Evaluating Task-Dependent Taxonomies for Navigation

Yuyin Sun
University of Washington
sunyuyin@cs.washington.edu

Adish Singla
ETH Zurich
adish.singla@inf.ethz.ch

Tori Yan
University of Washington
qiaoiyan@cs.washington.edu

Andreas Krause
ETH Zurich
krausea@ethz.ch

Dieter Fox
University of Washington
fox@cs.washington.edu

Abstract

Taxonomies of concepts are important across many application domains, for instance online shopping portals use catalogs to help users navigate and search for products. Task-dependent taxonomies, *e.g.*, adapting the taxonomy to a specific cohort of users, can greatly improve the effectiveness of navigation and search. However, taxonomies are usually created by domain experts and hence designing task-dependent taxonomies can be an expensive process: this often limits the applications to deploy generic taxonomies. Crowdsourcing based techniques have the potential to provide a cost-efficient solution to building task-dependent taxonomies. In this paper, we present the first quantitative study to evaluate the effectiveness of these crowdsourcing based techniques. Our experimental study compares different task-dependent taxonomies built via crowdsourcing and generic taxonomies built by experts. We design randomized behavioral experiments on the Amazon Mechanical Turk platform for navigation tasks using these taxonomies resembling real-world applications such as product search. We record various metrics such as the time of navigation, the number of clicks performed, and the search path taken by a participant to navigate the taxonomy to locate a desired object. Our findings show that task-dependent taxonomies built by crowdsourcing techniques can reduce the navigation time up to 20%. Our results, in turn, demonstrate the power of crowdsourcing for learning complex structures such as semantic taxonomies.

Introduction

Taxonomies are useful across many real-world applications and scientific domains. Online shopping portals such as Amazon use catalogs to organize their products in order to simplify the task of navigation and product search for their users. In scientific domains, many machine learning algorithms and intelligent agents including robots also use taxonomies to improve their performance on fundamental tasks such as object recognition (Zweig and Weinshall 2007; Marszałek and Schmid 2007; Wu, Lenz, and Saxena 2014; Ordonez et al. 2013), natural language understanding (Voorhees 1993; Resnik 1999; Bloehdorn, Hotho, and Staab 2005; Knight 1993), and others. Task-dependent taxonomies, *i.e.*, taxonomies adapted to a specific application/task or to a specific cohort of users, can be extremely beneficial (Deng et al. 2014). Such taxonomies are usually created by hiring domain experts, which is an expensive process and thus not possible for every single task. Alternatively, there are methods for creating taxonomies autonomously (Blei, Ng, and Jordan 2003; Blei et al. 2004). However, these methods lack the semantics/common sense of humans and often produce taxonomies that are inefficient in terms of their end-use. Hence, many applications (Lai et al. 2011; Deng et al. 2009; Zweig and Weinshall 2007) are limited to using generic taxonomies such as WordNet (Fellbaum 1998).

Crowdsourcing based techniques. Recently, several crowdsourcing-based techniques (Chilton et al. 2013; Bragg, Mausam, and Weld 2013; Sun et al. 2015) (with well-designed algorithms and prepared questions) have demonstrated that even non-experts have the potential to build task-dependent taxonomies. Chilton et al. and Bragg, Mausam, and Weld introduce techniques for creating taxonomies based on the co-occurrence of multi-label object annotation. Sun et al. take a Bayesian approach and propose an active learning algorithm for building taxonomies. Their approach can capture uncertainty over taxonomies by producing a distribution over these taxonomies instead of producing one single taxonomy as output. Most of the work (Bragg, Mausam, and Weld 2013; Sun et al. 2015) evaluates the quality of the output taxonomies using accuracy with respect to some gold-standard knowledge base such as WordNet (Fellbaum 1998). However, there is no quantitative evaluation of the end-to-end deployment of taxonomies on the task for which they were developed.

Research questions. The primary goal of our work is to understand whether crowdsourcing provides an effective solution to create taxonomies of knowledge. To this end, we seek to answer the following 3 research questions: (i) do task-dependent taxonomies built by crowdsourcing techniques improve the end-to-end performance compared to using generic taxonomies built by experts?; and (ii) does the uncertainty captured by probabilistic approaches further help to improve the efficiency of performing the navigation tasks? Answering these questions will in turn shed light on the third, more general research question: 3) *Can crowdsourcing-based techniques provide practical solutions to complex structural learning tasks such as building taxonomies?*

Our approach. In this paper, we present a quantitative study to measure the effectiveness of different taxonomies in the context of an end-to-end application. Specifically, we

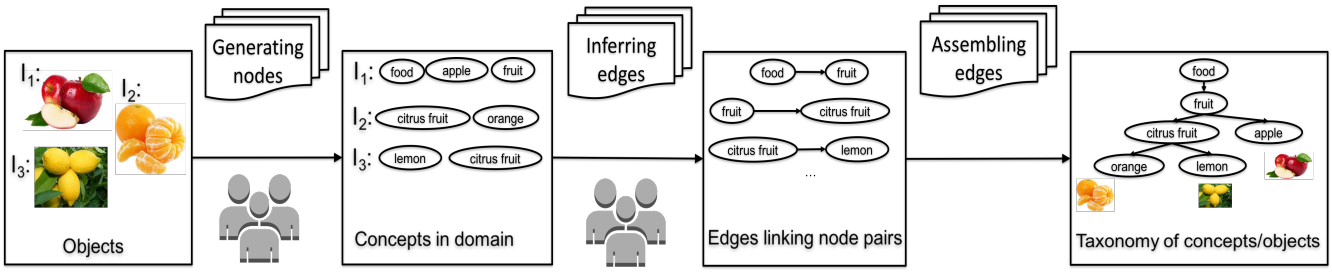


Figure 1: Illustration of the workflow of building task-dependent taxonomies via crowdsourcing. Collection of the tags / nodes and inference of the edges are done via crowdsourcing. Assembling edges to get a final taxonomy is performed by an algorithm.

compare the task-dependent taxonomies built by crowdsourcing techniques to a generic taxonomy built by experts. To start, we use a crowdsourcing based technique (Sun et al. 2015) to create a taxonomy using the AMT. Then, to perform the user study for quantitative evaluation, we design randomized behavioral experiments on AMT for a navigation task resembling many real-world applications, such as conducting a product search in online shopping portals. Participants in the user study differ from AMT workers who helped build the taxonomy we are evaluating. As part of our user study, we record various quantitative metrics such as the time of navigation, the number of clicks performed, and the search path taken by a participant to locate the target.

Our results. Our findings from the user study show that: (i) the task-dependent taxonomies built by crowdsourcing techniques can reduce the navigation time by up to 20%; and (ii) the uncertainty/distribution over the taxonomies can be exploited to help users perform navigation tasks more efficiently. These results affirmatively answer our research questions and confirm that crowdsourcing provides an effective solution to building task-dependent taxonomies. This, in turn, demonstrates that crowdsourcing can provide a practical solution for learning complex structure using responses to simple questions from non-experts. Furthermore, our design of behavioral experiments can serve as an example for crowdsourcing-based user studies of complex systems.

The rest of this paper is organized as follows: We will first introduce the background of building taxonomies via crowdsourcing. Next, we will present the data collection step and the experimental setup. Then, we will discuss the results from the user study. Finally, we will discuss related work and conclude.

Building Taxonomies via Crowdsourcing

We begin by reviewing background information on crowdsourcing based techniques to build taxonomies. The next section will use one of these techniques to build the taxonomy on our dataset for evaluation via a user study.

Different Components to Specify a Taxonomy

A task/domain is specified by a set of objects. A taxonomy represents a knowledge base that organizes the objects into a hierarchical tree structure. Two main components are necessary to specify a taxonomy completely: the set of nodes in the tree, and the set of edges connecting these nodes. Figure 1 illustrates the workflow involved in building a taxonomy on a toy example, as discussed further below.

Generating nodes. A natural approach of node generation is to represent the objects related to the task/domain for which the taxonomy is being built by a set of concepts or tags. These tags correspond to the nodes in the taxonomy tree. Each node is associated with a set of objects that is tagged by the node or any of its descendant nodes. Most crowdsourcing-based methods (Chilton et al. 2013; Sun et al. 2015) begin by collecting a set of tags for the objects (the first step in Figure 1). For example, CASCADE (Chilton et al. 2013) shows a set of objects via their text descriptions to the crowdsourcing workers and asks them to provide tags they will use to refer to these objects. Sun et al. take a similar approach. However, instead of showing the text descriptions, the authors show visual images of the objects and ask the workers to annotate them with tags.

Inferring edges. The next step is to specify the directed edges that connect these nodes (the second step in Figure 1). In the hierarchical tree of the taxonomy, each node is more general than any of its descendant nodes. Inferring edges in the tree is a challenging problem as it requires having global knowledge of the complete set of nodes. In the crowdsourcing setting, non-expert workers would not possess such a global knowledge and are better at answering questions that can be answered with local/partial knowledge. So the key to designing crowdsourcing based algorithms for building taxonomies is to break down the problem requiring global knowledge into simpler problems/questions requiring only local knowledge to answer. The algorithms then fuse the answers to these simple questions to generate the final global structure of the taxonomies (the last step in Figure 1).

Decomposing the Structure Learning Problem

We now categorize existing crowdsourcing techniques for building taxonomies into two main categories based on how they decompose the global structure learning problem into simpler problems and questions.

Indirect questions to infer edges. CASCADE (Chilton et al. 2013) and DELUGE (Bragg, Mausam, and Weld 2013) are representative techniques of this category, which use local questions without direct connection to the structure of the taxonomy. Chilton et al. introduce CASCADE, a workflow for creating taxonomies based on the co-occurrence of multi-label annotation of items. CASCADE takes a set of objects and the corresponding set of tags to generate simple ques-

tions/microtasks as follows: each microtask consists of one object and one tag. CASCADE then asks a worker to vote on whether the object fits the tag. In the fusion step, it creates nested categories representing the parent-child relationship of these tags. DELUGE improves the efficiency of this annotation process to reduce the number of questions required to learn a taxonomy with the same accuracy.

Direct questions to infer edges. Sun et al. take a Bayesian approach and propose an active learning algorithm for building hierarchies/taxonomies. Their approach directly asks questions related to the structure of the taxonomy. Specifically, they ask *path questions*, *e.g.*, "Should there be a path from 'apple' to 'food' in the target taxonomy?", stated informally as "Is 'apple' a type of 'food'?". These questions can be answered by crowdsourcing workers using only local/partial knowledge. Sun et al. provide a probabilistic model to efficiently maintain a distribution over all possible tree structures to represent the fused knowledge from the answers obtained from workers. At each iteration, the model updates the distribution based on the answer to a path question. Intuitively, if a worker gives a positive answer to a path, his/her answer will increase the probability of the taxonomy trees with that path; otherwise, the probabilities of the corresponding trees will be reduced.

Technique Used for Evaluation

In this paper, we are using Sun et al.'s method for building task-dependent taxonomies and then compare them to the generic taxonomy of WordNet. The key reasons for using this technique are two-fold. First of all, the experimental study performed by Sun et al. shows that this Bayesian approach is empirically more cost-efficient compared to CASCADE and DELUGE, for which the cost quantifies the number of questions asked to the participants in order to produce a taxonomy of desired accuracy. Second, this approach outputs a distribution over taxonomies that in turn enables various operations: (i) using MAP (maximum a posteriori probability) inference to compute the most representative taxonomy, and (ii) estimating the marginal likelihood of edges efficiently. These marginal likelihoods in turn allow to capture the uncertainty that naturally exists in semantic taxonomies, *e.g.,* the answer to the question "Is 'candy' a type of 'dessert'?" or "Is 'candy' a type of 'sugar'?" is inconsistent depending on the participants' beliefs. Our second research question concerns the use of these uncertainties, and whether it can improve the efficiency of performing the navigation tasks relative to the use of a single representative taxonomy.

Application Domain and Navigation Tasks

In this section, we discuss the application domain, the process of data collection for this domain, and the navigation tasks we use for evaluation.

Application Domain

In this paper, we consider the application domain of *kitchen* objects. This domain involves objects used in the kitchen in everyday life such as food items, kitchen appliances, cooking utensils, *etc*. We chose this domain for 2 reasons: (i) the navigation tasks in this domain represent simple everyday scenarios, ensuring that the AMT workers are familiar with the domain to perform the navigation tasks for evaluation, and (ii) it is a rich domain that captures the intricacies of real-world applications, such as online shopping portals.

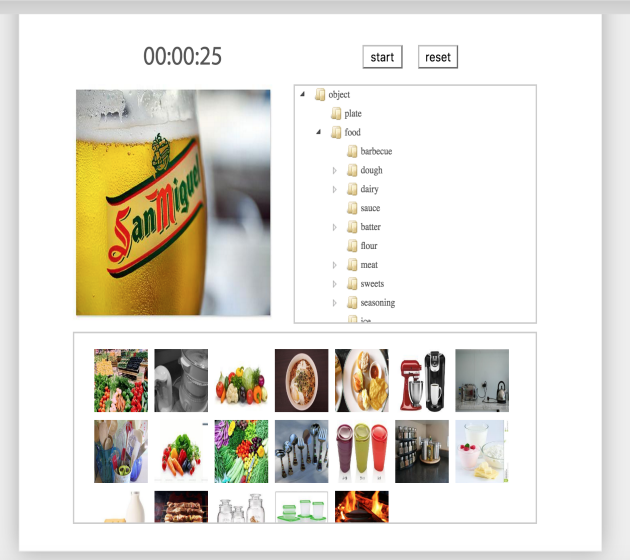


Figure 2: This figure shows the interface provided to the AMT workers to perform the navigation tasks: (i) the target object is shown in the *top-left* panel; (ii) the *top-right* panel shows the taxonomy – a navigation system; and (iii) the *bottom* panel shows a subset of the images associated with the current node clicked by the user.

Collecting objects for our domain. We begin by collecting the most frequently used keywords in the kitchen domain. We extract the words/phrases representing most commonly used kitchen concepts from WikiHow[1] articles using the Stanford parser (Klein and Manning 2003). We then do a two-step filtering process as follows: first, we keep the top 100 frequently occurring keywords as candidates[2]; then, we filter out the candidates that are not in WordNet[3]. This filtering process yields a set of about 70 *seeding* keywords — these are the concepts that represent the kitchen domain. We then use these keywords to search for images via Google's image search engine to get exemplary images. For each keyword, we collected about 50 images, thus giving us a total of $3,500$ images. These images represent the objects for the kitchen domain, *c.f.*, "Objects" in Figure 1. In the next section, we describe the different taxonomies that we build to organize these objects.

[1] http://www.wikihow.com

[2] The first filtering step and other parameters are set to control the cost of doing the experiments needed for this paper. However, we did not tune these parameters and the approaches can be applied to larger problems.

[3] This second filtering step is to have a reasonable WordNet taxonomy that can be used as a baseline taxonomy for the comparison.

Navigation Task

Figure 2 illustrates a navigation task for the kitchen domain using one of the taxonomies we evaluate. This is the same interface provided to AMT workers during our user studies. The goal of a navigation task is to localize a target object from the set of objects in the domain — the target object is shown in the top-left panel of Figure 2 and randomly picked from all candidate objects. All candidate objects are organized using a taxonomy represented by nodes and edges as discussed in the background section. The user has access to this taxonomy as the *navigation system*, shown in the top-right panel in Figure 2. Recall that each node in the taxonomy is associated with a set of objects that is tagged by this node or any of its descendant nodes. When a user clicks on a node, a random subset of the images belonging to that node (or any of its children) appears (the bottom panel of Figure 2). The user can search through these images to find the target. Or the user can use these images as an indicator of the kind of objects expected to appear under the current node, then click finer-grained nodes beneath the current node to look for the target. During the navigation task, the user can also backtrack the search path by clicking on a non-descendant node. The task finishes when the user finds the target object in the bottom panel and clicks on it. This navigation task resembles many real-world applications. For example, many online shopping portals, such as Amazon[4] and BestBuy[5], organize their products in a taxonomy to help their customers navigate and search efficiently.

Metrics. Our primary goal is to quantitatively measure the efficiency of the taxonomies as a navigation system. The natural metric is the time a user spends per task, *i.e.*, time to localize the target object using a given taxonomy. A shorter time suggests that the system is more efficient at guiding users to find the target object. Another metric of interest is the number of clicks performed before a user localizes the target object. Intuitively, this number captures whether a taxonomy matches the common sense. The number of backtracking steps done by a user while navigating the taxonomy tree is another metric we measure. A large number of backtracking steps indicates that the taxonomy might confuse its users.

Taxonomies Used for Evaluation

In this section, we present the specific taxonomies that are used for evaluation in our work. We compare three different taxonomies: 1) the WordNet taxonomy (Fellbaum 1998), 2) MAP (the most representative) taxonomy created by (Sun et al. 2015), and 3) the multifaceted taxonomy created by (Sun et al. 2015) to capture uncertainty.

WordNet Taxonomy

We would like to compare with a taxonomy created by domain experts. Ideally we want to use an expert-built task-dependent taxonomy, however, it is not available and difficult to create. Therefore, we decided to use an expert-built generic taxonomy in this work. We are using WordNet (Fellbaum 1998) as it is commonly deployed by various application domains. Moreover, many other expert-built taxonomies, such as BabelNet (Navigli and Ponzetto 2012) and WikiTax2WordNet (Ponzetto and Navigli 2009), use WordNet as the backbone and are built by adding more nodes or mapping information from other sources such as Wikipedia. Therefore, evaluating WordNet taxonomy gives us a reasonable indication of the results we expect from other expert-built generic taxonomies.

WordNet, a large taxonomy containing over 150,000 words, groups words that are roughly synonymous into *synsets*. These synsets are then connected to each other based on semantic relations, represented as a Directed Acyclic Graph (DAG). To generate a WordNet-based taxonomy for our domain, we extract a taxonomy from WordNet covering the 70 seeding keywords representing the concepts for kitchen domain as described below.

First, we locate the nodes in WordNet's DAG corresponding to the 70 keywords. Although all 70 keywords are in WordNet (note that we filtered the keywords to keep only the ones that are in WordNet), it is still a non-trivial task to locate the nodes in the DAG that are relevant to the kitchen domain because the same word may have different meanings *i.e.*, it could correpond to different synsets/nodes in WordNet. Therefore, whenever a keyword is mapped to multiple nodes in WordNet, we use the following heuristic to decide which synset/node in the WordNet has the semantics related to the kitchen domain. The heuristic will pick a synset if it has 'food', 'instrumentality' or 'appliance' as its ancestor synsets; otherwise, it picks the synset based on priority ordering defined by the match of the target keyword in the *lemmas* (semantic sense) of the synsets. It is worth noting that the crowdsourcing-based technique we use (Sun et al. 2015) faces a similar problem of polysemy in collecting tags in the first step, *c.f.*, Figure 1. For example, 'apple' could represent the fruit or the corporation. However, crowdsourcing based techniques resolve this naturally as the tag collection process involves showing images of the objects to the workers.

Then, after locating the nodes in the DAG for the 70 seeding keywords, we greedily find the minimum spanning tree in the DAG to cover these 70 nodes. To find this tree, the algorithm initializes a forest with 70 nodes. At each iteration, it picks a pair of nodes, adds their lowest common ancestor into the forest, and connects these two nodes with the new ancestor node. The algorithm then repeats this process until the forest becomes a single-rooted tree. After executing this algorithm, we have a total of 96 nodes. Figure 5 (a) shows this WordNet taxonomy corresponding to the 96 nodes; the 70 nodes corresponding to the seeding keywords of kitchen domain are marked in *Blue*.

Crowdsourcing-Based Taxonomies

Next, we use the method of Sun et al. for building crowdsourcing-based taxonomies. Following the workflow in Figure 1, we first collect tags for the object images corresponding to the 70 seeding keywords representing the kitchen

[4] http://www.amazon.com/gp/site-directory
[5] http://www.bestbuy.com/

domain.[6] We show the 70 representative object images to the AMT workers and ask them to provide 3 distinct tags for each image they see. Furthermore, each image is shown to 3 distinct workers. At the end, we aggregate all the tags, and we keep only the tags that are used by more than 5 workers to refer to these objects. We combine the newly collected tags with the original 70 keywords to get the final set of 104 tags/nodes. We then run the sequential version of the method proposed in (Sun et al. 2015) to learn the taxonomy over the set of these 104 nodes. As discussed in the previous section, the output of the method is a distribution over all taxonomy trees with 104 nodes.

MAP taxonomy. This is the taxonomy obtained by performing MAP (maximum a posteriori probability) inference and is the most representative taxonomy (Sun et al. 2015). Figure 5 (b) shows this MAP taxonomy corresponding to the 104 nodes; the 70 nodes corresponding to the seeding keywords of the kitchen domain are marked in *Blue*.

Multifaceted taxonomy. The MAP taxonomy ignores the uncertainty that naturally exists in semantic taxonomies. One reason for this uncertainty is the multifaceted classification (Priss 2008) problem as there are different criteria that can be considered by humans to classify and categorize objects. The most common example are online shopping portals, where products can be categorized by price, type, maker and so on. In order to incorporate this uncertainty, we create an augmented taxonomy to utilize the uncertainty to create more options for users to localize the target object in the taxonomy (Sifer 2006). We begin by computing the marginal likelihood of the edges from the distribution of the taxonomies obtained above — their Bayesian approach provides a computationally tractable method to compute these likelihoods. We then keep only the edges that have a marginal likelihood of more than 0.1. Next, we append the MAP taxonomy tree with these additional edges. As an example, *c.f.*, Figure 5, we obtain the edges from 'sugar' to 'candy' and 'dessert' to 'candy' among these high likelihood edges. The MAP taxonomy tree (Figure 5 (b)) already has an edge from 'dessert' to 'candy'. Hence, we append this tree with an edge from 'sugar' to 'candy'. It is important to note that the resulting structure after augmenting these edges is not a tree but a DAG. For the user studies, we consider showing it to users as a tree for easier interpretation and navigation. To convert this DAG to a tree, we do the following: For each node x with $n > 1$ parent nodes represented by the set π_x, we replicate the sub-tree rooted at x for n times and add this sub-tree as a child node to each of the parent nodes in π_x. This treatment to the DAG gives us a taxonomy with replicated nodes (as shown in Figure 5 (c)) and we call this taxonomy a Multifaceted taxonomy. In Figure 5 (c), the set of nodes that was replicated and added to the MAP taxonomy at different positions are highlighted in *Red*.

[6]Note that we have 50 images per seeding keyword. However, in order to make this tag collection process cost-efficient, we randomly selected 5 images per keyword, then grouped these 5 images into one big image, and used this as the representative object image to collect tags.

Structural Comparison of Taxonomies

We first compare the three taxonomies based on various structural and qualitative aspects.

Number of nodes. The MAP taxonomy (*c.f.*, Figure 5 (b)) contains 104 nodes compared to 96 nodes in the WordNet taxonomy (Figure 5 (a)). Note that both these taxonomies contain the same set of 70 seeding keywords we begin with. A careful examination reveals that the MAP taxonomy includes many *instance names* that specify some seeding keywords. For example, 'fish' is a seeding keyword, and 'salmon' is a particular type of 'fish'. The reason for seeing these instance names is that we show images corresponding to the 'fish' object to collect the tags, and upon seeing 'salmon' food dish in the image, some workers may annotate the image with the tag 'salmon'. These specific instance names do not exist in the WordNet taxonomy since we generate the taxonomy by adding only ancestor nodes of the seeding keywords in the DAG of the WordNet. Furthermore, if we remove all nodes related to these specific instance names from the MAP taxonomy, we are left with 91 nodes — a number slightly lower than the size of the WordNet taxonomy at 96 nodes. This suggests that non-experts might use a more compact set of tags/concepts to refer to the same collection of objects for a certain application domain.

Depth/width of taxonomy. The MAP and the Multifaceted taxonomies both have a depth of 6 and a width of 16. On the other hand, the WordNet taxonomy has a depth of 8 and a width of 8. These numbers suggest that non-experts tend to build shallower but wider trees, while experts might prefer creating deeper but narrower trees.

Words in taxonomy. Taxonomies built via crowdsourcing have more familiar keywords/concepts associated with the nodes. The WordNet taxonomy has more obscure/infrequently occurring keywords such as 'nutriment', 'instrumentality', 'concoction', and others. These keywords in the WordNet taxonomy might be unfamiliar to ordinary users, in particular non-native English speakers.

Uncertainty. The taxonomy in Figure 5 (c) captures some interesting information about the uncertainty of the edges. One example is the 'blender' node. As per the MAP taxonomy in Figure 5 (b), 'blender' is a type of 'kitchen appliance'. However, in the Multifaceted taxonomy, a 'blender' could also be a kind of 'container'. This uncertainty captures the notion of multifaceted classification: humans use different attributes to classify the same objects/ concepts. Some users classify 'blender' according to its essential functionality as an appliance, while others believe that'blender' can be used as a 'container'. Keeping two 'blender' nodes in one taxonomy thus potentially provides users with more choices in navigating the taxonomy to find the target objects. A user who believes 'blender' is a 'container' will not find the target object directly in the MAP taxonomy, leading to more clicks and backtracking steps. But the user can quickly locate 'blender' in the Multifaceted taxonomy. The taxonomy in Figure 5 (c) also illustrates other sorts of uncertain edges. For example, some users are confused about the relation between 'dough' (flour mixed with less water) and 'batter' (flour mixed with more water). In the Multifaceted taxonomy, we can see edges from 'dough' to 'batter' and from 'batter' to 'dough'. These

Table 1: Means $\pm$ standard deviations of time, the number of clicks, and the backtracking steps (bts) for finding one target. MF stands for Multifaceted taxonomy.

Metrics	Taxonomies		
	WordNet	MAP	MF
time(s)	60.2 ± 76.3	49.8 ± 57.2	47.9 ± 63.2
clicks	6.0 ± 6.7	4.4 ± 5.2	4.0 ± 5.7
bts	2.3 ± 3.3	1.3 ± 2.7	1.1 ± 3.0

Table 2: Means $\pm$ standard deviations of different metrics for finding a target from the nodes with uncertainty. MF stands for Multifaceted taxonomy.

Metrics	Taxonomies	
	MAP	MF
time(s)	49.1 ± 44.9	41.3 ± 44.6
clicks	4.3 ± 4.9	2.8 ± 2.5
bts	1.6 ± 2.5	0.8 ± 1.4

kind of cycles suggest that people see these two nodes as synonyms.

User Study for Quantitative Evaluation

We performed our user study on AMT by recruiting workers to perform navigation tasks. Each worker participating in the study is allowed to accomplish the navigation task only two times: one task uses the WordNet taxonomy, and the other task uses either the MAP taxonomy or the Multifaceted taxonomy. The order of these two tasks is chosen randomly. We do not allow workers to perform tasks with both the MAP and the Multifaceted taxonomies as they might learn one taxonomy from another, thereby affecting the validity of the study. We ran these online experiments on AMT for 3 days restricted to US workers with more than 95% approval rate, and over 800 distinct workers participated in the study. Considering only the tasks successfully accomplished by the workers, we have the following number of completed tasks: 730 using the WordNet taxonomy, 343 using the MAP taxonomy, and 436 using the Multifaceted taxonomy.

Task-Dependent vs. Generic Taxonomies

We first compare the efficiency of using task-dependent taxonomies to the generic taxonomy to answer the first research question (*viz.*, *Do task-dependent taxonomies built by crowdsourcing techniques improve the end-to-end efficiency compared to generic taxonomies built by experts?*) Among the three taxonomies we use, both the MAP and the Multifaceted are task-dependent taxonomies, and the WordNet taxonomy is generic.

Table 1 shows the summary statistics and the average performance of tasks for three metrics, including: time, number of clicks, and number of backtrackings per task, where backtracking indicates that a user navigated back up through the taxonomy after discovering a potential dead end. Figure 3 shows the distribution of the different metrics for the different types of taxonomies. Significance testing involves unpaired t-tests.

Results show that using the WordNet taxonomy performs significantly worse than using both the MAP and the Multifaceted taxonomies regarding all three metrics ($p < 0.05$). Among the three taxonomies, the Multifaceted taxonomy performs the best. It outperforms the WordNet taxonomy by 20% with respect to time, by 33% with respect to the number of clicks, and by 49% with respect to the number of backtrackings.

These findings show that non-expert workers at AMT can build useful task-dependent taxonomies. The task-dependent taxonomies are designed for the kitchen application domain and can therefore be deployed by navigation tasks to achieve efficient navigation performance. On the other hand, the generic taxonomy does not consider any task information. Although it is created by experts and is correct in its content, it does not provide the best possible navigation performance in the desired target domain.

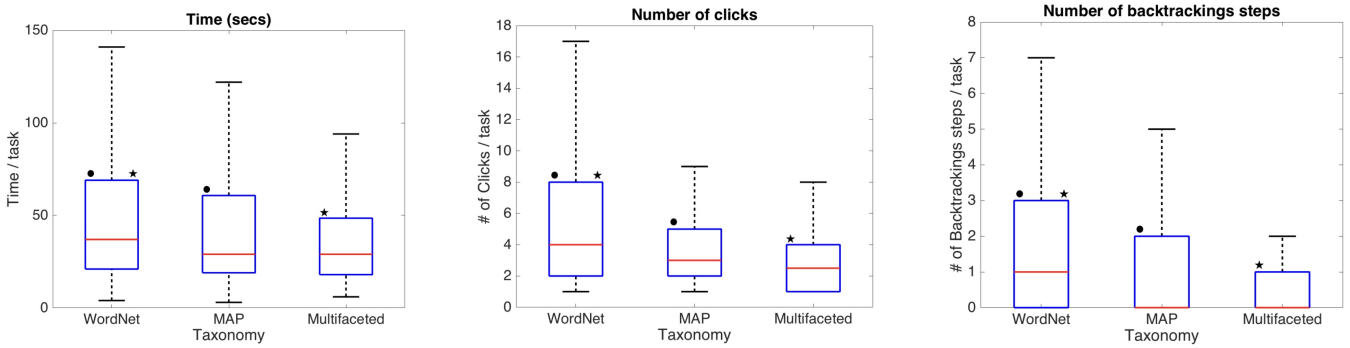
MAP vs. Multifaceted Taxonomies

We examined the tasks using MAP or Multifaceted taxonomies and tried to answer the second question (*viz.*, *Does the uncertainty captured by probabilistic approaches help?*). To determine the effect of using uncertainty information, we segmented all the 70 nodes into two groups (with and without uncertainty) based on whether there are replicated nodes in the Multifaceted taxonomy. We did this because: 1) the MAP and Multifaceted taxonomies are different only on these nodes, and 2) these nodes have high uncertainty, letting us evaluate the difference between using and not using uncertainty information in the taxonomies.

First, we compare the performance of the taxonomies on the group of nodes with uncertainty. The evaluation metrics of the two taxonomies are summarized in Table 2. The average time, the number of clicks, and the number of backtrackings using the Multifaceted taxonomy are consistently lower than using the MAP taxonomy. Among the three metrics, both the numbers of clicks and backtracking results are significantly different between the two taxonomies, with both p values < 0.01. The difference on the time metric is not as significant as the other two metrics given the p value of 0.218. We hypothesize that the time does not exactly reflect the time taken by users to navigate through the taxonomy because it also includes the time spent for searching for target images.

Next, we report the backtracking rates of the nodes with uncertainty in Table 3. The backtracking rates tell us how often a user has to backtrack at least once to localize the target object. Looking at the backtracking rate of MAP taxonomy, users backtrack more often on the nodes with high uncertainty than on the nodes without uncertainty (54% vs. 37%). This suggests that the uncertainty over nodes in the taxonomy does capture the uncertainty users have when they perform navigation tasks.

The results also show that using the Multifaceted taxonomy clearly outperforms the MAP taxonomy on searches relating to the nodes with uncertainty, as confirmed by a reduction of the average backtracking rate from 54% to 37% (about 30% marginal improvement). Although the paired t-test is not significant (p = 0.06), using the Multifaceted taxonomy



(a) Time (in secs) per task. $p^{\bullet} < 0.05$; $p^{\star} < 0.01$.

(b) Number of clicks per task. $p^{\bullet} < 0.001$; $p^{\star} < 0.0001$.

(c) Number of backtracking steps per task. $p^{\bullet} < 0.0001$; $p^{\star} < 0.0001$.

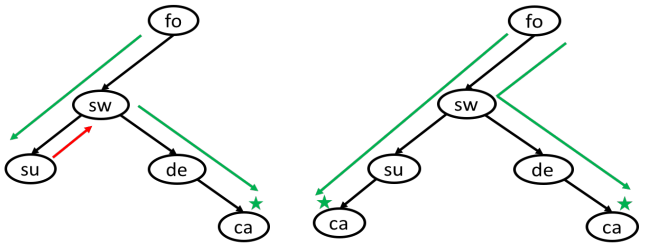
Figure 3: Boxplots showing the evaluation metrics by the types of taxonomies. The red bars represent the median, and the blue boxes represent first and third quartile of the distribution. Two statistical significance tests are performed: "$\bullet$" compares the WordNet taxonomy with the MAP taxonomy and "$\star$" compares the WordNet taxonomy with the Multifaceted taxonomy.

Table 3: The backtracking rates (%) of nodes with uncertainty when different taxonomies are used. MF stands for Multifaceted taxonomy. "w/ uncertainty" means the group of nodes with uncertainty, and "w/o uncertainty" indicates the group of nodes without uncertainty.

Node	BT-MAP(%)	BT-MF(%)
'sauce'	44	17
'flour'	100	8
'salad'	0	18
'grill'	67	100
'stove'	60	88
'blender'	75	89
'microwave'	100	90
'pot'	100	14
'fork'	0	0
'juice'	20	33
'candy'	67	17
'chicken'	50	25
'batter'	80	60
'dough'	33	20
w/ uncertainty	54 ± 31	37 ± 34
w/o uncertainty	37 ± 34	36 ± 27

outperforms the MAP taxonomy for most nodes. This is not surprising since Multifaceted provides alternative paths for uncertain nodes, thereby avoiding the backtracking required by the MAP taxonomy.

We examined the common patterns of user searching for target nodes with uncertainty using the MAP and Multifaceted taxonomies. Figure 4 shows how users look for *candy* using different taxonomies. We show the parts of the taxonomies relevant to the target node *candy*, and highlight the path taken by the users to look for *candy*. When doing so, about half of the users first go through *sugar* from *sweet* because they might believe that *candy* is a type of *sugar*. The other users take the path from *sweet* to *dessert*. The MAP taxonomy does not consider the uncertainty information and puts *candy* only as a child of *dessert*. Users going in the direction of *sugar* could not find the target in the MAP taxonomy and therefore have to backtrack and search the other alternative path, from *dessert* to *candy*. On the other hand, the Multifaceted taxonomy consider the uncertainty over the location of *candy*. Therefore, it puts candy under both possible locations. Hence, the users could find *candy* by taking either path in the Multifaceted taxonomy. Using uncertainty reduces the backtracking rate for sugar from 67% to 17%.



(a) The sub-tree for 'candy' from the MAP taxonomy.

(b) The sub-tree for 'candy' from the Multifaceted taxonomy.

Figure 4: Illustration of the search paths taken by a user to search for 'candy'. The following abbreviations are used: 'fo' for 'food', 'sw' for 'sweet', 'su' for 'sugar', 'de' for 'dessert', and 'ca' for 'candy'. The green edges indicate the path-taken by a user, the red edge indicates a backtracking, and $*$ indicate the target.

Related Work

Crowdsourcing for user studies. Crowdsourcing platforms provide easy, on-demand and low-cost access to a large pool of diverse participants, thereby opening up opportunities to conduct large-scale online user studies (Kittur, Chi, and Suh 2008; Mason and Suri 2012). In the computer science community, crowdsourcing-based user studies have been widely deployed to evaluate the performance of algorithms in various application domains, such as graphical perception design (Heer and Bostock 2010; Micallef, Dragicevic, and Fekete 2012), web search (Alonso and Baeza-Yates 2011; Chandar and Carterette 2012), recommendation systems (Maryam and Popescu-Belis 2012), computer vision (Deng et al. 2015), *etc.*

Evaluating ontologies. Taxonomies are usually major

components of an ontology; hence, the methodologies for evaluating ontologies significantly overlap with the evaluation of taxonomies. When a ground-truth ontology or knowledge base is available, precision-recall-based metrics inspired from information retrieval evaluations have been used (Maedche and Staab 2002). However, these methods are inadequate otherwise. On the other hand, data-driven methods (Porzel and Malaka 2004) evaluate target ontologies to match the knowledge included in some existing corpus or test data set. However, they do not quantify the quality/utility of a target ontology in terms their efficiency of performing the desired task. One orthogonal direction for evaluating taxonomies is to study the *cognitive feasibility* of the taxonomy by measuring participants' reactions to the relations entailed by the structure of the taxonomy (Evermann and Fang 2010). Building on the ideas of Evermann and Fang, (Mortensen, Musen, and Noy 2013) proposed a crowdsourcing-based user study to evaluate/verify target ontologies. However, (Mortensen, Musen, and Noy 2013; Evermann and Fang 2010) rely on microtask methods of ontology verification wherein participants answer simple binary questions, such as "Is every 'heart' an 'organ'?".

Task-driven evaluation of taxonomies. Task-driven evaluations (Brewster et al. 2004) directly measure the end-to-end performance of different taxonomies. These methods (Brewster et al. 2004) define simple microtasks, *e.g.*, binary questions to evaluate the accuracy and coverage of the taxonomy content. The downside of these methods is that only local knowledge of the target taxonomy usually suffices to answer these questions. Our approach is a type of task-driven evaluation method. However, unlike the traditional task-driven methods that focus on assessing the content via simple questions, our approach focuses on evaluating the efficiency of using a taxonomy for doing *inference*: We rely on a more sophisticated navigation task to assess the taxonomy as a whole, representing a knowledge base in a hierarchical structure.

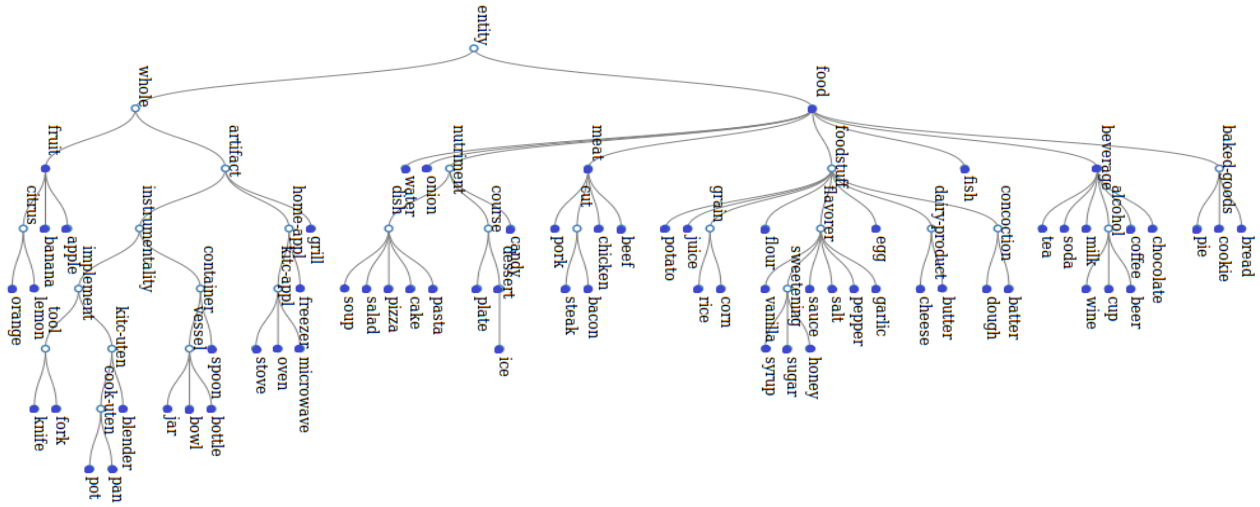
Conclusion

In this paper, we evaluated crowdsourcing-based techniques for building task-dependent taxonomies. Our experiments show that task-dependent taxonomies built by crowdsourcing techniques can significantly improve the efficiency of navigating and searching for target objects compared to using a generic taxonomy created by experts. The results also show that using multifaceted taxonomies to capture uncertainties over node positions reduces the number of clicks/backtracking steps performed by users to find objects. These results demonstrate that crowdsourcing-based techniques can be deployed for complex structural learning tasks such as building semantic taxonomies.

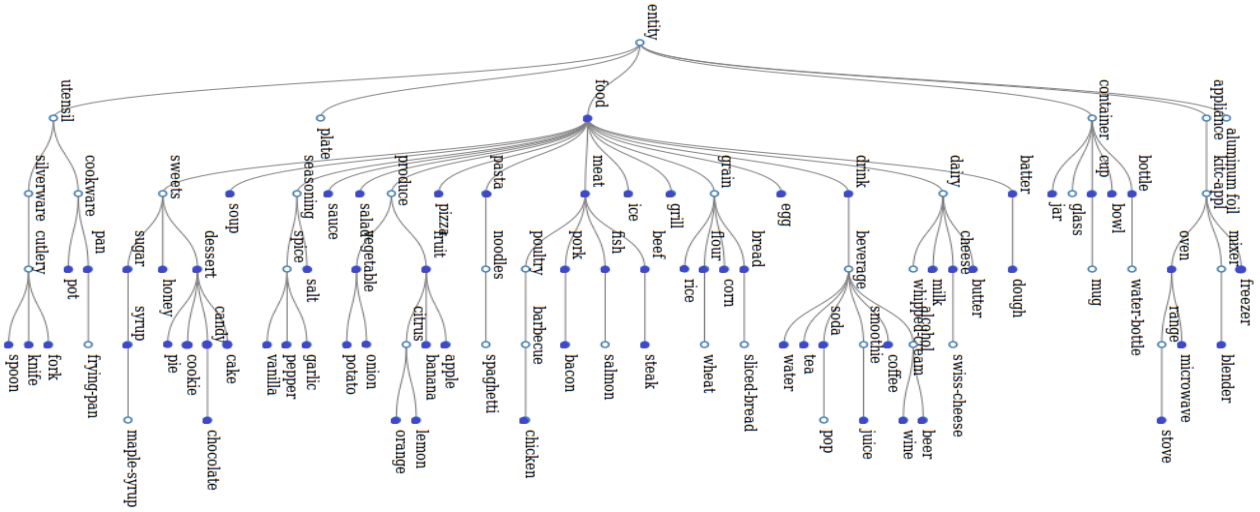
In future work, it would be interesting to extend the analysis to more application domains and evaluate other popular taxonomies. In particular, applying this analysis to domains with existing expert-built task-dependent taxonomies, such as Amazon shopping catalog, would be useful. Another interesting future direction would be evaluating the quality of taxonomies in doing other types of tasks besides navigation tasks, for example, evaluating the performance of using taxonomies to teach users the knowledge of the domain.

Acknowledgement

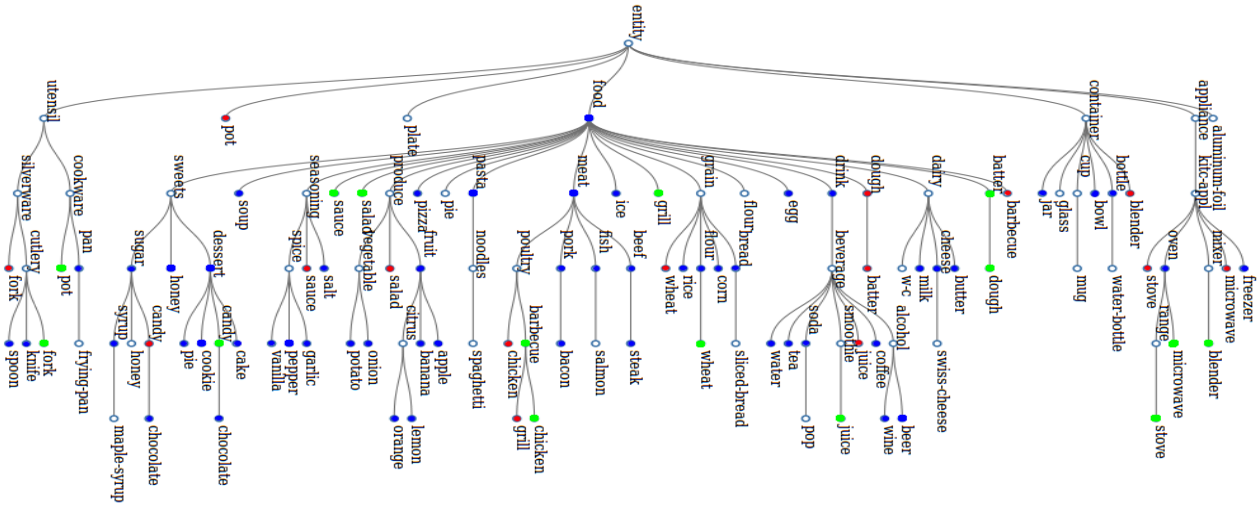
This work was funded in part by the Intel Science and Technology Center for Pervasive Computing, ONR grant N00014-13-1-0720, and the Nano-Tera.ch program as part of the Opensense II project. Adish Singla acknowledges support by a Facebook Graduate Fellowship.



(a) WordNet taxonomy for the kitchen domain.



(b) MAP taxonomy built by crowdsourcing (Sun et al. 2015) for the kitchen domain.



(c) Multifaceted taxonomy capturing uncertainity built by crowdsourcing (Sun et al. 2015) for the kitchen domain.

Figure 5: Three taxonomies for the kitchen domain used for evaluation. In (a) WordNet and (b) MAP taxonomy, the 70 nodes corresponding to the seeding keywords of the kitchen domain are marked in *Blue*. In (c) Multifaceted taxonomy, a small set of nodes with high uncertainty (highlighted in *Green*) are replicated and added to the MAP taxonomy at different positions (highlighted in *Red*). The following abbreviations are used in these taxonomies: 'kitc-appl' for 'kitchen appliance', 'cook-uten' for 'cooking utensil', 'w-c' for 'whipped cream', 'home-appl' for 'home appliance', and 'kitc-uten' for 'kitchen utensil'.

References

Alonso, O., and Baeza-Yates, R. 2011. Design and implementation of relevance assessments using crowdsourcing. In *Advances in information retrieval*. Springer. 153–164.

Blei, D.; Griffiths, T.; Jordan, M.; and Tenenbaum, J. 2004. Hierarchical topic models and the nested chinese restaurant process. In *NIPS*, volume 16, 17.

Blei, D.; Ng, A.; and Jordan, M. 2003. Latent dirichlet allocation. *JMLR* 3:993–1022.

Bloehdorn, S.; Hotho, A.; and Staab, S. 2005. An ontology-based framework for text mining. In *GLDV*.

Bragg, J.; Mausam; and Weld, D. 2013. Crowdsourcing multi-label classification for taxonomy creation. In *HCOMP*.

Brewster, C.; Alani, H.; Dasmahapatra, S.; and Wilks, Y. 2004. Data driven ontology evaluation. In *LREC 2004*.

Chandar, P., and Carterette, B. 2012. Using preference judgments for novel document retrieval. In *SIGIR*. ACM.

Chilton, L.; Little, G.; Edge, D.; Weld, D.; and Landay, J. 2013. Cascade: Crowdsourcing taxonomy creation. In *CHI*.

Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; and Fei-fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*.

Deng, J.; Russakovsky, O.; Krause, J.; Bernstein, M.; Berg, A.; and Fei-Fei, L. 2014. Scalable multi-label annotation. In *CHI*.

Deng, J.; Krause, J.; Stark, M.; and Fei-Fei, L. 2015. Leveraging the wisdom of the crowd for fine-grained recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.

Evermann, J., and Fang, J. 2010. Evaluating ontologies: Towards a cognitive measure of quality. *Information Systems* 35(4):391–403.

Fellbaum, C. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Heer, J., and Bostock, M. 2010. Crowdsourcing graphical perception: using mechanical turk to assess visualization design. In *CHI*, 203–212. ACM.

Kittur, A.; Chi, E.; and Suh, B. 2008. Crowdsourcing user studies with mechanical turk. In *CHI*, 453–456. ACM.

Klein, D., and Manning, C. 2003. Accurate unlexicalized parsing. In *ACL*.

Knight, K. 1993. Building a large ontology for machine translation. In *the Workshop on Human Language Technology*, 185–190. Association for Computational Linguistics.

Lai, K.; Bo, L.; Ren, X.; and Fox, D. 2011. A Scalable Tree-based Approach for Joint Object and Pose Recognition. In *AAAI*.

Maedche, A., and Staab, S. 2002. Measuring similarity between ontologies. In *Knowledge engineering and knowledge management: Ontologies and the semantic web*. Springer. 251–263.

Marszałek, M., and Schmid, C. 2007. Semantic hierarchies for visual object recognition. In *CVPR*.

Maryam, M., and Popescu-Belis, A. 2012. Using crowdsourcing to compare document recommendation strategies for conversations. In *RecSys, Recommendation Utility Evaluation (RUE' 2012)*.

Mason, W., and Suri, S. 2012. Conducting behavioral research on amazon's mechanical turk. *Behavior research methods* 44(1):1–23.

Micallef, L.; Dragicevic, P.; and Fekete, J. 2012. Assessing the effect of visualizations on bayesian reasoning through crowdsourcing. *IEEE Transactions on Visualization and Computer Graphics* 18(12):2536–2545.

Mortensen, J.; Musen, M.; and Noy, N. 2013. Developing crowdsourced ontology engineering tasks: an iterative process. In *Proceedings of the 1st International Conference on Crowdsourcing the Semantic Web-Volume 1030 (CEUR-WS'13)*, 79–88. CEUR-WS. org.

Navigli, R., and Ponzetto, S. 2012. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence* 193:217–250.

Ordonez, V.; Deng, J.; Choi, Y.; Berg, A.; and Berg, T. 2013. From large scale image categorization to entry-level categories. In *ICCV*, 2768–2775.

Ponzetto, S., and Navigli, R. 2009. Large-scale taxonomy mapping for restructuring and integrating wikipedia. In *IJCAI*, volume 9, 2083–2088.

Porzel, R., and Malaka, R. 2004. A task-based approach for ontology evaluation. In *ECAI Workshop on Ontology Learning and Population*.

Priss, U. 2008. Facet-like structures in computer science. *Axiomathes* 18(2):243–255.

Resnik, P. 1999. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence* 11:95–130.

Sifer, M. 2006. Filter co-ordinations for exploring multi-dimensional data. *Journal of Visual Languages & Computing* 17(2):107–125.

Sun, Y.; Singla, A.; Fox, D.; and Krause, A. 2015. Building hierarchies of concepts via crowdsourcing. In *IJCAI*.

Voorhees, E. 1993. Using wordnet to disambiguate word senses for text retrieval. In *SIGIR*.

Wu, C.; Lenz, I.; and Saxena, A. 2014. Hierarchical semantic labeling for task-relevant rgb-d perception. In *Proceedings of Robotics: Science and systems (RSS'14)*.

Zweig, A., and Weinshall, D. 2007. Exploiting object hierarchy: Combining models from different category levels. In *ICCV*, 1–8. IEEE.