Mining Data and Ontologies Seamlessly with RDF Hypergraphs

Haishan Liu
University of Oregon
Eugene, OR, 97403, USA
ahoyleo@cs.uoregon.edu

Dejing Dou
University of Oregon
Eugene, OR, 97403, USA
dou@cs.uoregon.edu

Paea LePendu
Stanford University
Stanford, CA, 94305, USA
plependu@stanford.edu

Ruoming Jin
Kent State University
Kent, OH, 44242, USA
jin@cs.kent.edu

Nigam Shah
Stanford University
Stanford, CA, 94305, USA
nigam@stanford.edu

ABSTRACT

In this paper, we address an interesting data mining problem of finding semantically associated itemsets (i.e., items connected via indirect links) on ontology-annotated datasets. Semantically rich datasets, such as those annotated by formal ontologies and represented in RDF or OWL, can be viewed as datasets embedded with domain knowledge. This new standard of representation provides a unique opportunity to mine the data and domain knowledge simultaneously. In this paper, we develop a way to represent both data and ontology in a unified graph representation, and we propose to use random walk with restart to efficiently calculate the similarity between items so as to determine their associations from very large size of data and ontologies. We show the proposed method is indeed capable of capturing semantically associated itemsets while seamlessly incorporate domain knowledge defined in ontologies through experiments performed with real life data and ontologies. The semantically associated itemsets discovered in our experiment is promising to provide valuable insights on interrelationship between medical concepts and other domain specific ones.

Keywords

Semantic Data Mining, Random Walk, Semantically Associated Itemset

1. INTRODUCTION

Researchers around the world are linking more and more data to ontologies. There are two major challenges facing researchers when it comes to mining extremely large sets of ontologies and data. The first is to leverage both the ontologies and the data in a systematic and scalable way. The second is to deal with errors in both ontologies and data since neither of them is perfect in reality. Previous research has not been sufficient to address these challenges. For example, some approaches have utilized ontologies in data mining, but the ontologies are typically partially used (mainly the subsumption relationship) on a specific portion of the task (most often concept aggregation). On the other hand, limited attempts have been made to check errors in large ontologies. Traditional approaches to sanitize knowledge bases by using an inference engine for consistency checking is hard to scale. The justification in an inconsistent ontology is normally done by identifying "minimal unsatisfiable subsets" [**?**, **?**] of statements that cause it to be inconsistent. Deriving all minimal unsatisfiable sets will help generate the "simplest" explanation (justification) for the errors.

We believe that with the advent of increasing amount of ontology-annotated data, new possibilities are opened up for both data mining and ontology development. Therefore our new method, which we call *semantic data mining*, focuses on drawing insights from both domain knowledge and data in a systematic way along all kinds of relationships defined by the ontology. It enables domain knowledge to be seamlessly brought into the data mining process, helping improve the quality of pattern discovered in a noisy environment. On the other hand, it also benefits the ontologies by having empirical substantiation from data to either bolster a priori ontological assertions, or detect potential errors therein.

Semantic data mining differs from other recent studies, by leveraging links between entities defined by ontologies—via annotations—to the mining algorithms explicitly in a unified model. With ontology-annotated data, this would require traversing links across the ontologies to infer implicit interconnections among the data. Hence, semantic data mining is inspired by a combination of graph-based representation [**?**], hypergraphs [**?**, **?**], and random walks [**?**, **?**, **?**, **?**]. This work extends our previous work that implements a hyper graph-based approach to learn associations from interlinked data (without ontologies) [**?**]. The hypergraph representation proposed by Hayes et al. [**?**] is a key innovation. We adopt this representation in our approach because properties in RDF triples can be represented as first class objects among all interrelated objects, enabling us to embed both the ontology and the data together and to serialize it into a bipartite graph representation for scalable processing.

The mining problem we aim to solve is an extension to one of

the most basic and useful data mining tasks, association rule mining [?]. Traditional association rule mining relies on co-frequencies within transactions. We look one step further to find indirectly associated items. This simple extension has far reaching applications in healthcare. For instance, consider a simple scenario illustrated by Swanson [?] years ago while studying Raynauld's syndrome. He noticed that the literature discussed Raynauld's syndrome (Z), a peripheral circulatory disease, together with certain changes of blood in human body (Y); and, separately, the consumption of dietary fish oil (X) was also linked in the literature to similar blood changes. But fish oil and Raynauld's syndrome were never linked directly in any previous studies. Swanson reasoned (correctly) that fish oil could potentially be used to treat Raynauld's syndrome, i.e., $X \rightsquigarrow Y \rightsquigarrow Z$. We call such indirectly associated items, (X, Z), *semantically associated itemsets*.

Our work makes the following main contributions: First, we employ a hypergraph representation to capture both ontologies and data. We can weight each hyperedge so that certain links (such as *is_a* or *may_treat* relationships) can carry appropriate strength. Next, we serialize the hypergraph and weighted hyperedges into a bipartite representation for efficient processing. Then, we implement highly efficient and scalable random walks with restart over the bipartite graph to generate frequent itemsets, including associations that may not necessarily be co-frequent. The discovered semantic associations can be used in turn to detect potential errors in ontologies. Finally, we evaluate the effectiveness of the results on well-known shopping cart benchmark datasets, as well as the large real-world healthcare dataset.

2. RELATED WORK

2.1 Ontology Annotated Datasets

Using formal ontologies to annotate data has become increasingly popular in many scientific domains. For instance, in the genetic domain, researchers curate literature to generate ontology-annotated data for different species of model organisms by linking specific proteins to various classes in the Gene Ontology (GO). These publicly available GO annotation databases [?] make enrichment analysis possible, which enables researchers to functionally profile sets of interesting genes identified by microarray experiments [?]. At Stanford, the National Center for Biomedical Ontology (NCBO) annotates large volumes of biomedical text for search and mining, a technology which won the 2010 ISWC Semantic Web Challenge Open Track [?, ?] and has been used, for instance, to profile disease research [?]. Finally, millions of patient electronic health records are being annotated using medical ontologies like SNOMED-CT and MedDRA in efforts to advance patient healthcare [?].

2.2 Ontologies in Data Mining

Staab and Hotho [?] were one of the earliest to utilize the idea of mapping terms in text to classes in an ontology and they essentially use the ontology to aggregate data and thus reduce feature dimensionality during clustering. Adryan et al. [?] enable cluster visualization for gene expression data by navigating various levels of Gene Ontology hierarchy. Shen et al. [?] use the ontology to check for consistency of association rules. Wen et al. [?] take into consideration the ontology hierarchy to offset biases toward overly-general terms in text mining.

2.3 Graphs in Data Mining with Ontologies

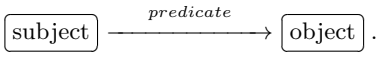
Kiefer et al [?] extends the SPARQL query language to enable creating and working with the data mining model. Lin et al. [?] also treats the RDF triple store as a datasource during mining and develops Relational Bayesian Classifiers (RBCs) that aggregate SPARQL queries. Similarly, Bicer et al [?] define kernel machines over RDF data where features are constructed by ILP-based dynamic propositionalization. In each case, RDF is merely a data model, paying little attention to the use of domain-specific knowledge in related ontologies.

3. MINING ONTOLOGY-ANNOTATED DATA USING HYPERGRAPHS

Besides the common graph-theoretic model of RDF as labeled, directed multi-graphs, Hayes has established that RDF can be also represented as hypergraphs (bipartite graphs) [?]. This result constitutes an important aspect of the theoretical basis of this paper and is discussed in detail below.

3.1 Hypergraph Representation for Ontologies and RDF Data

The RDF graph is defined as a set of RDF triples and can be visualized as a *directed labeled graph* as follows:



RDF makes the artificial distinction between resources and properties, which leads to incongruous graph representations. Consider the following example:

EXAMPLE 3.1. (**Discrepancy of the RDF directed labeled graph**) *A set of RDF statements describes the following relationships among people:* $\langle a\ collaborateWith\ b\rangle$, $\langle b\ coauthorWith\ c\rangle$, $\langle a\ influences\ d\rangle$ *and* $\langle d\ friendOf\ e\rangle$; *where a, b, c, d, and e are variables representing people. Futhermore,* $\langle coauthorWith\ subProperty\ collaborationWith\rangle$. *In this case, the graph representation mixes nodes and edges incongruously. See Figure* **??**.

As illustrated in Figure **??**, a hypergraph representation of the RDF statements avoids such discrepancies. A *hypergraph* [?] is a generalization of a traditional graph where edges, called hyperedges, can connect more than two vertices. If each edge in a hypergraph covers the same number of nodes, it's called an r-uniform hypergraph, r being the number of nodes on each edge. Any RDF Graph can be represented by a simple, ordered, 3-uniform hypergraph, in which an RDF triple corresponds to a hypergraph edge, the nodes being the subject, predicate and object in this order [?]. In this way, ontological statements can be represented in a coherent graph model (Fig. **??**(B)).

DEFINITION 3.1. (**Hypergraph**) *Formally, a hypergraph* $G = (V, E)$, *is a pair in which* V *is the vertex set and* E *is the*

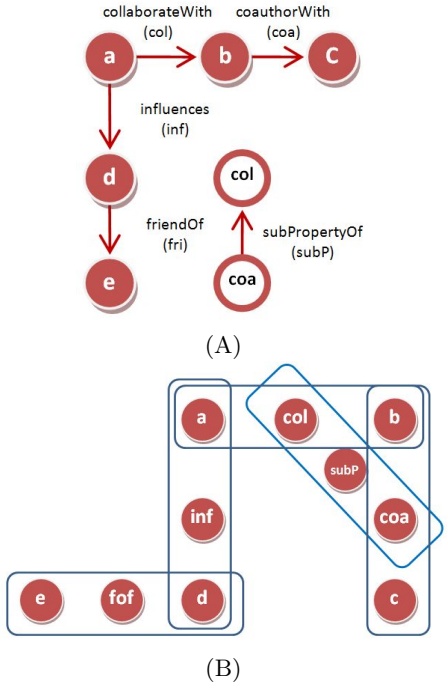


Figure 1: As a directed graph representation (A) the nodes and edges for *col* and *coa* are incongruously intermixed, where the hypergraph (B) seamlessly avoids the discrepancy.

hyperedge set where each $e \in E$ is a subset of V. A weighted hypergraph is a hypergraph that has a positive number $w(e)$ associated with each hyperedge e; called the weight of hyperedge e: Denote a weighted hypergraph by $G = (V, E, w)$.

Furthermore, A hypergraph $HG = (V, E)$ can be transformed to a *bipartite graph* BG as follows: the node sets V and E be the two parts of BG, and (v_1, e_1) is connected with an edge if and only if vertex v_1 is contained in the hyperedge e_1 in HG. In other words, the incidence matrix of HG can be viewed as the node adjacency (biadjacency) matrix of the bipartite graph. The proof that BG is indeed bipartite is straightforward and we omit it for lack of space.

Bipartite graphs are well-known from their efficient computational properties and we demonstrate next that the RDF bipartite graph can be efficiently mined for *semantically associated itemsets.*

3.2 Ontology-annotated Data as Bipartite RDF Graphs

There already exist methods for transforming data, such as those in relation databases, into RDF [**?**]. An ontology-annotation, as we see it, is a binary value representing whether some ontological concept (or class) is associated with some entity. Often, this means that some concept appears in some document and thus the ontology serves to index the document with related concepts [**?**]. Thus, we can think of ontology-annotations as a table, with each row representing an entity (e.g., a document), and each column is a class from some ontology. Cells having a "1" denotes that the document *mentions* the term defined by the class. RDF can be seen as a sparse matrix representation of this data (Table **??**). This idea can be easily extended to nominal-valued tables as well, or with other relationships besides *mentions* as we illustrate when discussing ontologies as bipartite graphs in the next section.

	f_1	$\cdots$	f_n
r_1 :	0	$\cdots$	1
$\vdots$	$\vdots$	$\ddots$	$\vdots$
r_m :	1	$\cdots$	0

(A)

s	p	o
<r_1	mentions	f_n>
<r_m	mentions	f_1>

(B)

Table 1: Ontology-annotated data (A) is a feature matrix attributing classes from ontologies, f_i, to entities such as documents, r_j, that is easily represented in sparse matrix form using RDF statements (B).

3.3 Incorporating Ontologies into Bipartite RDF Graphs

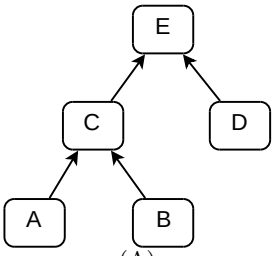
Given that ontology-annotated data links entities in the data to classes from ontologies, the RDF bipartite graph can also capture relations among classes in the ontology in the same representation. Thus, data mining algorithms will benefit directly from their seamless integration. The following example shows the combination of an ontology graph and a data graph:

EXAMPLE 3.2. (**Ontology and data in one bipartite RDF graph**) *Considering just the* rdfs:subClassOf *for illustration (other relationships are treated no differently), Figure* **??***(A,B) shows a small hierarchy for concepts* A–E. *Similarly, Figure* **??***(C) provides a set of ontology-annotated data. As illustrated earlier, the ontological statements as hyperedges are structurally no different than data-oriented RDF statements. Thus, Figure* **??** *illustrates the combined graph.*

Figure **??** illustrate another way to view the RDF bipartite graph in Figure **??** (A) by visually rearranging nodes. Our proposed semantic association mining algorithm is based on the idea of neighborhood formation using random walk with restart on column nodes (the right most part). Conversely, neighborhood formation on row nodes (the left most part) is able to facilitate row-oriented analysis such as classification and clustering. This shows the flexibility of the RDF bipartite graph in support of different mining tasks. Further discussion can be found in Section **??**.

We also distinguish paths in the RDF bipartite graph by assigning weights to those paths that represent different semantic relationships such as class subsumption, *part_of*, and other general or domain–specific properties such as *may_treat.*

DEFINITION 3.2. (**Data model for the combined RDF bipartite graph**) *The unified RDF bipartite graph of both*



(A)

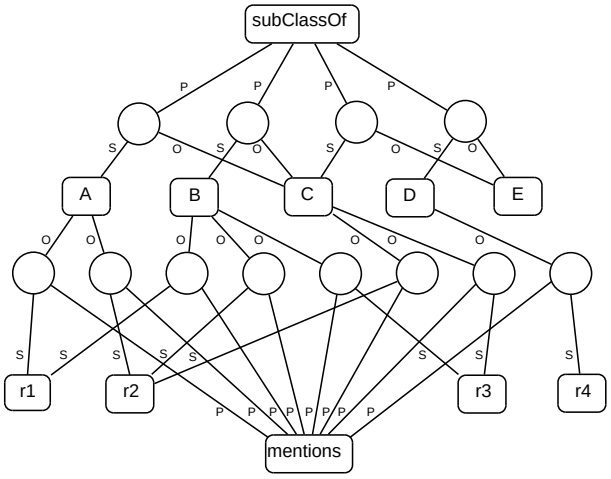
	A	B	C	D	E
$A:$	0	0	1	0	0
$B:$	0	0	1	0	0
$C:$	0	0	0	0	1
$D:$	0	0	0	0	1
$E:$	0	0	0	0	0

(B)

	A	B	C	D	E
$r_1:$	1	1	0	0	0
$r_2:$	1	1	1	0	0
$r_3:$	0	1	1	0	0
$r_3:$	0	0	0	1	0

(C)

Figure 2: Five concepts (*A*–*E*) are represented visually as a hierarchy (A) and also as a a hypergraph using the binary feature matrix (B), where a "1" denotes *rdfs:subClassOf*, which is similar to the ontology-annotated data (C), where "1" denotes *mentions*.



(A)

s	*p*	*o*
<A>	<subClassOf>	<C>
<B>	<subClassOf>	<C>
<C>	<subClassOf>	<E>
<D>	<subClassOf>	<E>
<r1>	<mentions>	<A>
<r1>	<mentions>	<B>
<r2>	<mentions>	<A>
<r2>	<mentions>	<B>
<r2>	<mentions>	<C>
<r3>	<mentions>	<B>
<r3>	<mentions>	<C>
<r4>	<mentions>	<D>

(B)

Figure 3: The RDF bipartite graph representation (A) easily combines both the ontology-annotated data with the ontological relationships (B) based on the information described in Figure ??.

data and ontology is defined as $G = \langle V_v \cup V_s, E\rangle$, where V_v denotes value nodes corresponding to RDF components (subject, predicate, or object), and V_s denotes statement nodes corresponding to RDF statements. More specifically, statement nodes can be further divided according to whether they are from data or ontology, i.e., $V_s = V_d \cup V_o$; the value nodes can be divided according to whether they represent rows or attributes in the data, i.e. $V_d = V_r \cup V_a$. The graph G can be represented in a biadjacency matrix $\mathbf{M}$, where $\mathbf{M}(i, j)$ is non-zero if there is an edge between $\langle V_{v_i}, V_{s_j}\rangle$. For an unweighted graph, the value can be 0/1, while for a weighted graph, any non-negative value.

The biadjacency matrix $\mathbf{M}$ can be split into vertical stripes by statement nodes V_s. For example, according to Figure **??**(B), the bipartite graph corresponding to lower 8 RDF statements representing the underlying transaction table can be modeled as the matrix $\mathbf{M}_d$ in Equation **??** (RDF statement nodes are labeled $s_1 \ldots s_8$ respectively); and the bipartite graph corresponding to upper 4 statements (labeled $s_9 \ldots s_{12}$) representing the subsumption hierarchy in the ontology can be modeled as the matrix $\mathbf{M}_o$ in Equation **??**.

To obtain the biadjacency matrix $\mathbf{M}$ of the combined RDF bipartite graph in Figure **??**, we can simply concatenate $\mathbf{M}_d$ and $\mathbf{M}_o$ horizontally: $\mathbf{M} = [\mathbf{M}_d\ \mathbf{M}_o]$. In general, If there are k different semantic relationships in the ontology, $\mathbf{M}_o$ can be further divided into more vertical stripes $\mathbf{M}_{o_i}, i = 1 \ldots k$, where $\mathbf{M}_{o_i}$ may represent, for example, the *part_of* lattice. Each $\mathbf{M}_{o_i}$ is distinguished from another by the respective weight. In this case, $\mathbf{M}$ is the horizontal concatenation of all the weighted vertical stripes as shown in Equation **??**. After the concatenation, $\mathbf{M}$ can be represented as the form shown in Equation **??**.

$$\mathbf{M}_d = \begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ A \\ B \\ C \\ D \\ E \end{array} \begin{array}{c} \begin{array}{cccccc} s_1 & s_2 & s_3 & \ldots & s_8 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & & 0 \\ 0 & 0 & 1 & & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & & 1 \\ 1 & 0 & 1 & & 0 \\ 0 & 1 & 0 & & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & & 1 \\ 0 & 0 & 0 & & 0 \end{bmatrix} \end{array} \quad (1)$$

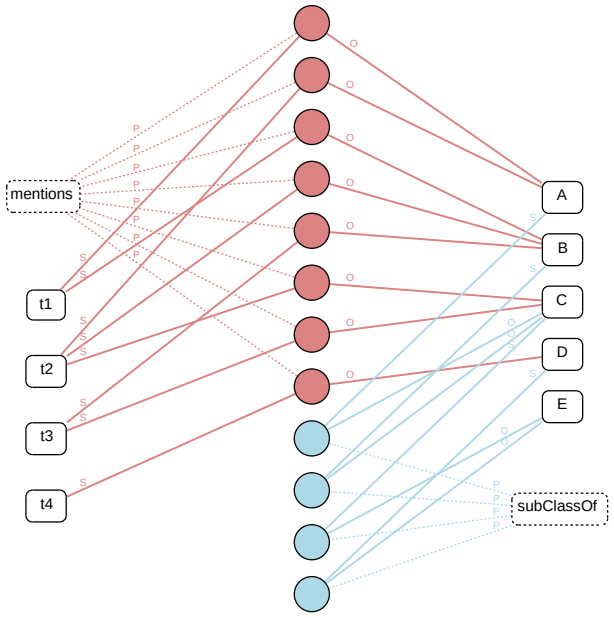


Figure 4: By grouping the nodes according to whether they are row elements or column elements in Figure ?? (B), the bipartite graph shown in Figure ?? (A) can be further transformed as shown in this figure.

$$\mathbf{M}_o = \begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ A \\ B \\ C \\ D \\ E \end{array} \begin{array}{c} \begin{array}{cccc} s_9 & s_{10} & s_{11} & s_{12} \end{array} \\ \left[\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{array}\right] \end{array} \tag{2}$$

$$\mathbf{M} = \begin{bmatrix} w_d\mathbf{M}_d & w_{o_1}\mathbf{M}_{o_1} & w_{o_2}\mathbf{M}_{o_2} & \dots \end{bmatrix} \tag{3}$$

$$\mathbf{M} = \begin{array}{c} \\ r \\ a \end{array} \begin{array}{c} \begin{array}{cccc} ds & os_1 & os_2 & \dots \end{array} \\ \left[\begin{array}{c|c|c|c} \mathbf{M}_{dr} & \mathbf{0} & \mathbf{0} & \dots \\ \hline \mathbf{M}_{da} & \mathbf{O}_1 & \mathbf{O}_2 & \dots \end{array}\right] \end{array} \tag{4}$$

The RDF bipartite graph as a unified representation for both ontologies and data serves as the basis for semantic data mining. Given this, the main research challenge is how to develop meaningful graph-based analysis method to utilize the information embedded in this representation. In this work, we focus on tackling one important data mining tasks, namely, *association mining*. We will show that, with prior knowledge encoded via ontologies, the unified RDF hypergraphs will enable us to discover hidden association between entities, between entities and ontological concepts, and between ontological concepts.

3.4 Similarity Ranking by Random Walk with Restart

Similar to the relevance score [**?**], we believe that two items have a strong semantic association if they are related to many similar objects. We apply random walks with restart (RWR) to capture this intuition from the bipartite graph. We choose the random walk approach to compute the relevance score because it gives nodes that are connected to many other nodes higher ranks, because the random walker has more paths to reach those nodes. The purpose of the periodic restart is to raise the chance that closely related nodes are visited more often than other nodes.

Intuitively, RWR in a bipartite graph works as follows: assume we have a random walker that starts from node a. For each step, the walker chooses randomly among the available edges from the current node. After each iteration, with probability c, it resets its position back to node a. The final steady-state probability that the random walker reaches node b is the similarity score of L with respect to a: $s(a, b)$. We denote the similarity score between entities e_1 and e_2 by $s(e_1, e_2)$, where $s(e_1, e_2) \in [0, 1]$ and $s(e_1, e_2) = 1$ if $e_1 = e_2$ such that an attribute node a in the unified graph $G = G_d \cup G_o$ and $a \in G_d \cap G_o$ we want to compute a similarity score $s(a, b)$ for all nodes $b(\neq a) \in G_d \cap G_o$.

In more detail, given the biadjacency matrix $\mathbf{M}$ in Equation **??** for the combined RDF bipartite graph G, we can construct the adjacency matrix $\mathbf{A}$ of G:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{M} \\ \mathbf{M}^T & \mathbf{0} \end{bmatrix}$$

The probability of a random walker taking a particular edge $\langle a, b\rangle$ from a node a while traversing the graph is proportional to the edge weight over the total weight of all outgoing edges from a, i.e., $P(a, b) = A(a, b)/\Sigma_{i=1}^{m+n} A(a, i)$. Therefore, the Markov transition matrix P of G is constructed as: $P = normc(A)$, where $normc(A)$ normalizes A such that every column sum up to 1.

First, we transform the input attribute node a into a $(k+n)\times 1$ query vector $\mathbf{q}_a$ with 1 in the a-th row and 0 otherwise. Second, we need to compute the $(k + n) \times 1$ stead-state probability vector $\mathbf{u}_a$ over all nodes in G. Last we extract the probabilities of the row nodes as the similarity score vectors. Note that $\mathbf{u}_a$ can be computed by an iterated method from the following lemma.

LEMMA 3.1. *Let c be the probability of restarting random-walk from the node a. Then the steady-state probability vector $\mathbf{u}_a$ satisfies*

$$\mathbf{u}_a = (1 - c)P_A\mathbf{u}_a + c\mathbf{q}_a \ . \tag{5}$$

The iterative update of $\mathbf{u}_a$ in the algorithm (inside the while loop) is modified from Lemma **??** while avoiding materializing $\mathbf{A}$ and $\mathbf{P}$ for scalability.

Algorithm 1 Calculate Semantic Association

Input: query attribute a, bipartite matrix M, restarting probability c, tolerant threshold ϵ
Output: $y = x^n$
$\mathbf{q}_a \Leftarrow \mathbf{0}$
$\mathbf{q}_a(a) = 1$ (set a-th element of $\mathbf{q}_a$ to 1)
while $|\Delta \mathbf{u}_a| > \epsilon$ **do**

$$\mathbf{u}_a = (1-c)\begin{bmatrix} normc(\mathbf{M})\mathbf{u}_a(k+1:k+n); \\ normc(\mathbf{M}^T)\mathbf{u}_a(1:k) \end{bmatrix} + c\mathbf{q}_a$$

end while
return $\mathbf{u}_a(1:k)$

4. EVALUATION AND RESULTS

We use two datasets to evaluate our approach. In the *shopping cart* dataset, our goal is to verify that associated items make sense. In the *healthcare* dataset, our goal is to evaluate the scalability of our methods using relationships other than just *is_a* and our preliminary results also show promise. See Table **??**.

4.1 Shopping Cart Dataset

4.1.1 Data

The shopping cart dataset contains purchase information on 100 grocery items for 2,127 shopping orders. The data tuples can be represented as 8,481 RDF statements. We introduce a small ontology to organize the grocery items into a subsumption hierarchy (see Figure **??** for example) with 28 internal nodes. Since the 100 grocery items are mostly at the leaf level, this results in a total of 127 new RDF statements to incorporate the ontology with the data.

4.1.2 Results

Items associated with the query term "toothbrush" are ranked by the strength of semantic association in Table **??**. When the ontology is ignored (Table **??** (A)), associated items are either hub nodes (with many edges linking to other items) or frequently co-occur with the query item, as in traditional association rule mining over transactional data. Conversely, using only the ontology graph (Table **??** (B)) returns the neighborhood of terms in the rdfs:subClassOf lattice. Combining both the ontology and data (Table **??** (C–E)), yields a variety of mixtures of these associations. In a rough sense, it conforms to the ratio of the size of ontology graph and data graph as well (see Table **??**).

Because we created the ontology to illustrate our point, we filter out all terms exclusive to the ontology in Table **??** to elicit top items associated with the query term "soup." The pair *"soup"-"cold remedies"*) (ranked 34th with data graph, not shown in the table) is one that we can find, which makes sense but never co-occurs in the data alone and would not be picked up by traditional methods.

4.2 Shopping Cart

4.2.1 Dataset

The shopping cart dataset is the same as we used in the case study of Chapter **??**. It contains purchase information on 100 grocery items (represented by boolean column headers) for 2,127 shopping orders (corresponding to tuples) from a Foodmart. We first construct an RDF bipartite graph from the dataset by transforming the table to 8481 RDF statements.

Besides, we manually create an ontology to organize the grocery items into a subsumption hierarchy. In this process, we introduce 28 parent nodes (the 100 grocery items appeared in the data are mostly at the leaf level) from which derive a total of 127 RDF statements. As the size of this dataset is fairly small, the calculation of similarity ranking for a given term is fast. In the following we highlight the effect of incorporation of ontology by comparing results obtained with and without ontologies.

4.2.2 Results

In Table **??**, results of items ranked by the strength of semantic association with regard to a query term "Toothbrush" under various combinations of parameters are demonstrated side-by-side for comparison. We first show the result ranked by co-frequency in Table **??**(A) as a baseline. Then, we observe that without using ontology, performing random walk with restart on the data graph (Table **??**(A)) starting from "toothbrush" yields similar results to the work reported in [**?**] based on random walk commute time similarity. Items ranked high in this setting where only the data graph is considered are typically either hub nodes (with many edges linking to other items) or co-frequent with the query item (many edges connecting them). Second, applying the same similarity ranking method solely on the ontology graph (Table **??**(C)) gives a list of association based on the graph-configuration of the ontological structure (in this case, the rdfs:subClassOf lattice). The items that are considered most similar to the query term "Toothbrush" is its immediate parent class "PersonalHygiene," followed by some most derived classes at the same level of "PersonalHygiene" and then siblings of "Toothbrush" itself. Next, Table **??**(D)–(F) demonstrate the results of mining on the combined graph with different ratios of weights assigned to ontology edges and data edges respectively. It is obvious that these results can be seen as a mix of the data-only and ontology-only results with various emphasis on the data or ontology. We can observe that when $w_o/w_d = 20$ the ontology and data appear to have equal significance in determining the ranking (w_o is the weight of ontology edges (i.e., rdfs:subClassOf) and w_d is the weight of data edges). In a rough sense, it conforms to the ratio of the size of ontology graph and data graph as well (see Table **??**). In reality, the appropriate ratio for the edge weights is not only dependent on the size of graphs but also the specific configuration of the graph (depth, average degree, etc). Moreover, specifying the ratio of prior knowledge in ontologies and inductive evidences in data that one wants to employ for discovering new patterns is a highly empirical process. Multiple pilot trials may need to be carried out for the optimal ratio before it is applied to the real application.

We notice that without any filtering on the ranked semantic associations from the combined graph, the list includes items that never appear in the transactional data. This is because typically the semantic annotation process links table attributes to their most specific matching concept in the ontology which are close to the leaf level. The incorporation

	# RDF triples	# *is_a* relationships	# other relationships
Shopping cart	8,481	127	0
Healthcare	10,000,257	1,048,604	43,780

Table 2: The *shopping cart* and *healthcare* datasets vary in size of data, size of ontology, and in the kinds of relationships defined by the ontology.

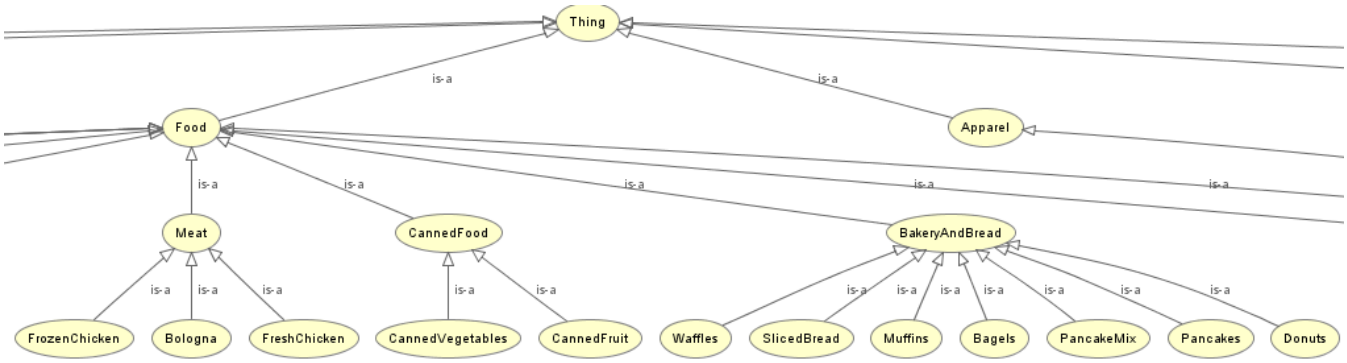


Figure 5: A portion of the manually created shopping cart ontology.

of ontology is to aid the mining process, therefore including in the result those parent nodes (e.g.,, "PersonalHygiene") that never appear in the data is counterintuitive. To overcome this, we can simply filter out those items exclusive to the ontology. Table **??** shows an example of filtered result given a query term "soup." The co-frequency of items are also listed for comparison.

4.3 Electronic Health Records

4.3.1 Dataset

In our second evaluation, we analyzed the electronic health records of real patients. The patient clinical note data are from Stanford Hospital's Clinical Data Warehouse (STRIDE). These records archive over 17-years worth of patient data comprising of 1.6 million patients, 15 million encounters, 25 million coded ICD9 diagnoses, and a combination of pathology, radiology, and transcription reports totaling over 9 million clinical notes (i.e., unstructured text). We obtained the set of drugs and diseases for each patient's clinical note by using a new tool, the *Annotator Workflow*, developed at the National Center for Biomedical Ontology (NCBO), which annotates clinical text from electronic health record systems and extracts disease and drug mentions from the electronic health records.

From this set of 1.6 million patients with annotated records, we vectorize texts and turned them into a huge bag-of-word representation, from which an RDF bipartite graph is constructed (including 148 million RDF statements, see Table **??**). we applied our algorithms to all previous records in the patient's timeline, looking at just the set of drugs and their semantically related diseases. Therefore, at a very simplistic level, the experiment result shows that strong semantically associated items in this context could possibly represent sets of drugs that could lead toward certain diseases.

One strength of the Annotator is the highly comprehensive and interlinked lexicon that it uses. It can incorporate the entire NCBO BioPortal ontology library of over 250 ontologies to identify biomedical concepts from text using a dictionary of terms generated from those ontologies. Terms from these ontologies are linked together via mappings. For this study, we specifically configured the workflow to use a subset of those ontologies that are most relevant to clinical domains, including Unified Medical Language System (UMLS) terminologies such as SNOMED-CT, the National Drug File (NDFRT) and RxNORM, as well as ontologies like the Human Disease Ontology. The resulting set of ontologies contains 1 million subsumption statements.

To highlight the capability of our method for incorporating multiple types of relationships, we also explore the "may_treat" relationship between drugs and diseases defined in NDFRT, for example, Thiabendazole may_treat Larva Migrans. Since we are interested in learning the interaction between drugs and diseases, may_treat is naturally a better indicator relationship to include while mining semantic associations than the subsumption relationship. Our results below illustrate this point.

4.3.2 Results

Before studying the drug-disease association, we carried out a similar test to that on the shopping cart dataset, in which we focus on studying the drug-drug and disease-disease association. To this purpose, we combine the subsumption hierarchy in the ontology graph with the data graph. Table **??** shows the ranked semantic association for the query term "Rofecoxib" (an active ingredient of some anti-inflammatory drugs) given different weight configuration to combine graphs. Without any preprocessing and prior knowledge about how the clinical notes are prescribed, the incorporation of subsumption relationship can be seen as a mean for denoising and enhancement of the data. Given the ratio of the size of

$w_o = 0$, $w_d = 1$		$w_o = 1$, $w_d = 0$		$w_o = 1$, $w_d = 1$		$w_o = 10$, $w_d = 1$	
item	p(%)	item	p(%)	item	p(%)	item	p(%)
Soup	0.42	PersonalHygiene	12.55	PersonalHygiene	0.74	PersonalHygiene	3.97
Cookies	0.41	Snack	0.86	Soup	0.41	NasalSprays	0.41
NasalSprays	0.38	Health	0.64	Cookies	0.4	Soup	0.34
Popcorn	0.32	Sponges	0.57	NasalSprays	0.37	Cookies	0.34
PaperWipes	0.29	Soap	0.57	Popcorn	0.31	Mouthwash	0.3
FrozenVegetables	0.29	Shampoo	0.57	FrozenVegetables	0.29	Popcorn	0.25
PersonalHygiene	0.26	NasalSprays	0.57	PaperWipes	0.28	FrozenVegetables	0.24
DriedFruit	0.25	Mouthwash	0.57	DriedFruit	0.25	PaperWipes	0.23
Milk	0.25	Conditioner	0.57	Milk	0.25	DriedFruit	0.22
Mouthwash	0.24	MealCourse	0.54	Mouthwash	0.23	Milk	0.21
(A)		(B)		(C)		(D)	

Table 3: Foodmart items ranked by the strength of semantic association to the query term "toothbrush." The ranking, $p(\%)$, denotes the steady-state probability.

$w_o = 0$, $w_d = 1$				$w_o = 1$, $w_d = 0$			
item	p(%)	item	p(%)	item	p(%)	item	p(%)
Cheese	0.38	Preserves	0.19	TVDinner	0.46	Sponges	0.06
Cookies	0.32	Juice	0.17	Pizza	0.46	Soap	0.06
DriedFruit	0.32	Lightbulbs	0.17	Pasta	0.46	Shampoo	0.06
Wine	0.24	PaperWipes	0.16	HotDogs	0.46	NasalSprays	0.06
CannedVegetables	0.23	Pizza	0.16	Hamburger	0.46	Mouthwash	0.06
FrozenVegetables	0.23	Nuts	0.16	FrenchFries	0.46	Conditioner	0.06
Cereal	0.22	Popcorn	0.16	DeliSalads	0.46	Ibuprofen	0.06
Milk	0.22	Chips	0.16	DeliMeats	0.46	ColdRemedies	0.06
ChocolateCandy	0.19	Eggs	0.16	Sunglasses	0.07	Aspirin	0.06
Waffles	0.19	TVDinner	0.15	Toothbrushes	0.06	Acetominifen	0.06

Table 4: Foodmart items ranked by the strength of semantic association to the query term "soup." The ranking, $p(\%)$, denotes the steady-state probability. Terms exclusive to the Foodmart ontology are filtered out.

the ontology to the size of data, the data graph in this test is more dominant in determining the ranking than in the shopping cart experiment. One can gradually change the ratio of w_o to w_d to strike a balance and achieve the optimal result.

To verify the drug–disease association and study the impact of different semantic relationships on finding such association, we carry out the following experiment. Table **??** illustrates the rankings of three associations (one per row) under different settings. The first element in the pair is the query item, which are all active ingredients of some prescription drugs, and the ranking shown in the table is for the second item, which are diseases. For example, arthritis is ranked as the 527th semantically associated item to Rofecoxib according to similarity ranking based only on data graph. All these item pairs are actually gold standard associations backed by known drug–disease relationships, we know the strength of semantic associations between them should be strong.

We observe that the ranking based on data graph alone is fairly high already, consider there are approximately 1 million concepts of interest. However, the results based on the combination of data and subsumption ("isa") graph are worse. It is because the subsumption hierarchies for drugs and diseases are largely separate structures. Therefore the subsumption relationships can only boost the association within the drug and disease hierarchies respectively, but obfuscate the cross-hierarchy associations that we aim to find between drugs and diseases. On the other hand, however, the association between these pairs can be exactly captured by the NDFRT "may_treat" relationship (e.g.,, NDFRT explicitly defines that Rofecoxib may_treat arthritis). When the "may_treat" graph is incorporated into the mining process, the ranking for the association is greatly boosted.

Conversely, we are also interested in learning whether the data graph can help discover patterns in the ontology graph. Figure **??** (left) shows a subgraph of the NDFRT "may_treat" relationship. Rofecoxib is asserted to treat two diseases, namely, dysmenorrhea and degenerative polyarthritis. And there are altogether 116 and 200 drugs that are known to treat dysmenorrhea and degenerative polyarthritis respectively (hence the in-degrees of the nodes). Applying our method on this graph with the query term "Rofecoxib" yields a similarity-ranked list having degenerative polyarthritis and dysmenorrhea as the top two items. Since this result is the exact ground truth, there is no improvement to be made with the incorporation of the data graph. Therefore, we alter the ground truth graph with some deliberately distorted information, as is shown in Figure **??** (right), so that the may_treat graph alone produces only inferior result. More specifically, we specify that Rofecoxib should treat hypertensive disease, the very diseases that is asserted to be treated by the most drugs (a total of 619). Then we add an imaginary drug to treat degenerative polyarthritis, dysmenorrhea, and hypertensive disease. In this way, the original direct connections between Rofecoxb and degenerative polyarthritis and dysmenorrhea become erroneously indirect and are obfuscated by some the noise of high degree nodes along the path. With this scenario, we hope to learn if the incorporation of data graph can correct the misinformation in ontologies.

ranked by co-frequency		w/ data only		w/ ontology only	
item	freq	item	p(%)	item	p(%)
PaperWipes	8	Soup	0.42	PersonalHygiene	12.55
Popcorn	7	Cookies	0.41	Snack	0.86
Soup	6	NasalSprays	0.38	Health	0.64
NasalSprays	6	Popcorn	0.32	Sponges	0.57
Cookies	6	PaperWipes	0.29	Soap	0.57
Spices	5	FrozenVegetables	0.29	Shampoo	0.57
Soda	4	PersonalHygiene	0.26	NasalSprays	0.57
Shrimp	4	DriedFruit	0.25	Mouthwash	0.57
FlavoredDrinks	4	Milk	0.25	Conditioner	0.57
Dips	4	Mouthwash	0.24	MealCourse	0.54
(A)		(B)		(C)	

$w_o = 1$, $w_d = 1$		$w_o = 10$, $w_d = 1$		$o_w = 20$, $o_d = 1$	
item	p(%)	item	p(%)	item	p(%)
PersonalHygiene	0.74	PersonalHygiene	3.97	PersonalHygiene	6.27
Soup	0.41	NasalSprays	0.41	NasalSprays	0.5
Cookies	0.4	Soup	0.34	Mouthwash	0.41
NasalSprays	0.37	Cookies	0.34	Shampoo	0.31
Popcorn	0.31	Mouthwash	0.3	Soup	0.29
FrozenVegetables	0.29	Popcorn	0.25	Cookies	0.29
PaperWipes	0.28	FrozenVegetables	0.24	Sponges	0.28
DriedFruit	0.25	PaperWipes	0.23	Health	0.27
Milk	0.25	DriedFruit	0.22	Conditioner	0.27
Mouthwash	0.23	Milk	0.21	Soap	0.25
(D)		(E)		(F)	

Table 5: Foodmart items ranked by the strength of semantic association (i.e., $p(\%)$, the steady-state probability), given the query term "Tooth Brush."

w/ data only						w/ onto only					
item	p(%)	freq	item	p(%)	freq	item	p(%)	freq	item	freq	p(%)
Cheese	0.38	98	Preserves	0.19	65	TVDinner	0.46	40	Sponges	21	0.06
Cookies	0.32	96	Juice	0.17	47	Pizza	0.46	46	Soap	0	0.06
DriedFruit	0.32	87	Lightbulbs	0.17	47	Pasta	0.46	29	Shampoo	34	0.06
Wine	0.24	63	PaperWipes	0.16	55	HotDogs	0.46	30	NasalSprays	21	0.06
CannedVegetables	0.23	67	Pizza	0.16	46	Hamburger	0.46	19	Mouthwash	28	0.06
FrozenVegetables	0.23	79	Nuts	0.16	60	FrenchFries	0.46	37	Conditioner	12	0.06
Cereal	0.22	56	Popcorn	0.16	39	DeliSalads	0.46	31	Ibuprofen	18	0.06
Milk	0.22	53	Chips	0.16	46	DeliMeats	0.46	37	ColdRemedies	33	0.06
ChocolateCandy	0.19	16	Eggs	0.16	51	Sunglasses	0.07	12	Aspirin	22	0.06
Waffles	0.19	51	TVDinner	0.15	40	Toothbrushes	0.06	13	Acetominifen	12	0.06

Table 6: Semantically associated items for the query term "Soup," by filtering out those items exclusive to the Foodmart ontology.

Table **??** shows the result of ranks of the associations between Rofecoxib and degenerative polyarthritis and dysmenorrhea. The ranks of the associations drastically drop to the 555th and 246th respectively on the noisy graph from the top two on the original ground truth graph. This is mainly due to the large node, hypertensive disease, in the middle of the connections. However, with the combined data and may_treat graph, we notice that the rank of Rofecoxib and degenerative polyarthritis increases to 263rd, while the rank of Rofecoxib and dysmenorrhea decreases to 1703rd. This shows that the data graph endorses more strongly the association between Rofecoxib and degenerative polyarthritis. Indeed, although Rofecoxib are known to treat both degenerative polyarthritis and dysmenorrhea, the former is a much more popular usage. A search on the National Library of Medicine's PubMed database[1] for "Rofecoxib and polyarthritis" returns 518 results, while "Rofecoxib and dysmenorrhea" only returns 29. This result shows that the data graph can help correct misinformation in ontologies to some extent, and in a sense, it also gives a clue of how prior beliefs fit with reality.

5. DISCUSSION

We have demonstrated that using the proposed combined RDF bipartite graph incorporates both domain knowledge and data based on the user's desired weights for each component for finding *semantically associated itemsets*.

Developing scalable algorithms for semantic data mining is critically important. In our work, the healthcare dataset has grown beyond 100 billion triples and the size of the ontologies used are also large (SNOMED-CT has nearly 400,000 classes). The RWR method we describe works well for query-based node similarity, but it will not scale to generate full pair-wise node similarities at this tremendous scale because a calculation of eigenvectors of the Laplacian is required to derive the similarity measures, which is very expensive on large graphs.

[1] `http://www.ncbi.nlm.nih.gov/`

rank	w/ data only	w/ onto only	$w_o = 10000, w_d = 1$
1	reflux	valdecoxib	reflux
2	medical history	meloxicam	obstruction
3	history of previous events	celecoxib	injury
4	diagnosis	parecoxib	valdecoxib
5	pharmaceutical preparations	etoricoxib	medical history
6	blood and lymphatic system disorders	deracoxib	foreign body sensation
7	disease	lumiracoxib	history of previous events
8	infantile neuroaxonal dystrophy	firocoxib	adverse effects
9	today	nabumetone	celecoxib
10	hypersensitivity	macrolides	actual hypothermia

Table 7: Results of Health items ranked by the strength of semantic association, given the query term "Rofecoxib."

	w/ data only		w/ data and "isa"		w/ data and "may_treat"	
	p(%)	rank	p(%)	rank	p(%)	rank
$\langle Rofecoxib, degenerative\ polyarthritis \rangle$	0.006	527	0.004	632	0.51	13
$\langle valdecoxib, degenerative\ polyarthritis \rangle$	0.007	613	0.005	695	0.63	17
$\langle troglitazone, diabetes \rangle$	0.006	478	0.005	514	0.44	11

Table 8: Rankings of three semantic associations in health data under different settings.

While non-trivial practical problems associated with very large graphs cannot be completely avoided, our work makes it possible to leverage decades of work on graph-based methods in this effort to mine semantically associated data. The general strategy going forward is to employ approximation and develop parallelizable algorithms. Lin and Cohen [**?**] proposed an approximation to a eigenvalue-weighted linear combination of all the eigenvectors, which can be achieved by performing a small number of matrix-vector multiplications. The procedure results in a more scalable method called *power iteration clustering* that finds a very low-dimensional data embedding using truncated power iteration on a normalized pair-wise similarity matrix of the data points. Zhao et al. [**?**] described the idea of *graph coordinate systems*, which provides a fast embedding of large graphs into a hyperbolic space. The method parallelizes easily and efficiently locate shortest paths between node pairs, which relates well to the notion of commute time distance, which our RWR method seeks to elicit. Savas and Dhillon [**?**] introduced a novel framework called *clustered low rank matrix approximation for massive graphs*. After a few intermediate steps, they are able to finally project an a optimal, low rank approximation of the entire graph, thus including connections or edges between vertices from different clusters. We intend to extend these ideas to ontology-annotated hypergraphs.

The weighted hyperedges provide a great deal of flexibility to users who may prefer domain knowledge over data (or vice versa) and opens up new research questions on how to optimally configure learning algorithms. In reality, the appropriate ratio for the edge weights is not only dependent on the size of graphs but also the graph configuration (depth, average degree, etc). Moreover, allowing users to specify the ratio of prior knowledge in ontologies versus inductive evidence from data enables us to discover empirically optimal configurations. We have performed exhaustive feature selection on classification algorithms for our healthcare dataset in the past, and we would also explore a few permutations on these hyperedge weights going forward. We can draw upon other works, such as, Tian et al. [**?**], who proposed a semi-supervised approach for classifying nodes in a graph based on a relatively small labeled set.

The RDF bipartite graph representation has limited expressivity compared to OWL itself. For example, domain, range and cardinality constraints are not straightforward to model. One possible approach is to model domain constraints by explicitly describing the desired or acceptable walk (traversal sequence) in the RDF hypergraph. In this case, the recently proposed *regular traversal expression* [**?**] technique may apply. However, their fast power-iteration approach for computing the stationary probability may not be applicable any more due to the label sequence constraint, but the Monte-Carlo simulation of the random walk may help to approximate the similarity measure.

We have showcased the utility of the RDF bipartite graph in mining *semantically associated itemsets* and will explore other data mining tasks as well. For example, the the classification task can be formulated as discovering the correct labels for unlabeled vertices in the weighted bipartite graph. Conceptually, the pair of nodes being "close to" one another shall share the same labels, which are consistent to one basic principle of semi-supervised learning, so the challenge is to define the closeness between two nodes. The clustering task can be viewed as a neighborhood formation for vertices on the data partition. Basically, for any closely related nodes, we group them together. Again, with an explicit similarity measure, possibly even the classical k-means in this case, could be directly applied to cluster the vertices.

6. CONCLUSION AND FUTURE WORK

We approach the discovery of indirectly associated items from ontology-annotated data, called *semantically associated itemsets*, by proposing a new semantic data mining technique. Our method uses hypergraphs and random walks with restart over a bipartite graph serialization to discover associations that cannot be found by methods that rely on co-frequent items because it utilizes both the ontology and the data at the same time. Moreover, we allow users the ability to customize the weight of each component, giving flexibility in how strongly the role of the ontology plays over the

	w/ noisy may_treat only		w/ data and noisy may_treat	
	p(%)	rank	p(%)	rank
$\langle Rofecoxib, degenerative\ polyarthritis \rangle$	3.60e-3	555	8.14e-3	263
$\langle Rofecoxib, dysmenorrhea \rangle$	1.54e-2	246	1.26e-3	1703

Table 9: Rankings of associations on the noisy may_treat graph (Figure ?? right) between Rofecoxib and two diseases derived with and without data.

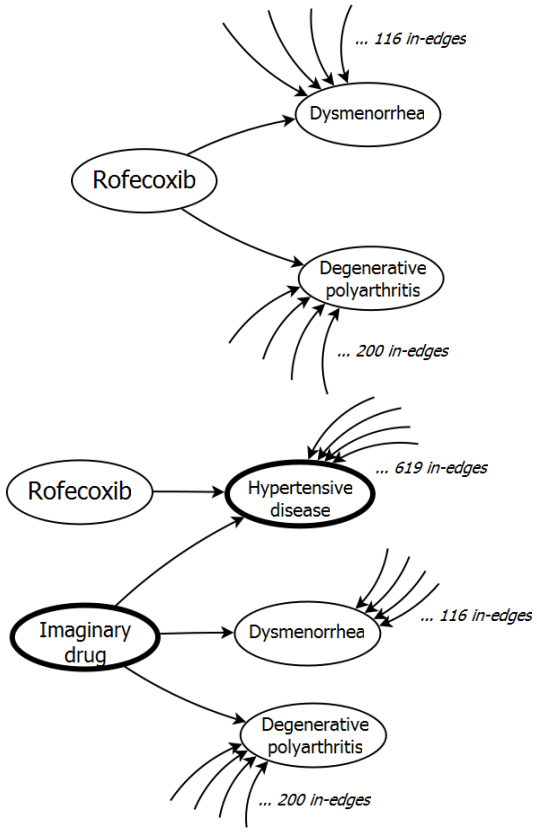


Figure 6: The may_treat subgraph before and after distortion: The left-hand side of the figure shows the may_treat subgraph of ground truth relationships between the drug Rofecoxib and two diseases. The right-hand side shows the may_treat subgraph with some deliberately distorted information.

data, or vice-versa. Our evaluations show that the method discovers indirect associations and that it scales to datasets that are large in multiple says: the ontology can be large and the data can be large.

In future work, we will explore algorithms that suggest appropriate weights to apply to the components of the hypergraph. Also, we will implement methods for clustering and classification tasks within this framework. We will focus on our healthcare datasets mainly because they are both large and complex, but also because of the enormous potential in advancing the state-of-the-art in clinical informatics and improving the quality of care for millions of patients.