a| + |\downarrow b| - 2 \max_{c \in a \wedge b} |\downarrow c|$. We can measure the change in distance between $a, b \in P$ induced by f as the **distance discrepancy**

$$\delta(a,b) := |\bar{d}_l(a,b) - \bar{d}_l(f(a), f(b))|,$$

where $\bar{d}_l(a,b) := \frac{d_l(a,b)}{\text{diam}_d(\mathcal{P})} \in [0,1]$ is the normalized lower distance between a and b in $\mathcal{P}$ given the diameter $\text{diam}_d(\mathcal{P}) := \max_{a,b \in P} d(a,b)$. We can measure the entire amount of distance discrepancy at a node $a \in P$ compared to all the other anchors $b \in P$ by summing

$$\delta_f(a) := \sum_{b \in P} \delta(a,b) = \sum_{b \in P} |\bar{d}_l(a,b) - \bar{d}_l(f(a), f(b))|,$$

yielding the discrepancy $\delta(f) := \sum_{a \in P} \delta_f(a)$ of the alignment.

Consider the example in Fig. 8, with the partial alignment function f as shown, mapping only certain nodes $\{B, E, G\}$ from $\mathcal{P}$ to $\mathcal{P}'$. Then we have e.g. the lower normalized distance between nodes E and G as $\bar{d}_l(E,G) = 1/3$; the distance discrepancy between the two nodes E, G in virtue of f as $\delta(E,G) = |1/3 - 3/5| = .267$; the entire distance discrepancy at the node E as $\delta_f(E) = 2/5$; and finally the distance discrepancy for the entire alignment as $\delta(f) = .47$.

6 Future Work

Our work continues across the range of tasks outlined here, and includes a number of future targets:

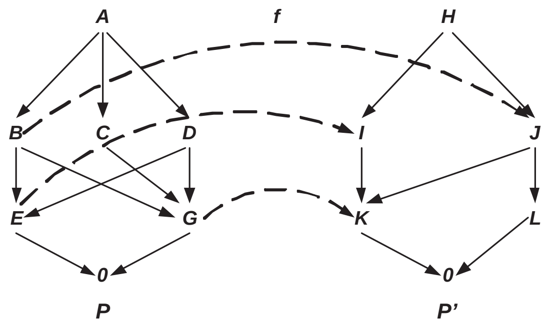


Figure 8: An example alignment.

- A characterization of semantic similarities S in terms of order metrics.
- Induction of new alignment links based on searching for low-discrepancy mappings within the space of order morphisms.
- Characterization of ontology link types in terms of hierarchical structure, factoring transitive and non-transitive link types.
- Identification of measures of centroid and dispersion in ontology clustering tasks.
- Anomaly detection in concept lattice based on correlation of interval rank and extent/intent size.

References

[1] Aho, AV; Garey, MR; and Ullman, JD: (1972) "The Transitive Reduction of a Directed Graph", *SIAM Journal of Computing*, v. **1**:2, pp. 131-137

[2] Ashburner, M; Ball, CA; and Blake, JA et al.: (2000) "Gene Ontology: Tool For the Unification of Biology", *Nature Genetics*, v. **25**:1, pp. 25-29

[3] A Butanitsky and G Hirst: (2006) "Evaluating WordNet-based Measures of Lexical Semantic Relatedness", *Computational Linguistics*, v. **32**:1, pp. 13-47

[4] A Clauset, C Moore, M Newman: (2006) "Structural Inference of Hierarchies in Networks", in: *Proc. 23rd Int. Conf. on Machine Learning*

[5] A Clauset, C Moore, M Newman: (2008) "Hierarchical Structure and the Prediction of Missing Links in Networks", *Nature*, v. **453**, pp. 98-101, `doi:10.1038/nature06830`

[6] T Cormen, CE Leiserson, RL Rivest: (1990) *Introduction to Algorithms*, MIT Press, Cambridge MA

[7] BA Davey, HA Priestly: (1990) *Introduction to Lattices and Order*, Cambridge UP, Cambridge UK, 2nd Edition

[8] Euzenat, Jérôme and Shvaiko, P: (2007) *Ontology Matching*, Springer-Verlag, Hiedelberg

[9] Fellbaum, Christiane, ed.: (1998) *Wordnet: An Electronic Lexical Database*, MIT Press, Cambridge, MA

[10] Ganter, Bernhard; Stumme, Gerd; and Wille, Rudolf, eds.: (2005) *Formal Concept Analysis: Foundations and Applications*, Springer-Verlag

[11] Ganter, Bernhard and Wille, Rudolf: (1999) *Formal Concept Analysis*, Springer-Verlag

[12] Joslyn, Cliff: (2004) "Poset Ontologies and Concept Lattices as Semantic Hierarchies", in: *Conceptual Structures at Work, Lecture Notes in Artificial Intelligence*, v. **3127**, ed. Wolff, Pfeiffer and Delugach, pp. 287-302, Springer-Verlag, Berlin

[13] CA Joslyn, B Baddeley, J Blake, C Bult, M Dolan, R Riensche, K Rodland, A Sanfilippo, A White: (2009) "Automated Annotation-Based Bio-Ontology Alignment with Structural Validation", *Proc. Int. Conf. on Biomedical Ontology (ICBO 09)*

[14] CA Joslyn, A Donaldson, P Paulson: (2008) "Evaluating the Structural Quality of Semantic Hierarchy Alignments", *Int. Semantic Web Conf. (ISWC 08)*, `http://dblp.uni-trier.de/db/conf/semweb/iswc2008p.html#JoslynDP08`

[15] Joslyn, Cliff; Mniszewski, Susan; Fulmer, Andy, and Heaton, G: (2004) "The Gene Ontology Categorizer", *Bioinformatics*, v. **20**:s1, pp. 169-177

[16] CA Joslyn, SM Mniszewski, SA Smith, PM Weber: (2006) "SpindleViz: A Three Dimensional, Order Theoretical Visualization Environment for the Gene Ontology", in: *Joint BioLINK and 9th Bio-Ontologies Meeting (JBB 06)*, `http://www.bio-ontologies.org.uk/2006/download/Joslyn2EtAlSpindleviz.pdf`

[17] CA Joslyn, P Paulson, KM Verspoor: (2008) "Exploiting Term Relations for Semantic Hierarchy Construction", in: *Proc. Int. Conf. Semantic Computing (ICSC 08)*, pp. 42-49, IEEE Computer Society, Los Alamitos

[18] CA Joslyn, A Pogel, S Schmidt: (2008) "Ordered Set Interval Rank for Knowledge Systems Analysis and Visualization", in preparation

[19] McBride, Brian: (2002) "Jena: A Semantic Web Toolkit", *IEEE Internet Computing*, v. **6**:6, pp. 55-59

[20] Monjardet, B: (1981) "Metrics on Partially Ordered Sets - A Survey", *Discrete Mathematics*, v. **35**, pp. 173-184

[21] Nuutila, Esko and Soesalon, Eljas: (1994) "On Finding the Strongly Connected Components in a Directed Graph", *Information Processing Letters*, v. **49**, pp. 9-14

[22] C Orum, CA Joslyn: (2009) *Valuations and Metrics on Partially Ordered Sets*, in: *Discrete Mathematics*, `http://arxiv.org/abs/0903.2679v1`, submitted

[23] Verspoor, KM; Cohn, JD; Mniszewski, SM; and Joslyn, CA: (2006) "A Categorization Approach to Automated Ontological Function Annotation", *Protein Science*, v. **15**, pp. 1544-1549