Resource-Bounded Crowd-Sourcing of Commonsense Knowledge

Yen-Ling Kuo and **Jane Yung-jen Hsu**
Department of Computer Science and Information Engineering
National Taiwan University
yjhsu@csie.ntu.edu.tw

Abstract

Knowledge acquisition is the essential process of extracting and encoding knowledge, both domain specific and commonsense, to be used in intelligent systems. While many large knowledge bases have been constructed, none is close to complete. This paper presents an approach to improving a knowledge base efficiently under resource constraints. Using a guiding knowledge base, questions are generated from a weak form of similarity-based inference given the glossary mapping between two knowledge bases. The candidate questions are prioritized in terms of the *concept coverage* of the target knowledge. Experiments were conducted to find questions to grow the Chinese ConceptNet using the English ConceptNet as a guide. The results were evaluated by online users to verify that 94.17% of the questions and 85.77% of the answers are good. In addition, the answers collected in a six-week period showed consistent improvement to a 36.33% increase in concept coverage of the Chinese commonsense knowledge base against the English ConceptNet.

1 Introduction

Building large knowledge bases involves significant costs and time. Knowledge acquisition is the essential process of extracting and encoding knowledge, both domain specific and commonsense, to be used in intelligent systems.

There are three major approaches to knowledge acquisition in developing the major knowledge sources available to date. First, the knowledge base may be carefully crafted by a team of knowledge engineers, for example, WordNet [Miller, 1995] and Cyc [Lenat, 1995]. This approach is the most labor-intensive and time-consuming. Second, knowledge may be extracted automatically via text mining or machine learning techniques [Schubert, 2002; Etzioni *et al.*, 2004; Carlson *et al.*, 2010]. While this approach is powerful and effective, it requires the most computational resources and is overly sensitive to syntactical changes of the underlying web corpus. Third, knowledge may be crowd-sourced from volunteer web users [Singh *et al.*, 2002; Chklovski, 2003] or human computation games such as Verbosity [von Ahn *et al.*, 2006] and Virtual Pets [Kuo *et al.*, 2009]. The crowd-sourcing approach succeeded in building useful knowledge bases within a relatively short time and with extremely low costs.

In general, the coverage of facts and relations in a knowledge base is limited by the human and/or computational resources avaialble for knowledge acquisition. The problem was addressed by using incomplete reasoning [Fox, 1981] or finding heuristics to prune the search space of world knowledge. Heuristics often require deeper understanding of the target domain and do not apply to another domain. In particular, online users are often only willing to spend a short time on a specific website/game, and the knowledge bases built collaboratively are often noisy. Previous research also found that unguided crowd-sourcing suffers from high redundancy and contributes little to the knowledge base [Chklovski and Gil, 2005; Kuo and Hsu, 2010].

Based on a framework for knowledge acquisition through the process of question answering from online users, this research investigates the problem of "How do we crowdsource commonsense knowledge effectively within the resource limitation?" The key is to identify the most *productive* questions used to acquire answers. The proposed question generation procedure utilizes a *guiding knowledge base*. New questions are generated using a weak form of inference, i.e. *similarity* under incomplete information, computed from the guiding knowledge base. The candidate questions are sorted according to their expected contributions to the knowledge base. It is therefore possible to optimize question selection based on the resource bounds by incrementally adding the acquired question-answer pairs to the knowledge base.

This paper starts by defining the *resource-bounded knowledge acquisition* problem and proposing an approach to automated question generation for knowledge acquisition. We then present an experiment on expanding the Chinese commonsense knowledge base using the OMCS ConceptNet [Havasi *et al.*, 2007] as a guide. The experimental results were evaluated in both *concept coverage* and *quality*. The answers collected in a six-week period showed consistent improvement in concept coverage of the Chinese commonsense knowledge base, and the answers were verified by online users for quality.

2 Resource-bounded Knowledge Acquisition

Crowd-sourcing of commonsense knowledge can be viewed as a process to ask online users for sentences that can be put into the knowledge base. In this section, we review the knowledge acquisition problem through a question answering framework and then introduce the *resource bound* in knowledge acquisition. With the consideration of available resources, it is possible to design an efficient approach to acquiring sentences.

2.1 Notations

Before giving the notations, we first present the framework used in this paper. A commonsense *knowledge base* contains sentences acquired from *agents* (e.g. experts, online users, and machine learning algorithms) and an *inference algorithm* can be applied on the knowledge base for reasoning. The *questions* are used for asking agents for *answers* to add into the knowledge base.

Knowledge base Let $\mathcal{K}$ be the knowledge base that contains grounded sentences and is a subset of world facts. Given a set of concepts C and a set of relations R, every sentence is a triple (s, r, o), where $s \in C$, $o \in C$, and $r \in R$. Every triple in $\mathcal{K}$ must be associated with either $true$ or $false$.

We use i to denote the inference algorithm that can be performed on $\mathcal{K}$. If an inference algorithm i can derive sentence μ from $\mathcal{K}$, we write $\mathcal{K} \vdash_i \mu$.

Question and answer As in Learner [Chklovski, 2003], we use the complement of a concept in a sentence as a feature f of that concept. We use F to denote the feature set. A concept s in triple (s, r, o) is associated with a right feature f_{right}. Similarly, an concept o in triple (s, r, o) is associated with a left feature f_{left}. Therefore, a question q is asking agents for the associated concepts of a feature such as "(___, `IsA`, basketball player)?" We use $\mathcal{Q}$ to denote the set of possible questions whose size is $2|C||R|$.

For any question set $Q \subseteq \mathcal{Q}$, let $Answer(Q)$ be the answers returned by a set of agents A and $Answer(Q) \subset \mathcal{X}$. If no agent has answer to the questions, $Answer(Q)$ would be an empty set $\emptyset$.

2.2 Problem definition

Since the sentences in a knowledge base forms the basis for the reasoning process of an intelligent system. The acquisition of commonsense knowledge can be considered as the following formulation:

Definition 1. *Given a commonsense knowledge base $\mathcal{K}$, the knowledge acquisition problem is using $Q \subseteq \mathcal{Q}$ to find a set of sentences $Answer(Q)$ such that $|\{\alpha : (\mathcal{K} \wedge Answer(Q)) \vdash_i \alpha\}| > |\{\alpha : \mathcal{K} \vdash_i \alpha\}|$. If no such question set Q exists, the $\mathcal{K}$ contains sufficient knowledge for programs to use.*

Definition 1 implies that we aim to add new sentences to $\mathcal{K}$ instead of the sentences that are already in $\mathcal{K}$ or can be inferred from $\mathcal{K}$. The most intuitive method to find such sentences is to try all the possible questions, i.e. $Q = \mathcal{Q}$. Therefore, it always takes a lot of time to to ask agents all the questions if we build a knowledge base from scratch.

Resource bound In building large commonsense knowledge base, it is impratical to enumerate all the possible questions ($|\mathcal{Q}| = 2|C||R|$, this number is over millions for existing facts), since we can only use a fixed amount of resources to get answers. *The resource limitation in knowledge acquisition comes from the number of agents we can access and the computing power these agents have*. For example, the knowledge base constructors only have a fixed amount of money to recruit a fixed number of workers on Amazon Mechanical Turk[1] and each of them can answer at most n questions within a short period of time. To quantify the resource limitation in the knowledge acquisition process, we compute the number of questions that can be answered by the agents in time T:

Definition 2. *Given a set of available agents $A = \{a_1, a_2, ..., a_m\}$ and the number of questions $N = \{n_1, n_2, ..., n_m\}$ they can answered in time T, the* resource bound Θ *in the knowledge acquisition process is $\sum_{i=1}^{m} n_i$.*

Note that n_i in definition 2 is a small constant compared to the number of possible questions, i.e. $n_i \ll |\mathcal{Q}|$. Since there is always only a fixed number of agents involved in the crowd-sourcing of common sense, the resource bound Θ is also far less than $|\mathcal{Q}|$.

With the resource bound Θ, we cannot put all the possible questions into the question set Q to get a good enough $Answer(Q)$. So, the knowledge acquisition problem turns to the resource-bounded case as the following definition:

Definition 3. *Given a commonsense knowledge base $\mathcal{K}$ and a resource bound Θ, the resource-bounded knowledge acquisition problem is using $Q \subseteq \mathcal{Q}$ to find a set of sentences $Answer(Q)$ such that $|\{\alpha : (\mathcal{K} \wedge Answer(Q)) \vdash_i \alpha\}| > |\{\alpha : \mathcal{K} \vdash_i \alpha\}|$ within time T where $|Q| \leq \Theta$*

In the resource-bounded case, we aim to find a question set Q whose size is smaller or equal to Θ such that the acquired answers infer more new sentences for the knowledge base. With no assumption or background knowledge of the knowledge domains, the question set could be any combination of $2|C||R|$ possible questions. If we randomly choose one combination of questions, it is easy to get $\emptyset$ or the sentences already in $\mathcal{K}$. Many people try to find a productive question set Q to elicit new sentences from agents. However, most of them are heuristics and require a lot of domain knowledge or skills to codify the question set. It is hard to apply these heuristics to another domain or large commonsense knowledge bases developed from general public.

2.3 Ideas for solving this problem

Instead of using heuristics to reduce the size of question set, we think the key component of finding a productive question set is to estimate the answers of a question and their inference results before asking for answers from agents. Using the estimated answers and inference results, we are able to put the most productive questions into the question set so that we can get most new sentences within the resource bound.

In next section, we introduce a *guiding knowledge base* that help us conduct the estimation for the concepts in a target domain and provide an algorithm to automatically generate questions that are plausible to draw answers with the most

[1]https://www.mturk.com

new inferred sentences. The concept coverage of the knowledge base after the crowd-sourcing process can then be computed using the overlaps with the guiding knowledge base to quantify the improvement in inference results.

3 Knowledge Base Approximation

Consider the problem of estimating the answers and their inference results. If we have another knowledge base containing the target domain knowledge and it shares some concepts and relations with the $\mathcal{K}$, we can use the sentences and inference results in another knowledge base as plausible answers to the questions. Therefore, we can compute the differences of the two knowledge bases to find a productive question set Q such that we can elicit answers from agents to fill in the gaps. We say that this process is *knowledge base approximation* and another knowledge base is a *guiding knowledge base* $\mathcal{K}_g$.

3.1 Similarity as weak inference

Before giving the algorithms to find a productive question set, we illustrate the inference algorithm used in the knowledge base approximation process.

KB matrix

In order to compare the answers and inference results with other knowledge bases, we put all the sentences in $\mathcal{K}$ to a "KB matrix" where the $(i, j)_{th}$ entry of the matrix is the truth assignment of the sentence that is consisted of i_{th} concept in C and j_{th} feature in F. For example, table 1 is a sub-matrix of a KB matrix.

Semantics of KB matrix For any two rows i, j in the KB matrix, we can find that the sentence (c_i, f) in an inference rule can be replaced by sentence (c_j, f) and gives plausible inference results if c_i and c_j have the same truth assignments for the same feature. For example, the sentences $PartOf(fur, cat)$ and $IsA(cat, pet)$ in modus ponens rule

$$\frac{PartOf(fur, cat) \rightarrow IsA(cat, pet), PartOf(fur, cat)}{IsA(cat, pet)}$$

can be replaced by $PartOf(fur, dog)$ and $IsA(dog, pet)$. After the replacement,

$$\frac{PartOf(fur, cat) \rightarrow IsA(dog, pet), PartOf(fur, dog)}{IsA(dog, pet)}$$

is still a plausible inference since dog and cat share the same truth assignments for features $PartOf(fur, ?)$ and $IsA(?, pet)$.

From the above observation, we can find that two concepts will get involved in the same inference rules if they have shared features and vice versa. In this paper, *the similarity of two concepts is defined with respect to the shared features of the two concepts*. Therefore, we can transfer the inference results of the concept which is similar to our targeted concepts instead of doing the complete inference for every sentence in the knowledge base. Using similarity as a weak form of inference, we are able to perform the same inference procedure in different knowledge bases. The KB matrix is a good fit for computing similarity of two concepts.

Similarity measure for large and noisy KB

For large commonsense knowledge bases that are constructed by crowd-sourcing techniques, the confidence of a sentence in knowledge base may not be 100%. Therefore, we relax the constraint of truth assignments of KB matrix to real numbers, which indicate the confidence of being true for the sentences and are in range $[-1, 1]$.

Since the size of knowledge base is always very large, the KB matrix must be large and sparse. As in AnalogySpace [Speer *et al.*, 2008], we apply truncated singular value decomposition (truncated SVD) on KB matrix to smooth the noisy data in the knowledge base. The concepts are then transformed to a k-dimensional vector space spanned by eigen-features. In the vector space spanned by eigen-features, the proximity of two concepts represents their level of overlaps in features. Therefore, the similarity of two concept vectors $\vec{c_1}$ and $\vec{c_2}$ is defined as follows:

$$Sim(\vec{c_1}, \vec{c_2}) = \frac{\vec{c_1} \cdot \vec{c_2}}{\|\vec{c_1}\|\|\vec{c_2}\|}$$

3.2 Acquisition via KB approximation

Now that we have similarity measure for any two concepts in a knowledge base, we can find the inference results of a concept. By measuring the differences of inference results in $\mathcal{K}$ and $\mathcal{K}_g$, we are able to create questions that would draw new sentences for $\mathcal{K}$ to fill up the gaps.

Concept coverage

In order to compare the inference results of two knowledge bases, sub-KB matrices are created with the glossary mapping of concepts/relations in the two knowledge bases. The inference results of a concept are reflected on its similar concepts because the similar concepts are plausible to involve in the same inference rules. If the top n similar concepts of a concept is the same in different knowledge bases, we say that the *coverage* of the concept is the same in the two knowledge bases. If its coverage is different in two knowledge base, we can always find a new sentence to improve the concept coverage. The coverage of each concept is evaluated using algorithm 1 where the number n of similar concepts is specified because we only compare the similar concepts.

Question set generation

For the concepts with low coverage scores, we borrow the features from its mapped concepts in guiding knowledge base $\mathcal{K}_g$ to generate new questions. The answers to the generated questions are guaranteed to cover at least the sentences found in $\mathcal{K}_g$. Using algorithm 2, we include the questions generated from the concepts with lowest coverage scores into Q and ensure that the size of Q is within the resource bound Θ. Therefore, the generated questions are the ones that would draw answers to approximate the inference results in the guiding knowledge base.

4 Evaluation

In order to evaluate the proposed approach to resource-bounded crowdsourcing of commonsense knowledge, we performed experiments using the ConceptNet [Havasi *et al.*, 2007], a multi-lingual commonsense knowledge base with APIs for accessing and contributing data.

Table 1: A sample part of a KB matrix.

	`IsA`(__, pet)	`AtLocation`(__, home)	`CapableOf`(__, fly)	`MadeOf`(__, metal)	`PartOf`(fur, __)
cat	True	True	False	False	True
dog	True	?	False	?	True
airplane	False	False	True	True	?
toaster	?	True	?	True	?

Algorithm 1 Concept Coverage $(\mathcal{K}, \mathcal{K}_g, Map, n)$

Require: A knowledge base $\mathcal{K}$, a guiding knowledge base $\mathcal{K}_g$, a mapping $Map : \mathcal{K} \to \mathcal{K}_g$ of concepts/realtions of the two knowledge bases, and n to indicate the number of similar concepts
Ensure: Coverage score of target concepts

Use the domain and range of Map to create KB matrices M and M_g of $\mathcal{K}$ and $\mathcal{K}_g$
for all concept $c \in M$ **do**
  Get top n similar concepts of c in M to form set S
  for all concept $c_i \in Map(c)$ **do**
    Get top n similar concepts of c_i in M_g to form set S_i
  end for
  Coverage score of $c = \frac{|(\bigcup_i S_i) \cap S|}{|S|}$
end for

Algorithm 2 Generate Questions $(\mathcal{K}, \mathcal{K}_g, Map, \Theta)$

Require: A knowledge base $\mathcal{K}$, a guiding knowledge base $\mathcal{K}_g$, a mapping $Map : \mathcal{K} \to \mathcal{K}_g$ of concepts/realtions of the two knowledge bases, and a resource bound Θ
Ensure: A question set Q and $|Q| \leq \Theta$

Sort the concepts according to their coverage scores and store them into a list L
Create KB matrix M and M_g of $\mathcal{K}$ and $\mathcal{K}_g$
for all concept $c \in L$ **do**
  for all concept $c_i \in Map(c)$ **do**
    Get features F of c_i in M_g but not in M
    for all feature $f \in F$ **do**
      if $|Q| \geq \Theta$ **then**
        return Q
      end if
      Get relation r from f
      Create question q using c and r
      Add q to Q
    end for
  end for
end for
return Q

4.1 Experimental setup

The ConceptNet corpora contains a growing collection of over one million sentences in English, as well as about a quarter million sentences in Chinese and Portuguese, respectively. The English and Portuguese corpora were collected from over 15,000 contributors at the OMCS website[2] within a span of 10 years [Singh *et al.*, 2002]. In contrast, the sentences in the Chinese ConceptNet were collected and verified via question-answering between players of the Virtual Pets game [Kuo *et al.*, 2009].

Given that the English ConceptNet is currently the biggest corpus, we decided to leverage it as the guiding knowledge base in generating questions for knowledge acquisition for Chinese ConceptNet by crowdsourcing. In what follows, we illustrate each component and its setting in our experiments with the ConceptNet.

Knowledge Representation: ConceptNet

The ConceptNet represents all collected sentences as a directed graph. The nodes of this graph are *concepts*, and its labeled edges are *relations* of commonsense knowledge connecting two concepts. There are over 20 relation types defined in ConceptNet. For example,

- `UsedFor(a, b)`, e.g. [Spoon] is used for [eating].
- `IsA(a, b)`, e.g. [Dog] is an [animal].

The KB matrix and vector space representation of the English and Chinese ConceptNet are constructed using the Divisi library[3]. The number of singular values k in the truncated SVD is set to 50 for building the vector space. The glossary mapping table for evaluating concept coverage is created from a Chinese-English dictionary. The number n of similar concepts considered in question generation is set to 20 in our evaluation. Table 2 summarizes the knowledge bases involved in this experiment.

Table 2: Statistics of ConceptNet

	English ConceptNet	Chinese ConceptNet
# of concepts	321,993	93,138
# of distinct sentences	569,596	252,319
# of relation types	20	20
% overlaps in concepts	–	9.69%
% overlaps in relations	–	100%

[2]http://openmind.media.mit.edu/
[3]http://csc.media.mit.edu/divisi

Acquisition method: Virtual Pets

The questions generated by algorithm 2 were added to the Virtual Pets [Kuo *et al.*, 2009], a community-based game for collecting Chinese common sense. The players in Virtual Pets answer questions such as "*Spoon* is used for ___?"

The players in Virtual Pets are the agents defined in knowledge acquisition process. They may ask/answer their pets questions, vote good or bad to an answer they get, or use the commonsense points to exchange food for their pets. In average, there were 55 active players per day and each of them answers 3.16 questions in one day. Therefore, the resource bound Θ in Virtual Pets is the number of questions that can be answered by the users in one day, i.e. $\Theta \approx 55 \times 3.16 = 173.8$. Since the collaboratively-built knowledge base requires answers verified by consensus (i.e. duplicated answers), the actual resource bound Θ is smaller than 173.8 per day. If we try every possible question, it may take many years to get one answer for each question.

In order not to interfere in the interaction flow of the game, we created a virtual player to ask the generated questions to other players. We randomly selected 1/3 of the active players every day to ask them the generated questions from the virtual player. If the players think the question they are asked does not make sense, they can simply pass it and report it to the system. For different players, they may get same questions in the same day so that we can get the concensus answers from these players.

4.2 Experimental result

In this experiment, we asked 480 questions in Virtual Pets and took 6 weeks to collect answers to these questions. Both the generated questions and collected answers were verified by players to evaluate their qualities. We also put the collected sentences to the Chinese ConceptNet in the end of 5th and 6th week to see how the generated questions improve the concept coverage within the resource limitation of six weeks.

Concept coverage

The first part of experiments looks into the coverage score of each concept in the Chinese ConceptNet before we conducted the proposed question generation process. There are 399 concepts whose coverage score ≥ 0.5 and 8,634 concepts whose coverage score < 0.5. Since the coverage score indicates the differences of inference results between Chinese and English ConceptNet, the result shows that the inference results of over 95% of concepts in Chinese ConceptNet were different from their mapped concepts in English ConceptNet.

Table 3: Example of concepts with different coverage scores

coverage score $>= 0.5$	coverage score < 0.5
動物 (animal)	華盛頓 (Washington)
建築物 (building)	經濟學 (economics)
食物 (food)	吸血鬼 (vampire)
房間 (room)	星座 (constellation)
空氣 (air)	巧克力 (chocolate)

When we compared the concepts with different coverage scores (see examples in table 3), we can easily find that most concepts with high coverage scores are generic concepts such as food and animal, whereas the concepts with low coverage scores are specific and culture-dependant concepts such as chocolate and vampire. We think it is the game design make players in a game tend to create easy and generic questions. If the questions are easy, they can get game points within a short period of time. With English ConceptNet as a guiding knowledge base, it is much easier to create specific questions to ask players.

Quality of the generated questions

In the experiment, the question generation algorithm was used to produce 38,285 questions, with only 3,743 of which already in the Virtual Pets. This suggests that the proposed approach identifies mostly new features rather than existing features in the knowledge base for question generation.

Given the resource bound, 480 questions generated from the lowest coverage scores are selected to ask players. These questions are mixed with the questions generated by human players. We recorded the bad questions players reported in the six weeks and make a comparison between the two question sets in table 4. The percentage of good questions generated by our algorithm is 94.17% which is comparable to the ones generated by human players (93.90%). Our result showed that it is feasible to use our approach to automatically generate questions. These generated questions are demonstrated to be as good as the questions created by human players.

Table 4: Quality of the generated questions

	By our algorithm	By players
# of questions	480	4,329
# of bad questions	28	264
% of good questions	94.17%	93.90%

Quality of the acquired sentences

Within the resource bound, i.e. $\Theta \approx 480$ in 6 weeks, we collected 3,788 distinct sentences. All collected sentences were shuffled to be voted. Each sentence is rated as either good or bad by 3 randomly selected players, and it is treated as a good sentence if two or more players rated the sentence as good. Otherwise, it is considered as a bad sentence. There are 3,249 out of the 3,788 sentences are rated as good sentences by players (precision = 85.77%). Compared to the answers to player-generated questions (80% for answer count ≥ 2 [Kuo *et al.*, 2009]), we think questions generated by our algorithm are able to draw more good answers than the questions created by human players. In addition, all of these collected sentences are new to the Chinese ConceptNet since we generated questions from concepts with low coverage scores. Therefore, the collected sentences have more expected contribution to the knowledge base than the sentences collected from the original collection process.

Coverage improvement

Table 5 summarizes the improvement in concept coverage from week 5 to week 8 in the proposed crowdsourcing process guided by the English ConceptNet. Taking week 0 as the baseline, the improvement is measured in terms of the number of concepts whose coverage score is < 0.5, denoted by $|c^-|$, and the corresponding percentage of improvement, denoted by Δ. Concept coverage improves by 33.51% at week 5, and steadily improved to a 37.02% increase by the 8th week.

Table 5: Improvements in concept coverage

	week 0	week 5	week 6	week 7	week 8
$\|c^-\|$	8,630	5,783	5,495	5,450	5,435
Δ	–	33.51%	36.33%	36.85%	37.02%

In addition to improving the original concepts in Chinese ConceptNet, 402 new concepts were introduced to the knowledge base. The statistics show that we can incrementally add new question-answer pairs so that the target knowledge base will approximate the guiding knowledge base. However, it is interesting to observe that some concepts with low coverage scores may remain due to cultural differences and insufficient data in the guiding KB.

5 Conclusion

This paper introduced the problem of resource-bounded crowdsourcing of commonsense knowledge and proposed an approach to finding a productive question set automatically. By leveraging a guiding knowledge base containing the target domain knowledge, we are able to improve the acquisition efficiency and concept coverage according to a glossary mapping table between the two knowledge bases.

Experiments have been conducted to generate questions for growing commonsense knowledge bases, such as the Chinese ConceptNet collected via Virtual Pets, using English ConceptNet as a guide. The results showed that the generated questions and their collected sentences are as good as the original collection process (precision = 94.17% for questions and 85.77% for answers) with an improvement in concept coverage by 37.02%. The proposed approach can be embedded into a perpetual crowdsourcing process to improve the acquisition efficiency of building large commonsense knowledge bases. It is also easy to target specific domains in acquisition of commonsense knowledge within the resource bound.

Acknowledgments

This research was supported by a grant from the National Science Council of Taiwan, NSC 99-2221-E-002-139-MY3.

References

[Carlson *et al.*, 2010] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. Toward an architecture for never-ending language learning. In *Twenty-Fourth Conference on Artificial Intelligence (AAAI10)*, 2010.

[Chklovski and Gil, 2005] Timothy Chklovski and Yolanda Gil. An analysis of knowledge collected from volunteer contributors. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05)*, 2005.

[Chklovski, 2003] Timothy Chklovski. Learner: a system for acquiring commonsense knowledge by analogy. In *Proceedings of the 2nd International Conference on Knowledge Capture*, 2003.

[Etzioni *et al.*, 2004] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Methods for domain-independent information extraction from the web: an experimental comparison. In *Proceedings of the 19th national conference on Artifical intelligence*, AAAI'04, pages 391–398, 2004.

[Fox, 1981] Mark S. Fox. Reasoning with incomplete knowledge in a resource-limited environment: integrated reasoning and knowledge acquisition. In *In Proceedings of the 7th International Joint Conferences on Artificial Intelligence (IJCAI-81)*, pages 313–318, 1981.

[Havasi *et al.*, 2007] Catherine Havasi, Rob Speer, and Jason Alonso. ConceptNet 3: A flexible, multilingual semantic network for common sense knowledge. In *Recent Advances in Natural Language Processing*, Borovets, Bulgaria, September 2007.

[Kuo and Hsu, 2010] Yen-Ling Kuo and Jane Yung-jen. Hsu. Goal-oriented knowledge collection. In *2010 AAAI Fall Symposium on Common Sense Knowledge (FSS10)*, 2010.

[Kuo *et al.*, 2009] Y. L. Kuo, J. C. Lee, K. Y. Chiang, Rex Wang, Edward Shen, C. W. Chan, and Jane Y.-j. Hsu. Community-based game design: experiments on social games for commonsense data collection. In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, 2009.

[Lenat, 1995] Douglas B. Lenat. CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38, 1995.

[Miller, 1995] George A. Miller. WordNet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[Schubert, 2002] Lenhart Schubert. Can we derive general world knowledge from texts? In *Proceedings of the 2nd International Conference on Human Language Technology Research (HLT02)*, 2002.

[Singh *et al.*, 2002] Push Singh, Thomas Lin, Erik T. Mueller, Grace Lim, Travell Perkins, and Wan L. Zhu. Open mind common sense: Knowledge acquisition from the general public. In *On the Move to Meaningful Internet Systems, 2002 - DOA/Coop IS/ODBASE 2002 Confederated International Conferences DOA, CoopIS and ODBASE 2002*, 2002.

[Speer *et al.*, 2008] Robert Speer, Catherine Havasi, and Henry Lieberman. AnalogySpace: Reducing the dimensionality of common sense knowledge. In *Proceedings of AAAI-2008*, 2008.

[von Ahn *et al.*, 2006] Luis von Ahn, Mihir Kedia, and Manuel Blum. Verbosity: A game for collecting commonsense knowledge. In *ACM Conference on Human Factors in Computing Systems (CHI Notes)*, pages 75–78, 2006.