Resolving Task Specification and Path Inconsistency in Taxonomy Construction

Hui Yang
Department of Computer Science
Georgetown University
37th and O street NW
Washington, DC, 20057
huiyang@cs.georgetown.edu

Abstract

Taxonomies, such as Library of Congress Subject Headings and Open Directory Project, are widely used to support browsing-style information access in document collections. We call them browsing taxonomies. Most existing browsing taxonomies are manually constructed thus they could not easily adapt to arbitrary document collections. In this paper, we investigate both automatic and interactive techniques to derive taxonomies from scratch for arbitrary document collections. Particular, we focus on encoding user feedback in taxonomy construction process to handle task-specification rising from a given document collection. We also addresses the problem of path inconsistency due to local relation recognition in existing taxonomy construction algorithms. The user studies strongly suggest that the proposed approach successfully resolve task specification and path inconsistency in taxonomy construction.

1 Introduction

Taxonomies, such as Library of Congress Subject Headings (LCSH, 2011) and Open Directory Project (ODP, 2011), are widely used to support browsing-style information access in document collections. We call them browsing taxonomies. Browsing taxonomies are tree-structured hierarchies built upon a given document collection. Each term in a browsing hierarchy categorizes a set of documents related to this term. Driven by their needs, users can navigate through a the hierarchical structure of a browsing taxonomy to access particular documents. A browsing taxonomy can benefit information access via (1) providing an overview of (important) concepts in a document collection, (2) increasing the visibility of documents ranked low in a list (e.g. documents ordered by search relevance), and (3) presenting together documents about the same concept to allow more focused reading.

Most existing browsing taxonomies are manually constructed thus they could not easily adapt to arbitrary document collections. However, it is not uncommon that document collections are given ad-hoc for specific tasks, such as search result organization in for individual search queries (Carpineto et al., 2009) and literature investigation for a new research topic (Chau et al., 2011). There is a necessity to explore automatic or interactive techniques to support *quick* construction of browsing taxonomies for arbitrary document collections.

Most research on automatic taxonomy construction focuses on identifying local relations between concept pairs (Etzioni et al., 2005; Pantel and Pennacchiotti, 2006). The infamous problem of *path inconsistency*, which are usually caused by the local nature of most relation recognition algorithms when building a taxonomy, commonly exists in current research. Oftentimes, when a connecting concept for two pairs of parent-child concepts has multiple senses or represent mixed perspectives, the problem shows up. For example, while *financial institute*→*bank* and *bank*→*river bank* are correct; the path *financial institute*→*bank*→*river bank* is semantically inconsistent.

Proceedings of the 3rd Workshop on the People's Web Meets NLP, ACL 2012, pages 20–28,
Jeju, Republic of Korea, 8-14 July 2012. ©2012 Association for Computational Linguistics

In this paper, we propose a semi-supervised distance learning method to construct task-specific taxonomies. Assuming that a user is present to construct a taxonomy for browsing, the proposed approach directly learns semantic distances from the manual guidance provided by the user to predict semantically meaningful browsing taxonomies. Moreover, We tackle path inconsistency by posing constraints over root-to-leaf paths in a hierarchy to ensure concept consistency within paths

The contributions of our work include:

- It offers an opportunity for handling task specifications.
- Unlike most algorithms, our work takes care of path consistency during taxonomy construction.

The remainder of this paper is organized as follows: Section 2 describes the related work. Section 3 details the proposed automated algorithm for taxonomy construction. Section 4 presents the interactive algorithm to incorporate user feedback under a supervised semantic distance learning framework. Section 5 describes the evaluation and Section 6 concludes the paper.

2 Related Work

Most research conducted in the NLP community focuses on extracting local relations between concept pairs (Hearst, 1992; Berland and Charniak, 1999; Ravichandran and Hovy, 2002; Girju et al., 2003; Etzioni et al., 2005; Pantel and Pennacchiotti, 2006; Kozareva et al., 2008). More recently, more attention has been paid in building full taxonomies. For example, (Kozareva and Hovy, 2010) proposed to connect local concept pairs by finding the longest path in a subsumption graph. Both (Snow et al., 2006) and (Yang and Callan, 2009) incrementally grew taxonomies by adding new concepts at optimal positions within the existing structures. Specifically, Snow et al. estimated conditional probabilities by using syntactic parse features and decided taxonomic structure via maximizing overall likelihood of taxonomy. Yang and Callan proposed the ME framework to model the semantic distance $d(c_x, c_y)$ between concepts c_x and c_y as a weighted combination of numerous lexical and semantic features: $\sum_j \text{weight}_j * \text{feature}_j(c_x, c_y)$ and determine the taxonomic structure by minimizing overall distances.

An advantage in ME is that it allows manipulations to concept positions by incorporating various constraints to taxonomic structures. For example, ME handled concept generality-specificity by learning different distance functions for general concepts (located at upper levels) and specific concepts (located at lower levels) in a taxonomy.

In the Information Retrieval (IR) community, browsing taxonomies. also often called browsing hierarchies or Web directories, has been studied as an alternative to the ranked list representation for search results by the Information Retrieval (IR) community. The proposed forms of browsing structures include topic clusters (Cutting et al., 1992) and monothetic concept hierarchies (Sanderson and Croft, 1999; Lawrie et al., 2001; Kummamuru et al., 2004; Carpineto et al., 2009). The latter uses single concepts to represent documents containing them and organizes the concepts into hierarchies; they are in fact taxonomies. The major drawback of these approaches is that they often fail to produce meaningful taxonomic structures due to neglecting the semantics among concepts. For instance, (Sanderson and Croft, 1999) used document frequency and (Lawrie et al., 2001) used conditional probability to derive *is-a* relations. Moreover, they also suffer from path inconsistency when building full taxonomies.

3 Browsing Taxonomy Construction

To build browsing taxonomy for a document collection, the first step is to extract the concepts. We take a simple but effective approach. We exhaustively examine the collection and output a large set of terms, formed by nouns, noun phrases, and named entities occurring $>$5 times in the collection. We then filter out invalid terms due to part-of-speech errors or misspelling by removing terms that occur $<$4 times out of the top 10 returned snippets when submitting the term to *google.com* as a search query. We further conflate similar terms into clusters using LSA (Bellegarda et al., 1996) and select the most frequent terms as concepts from each term group. We select the N most frequent concepts to form the concept set C. N usually ranges from 30 to 100. We assume that C contains all concepts in the browsing taxonomy;

even when an important concept for the collection is missing, we will "make do" with C. This may lead to some errors, but can be later corrected by users through proposing new concepts interactively (Section 4).

This section presents how to automatically build taxonomies. We introduce the semantic distance learning method in Section 3.1 and present how to achieve path consistency control in Section 3.2.

3.1 Semantic Distance Learning

To support browsing in arbitrary collections, in this paper, we propose to incorporate task specification in a taxonomy. One way to achieve it is to define task-specific distances among concepts. Moreover, through controlling distance scores among concepts, we can enforce path consistency in taxonomies. For example, when the distance between *financial institute* and *river bank* is big, the path *financial institute→bank→river bank* will be pruned and the concepts will be repositioned. Inspired by ME, we take a distance learning approach to deal with path consistency (Section 3) and task specification (Section 4) in taxonomy construction. In this section, we demonstrate how to estimate semantic distances from training data.

We assume that there are some underlying feature functions that measure semantic dissimilarity for two concepts from various aspects and a good semantic distance is a combination of all features. Different from ME, we model the semantic distance between concepts (c_x, c_y) as a Mahalanobis distance (Mahalanobis, 1936):

$$d_{c_x,c_y} = \sqrt{\Phi(c_x, c_y)^T W^{-1} \Phi(c_x, x_y)} \qquad (1)$$

$d_{c_x,c_y} = \sqrt{\Phi(c_x, c_y)^T W^{-1} \Phi(c_x, x_y)}$, where $\Phi(c_x, c_y)$ represents the set of pairwise underlying feature functions, where each feature function is $\phi_k : (c_x, c_y)$ with k=1,...,$|\Phi|$. W is a weight matrix, whose diagonal values weigh the underlying feature functions. When only diagonal values of W are taken into account, W is equivalent to assigning weights to different axes in the random vectors.

Note that a semantic distance is still a distance metric. One important characteristic of a valid distance metric is that it must represent valid clustering partitions, which means that the clustering partitions represented by the distance metric should be consistent. Therefore, certain constraints need to be satisfied. An obvious one is that concepts in the same cluster should have smaller distance scores than those in different clusters. Moreover, a valid distance metric should be non-negative and satisfy the triangle inequality. To ensure such regularities, we need to constrain W to be positive semi-definite (PSD) (Bhatia, 2006):

$$W \succeq 0.$$

Since we assume that a good semantic distance is a combination of all these features, we can decompose the task of semantic distance learning into two subtasks - identifying good features and learning the weight matrix from training data.

In our approach, we employ a wide range of features to cover various aspects in measuring dissimilarity between concepts. Given two concepts c_x and c_y, a feature is defined as a function $\phi : (c_x, c_y)$ enerating a value within [0,1]. In total, we used 31 features, including *lexical-syntactic patterns, contextual, co-occurrence, syntactic dependency*, and *definitions*.

Similar to the linguistic approaches, we use *lexical-syntactic patterns* to evaluate relations among concepts. Our patterns include hypernym patterns such as "c_x*, and other* c_y", sibling patterns such as "c_x *and* c_y", and part-of patterns such as "c_x *consists of* c_y". Each feature returns a boolean value of wether it can find instances for the pattern in text.

Besides patterns, we used more semantic features. For example, since word meanings can be inferred from and represented by contexts, we develop several *contextual* features. One is *Local Context KL-Divergence*, which measures the Kullback-Leibler divergence between two unigram language models built for c_x and c_y upon all left two and right two words surrounding them. Moreover, we formulate the *co-occurrence* features as point-wise mutual information between (c_x, c_y):

$$pmi(c_x, c_y) = log \frac{Count(c_x, c_y)}{Count(c_x) Count(c_y)},$$

where $Count(.)$ is defined as the number of documents or sentences containing the concept(s), or n as in "Results 1-10 of about n for term" appearing on the first page of Google search results for querying c_x, c_y, or $c_x c_y$.

We also generate *syntactic dependency* features via syntactic parse[1] and semantic role labeling[2]. For example, we measure how many overlaps exist between c_x's and c_y's modifiers. Lastly, we measure *definition overlaps* between c_x and c_y by counting the number of nonstop word overlaps between their definitions obtained by querying *google.com* with "define:c_x" and "define:c_y".

To achieve a comprehensive distance measure for concepts, we propose to effectively combine these features. Our goal is to find a parametric distance metric functions which allows combining various features and assigning different weights for them. It also needs to produce distances that satisfy non-negativity and triangle inequality.

We further estimate W by minimizing the squared errors between the semantic distances d generated from the training data and the expected value $\hat{d}$. Moreover, we constrain W to be PSD. The parameter estimation is:

$$\min_{W} \sum_{x=1}^{|\mathbf{C}|} \sum_{y=1}^{|\mathbf{C}|} \left(d_{c_x,c_y} - \sqrt{\Phi(c_x,c_y)^T W^{-1} \Phi(c_x,c_y)} \right)^2 \tag{2}$$

subject to $W \succeq 0$. The optimization can be done by any standard semi-definite programming (SDP) solver. We used (Sedumi, 2011) and (Yalmip, 2011) to perform the optimization.

In our framework, the major source of training data is user feedback. Another source is existing hierarchies such as WordNet (Fellbaum, 1998) and ODP (ODP, 2011) (Section 3). Nonetheless, we obtain the semantic distance for a concept pair (c_x, c_y) in training data by summing up edge weights along the shortest path from c_x to c_y in a training hierarchy. The edge weight can be assigned based on the types of relations that an edge represent as in Section 4.1.

The learned model W can be used to predict distance scores for testing concept pairs by applying Eq. 1 on them.

3.2 Resolving Path Inconsistency

With the pair-wise semantic distances, we are ready to build the full taxonomy. As in ME, we also take an incremental taxonomy construction framework, where concepts are inserted one at a time. Particularly, we propose that at each insertion, a concept c_z is tried as either a parent or a child concept to all existing nodes in the current partial taxonomy T^n. The evaluation of the best position depends on the semantic distances between c_z and all other concepts in the taxonomy.

To enforce consistency along a path from the root to a leaf in a taxonomy, we propose to require all concepts on the path to be about the same topic. They need to be coherent no matter how far away two concepts are apart in this path. We achieve this by enforcing the sum of semantic distances in a path to be as small as possible. Particularly, when a new concept c_z is added into a taxonomy T, we require that the optimal root-to-leaf path $\hat{P}$ containing c_x should satisfy the following condition:

$$\hat{P_{c_z}} = \arg\min_{P'_{c_z}} \sum_{c_x, c_y \in P'_{c_z}, x<y} d(c_x, c_y) \tag{3}$$

where P_{c_z} is a root-to-leaf path including c_z, $x < y$ defines the order of the concepts so we only compute a pair-wise distance between two concepts once.

To incorporate path consistency into taxonomy construction, we introduce a variable $\lambda \in [0, 1]$ to control the contributions from overall semantic distance minimization (as in ME) and path distance minimization. We formulate the optimization as:

$$\min \lambda u + (1 - \lambda) v \tag{4}$$

subject to $u = |\sum_{c_x, c_y \in C^n \cup \{c_z\}, x<y} d(c_x, c_y) - \sum_{c_x, c_y \in C^n, x<y} d(c_x, c_y)|$, $v = \sum_{c_j, c_k \in P'_{c_z}, j<k} d(c_j, c_k)$, $0 \leq \lambda \leq 1$, where u denotes "minimization of overall semantic distance", v denotes the "path consistency", and C^n is the concept set for the n^{th} partial taxonomy.

4 Resolving Task Specification

Give an arbitrary document collection and its concept set C, most concepts can be organized nicely according to the automatic algorithm proposed in Section 3. However, for concepts with multiple perspectives, we need to decide which perspective the

[1] Done by Minipar: http://www.cs.ualberta.ca/lindek/minipar.htm.

[2] Done by Assert: http://cemantix.org/assert/.

task wants to keep in the browsing taxonomy. Moreover, Section 3 learns distance functions from WordNet and ODP, which suggests that the algorithm will roughly follow how WordNet and ODP define relations. In practice, a task may require completely different organizations, e.g., by question-answer pairs or by topics. The ever-changing task specifications can only be captured by the user/constructor who adjusts a browsing taxonomy to suit the requirements.

This section studies how to incorporate task specifications in the taxonomy construction. Particularly, how to allow the machine learning algorithm to learn from the user, and how to produce a task-specific browsing taxonomy according to the user's guidance. The framework is expected to produce taxonomies that reflect personal preferences as a consequence of learning from manual guidance.

We present a general framework that enables taxonomy construction taking into account user-defined concept organization. Basically, to guide how to organize the concepts, a user trains the supervised distance learning model using a taxonomy construction tool that supports editing functions such as dragging and dropping, adding, deleting, and renaming nodes that allows the user to intuitively modify a taxonomy.

Particularly, an initial taxonomy is constructed by the automatic taxonomy construction framework presented in Section 3. Starting from the initial taxonomy, a user can teach the machine learning algorithm by providing manual guidance to it. The algorithm learns from the manual guidance and adjusts the distance learning function and modifies the taxonomy accordingly. When a user put c_x under c_y, it indicates that the user wants a relation demonstrated by $c_x \rightarrow c_y$ to be true in this taxonomy. We capture the user inputs as *manual guidance* and make use of it to adjust the distance learning model to organize other concepts agreeing with the user. The teaching and the learning alternate until the user is satisfied with the taxonomy. The resulting taxonomy contains both the user's inputs and the machine's adjusted organization for the concepts.

4.1 Collecting and Learning from Manual Guidance

The most challenging part of incorporating manual guidance in the machine learning process is how to translate it into a format that the machine can easily understand and incorporate into its learning models. In this research, browsing taxonomies are tree structures. Trees. however, are not straightforward for a machine learning algorithm to manipulate. In order to capture the changes between each version of the manual editions, the learning algorithm needs both the training and the test data to be in a format which is easy to handle. Matrix representation can be easily understood and manipulated by many machine learning algorithms. We therefore convert taxonomies from trees to matrices and use a matrix representation for all the intermediate editions in the taxonomy construction process.

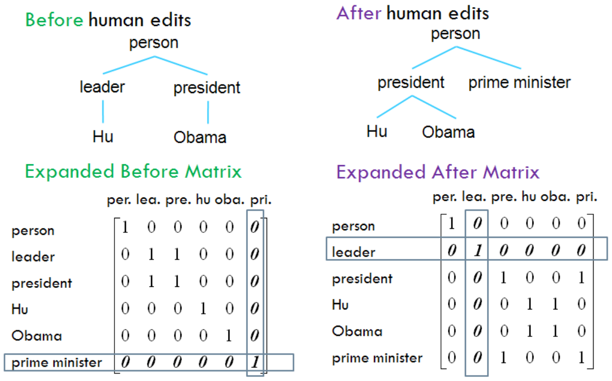
We propose to convert a taxonomy from a tree to matrices of neighboring nodes and represent the differences in matrices before and after human edits as *manual guidance*. We then train the learning framework to adjust to it and make predictions for unorganized concepts.

We represent the organization of concepts before a user's modifications as a *before matrix*; likewise, the new organization of concepts after her modifications is represented as a *after matrix*. Given these two matrixes, *manual guidance* is a submatrix in *after matrix* that shows the differences between *before matrix* and *after matrix*.

We compare the before matrix *A* and the after matrix *B* to derive the manual guidance *M*. The manual guidance is not simply the matrix difference between the before matrix and the after hierarchy matrix. It is part of the after matrix because it is the after matrix that indicates where the user wants the taxonomy to develop. We define manual guidance M as a submatrix which consists of some entries of the *after matrix* B; at these entries, there exist differences between the *before matrix* A and the *after matrix* B.

For simple cases when the set of concepts remain unchanged before and after human modifications, the above definition and calculation of manual guidance work. However, oftentimes the user adds, deletes or renames concepts, and the concept set changes. When the concept set changes, the above definition of manual guidance M needs a slight alteration.

Figure 1 shows an example taxonomy whose concept set changes. The original concept set before



$$\begin{array}{l|cccccc}
 & \text{per.} & \text{lea.} & \text{pre.} & \text{hu} & \text{oba.} & \text{pri.} \\
\hline
\text{person} & 1 & 0 & 0 & 0 & 0 & \mathit{0} \\
\text{leader} & 0 & 1 & 1 & 0 & 0 & \mathit{0} \\
\text{president} & 0 & 1 & 1 & 0 & 0 & \mathit{0} \\
\text{Hu} & 0 & 0 & 0 & 1 & 0 & \mathit{0} \\
\text{Obama} & 0 & 0 & 0 & 0 & 1 & \mathit{0} \\
\text{prime minister} & \mathit{0} & \mathit{0} & \mathit{0} & \mathit{0} & \mathit{0} & \mathit{1}
\end{array}$$

$$\begin{array}{l|cccccc}
 & \text{per.} & \text{lea.} & \text{pre.} & \text{hu} & \text{oba.} & \text{pri.} \\
\hline
\text{person} & 1 & \mathit{0} & 0 & 0 & 0 & 0 \\
\text{leader} & \mathit{0} & \mathit{1} & \mathit{0} & \mathit{0} & \mathit{0} & \mathit{0} \\
\text{president} & 0 & \mathit{0} & 1 & 0 & 0 & 1 \\
\text{Hu} & 0 & \mathit{0} & 0 & 1 & 1 & 0 \\
\text{Obama} & 0 & \mathit{0} & 0 & 1 & 1 & 0 \\
\text{prime minister} & 0 & \mathit{0} & 1 & 0 & 0 & 1
\end{array}$$

Figure 1: A taxonomy before and after human modifications (concept set changes; relation type = $sibling$).

the human modification is {*person, leader, president, Hu, Obama*}. The taxonomy's *before matrix* A is:

$$A = \begin{array}{cccccc}
 & person & leader & president & Hu & Obama \\
person & 1 & 0 & 0 & 0 & 0 \\
leader & 0 & 1 & 1 & 0 & 0 \\
president & 0 & 1 & 1 & 0 & 0 \\
Hu & 0 & 0 & 0 & 1 & 0 \\
Obama & 0 & 0 & 0 & 0 & 1
\end{array}$$

The user modifies the taxonomy at several places. In particular, *leader* is deleted, *Hu* is moved to be under *president*, and *prime minister* is inserted as a new concept into this taxonomy. Therefore the concept set changes to {*person, president, Hu, Obama, prime minister*}. The *after matrix* B is:

$$B = \begin{array}{cccccc}
 & person & president & Hu & Obama & PM \\
person & 1 & 0 & 0 & 0 & 0 \\
president & 0 & 1 & 0 & 0 & 1 \\
Hu & 0 & 0 & 1 & 1 & 0 \\
Obama & 0 & 0 & 1 & 1 & 0 \\
PM & 0 & 1 & 0 & 0 & 1
\end{array}.$$

Since the concept sets before and after the human modifications change, we cannot simply use matrix subtraction to get the difference between the before and after matrices. Suppose the concept set in the taxonomy before the modifications is C_A, and the concept set after modifications is C_B, we define an expanded set of concepts C_E as the union of C_A and C_B.

For taxonomies with concept changes, we define the manual We then define manual guidance M as a submatrix which consists of some entries of the *after matrix* B; at these entries, there exist differences from the *expanded before matrix* A' to the *expanded after matrix* B'. The expanded rows and columns in A' and B' are filled with 0 for non-diagonal entries, and 1 for diagonal entries. Note that the concepts corresponding to these entries should exist in C_B, the unexpanded set of concepts after human modifications. Formally,

$$M = B[r; c]$$

where $r = \{i : b_{ij} - a_{ij} \neq 0, c_i \in C_B\}$, $c = \{j : b_{ij} - a_{ij} \neq 0, c_j \in C_B\}$, a_{ij} is the $(i,j)^{th}$ entry in A', and b_{ij} is the $(i,j)^{th}$ entry in B'.

For the example in Figure 1, the manual guidance M is:

$$M = B[2,3,4,5;2,3,4,5] = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}.$$

Based on M, we can create training data $D = 1 - M$, for the supervised distance learning algorithm, which aims to learn a good model which best preserves the regularity defined by the task and the user using the techniques proposed in Section 3.1.

5 Evaluation

To evaluate the effectiveness of our approach, We conducted two user studies, one to evaluate browsing effectiveness and another to evaluate quality of taxonomies. Five users (graduate students and relatives of the authors) in the first study were asked to construct browsing taxonomies with a task in mind - "writing a survey paper about the collection".

In the second study (24 graduates and undergraduates), we compared taxonomies constructed by different users to identify where mixed perspectives in taxonomies come from in Section 5.3. We also investigated whether the differences are due to self-inconsistency in Section 5.4. Moreover, we manually select relations violating path consistency and report our approach's ability to handle path consistency in Section 5.2.

5.1 Datasets

To show that task-specific taxonomies are more suitable for browsing than general taxonomies, we compared excerpts of the official North America Industry Classification Systems (we call them NAICS-

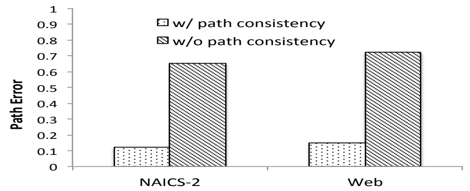


Figure 2: Path error w/ and w/o path consistency control.

1) with comparable taxonomies derived by techniques presented in this paper (we call them NAICS-2). Since the original collection used to build official NAICS taxonomies is not available, we created document collections by crawling search results from *google.com* for concepts in NAICS-1 excerpts. The participants worked on the collection to create NAICS-2 taxonomies. Each NAICS-1 or NAICS-2 taxonomy contains about 40 concepts.

We also evaluate our techniques on Web search result organization. Five Web datasets were created by submitting 4 to 5 queries[3] to and collecting the returned Web documents from search engines Bing and Google. Around 100 Web documents and 40 concepts are collected for a topic. We manually judged relevant documents for each topic.

5.2 Path Consistency

To evaluate how well our method can handle path inconsistency, we compare the path error rate before and after applying path consistency control. The evaluation is only conducted for the automated algorithm (Section 3) on the NAICS-2 and Web datasets. No user study is involved.

Two human assessors manually evaluated the path errors[4] in a taxonomy by the following procedure: (1) Starting from the root concept, perform a depth-first traverse in the taxonomy; (2) along each path, count the number of wrong ancestor-descendant pairs due to word sense ambiguity or mixed perspectives; (3) sum up the errors that both assessors agree and normalize them by the taxonomy size. Note that path errors are evaluated for concepts are *not* immediately connected, whereas differences due to mixed perspectives (Section 5.3) refer to immediate relations. Figure 2 shows that with path consistency control, we can statistically significantly reduce path errors due to word sense ambiguity and mixed perspectives by 500% (p-value$<.001$, t-test). It strongly indicates that our technique to control path inconsistency in taxonomy construction is effective.

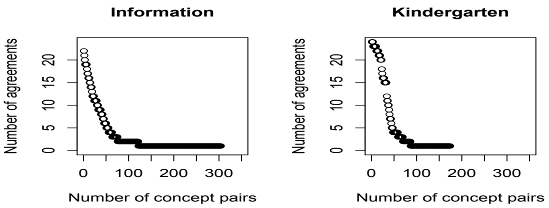


Figure 3: Agreements among participants for the parent-child pairs for datasets *information* and *kindergarten*.

5.3 Mixed Perspectives in Taxonomies

To better understand mixed perspectives in taxonomies constructed, we look for commonality and differences among the taxonomies constructed by the 24 participants for the same topic in the second user study. We break each taxonomy into parent-child pairs, and count how many participants agreed on a pair. The agreements range from 1 to 24. The taxonomies we examined are NAICS-2 and Web.

We plot the number of agreements for every concept pair and observe a long-tail power-law distribution for all datasets. Figure 3 shows that for the dataset "information", which contains about 300 unique concept pairs, while in "kindergarten", more than 200 unique concept pairs exist. This suggests that people use rich and diverse expressions to construct taxonomies and organize information differently within them. Although commonality (can be as high as 24 out of 24) and differences co-exist in taxonomies created for the same topic, the differences are much more dominate than the commonality.

We manually break down the types of differences in producing parent-child pairs into the following categories: *mixed parents* (a concept has different parent concepts due to word sense ambiguity), *mixed ancestors* (a concept is assigned to grandparents, not the direct parent), *mixed relation types* (a pair show relations other than *is-a*, such as part-of and affiliation), *new concepts* (participants add new concepts), *morphological differences* (plurals, -tion, etc), *errors* (clearly wrong relations, e.g.,

[3] E.g., queries "trip to DC", "Washington DC", "DC", and "Washington" were submitted for the topic "plan a trip to DC".

[4] Other types of errors were ignored in the assessment.

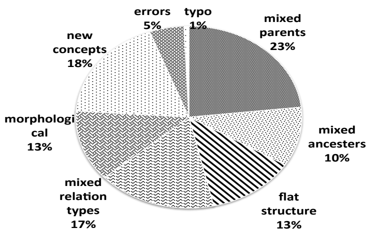


Figure 4: Sources of differences in NAICS-2 and Web.

infant→school director), *flat structure* (some participants liked to assign a large portion of concepts as children to the root), and *typo*.

Figure 4 illustrates the break-down of various types of differences. *Mixed parents* is the largest contributor with 23% share, followed by *new concepts* (18%) and *mixed relation types* (17%). Among all the types, *mixed parents*, *new concepts*, and *mixed relation types* indicate mixed perspectives or word sense ambiguity; in total they contribute about 58% differences in taxonomies. *Flat structure* and *mixed ancestors* are about confusions in taxonomy topology, which contribute about 23% differences. Other differences due to morphological changes, typos and errors contribute about 19% differences. The break-down reveals that mixed perspective, one of main foci in this paper, is indeed the biggest source of difference in taxonomy construction.

5.4 Self-agreement

Another doubt is that maybe the differences come from randomness? To find out if the variations among taxonomies is due to randomness, we designed a repeat phase in the second user study. We randomly invited 12 participants to repeat the same tasks in the same order 3 weeks[5] after the initial phase and compare the taxonomies constructed in both phases for the NAICS-2 and Web datasets.

We use Fragment-Based Similarity (FBS) proposed by (Yang, 2011) to calculate the similarity between taxonomies constructed in the initial phase and in the repeat phase by the same participant. FBS for two taxonomies T_i and T_j is calculated as: $FBS(T_i, T_j) = \frac{1}{max(U,V)} \sum_{p=1}^{m} sim_{cos}(t_{ip}, t_{jp})$, where U and V is the number of concepts in T_i and T_j respectively, m is the number of matched pairs based on the highest cosine similarity, sim_{cos} is the cosine similarity between vectors for subtrees of concepts t_ip and t_jp.

Self agreement (in FBS)	Max	Min	Average
per participant per dataset	1	0.37	0.74
per participant	0.81	0.63	0.74
per dataset	0.95	0.62	0.74

Table 1: Self-agreement; measured in FBS.

Table 1 indicate the self-agreement between taxonomies for any participant and/or any topic. The max self-agreement is as high as 1. The average self-agreement is 0.74, which is high at the range of FBS. It suggests that the participants are quite self-consistent when constructing taxonomies at different times. It builds the foundation for our study on multiple perspectives in taxonomy construction.

6 Conclusion

This paper explores techniques to *quickly* derive *task-specific* taxonomies supporting browsing in arbitrary document sets. It addresses two issues in taxonomy construction: path inconsistency due to word sense ambiguity and mixed perspectives, and task specifications in arbitrary collections. We tackle both issues in a supervised distance learning framework via minimizing distances along a path and using user inputs as training data, respectively. The user studies strongly suggest that the proposed techniques are highly effective in constructing browsing taxonomies as well as handling path consistency.

References

J. R. Bellegarda, J. W. Butzberger, Yen-Lu Chow, N. B. Coccaro, and D. Naik. 1996. A novel word clustering algorithm based on latent semantic analysis. In *Proceedings of the Acoustics, Speech, and Signal Processing, 1996. on Conference Proceedings., 1996 IEEE International Conference - Volume 01*, ICASSP '96, pages 172–175, Washington, DC, USA. IEEE Computer Society.

Matthew Berland and Eugene Charniak. 1999. Finding parts in very large corpora. In *Proceedings of the 27th Annual Meeting for the Association for Computational Linguistics (ACL 1999)*.

Rajendra Bhatia. 2006. *Positive definite matrices (princeton series in applied mathematics)*. Princeton University Press, December.

[5] The three week period ensured that participants only had limited memory of the details about the tasks.

Claudio Carpineto, Stefano Mizzaro, Giovanni Romano, and Matteo Snidero. 2009. Mobile information retrieval with search results clustering: Prototypes and evaluations. *Journal of American Society for Information Science and Technology (JASIST)*, pages 877–895.

Duen Horng Chau, Aniket Kittur, Jason I. Hong, and Christos Faloutsos. 2011. Apolo: making sense of large network data by combining rich user interaction and machine learning. In *CHI*, pages 167–176.

Gouglass R. Cutting, David R. Karger, Jan R. Petersen, and John W. Tukey. 1992. Scatter/Gather: A cluster-based approach to browsing large document collections. In *Proceedings of the fifteenth Annual ACM Conference on Research and Development in Information Retrieval (SIGIR 1992)*.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the web: an experimental study. In *Artificial Intelligence, 165(1):91-134, June*.

Christiane Fellbaum. 1998. *WordNet: an electronic lexical database*. MIT Press.

Roxana Girju, Adriana Badulescu, and Dan Moldovan. 2003. Learning semantic constraints for the automatic discovery of part-whole relations. In *Proceedings of the Human Language Technology Conference/Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT/NAACL 2003)*.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING 1992)*.

Zornitsa Kozareva and Eduard Hovy. 2010. A semi-supervised method to learn and construct taxonomies using the web. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1110–1118, Cambridge, MA, October. Association for Computational Linguistics.

Zornitsa Kozareva, Ellen Riloff, and Eduard Hovy. 2008. Semantic class learning from the web with hyponym pattern linkage graphs. In *Proceedings of the 46th Annual Meeting for the Association for Computational Linguistics (ACL 2008)*.

Krishna Kummamuru, Rohit Lotlikar, Shourya Roy, Karan Singal, and Raghu Krishnapuram. 2004. A hierarchical monothetic document clustering algorithm for summarization and browsing search results. *Proceedings of the 13th conference on World Wide Web WWW 04*, page 658.

Dawn Lawrie, W. Bruce Croft, and Arnold Rosenberg. 2001. Finding topic words for hierarchical summarization. In *Proceedings of the 24th Annual ACM Conference on Research and Development in Information Retrieval (SIGIR 2001)*, pages 349–357.

LCSH. 2011. Library of congress subject headings. `http://www.loc.gov/`.

P. C. Mahalanobis. 1936. On the generalised distance in statistics. In *Proceedings of the National Institute of Sciences of India 2 (1): 495*.

ODP. 2011. Open directory project. `http://www.dmoz.org/`.

Patrick Pantel and Marco Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 44th Annual Meeting for the Association for Computational Linguistics (ACL 2006)*.

Deepak Ravichandran and Eduard Hovy. 2002. Learning surface text patterns for a question answering system. In *Proceedings of the 40th Annual Meeting for the Association for Computational Linguistics (ACL 2002)*.

Mark Sanderson and W. Bruce Croft. 1999. Deriving concept hierarchies from text. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1999)*.

Sedumi. 2011. `http://sedumi.mcmaster.ca`.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (ACL/COLING 2006)*.

Yalmip. 2011. `http://users.isy.liu.se/johanl/yalmip`.

Hui Yang and Jamie Callan. 2009. A metric-based framework for automatic taxonomy induction. In *Proceedings of the 47th Annual Meeting for the Association for Computational Linguistics (ACL 2009)*.

Hui Yang. 2011. *Personalized Concept Hierarchy Construction*. Ph.D. thesis, Carnegie Mellon University. `http://www.cs.cmu.edu/~huiyang/publication/dissertation.pdf`.