Automatic Metadata Generation Using Associative Networks

MARKO A. RODRIGUEZ
JOHAN BOLLEN
and
HERBERT VAN DE SOMPEL
Los Alamos National Laboratory

In spite of its tremendous value, metadata is generally sparse and incomplete, thereby hampering the effectiveness of digital information services. Many of the existing mechanisms for the automated creation of metadata rely primarily on content analysis which can be costly and inefficient. The automatic metadata generation system proposed in this article leverages resource relationships generated from existing metadata as a medium for propagation from metadata-rich to metadata-poor resources. Because of its independence from content analysis, it can be applied to a wide variety of resource media types and is shown to be computationally inexpensive. The proposed method operates through two distinct phases. Occurrence and cooccurrence algorithms first generate an associative network of repository resources leveraging existing repository metadata. Second, using the associative network as a substrate, metadata associated with metadata-rich resources is propagated to metadata-poor resources by means of a discrete-form spreading activation algorithm. This article discusses the general framework for building associative networks, an algorithm for disseminating metadata through such networks, and the results of an experiment and validation of the proposed method using a standard bibliographic dataset.

Categories and Subject Descriptors: H.3.1 [**Content Analysis and Indexing**]; H.3.7 [**Digital Libraries**]

General Terms: Algorihms, Experimentation

Additional Key Words and Phrases: Keywords: Associative networks, particle-swarms, metadata generation

This research was financially supported by the Research Library at the Los Alamos National Laboratory.

Author's address: M. A. Rodriguez, STB-RL: Digital Library Research and Prototyping Team, Los Alamos National Laboratory, Los Alamos, NM87545; email: marko@lanl.gov.

© 2009 ACM 1046-8188/2009/02-ART7 $5.00 DOI 10.1145/1462198.1462199 http://doi.acm.org/10.1145/1462198.1462199

ACM Reference Format:
Rodriguez, M. A., Bollen, J., and Van De Sompel, H. 2009. Automatic metadata generation using associative networks. ACM Trans. Inform. Syst. 27, 2, Article 7 (February 2009), 20 pages. DOI = 10.1145/ 1462198.1462199 http://doi.acm.org/10.1145/1462198.1462199

1. INTRODUCTION

Resource metadata plays a pivotal role in the functionality and interoperability of digital information repositories. However, in spite of its value, high-quality metadata is difficult to come by [Duval et al. 2002]. Ward [2003] demonstrates that although as many as 15 possible metadata properties can theoretically be included in the widely used Dublin Core standard,[1] few are frequently used in collections whose metadata are generally created by the authors themselves.[2] The problem of poor and incomplete metadata is expected to worsen as repositories are applied to materials collected beyond the traditional, centralized methods of publication and start to obtain data from Web pages, blogs, personal multimedia collections, and collaborative tagging environments.

Metadata is a costly resource to create, maintain, and/or recover manually. There has therefore been significant research on automated metadata generation (e.g., by extracting metadata from the content of resources). Natural language processing [Yang and Lee 2005] and document image analysis techniques [Giurida and Shek 2000; Sebastiani 2002; Han et al. 2003; Mao et al. 2004] may extract keywords, subject categories, author, and citations (e.g., CiteSeer[3]) from manuscripts. Furthermore, in Greenberg [2004], two metadata generators are demonstrated that successfully harvest and extract metadata from existing resource source and content. Such content-based techniques are much less efficient for multimedia resources, for example, video, music, images, and datasets. Reliable content analysis for such data is still an active research area and existing methods generally yield little content-related metadata. In addition, content-based approaches can be prohibitively expensive in computational terms [Kuwano et al. 2004].

For the reasons outlined above, methods for the generation of metadata that do not rely on resource content have generated considerable interest. The recent growth in applications of “folksonomies” (i.e., community-based “tagging” [Mathes 2004; Golder and Huberman 2006]), has been, to some extent, inspired by the shortcomings of existing metadata generation methods. Unfortunately, human tagging works well only in situations where the number of participants greatly exceeds the number of resources to be tagged and where there is no requirement for controlled vocabularies or standardized metadata formats.

In this article, we propose a system for automated metadata generation that starts from a common scenario: a heterogeneous repository contains resources for which varying degrees of metadata are available. Some resources have been imbued with rich, vetted metadata, whereas others have not. However, if it can

[1] The Dublin Core 1.1 specification is available at `http://dublincore.org/documents/1999/07/02/dces/`

[2] The most frequently used are *creator*, *identifier*, *title*, *date*, and *type*.

[3] CiteSeer available at http://citeseer.ist.psu.edu/

be assumed that resources that are "similar" (e.g., similar in publication venue, authorship, date, citations) are more likely to have shared metadata, then the problem of metadata generation can be reformulated as one of extrapolating metadata from metadata-rich to related, but metadata-poor resources. This article's experiment focuses on identifying which aspects of metadata similarity are best used to extrapolate resource metadata in a bibliographic dataset.

As a case in point, Naaman et al. [2005] describe a method to support the annotation of personal photograph collections. Once a user has annotated a photograph its metadata is automatically transferred to photographs taken at similar times and locations. For example, a user photographs a group of friends at 3:45 P.M. Another photograph is made at 3:47 P.M. Since the second photograph was taken only two minutes after the first, it is likely that it depicts a similar scene. The system therefore transfers metadata from photograph 1 to photograph 2. Similarly, Prime-Claverie et al. [2005] propose a method of Web page metadata propagation using cocitation networks. The general idea is that if two Web pages cite other Web pages in common, then the probability that they share similar metadata is higher. The user can later correct and augment any transferred metadata.

The mentioned systems are strongly related to collaborative filtering [Herlocker et al. 2004]. Collaborative filtering systems are commonly employed in online retail systems to recommend items of interest to individual users. Using the principle that similar users are more likely to appreciate similar items, users are recommended items that are missing from their profiles but occur in the profiles of similar users. The collaborative filtering process can thus be regarded as an instance of metadata propagation. If users are considered resources and their profiles are considered resource metadata, it can be said that collaborative filtering systems "recommend" metadata from one resource to another based on resource similarity.

A generalization of the above metadata propagation systems can be made in terms of the following elements:

(1) A mechanism to generate resource relations, that is, assess their similarity.
(2) The determination of a metadata-rich subset of the repository's collection that can serve as a reference set.
(3) A means of propagating metadata from the metadata-rich reference set to a metadata-poor subset of the collection using the established resource relations as a substrate.

Such systems for the generation of metadata can be said to operate on a "Robin Hood" principle; they take from metadata-rich resources and give to metadata-poor resources, with the exception that metadata is not a zero-sum resource. This mode of operation has a number of desirable properties. First, it reduces the need for the costly generation of metadata; metadata is automatically extrapolated from an existing metadata-rich reference collection to a metadata-poor subset. Second, resource relations can be defined independent of content and metadata extrapolation can thus be implemented for wide range of heterogeneous resources, for example, audio, video, and images.

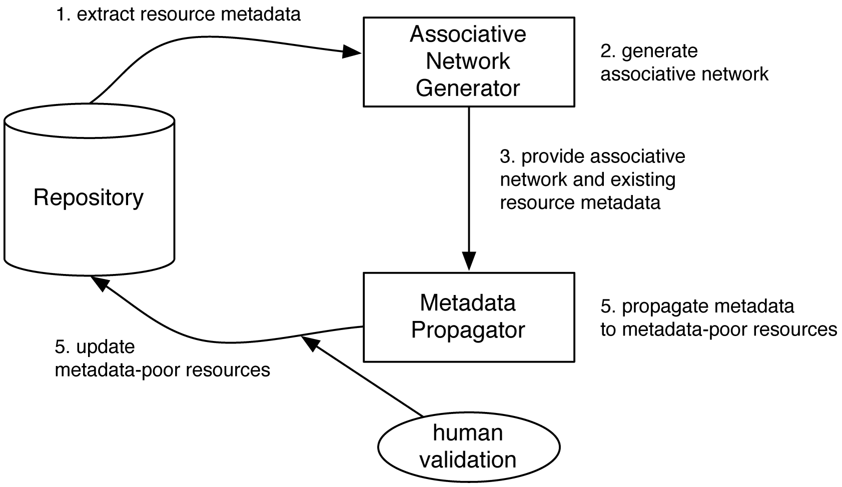

1. extract resource metadata
Associative Network Generator
2. generate associative network
3. provide associative network and existing resource metadata
Repository
Metadata Propagator
5. propagate metadata to metadata-poor resources
5. update metadata-poor resources
human validation

Fig. 1. System outline.

This article outlines a proposal for a metadata propagation system designed for scholarly repositories that takes advantage of the multiple means by which two resources can be related (e.g. cocitation, citation, coauthor, cokeyword, etc.). Figure 1 presents the outline of the proposed system's components and processing stages. First, resource metadata is extracted from the collection of a repository. Second, an associative multirelational network (i.e., a directed labeled graph) of resource relations is derived from a subset of the available metadata. Third, a metadata-rich subset of the collection is selected to serve as a reference data set. Fourth, and finally, metadata is propagated (i.e., extrapolated) from the metadata-rich reference set to all other metadata-poor resources over the associative network of resources after which the repository is updated. Human validation can vet the results of the metadata extrapolation before insertion into the repository occurs.

It is important to emphasize that this system requires the existence of some preliminary metadata both for the construction of resource relations and for metadata propagation. Furthermore, the quality or accuracy of the preliminary metadata is important in ensuring successful results (i.e., to avoid a "garbage in, garbage out" scenario). However, the metadata being propagated can be different from the metadata used to generate resource relations. For instance, in the manuscript domain, the propagation of keyword metadata may be most efficient along resource relations derived from citation metadata. Therefore, two aspects affect the efficiency of metadata propagation: the type of resource relations and the algorithm used to propagate metadata. It is important to note that no new metadata values are created in the model proposed in this article. While it is important for resources to maintain metadata, this method only propagates preexisting metadata values and thus does not increase the discriminatory aspects that metadata should and generally does provide. While like resources should have similar metadata, variations should also exist to make sure that a resource's metadata accentuates the unique characteristics of the resource.

This article will first discuss two algorithms to define sets of resource relations and represent these relations in terms of associative networks. It will then formally define a metadata propagation algorithm that can operate on the basis of the generated resource relations. Finally, the proposed metadata generation system is validated using a modified version of the KDD Cup 2003 High-Energy Physics bibliographic dataset (hep-th 2003).[4] While it is theoretically possible for this method to work on other resource types (e.g., video, audio) as it doesn't require an analysis of the content of the resources, only their metadata, it is only speculated that the results of such a method would be viable in these other, nontested, domains.

2. CONSTRUCTING AN ASSOCIATIVE NETWORK OF REPOSITORY RESOURCES

An associative network is a network that connects resources according to some measure of similarity. An associative network is represented by the data structure $G = (N, E, W)$ where N is the set of resources, $E \subseteq N \times N$ the set of directed relationships between resources, and W is the set of weight values for all edges such that $|W| = |E|$. Any edge $e_{i,j,\mu}$ with corresponding weight $w_{i,j,\mu}$ expresses that there exists a directed weighted relationship constructed using properties of type μ from resource n_i to resource n_j. The explicit representation of μ is necessary because an associative network can be constructed according to different properties (i.e., authorship, citations, keywords, etc.). As will be demonstrated, certain network μ relationships are better (in terms of precision and recall) at propagating certain property types than others.

The remainder of this section will describe two associative network construction algorithms. One is based on *occurrence* metadata where a resource is considered similar to another if there is a direct reference from one resource to the other (e.g., a direct citation). The other algorithm is based on *cooccurrence* metadata and thus, considers two resources to be similar if they share similar metadata. That is, two resources are deemed similar if the same metadata values occur in both their properties (i.e., same authors, same keywords, same publication venue, etc.). Depending on how the repository represents its metdata some property types will be direct reference properties and others will have to be infered through indirect, cooccurence algorithms.

2.1 Occurrence Associative Networks

An associative network can be constructed if direct references connect one resource to another. The World Wide Web, for instance, is an associative network based on occurrence data because a Web page makes a direct reference to another Web page via a hyperlink (i.e., the *href* HTML tag). For manuscript resources, occurrence information usually exists in citations. For instance, if resource n_i references (i.e., cites) resource n_j then their exists an edge $e_{i,j,\text{cite}}$. One potential algorithm for determining the edge weight is to first determine how many other citations resource n_i currently maintains. That is, if resource n_i also

[4] hep-th 2003 available at: `http://www.cs.cornell.edu/projects/kddcup/datasets.html`

Table I. Keyword Metadata for Resources n_i and n_j

Resource	Keyword-1	Keyword-2	Keyword-3
n_i	**repository**	**metadata**	particle
n_j	images	**repository**	**metadata**

cites 50 other resources then resource n_i is $\frac{1}{50}$ as similar to n_j, $w_{i,j,\text{cite}} = \frac{1}{50}$. Similarly, if resource n_i only cites resource n_j, then the strength of tie to resource n_j is greater, $w_{i,j,\text{cite}} = 1.0$. The general equation is defined by Equation (1) where the function meta(n_i, cite) returns the set of all citations for resource n_i. This equation holds only if resource $n_j \in$ meta(n_i, cite). Equation (1) makes use of the μ notation in order to generalize the equation for use with any direct reference property types.

$$w_{i,j,\mu} = \frac{1}{\left|\text{meta}(n_i, \mu)\right|} \ : \ n_j \in \text{meta}(n_i, \mu). \tag{1}$$

The running time of the algorithm to construct an associative network based on direct, occurrence property types is $O(N)$ since each resource must be checked once and only once for direct reference to other resources.

2.2 Cooccurrence Associative Networks

Cooccurrence networks are created when resources share the same metadata property values. For instance, if two resources share the same keyword, author, or citation values then there exists some degree of similarity. For a cooccurrence network the edge weight for any two resources, $w_{i,j,\text{co}\mu}$ and $w_{j,i,\text{co}\mu}$, is a function of the amount of metadata properties of type μ that n_i and n_j share in common. A specific example of this could be a cokeyword associative network created when two resources have similar keywords. For example, suppose the resource nodes n_i and n_j have the list of keyword properties presented in Table I.

In Table I, resource n_i and n_j share two keywords in common, namely *repository* and *metadata*. The edge weight between these two resources is a function of the amount of keywords they share in common, Equation (2), and the size of the keyword count of both resources. Therefore, according to Equation (3), the edges connecting resource n_i to n_j and n_j to n_i have a weight of $w_{n_i,n_j,\text{cokey}} = w_{n_j,n_i,\text{cokey}} = 0.5$.

$$\text{co}(n_i, n_j, \mu) = \text{meta}(n_i, \mu) \cap \text{meta}(n_j, \mu), \tag{2}$$

so that

$$w_{i,j,\text{co}\mu} = w_{j,i,\text{co}\mu} = \frac{|\text{co}(n_i, n_j, \mu)|}{[|\text{meta}(n_i, \mu)| + |\text{meta}(n_j, \mu)|] - |\text{co}(n_i, n_j, \mu)|}. \tag{3}$$

Notice that the cooccurrence algorithm in Equation (3) returns a coμ representation. This means for keyword properties, the returned weight is a cokeyword similarity weight. Similarly, for authorship metadata, the returned weight is a coauthorship weight. The running time of the algorithm to construct a cooccurrence network is $O(\frac{N^2-N}{2})$ since each resource's μ-properties must be checked against every other resource's μ-properties (N^2), except itself ($-N$), once and only once ($\frac{1}{2}$).

3. METADATA PROPAGATION ALGORITHM

Reconstructing the metadata for a metadata-poor collection of resources is dependent not only on the associative network data structure, but also upon the use of a metadata propagation algorithm. The algorithm chosen is a derivative of the particle-swarm algorithm [Rodriguez 2007]. Particle-swarm algorithms are a discrete form of the spreading activation algorithms [Collins and Loftus 1975; Cohen and Kjeldsen 1987; Crestani 1997; Crestani and Lee 2000; Huang et al. 2004]. Because particles are indivisible entities, it is easy to represent metadata properties as being encapsulated inside a particle. These metadata particles are then propagated over the edges of the associative network. Upon reaching a resource node that is missing a particular property type, the particle recommends its property value to the visited resource. This section will formally describe the metadata propagation algorithm before discussing the results of an experiement using a bibliographic dataset.

Every resource node in an associative network is supplied with a single particle, $p_i \in P$, such that $|P| = |N|$. The particle p_i encapsulates all the metadata properties of a particular resource n_i. Therefore, $\text{meta}(n_i, \mu) \equiv \text{meta}(p_i, \mu)$ for all μ. Particle p_i has a reference to its current node $c_i \in N$ such that at $t = 0$, $c_i = n_i$. The particle p_i begins its journey ($t = 0$) at its home node, n_i, and traverses an outgoing edge of n_i. Particle edge traversal is a stochastic process that requires the outgoing edge weights of each node to form a probability distribution. Therefore, the set of outgoing edge weights of relation type μ for n_i, $\text{out}(n_i, \mu)$, must be normalized as represented in Equation (4) and Equation (5). Unlike Equation (3), for cooccurrence edges, these equations do not guarantee that $w_{i,j,\mu} = w_{j,i,\mu}$.

$$w_{i,j,\mu} = \frac{w_{i,j,\mu}}{\sum_{\forall k \in e_{i,k,\mu}} w_{i,k,\mu}} \tag{4}$$

such that

$$\sum_{\forall j \in e_{i,j,\mu}} w_{i,j,\mu} = 1.0. \tag{5}$$

The function $\theta(\text{out}(n_i, \mu))$ is defined such that it takes a set of outgoing edges of relation type μ of node n_i and returns a single node n_j based upon the outgoing edge weight probability distribution, where $e_{i,j,\mu} \in \text{out}(n_i, \mu)$. This is how a particle traverses an associative network.

The particle p_i also has an associated energy value $\epsilon_i \in [0, 1]$. Each time an edge is traversed, the particle p_i decays its energy content, ϵ_i, according to a global decay value, $\delta \in [0, 1]$. Particle energy decay over discrete time t is represented in Equation (6). The rational for decay is based on the intuition that the metadata property values of a particular particle become less relevant the further the particle travels away from its source node (c_i at $t = 0$). Therefore, the further a particle travels in the network, the more that particle's energy value (or recommendation influence), ϵ, is decayed.

$$\epsilon_i(t + 1) = (1 - \delta)\epsilon_i(t). \tag{6}$$

The energy value of a particle defines how much recommendation influence a particle's metadata property values has on a visited metadata-poor node. Each

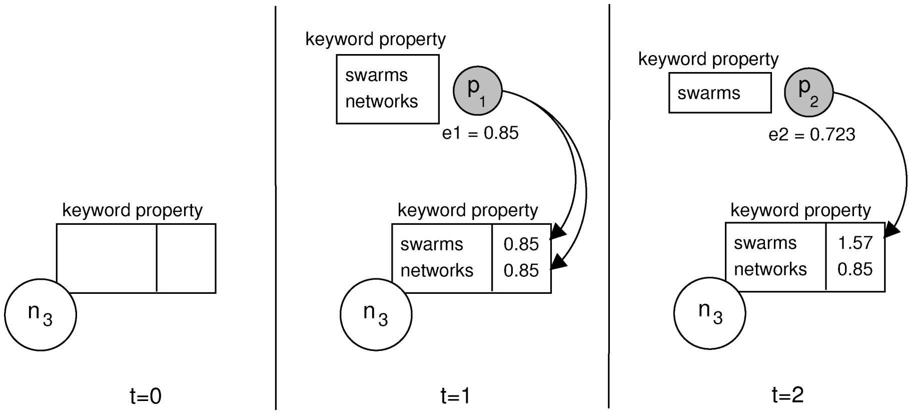


Fig. 2. Particles recommending metadata information to a metadata-poor node.

time a particle traverses a node with missing metadata properties, it not only recommends its metadata property values to that node, but also increments the appropriate property value with its current energy value ϵ_i. In Figure 2, at $t = 0$, before the propagation algorithm has been executed, resource n_3 has no keyword values. Therefore, when particle p_1 reaches n_3 at $t = 1$, particle p_1 recommends its keyword property values (keyword={swarm, algorithms}) to node n_3 with an influence of $\epsilon_1 = 0.85$. At $t = 2$, particle p_2, with $\epsilon_2 = 0.723$, recommends its keyword property (keyword={swarm}) to node n_3. Notice that the recommendation of the keyword property value "swarm" is reinforced each time that property value is presented to n_3.

The function of a single particle, p_i, at a particular node, n_j, is represented in pseudo-code in Algorithm 1 where rec(n_j, μ) returns the set of previous property values to n_j for a property of type μ.

If Algorithm 1 is called recommend Meta(n_j, p_i), then the full particle propagation algorithm can be described by the pseudo-code in Algorithm 2. The process of moving metadata particles through the associative network and recommending metadata-poor nodes metadata property values continues until some desired t is reached or all particle energy in the network has decayed to 0.0, $\sum_{\forall i} \epsilon_i \cong 0.0$.

4. AN EXPERIMENT USING THE 2003 HEP-TH BIBLIOGRAPHIC DATASET

This section will present the results of the proposed metadata generation system when attempting to reconstruct an artificially atrophied bibliographic dataset. By artificially reducing the amount of metadata in the full bibliographic dataset, it is possible to simulate a metadata-poor environment and at the same time still be able to validate the results of the metadata propagation algorithm. The section is outlined as follows. First, the dataset used for this experiment is described. Second, a short review of the validation metrics (precision, recall, and F-score) is presented. Third, the various system parameters are discussed. Finally, the results of the experiment are presented as a validation of the systems use for manuscript-based digital library repositories.

```
Input: recommendMeta($n_j, p_i$)
# $p_i$ updates the metadata of $n_j$ for all property types;
foreach ($\mu-$property) do
    # first ensure that $n_j$ is metadata-poor at the particular $\mu$-property;
    if ($|$meta$(n_j, \mu)| == 0$) then
        # update the metadata-poor node's $\mu$-property with the $\mu$ property value of $p_i$;
        foreach ($x \in$ meta$(p_i, \mu)$) do
            found = $false$;
            # if property value already exists, increment its energy value with $e_i$;
            foreach ($y \in$ rec$(n_j, \mu)$) do
                if ($x ==$ value$(y)$) then
                    energy$(y)$ = energy$(y) + \epsilon_i$;
                    found = $true$;
                end
            end
            # if no recommended value exists, add to $n_j$'s recommendations;
            if (!found) then
                addRec$(x, \epsilon_i) = x$;
            end
        end
    end
end
```

Algorithm 1. Particle p_i recommending metadata properties values to n_j

Further research into other domains besides manuscripts will demonstrate the validity of this method for other resource types.

The dataset used to validate the proposed system is a modified version of the hep-th 2003 bibliographic dataset for high energy physics and theory [McGovern et al. 2003].[5] A modified version of the hep-th dataset, as used in de Lin and Chalupsky [2004], is represented as a semantic network containing manuscripts (29,014), authors (12,755), journals (267), organizations (963), keywords (40), and publication date in year/season pairs (60). These nodes are then connected according to the following semantics:

—writes(a,m): author a wrote manuscript m

—date_published(m,d): manuscript m was published on date d

—organization_of(a,o): author a works for organization o

—published_in(m,j): manuscript m was published in journal j

[5] The details of how the hep-th dataset was created can be found in the disseminated README (`http://kdl.cs.umass.edu/data/hepth/hepth-README.txt`) of the dataset, and a few specifics are quoted here: "Object and link properties such as title, authors, journal (if published), and various dates were extracted from the abstract files with additional date information coming from the slacdates citation tarball. Because institutions were often not presented in a consistent format, the email domain of the submitter (if available) was used as a surrogate for institution. Because many authors had no associated email address, domain information is not available for all authors. Consolidation was performed on journal names, domains, and author names. A nominal amount of hand-cleaning to correct spelling or formatting problems was also performed."

```
Input: propagate($\mu$)
# $\delta$ is a global energy decay value ;
$\delta = 0.15$ ;
# create a particle for each node ;
foreach ($n_i \in N$) do
    $\text{meta}(p_i, \mu) = \text{meta}(n_i, \mu) \;:\; \forall \mu$ ;
    $\epsilon_i = 1.0$ ;
    $c_i = n_i$ ;
end
# propagate metadata particles throughout $\mu$ network ;
$t = 0$ ;
while ($\sum_{\forall p_i \in P} \epsilon_i > 0.0$ && $t <$ maxSteps) do
    foreach ($p_i \in P$) do
        # if $c_i$ has no outgoing edges, freeze the particle ;
        if ($|\text{out}(c_i, \mu)| > 0$) then
            $c_i = \theta(\text{out}(c_i, \mu))$ ;
            $\epsilon_i = \epsilon_i * (1 - \delta)$ ;
            # do not recommend metadata to the particle's home node ;
            if ($c_i\ != n_i$) then
                recommendMeta($c_i, p_i$) ;
            end
        end
    end
    $t = t + 1$ ;
end
```

Algorithm 2. Propagating metadata particles through an associative network of type μ

—cites(m_x,m_y): manuscript m_x cites manuscript m_y
—keyword_of(m,k): manuscript m has keyword k

For the purposes of this experiment, the semantic network from de Lin and Chalupsky [2004] was transformed into a list of manuscripts and their associated metadata property name/value pairs. These manuscript properites include: authors, date of publication, citations, keywords, publishing journal, and organizations. Of the 29,014 manuscript nodes, different occurrence and cooccurrence algorithms were used to construct the following associative networks:

(1) citation: manuscript m_i maintains an edge to manuscript m_j if m_i cites m_j (27,240 edges)
(2) coauthor: manuscripts maintain an edge if they share authors (724,406 edges)
(3) cocitation: manuscripts maintain an edge if they share citations (23,089,616 edges)
(4) cokeyword: manuscripts maintain an edge if they share keywords (12,418,172 edges)
(5) coorganization: manuscript maintain an edge if they share organizations (33,947,083 edges)

Though not explored empirically, it is worth noting that link prediction algorithms can be employed to resolve issues relating to edge sparsity in the network. In particular, the methods proposed in Liben-Nowell and Kleinberg [2003] and Huang et al. [2005] are such algorithms.

4.1 A Review of Precision, Recall, and F-Score

The results of the metadata generation experiment are evaluated according to the F-score measure so therefore, it is important to provide a quick review of *precision*, *recall*, and *F-score* within the framework of the notation presented thus far. For a particular property μ, precision is defined as the amount of property values of type μ received that were relevant relative to the total number of property values retrieved overall. This is represented in Equation (7), where the function $\text{rec}(n_i, \mu)$ returns the set of recommended property values for resource n_i of type μ, while $\text{meta}(n_i, \mu)$ returns the set properties values of type μ previously existing for resource n_i. Since the validation is against an artificially atrophied resource set, the recommended property values are checked against the previously existing property values (prior to artificial atrophy).

$$Pr(\mu) = \frac{|\text{meta}(n_i, \mu) \cap \text{rec}(n_i, \mu)|}{|\text{rec}(n_i, \mu)|}. \tag{7}$$

Recall, Equation (8), on the other hand, is defined as the proportion of relevant property values received to the total amount of relevant property values possible. For example, if resource n_i previously (before artificial atrophy) had the property value keyword={swarm} and is recommended the property value keyword={swarm}, then there is a 100% recall. On the other hand, if resource n_i previously had the property values keyword={swarm, network} and is recommended the property value keyword={swarm}, then there is a 50% recall, whereas its precision is 100% in both cases.

$$Re(\mu) = \frac{|\text{meta}(n_i, \mu) \cap \text{rec}(n_i, \mu)|}{|\text{meta}(n_i, \mu)|}. \tag{8}$$

Precision and recall tend to be inversely related, $Pr \approx \frac{1}{Re}$. This inverse relationship is understood best when examining the extreme cases. If every possible property value was provided to a resource ($|\text{rec}(n_i, \mu)| \rightarrow \infty$), and that resource originally only had one property value ($|\text{meta}(n_i, \mu)| = 1$), then the recall would be 100% while the precision would be near 0%. At the opposite extreme, if a resource previously had every possible property value in its original metadata ($|\text{meta}(n_i, \mu)| \rightarrow \infty$) and was recommend only one property value ($|\text{rec}(n_i, \mu)| = 1$), then the precision would be 100%, but the recall would be near 0%. While, in some systems, precision and recall can be inversely related, it is the goal of information retrieval systems that are validated according to this criterion to achieve both high precision and recall values.

Finally, F-score, Equation (9), can be used to combine precision and recall into a single measure. Note that different associative networks will perform differently for different property types. For instance, cocitation networks will, intuitively, preform better at propagating keyword values than coorganization networks. Therefore, the F-score measure will be represented as $F(\mu_x, \mu_y)$ in order to express the F-score of a network created from metadata properties of type μ_y propagating property values of type μ_x. Precision and recall can be represented in a similar fashion though the results of the experiment to follow are expressed according to the F-score measure only.

$$F(\mu_x, \mu_y) = \frac{2 \cdot Pr(\mu_x) \cdot Re(\mu_x)}{Pr(\mu_x) + Re(\mu_x)}. \tag{9}$$

4.2 Experiment Parameters

The experiment was set up to determine various F-scores, $F(\mu_x, \mu_y)$, where $\mu_x \in$ {auth, cite, date, jour, key, org} and $\mu_y \in$ {cite, coauth, cocite, cokey, coorg}. This means that for every type of associative network generated, an F-score for each metadata property type was determined. Since the hep-th 2003 bibliographic dataset is a metadata-rich dataset, it was necessary to destroy a percentage of the metadata to test whether or not the metadata generation algorithm could reconstruct the property values for the selected metadata-poor resources. Therefore, the tunable parameter, density, $\partial \in [0.01, 0.9]$, was created. The density of the network metadata ranges from 1% of the network resources containing metadata to 99% of the resources. Given the percentage parameter, resources were randomly selected for atrophy before the metadata propagation algorithm was run.

With the potential for 99% of the network containing metadata, the propagation of metadata to the lacking 1% would be overwhelming (a high recall with a low precision). In order to allow nodes to regulate the amount of metadata property values they accept, a second parameter exists. The percentile parameter, $\rho \in [0, 1]$, determines the energy threshold for property value recommendations. Since each rec(n_i, μ) entry has an associated energy value (recommendation influence), a range from 0th percentile, meaning all provided property values are accepted to 100th percentile, meaning only the top energy property value is accepted, is used. The pseudo-code for the experimental set-up is presented in Algorithm 3. In Algorithm 3, killMeta(), acceptMeta(), and calculateF() do not have accompanying pseudo-code.

The general expected trend is that as the density of the network increases, the recall increases and the precision decreases. With more property values being propagated, any metadata-poor record, on average, will receive more recommendations than are needed. For instance, a manuscript only has one publishing journal, therefore, a recommendation of 100 journals is going to yield a very low precision (0.01). To balance this effect, a percentile increase will tend to increase the precision of the algorithm at the expense of recall. When only the highest energy recommendations are accepted, the probability of rejecting a useful recommendation increases. In the case of journal propagation, if only the 100th percentile recommendation is accepted, then only the highest energy recommendation is accepted. If this journal recommendation is the correct publishing venue, then there is 100% recall and precision. If not, then there is 0% recall and precision. Depending on the number of values needed to fill a particular property, different ρ values will be most suitable than others.

```
Input: experiment()
# run the metadata propagation algorithm for each associative
network type ;
foreach ($\mu_y \in$ [coauth, cocite, cokey, cite]) do
    loadNetwork($\mu_y$) ;
    foreach ($\mu_x \in$ [auth, cite, date, jour, key, org]) do
        # atrophy a randomly selected percentage of the
        network ;
        for ($\partial = 0.01,\ \partial < 1.0,\ \partial = \partial + 0.2$) do
            killMeta($1 - \partial$) ;
            propagateMeta($\mu_x$) ;
            # allow metadata-poor resources to accept only a
            certain percentage of their recommended property
            values ;
            for ($\rho = 0.0,\ \rho <= 1.0,\ \rho = \rho + 0.1$) do
                acceptMeta($\rho$) ;
                calcuateF($\mu_x, \mu_y$) ;
            end
        end
    end
end
```

Algorithm 3. Determining the F-score for the various experimental parameters

4.3 The Results

This section presents the results of the experiment outlined previously in Algorithm 3. For every associative network type and for every metadata type, a F-score matrix was determined for every combination of ∂ (density) and ρ (percentile). These F-score values were calculated as the average over 20 different runs of the experiment. Tables II and III provide the max and mean F-scores for each network/metadata pair over the entire ∂/ρ set. Note that the bold faced values are those μ_x/μ_y pairs for which a landscape plot is provided. The italicized values are experimental anomalies since the same metadata that was used to generate the associative network is also the same metadata being propagated. For all other combinations, metadata of a particular μ type exists to create an associative network and metadata properties of a different μ type is being propagated over those edges. For instance, a coauthorship network is used to propagate citation property values.

The following landscape plots expose the relationship between ∂ and ρ. A short explanation of the intuition behind each plot is also provided.

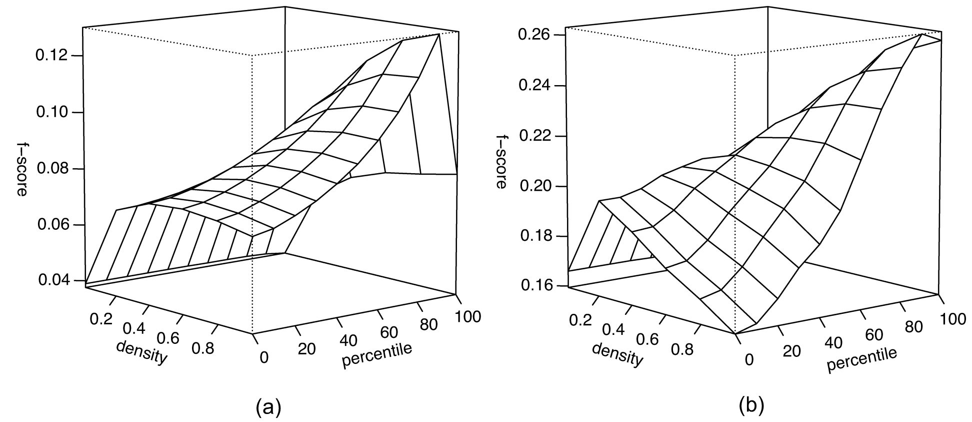
Intuitively, it makes sense that a coauthorship network would perform well when propagating citation, journal, keyword, and organization property values, which are represented in Figure 3(a), Figure 3(b), Figure 4(a), Figure 4(b) respectively. The performance is a result of the fact that collaborating authors tend to cite themselves, publish in similar journals, write about similar topics, and are within similar organizations. Notice the effect that percentile (ρ) has on Figure 3(a) as opposed to Figure 3(b). Since there tend to exist many citation property values (manuscripts cite many manuscripts), lower percentile values ($\rho \approx 0$) ensures that there is a high recall. When $\rho = 1.0$, only the top citation is

Table II. Max F-Scores

Network/Metadata	Author	Citation	Date	Journal	Keyword	Organization
citation	**0.1829**	*0.1757*	0.0606	**0.2438**	**0.3913**	**0.2782**
coauthor	*0.6218*	**0.1300**	0.0717	**0.2630**	**0.2795**	**0.6457**
cocitation	0.0770	*0.1821*	*0.0780*	**0.2081**	**0.2213**	0.1350
cokeyword	0.0073	0.0248	0.0472	**0.1904**	*0.8689*	0.0420
coorganization	0.0709	0.0236	0.0508	**0.1918**	0.1180	*0.5000*

Table III. Mean F-Scores

Network/Metadata	Author	Citation	Date	Journal	Keyword	Organization
citation	**0.1367**	*0.1327*	0.0441	**0.2133**	**0.3246**	**0.2218**
coauthor	*0.2848*	**0.0780**	0.0548	**0.2004**	**0.1958**	**0.3935**
cocitation	0.0338	*0.0697*	0.0539	**0.1554**	**0.1509**	0.0768
cokeyword	0.0032	0.0160	0.0385	**0.1468**	*0.3240*	0.0330
coorganization	0.0312	0.0145	0.0392	**0.1410**	0.0909	*0.1554*



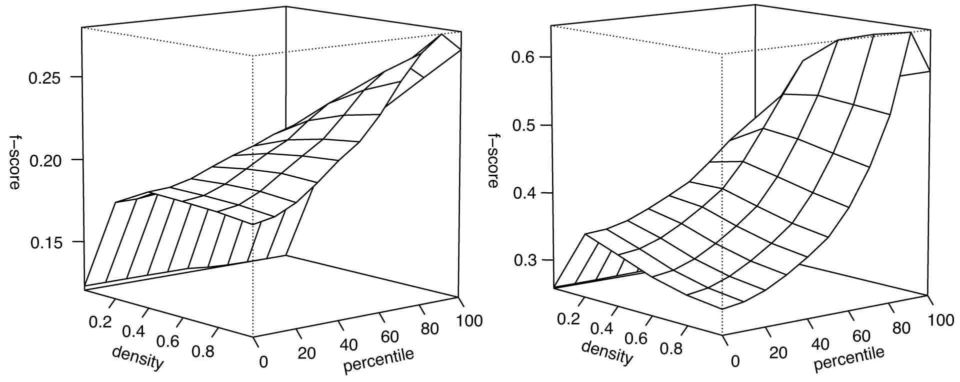
(a)

(b)

Fig. 3. Coauthorship network propagating (a) citation $F(\text{cite}, \text{coauth})$ and (b) journal F(jour, coauth) metadata.

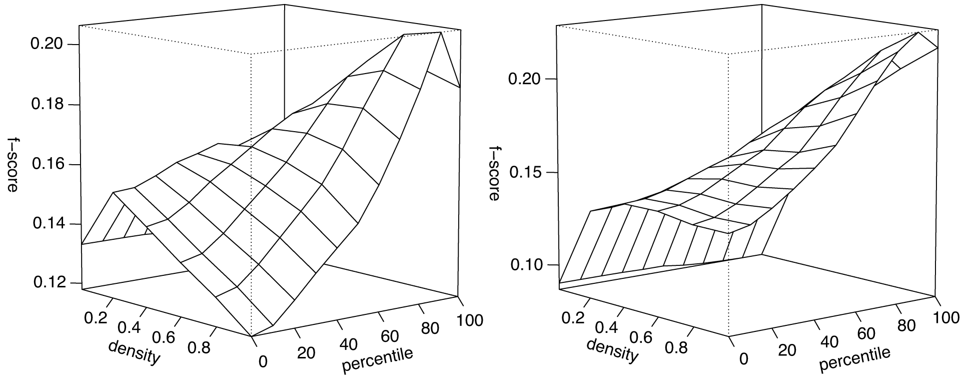
accepted and therefore the F-score drops (very poor recall). On the other other hand, in Figure 4, when $\rho = 0.0$, there are many journal recommendations. This is not desirable since a journal property only has one value (a manuscript is published in only one venue). Therefore, at $\rho = 1.0$, only one journal value is accepted into the resource's journal property. In situations where few property values are expected, the F-score is best with a high ρ.

A cocitation network, Figure 5, performs best with journal and keyword properties. This means that manuscripts are likely to cite other manuscripts with similar journal venues and since citation tends to be within the same subject domain, the probability of similar keyword metadata increases. Again, notice the effect of ρ on journal metadata propagation. The shape of the Figure 5(a) graph nearly mimics the shape of Figure 3(b). Likewise, for Figure 5(b) and Figure 4(a). Again, the expected property value number is a major factor in determining the system's ρ parameter.



(a) (b)

Fig. 4. Coauthorship network propagating (a) keyword $F(\text{key}, \text{coauth})$ and (b) organization $F(\text{org}, \text{coauth})$ properties.



(a) (b)

Fig. 5. Cocitation network propagating (a) journal $F(\text{jour}, \text{cocite})$ and (b) keyword $F(\text{key}, \text{cocite})$ properties.

A citation network, like a cocitation network, performs well for author, journal, keyword, and organizational properties (Figure 6 and Figure 7). It is interesting to note how much better a citation network works for $\rho \approx 0.0$. Since a citation network isn't symmetric, there is a chance that a particle will reach a dead end. When a particle reaches a dead end, it no longer recommends property values. Furthermore, citations are in a hierarchy with more recent publications being at the top of the hierarchy (manuscripts can not cite forward in time). Particles therefore trickle down the hierarchy via a single, non-recurrent path from top to bottom. This "plinko ball" effect is represented in Figure 8. The lack of recurrence in citation networks tends to produce a high precision with a lower recall. High precision and low recall is exactly what a low ρ produces. Therefore, since the topology of the citation network yields the same effect, the effect of ρ as $\rho \to 0.0$ isn't as pronounced.

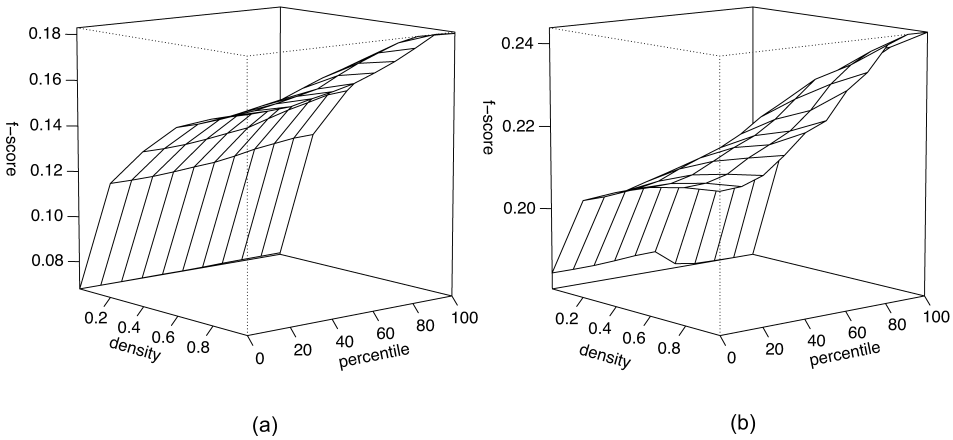


Fig. 6. Citation network propagating (a) author $F(\text{auth}, \text{cite})$ and (b) journal $F(\text{jour}, \text{cite})$ properties.

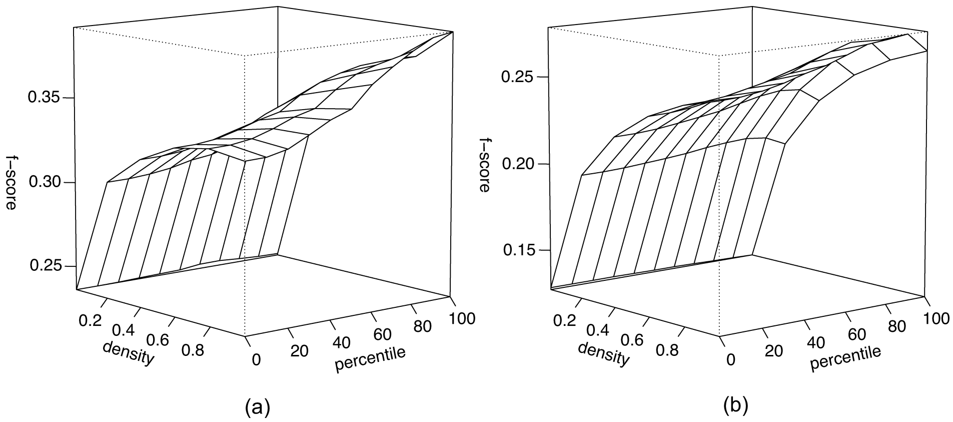


Fig. 7. Citation network propagating (a) keyword $F(\text{key}, \text{cite})$ and (b) organization $F(\text{org}, \text{cite})$ properties.

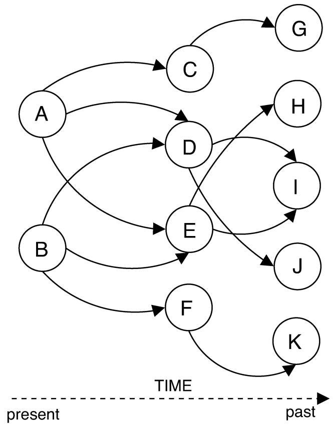


Fig. 8. Citation networks are non-recurrent networks.

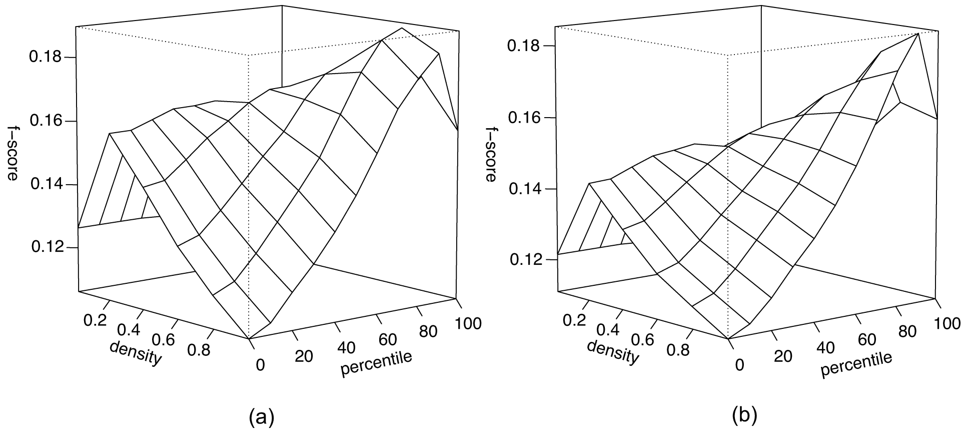


Fig. 9. Cokeyword and coorganization networks propagating (a) journal F(jour, cokey) and (b) F(jour, coorg) properties, respectively.

As can be noticed from Table II, Table III, and Figure 7(a), the keyword property performs best in a citation network. A direct reference from one document to another is a validation of the similarity between documents with respect to subject domain. Therefore, the tendency for citing documents to contains similar keyword values is high. For instance, refer to the citations of this article (references in this manuscript's bibliography). Every cited manuscript is either about automatic metadata generation, bibliographic networks, or network analysis.

A cokeyword network does not perform well for most property types except the journal property, Figure 9(a). This makes sense since manuscripts on similar topics are likely to be published in similar journals.

5. FUTURE WORK

This article has provided a preliminary exploration of metadata generation in terms of metadata property propagation within an associative network of repository resources. Further research in this area may prove useful for other network types such as those generated from other metadata properties. For instance, it may be of interest to study the effect of this algorithm on usage networks [Bollen et al. 2005]. Usage metadata, unlike citation and journal metadata, is applicable to every accessible resource. It would be interesting to see what cousage means for a particular genera of resources by determining which metadata properties these networks are best at propagating.

A variety of propagation algorithms may also be explored. It is assumed that a particle will take only edges of a particular μ type for the duration of their life span. Different path types might be an important aspect of increasing the precision and recall performance of this method. For instance, keyword metadata that first propagates over coauthorship edges and then over cocitation edges might provide better results. Methods to implement such propagation algorithms have been presented in Rodriguez [2007, 2008]. Also, different edge types can be merged such that all co-keyword and co-authorship edges are collapsed to form a single edge.

What has been presented in this study is the results of this algorithm without the intervention of any human components (besides the initial creation of metadata through the hep-th dataset creation process). Future work that studies this method with the inclusion of humans that help to validate and "clean" the recommended metadata would be telling of how much this method is able to speed up the process of generating accurate and reliable metadata for metadata-poor resources. Such an analysis is left to future research.

Finally, multiplicative effects due to particle interaction may effect the results of the algorithm. For instance, if two particles, p_i and p_j, meet at a particular node, n_k, and p_i and p_j have similar metadata then the footprint they leave at n_k should be more noticeable. Because two different metadata sources are supplying the same property values, there is an increased probability of that recommended metadata value being correct. Currently, only a summation is being provided. It may be interesting to multiply this summation by the number of unique particles that provided energy for a particular recommended metadata value. The variations of this preliminary framework will be explored in future work.

6. CONCLUSION

Automatic metadata generation is becoming an increasingly important field of research as digital library repositories become more prevalent and move into the arena of less strongly controlled, decentralized collections (e.g., arXiv and CiteSeer). The creation and maintenance of high-quality, detailed metadata is hampered on numerous levels. Manual metadata creation methods are costly. Recent efforts to leverage the collective power of social tagging (i.e., "folksonomies") may address some of the shortcomings of the manual creation of metadata and result in viable models for online resources that do not require strongly controlled vocabularies and metadata ontologies. However, it is doubtful that "folksonomies" can be generalized to situations that require vetted, well-standardized metadata. The automated creation of metadata on the basis of content-analysis is a promising alternative to the manual creation of metadata. It is conceivably more efficient in situations where textual data is available and allows for more formal control of the type and nature of metadata that is extracted. However, it can be unreliable for nontext resources, yield low-quality metadata and can be computationally expensive.

This article proposed another possible component of the metadata generation toolkit which may complement and support the above mentioned approaches. Instead of creating new metadata, metadata is propagated from a metadata-rich subset of the collection to similar, but metadata-poor subsets. The substrate for this extrapolation is an associative network of resource relations created from other available metadata. Metadata propagation may provide a computationally feasible means of generating large amounts of metadata for heterogeneous resources which can later be fine-tuned by manual intervention or cross-validation with content-based methods. The article finally provided experimental results using the High-Energy Physics bibliographic data set (hep-th 2003). Human intervention may play an important role in fine-tuning the metadata propagation algorithm. The results of this experiment are promising and there still exists a range of potential modifications to this basic framework that may lead to even better results.

ACKNOWLEDGMENTS

The modified hep-th 2003 bibliographic dataset was generously provided by Shou-de Lin and Jennifer H. Watkins provided editorial assistance. Finally, the hep-th 2003 database is based on data from the arXiv archive and the Stanford Linear Accelerator Center SPIRES-HEP database provided for the 2003 KDD Cup competition with additional preparation performed by the Knowledge Discovery Laboratory, University of Massachusetts Amherst.

REFERENCES

BOLLEN, J., DE SOMPEL, H. V., SMITH, J., AND LUCE, R. 2005. Toward alternative metrics of journal impact: a comparison of download and citation data. *Inform. Process. Manage. 41,* 6, 1419–1440.

COHEN, P. R. AND KJELDSEN, R. 1987. Information retrieval by constrained spreading activation in semantic networks. *Inform. Process. Manage. 23,* 4, 255–268.

COLLINS, A. AND LOFTUS, E. 1975. A spreading activation theory of semantic processing. *Psych. Rev. 82*, 407–428.

CRESTANI, F. 1997. Application of spreading activation techniques in information retrieval. *Artificial Intell. Rev. 11,* 6, 453–582.

CRESTANI, F. AND LEE, P. L. 2000. Searching the web by constrained spreading activation. *Inform. Process. Manage. 36,* 4, 585–605.

DE LIN, S. AND CHALUPSKY, H. 2004. Issues of verification for unsupervised discovery systems. In *Proceedings of the ACM/SIKDD International Conference on Knowledge Discovery and Data Mining (KDD'04) Workshop on Link Discovery*.

DUVAL, E., HODGINS, W., SUTTON, S., AND WEIBEL, S. L. 2002. Metadata principles and practices. *D-Lib Mag. 8,* 4.

GIURIDA, G. AND SHEK, E. YANG, J. 2000. Knowledge-based metadata extraction from postscript file. In *Proceedings of the International Conference on Digital Libraries*. ACM, 77–84.

GOLDER, S. A. AND HUBERMAN, B. A. 2006. Usage patterns of collaborative tagging systems. *J. Inform. Sci. 32,* 2, 198–208.

GREENBERG, J. 2004. Metadata extraction and harvesting: A comparison of two automatic metadata generation applications. *J. Intern. Catalog. 6,* 4, 59–82.

HAN, H., GILES, C. L., MANAVOGLU, E., AND ZHA, H. 2003. Automatic document metadata extraction using support vector machines. In *Proceedings of the Joint Conference on Digital Libraries (JCDL'03)*. ACM.

HERLOCKER, J. L., KONSTAN, J. A., TERVEEN, L. G., AND RIEDL, J. T. 2004. Evaluating collaborative filtering recommender systems. *ACM Trans. Inform Syst. 22,* 1, 5–53.

HUANG, Z., CHEN, H., AND ZENG, D. 2004. Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Trans. Informa. Syst. 22,* 1, 116–142.

HUANG, Z., LI, X., AND CHEN, H. 2005. Link prediction approach to collaborative filtering. In *Proceedings of the Joint Conference on Digital Libraries (JCDL'05)*. ACM.

KUWANO, H., MATSUO, Y., AND KAWAZOE, K. 2004. Reducing the cost of metadata generation by using video/audio indexing and natural language processing techniques. *NTT Technical Review*, 68–74.

LIBEN-NOWELL, D. AND KLEINBERG, J. 2003. The link prediction problem for social networks. In *Proceedings of the 12th International Conference on Information and Knowledge Management*. ACM, 556–559.

Mao, S., Kim, J. W., and Thoma, G. R. 2004. A dynamic feature generation system for automated metadata extraction in preservation of digital materials. In *Proceedings of the 1st International Workshop on Document Image Analysis for Libraries (DIAL'04)*. IEEE.

Mathes, A. 2004. Folksonomies—cooperative classification and communication through shared metadata. Computer Mediated Communication - LIS590CMC (graduate course), Univerity of Illinois Urbana-Champaisn.

McGovern, A., Friedland, L., Hay, M., Gallagher, B., Fast, A., Neville, J., and Jensen, D. 2003. Exploiting relational structure to understand publication patterns in high-energy physics. *ACM SIGKDD Explor. Newslett. 5,* 2 (December), 165–172.

Naaman, M., Yeh, R. B., Garcia-Molina, H., and Paepcke, A. 2005. Leveraging context to resolve identity in photo albums. In *Proceedings of the 5th Joint Conference on Digital Libraries (JCDL'05)*.

Prime-Claverie, C., Beigbeder, M., and Lafouge, T. 2005. Metadata propagation in the Web using co-citations. In *Proceedings of the IEEE/ACM International Conference on Web Intelligence*. IEEE Computer Society.

Rodriguez, M. A. 2007. Social decision making with multi-relational networks and grammar-based particle swarms. In *Proceedings of the 40th Annual Hawaii International Conference on Systems Science (HICSS'07)*.

Rodriguez, M. A. 2008. Grammar-based random walkers in semantic networks. *Knowledge-Based Systems [in press]*.

Rorvig, M., Jeong, K.-T., Pachlag, A., Anusuri, R., and Oyarce, S. 2002. Content based image retrieval by integration of metadata encoded multimedia (image and text) features. In *Proceedings of the World Wide Web Conference (WWW'02)*. ACM.

Sebastiani, F. 2002. Machine learning in automated text categorization. *ACM Comput. Surv. 34,* 1, 1–47.

Ward, J. 2003. A quantitative analysis of unqualified dublin core metadata element set usage within data providers registered with the open archives initiative. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'03)*. IEEE Computer Society, 315–317.

Yang, H.-C. and Lee, C.-H. 2005. Automatic metadata generation for Web pages using a text mining approach. In *Proceedings of the International Workshop on Challenges in Web Information Retrieval and Integration*. IEEE, 186–194.

Received January 2007; revised February 2008; accepted June 2008