WebChild 2.0: Fine-Grained Commonsense Knowledge Distillation

Niket Tandon
Allen Institute for Artificial Intelligence
Seattle, WA, USA
`nikett@allenai.org`

Gerard de Melo
Rutgers University
Piscataway, NJ, USA
`gdm@demelo.org`

Gerhard Weikum
Max Planck Institute for Informatics
Saarbrücken, Germany
`weikum@mpi-inf.mpg.de`

Abstract

Despite important progress in the area of intelligent systems, most such systems still lack commonsense knowledge that appears crucial for enabling smarter, more human-like decisions. In this paper, we present a system based on a series of algorithms to distill fine-grained disambiguated commonsense knowledge from massive amounts of text. Our WebChild 2.0 knowledge base is one of the largest commonsense knowledge bases available, describing over 2 million disambiguated concepts and activities, connected by over 18 million assertions.

1 Introduction

With the continued advances in natural language processing and artificial intelligence, the general public is increasingly coming to expect that systems exhibit what may be considered *intelligent* behavior. While machine learning allows us to learn models exploiting increasingly subtle patterns in data, our systems still lack more abstract, generic forms of *commonsense knowledge*. Examples of such knowledge include the fact that fire causes heat, the property of ice being cold, as well as relationships such as that a bicycle is generally slower than a car. Previous work in this area has mostly relied on handcrafted or crowdsourced data, consisting of ambiguous assertions, and lacking multimodal data. The seminal work on ConceptNet (Havasi et al., 2007), for instance, relied on crowdsourcing to obtain an important collection of commonsense data. However, it conflates different senses of ambiguous words (e.g., "*hot*" in the sense of temperature vs. "*hot*" in the sense of being trendy). It also lacks fine-grained details such as specific kinds of properties, comparisons between objects, and detailed knowledge of activities. We attempt to fill these significant gaps.

This paper presents automated methods targeting the acquisition of large-scale, semantically organized commonsense knowledge. This goal poses challenges because commonsense knowledge is: (i) *implicit and sparse*, as humans tend not to explicitly express the obvious, (ii) *multimodal*, as it is spread across textual and visual sources, (iii) *affected by reporting bias*, as uncommon facts are reported disproportionally, and (iv) *context dependent*, which implies, among other things, that at an abstract level it is perhaps best described as merely holding with a certain confidence. Prior state-of-the-art methods to acquire commonsense are either not automated or based on shallow representations. Thus, they cannot automatically produce large-scale, semantically organized commonsense knowledge.

To achieve this challenging goal, we divide the problem space into three research directions.

- Properties of objects: acquisition of properties like `hasSize`, `hasShape`, etc. We develop a transductive approach to compile semantically organized properties.
- Relationships between objects: acquisition of relations like `largerThan`, `partOf`, `memberOf`, etc. We develop a linear-programming based method to compile comparative relations, and, we further develop a method based on statistical and logical inference to compile part-whole relations.
- Their interactions: acquisition of knowledge about activities such as *drive a car*, *park a car*, etc., with expressive frame based representation including temporal and spatial attributes. For this, our Knowlywood approach is based on semantic parsing and probabilistic graphical models to compile activity knowledge.

Together, these methods result in the construction of a large, clean and semantically organized Commonsense Knowledge Base that we call the WebChild 2.0 knowledge base.

2 Web-Scale Fact Extraction

Our framework consists of two parts. First, we rely on massive amounts of text to extract substantial amounts of raw extractions. Subsequently, in Section 3, we will present a series of algorithms to distill fine-grained disambiguated knowledge from raw extractions of this sort. For details on the input dataset leading to raw extractions, refer to the original publications, cited through each subsection in Section 3.

Pattern-Based Information Extraction. For knowledge acquisition, it is well-known that one can attempt to induce patterns based on matches of seed facts, and then use pattern matches to mine new knowledge. Unfortunately, this bootstrapping approach suffers from significant noise a) when using a very large number of seeds, and b) when applied to large Web-scale data (Tandon et al., 2011), which appears necessary to mine adequate amounts of training data. Specifically, for many of our extractions, Google's large Web N-Gram dataset, which provides Web-scale data. Additionally, many of our experiments use a large subset of ConceptNet as seed facts, resulting in a seed set that is many orders of magnitude larger than the typical set of perhaps 10 seeds used in most other bootstrapped information extraction systems. This problem is exacerbated by the fact that we are aiming at commonsense knowledge, which is not typically expressed using explicit relational phrases. We thus devise a custom bootstrapping approach designed to minimize noise when applied to Web-scale data.

We assume that we have a set of relations $\mathcal{R}$, a set of seed facts $S(r)$ for a given $r \in \mathcal{R}$, as well as a $\mathrm{domain}(r)$ and $\mathrm{range}(r)$, specified manually to provide the domain and range of a given r as its type signature. For pattern induction, we look for co-occurrences of words in the seed facts within the n-grams data (for n = 3,4,5). Any match is converted into a pattern based on the words between the two occurrences, e.g. "*that apple is red*" would become "*<x> is <y>*".

Pattern Scoring. The acquired patterns are still rather noisy and moreover very numerous, due to our large set of seeds. To score the reliability of patterns, we invoke a ranking function that rewards patterns with high distinct seed support but also discounts patterns that occur across multiple dissimilar relations (Tandon et al., 2011). The intuition is that a good pattern should match many of the seed facts (reward function), but must be discounted at matching too many relations (discount function). As, e.g., the pattern "*<x> and <y>*" is unreliable because it matches seeds from too many relations.

This discount must be softened when the pattern matches related relations. To allow for this, we first define a relatedness score between relations. We can either provide these scores manually, or consider Jaccard overlap statistics computed directly from the seed assertion data. Let p be a candidate pattern and $r \in \mathcal{R}$ be the relation under consideration. We define $|S(r,p)|$ as the number of distinct seeds $s \in S(r)$ under the relation r that p matches. We then define the discount score of the pattern p for relation r as:

$$\phi(r,p) = \sum_{r' \in \mathcal{R}, r' \neq r} \frac{|S(r,p)|}{|S(r)|} - (1 - \mathrm{sim}(r,r')) \frac{|S(r',p)|}{|S(r')|}$$

where $\mathrm{sim}(r,r')$ is the similarity between relations r and r'. The final score is a combination of the discount score $\phi(r,p)$ and the judiciously calculated reward score. At the end, we choose the top-k ranked patterns as the relevant patterns for the extraction phase.

Assertion Extraction. We apply the chosen patterns to find new occurrences in our (Google Web N-grams) data. For instance, "*<x> is <y>*" could match "*the sun is bright*", yielding ("*sun*", "*bright*") as an assertion for the `hasProperty` relation. To filter out noise from these candidate assertions, we check if the extracted words match the required domain and range specification for the relation, using WordNet's hypernym taxonomy. Finally, we rank the candidate assertions analogously to the candidate patterns, but treating the patterns as seeds.

3 Distilling Fine-Grained Knowledge

The techniques described above provide a sizable set of extractions that serve as the the basis for WebChild 2.0. Next, we consider a family of algorithms that take raw extractions of this form and exploit them to distill detailed fine-grained knowledge.

3.1 Fine-Grained Properties

The first algorithm we consider (Tandon et al., 2014a) aims at compiling a large and clean set of fine-grained commonsense properties, connecting noun senses with adjective senses by a variety of relations. In contrast to prior work that only dealt with a generic `hasProperty` relation, we use 19 different (sub-)relations such as `hasShape`, `hasSize`, `hasTaste`, `hasAbility`, `evokesEmotion`, etc. This list is systematically derived from WordNet based on its `attribute` information.

Moreover, our goal is to distinguish the specific senses (e.g. $\texttt{green}^2_a$, or `green#a#2` refers to the second sense of WordNet adjective `green`) of the arguments of these relations as well. For example, for ⟨*plant* `hasProperty` *green*⟩, there are two competing interpretations with very different meanings: ⟨*industrial-plant* `hasQuality` *green-environmental*⟩ vs. ⟨*botanical-plant* `hasColor` *green-color*⟩.

We start out by constructing the range and domain of the property relations with a small set of seed examples. Such seeds would normally be gathered manually, but in our work, we observed that an ensemble of two very different, automated, and noisy sources can also produce high-quality seeds. We construct a graph where the nodes are words and word senses and the edge weights are computed based on taxonomic and distributional similarities (these edge weights are depicted in Figure 1). We then use a judiciously designed form of label propagation (Talukdar and Crammer, 2009) to learn the domain set, the range set, and the extension of such relations, at large scale and in terms of specific word senses. An example of a range graph is given in Figure 1. The highlighted seed nodes mark specific senses of words as pertaining to a specific relation (e.g., `hasTemperature`). Via label propagation, we can infer such information for additional nodes, such as the "pleasantly cold" sense of "*crisp*" (for the range of the relation), but not other irrelevant senses of "*crisp*". The same label propagation technique can then also be applied to infer entire relation tuples.

Our graph-based semi-supervised approach is generic enough to extract any type of fine-grained sub-property or attribute, for which we need only a few seeds to begin.

3.2 Comparative Knowledge

The second algorithm (Tandon et al., 2014b) aims at extracting and organizing large-scale comparative commonsense knowledge. Prior to our work, semantically organized comparative commonsense knowledge had not been studied or compiled before.

We first gather a set of new raw extractions using patterns targeting comparisons. These include the word "*to be*" followed by comparative forms of adjectives (e.g. "*smaller than*", "*more educated than*"). While we again follow a Web-scale extraction strategy as in Section 2, note that these patterns are generic in the sense that they cover all words identified as adjectives. Thus, this constitutes a form of open information extraction for comparative knowledge.

The next step is to refine and extend these extractions. The constituents of a comparative assertion are strongly related (e.g., *car*, *fast*, and *bike* in ⟨*car* `faster than` bike⟩); our method builds upon this observation to jointly disambiguate and connect these assertions, while inferring additional ones. Disambiguation is important because relationships such as "*richer than*" could refer to financial assets or to calories. The joint algorithm is also necessary to exploit dependencies and transitivity between extractions (e.g., "*is smaller than*", "*is larger than*", "*is not bigger than*"), with the goal of validating input extractions as well as inferring new relationships. This is achieved via a custom Integer Linear Program with constraints accounting for the coherence and logical consistency of the interpretation

3.3 Detailed Part-Whole Relationships

The third algorithm (Tandon et al., 2016) focuses on extracting and organizing large-scale part-whole commonsense knowledge. In contrast to prior work, our algorithm distinguishes `physicalPartOf` (e.g., ⟨`wheel physicalPartOf bike`⟩), `memberOf` (e.g., ⟨`cyclist memberOf team`⟩), and `substanceOf` (e.g., ⟨`rubber substanceOf wheel`⟩), and the arguments are disambiguated. We also estimate *cardinality*, describing the typical number of parts in a whole, and *visibility* information, i.e., whether the part can be perceived visually.

We again rely on raw assertions extracted from text, but distill these via statistical scoring combined with logical inference to account for tran-

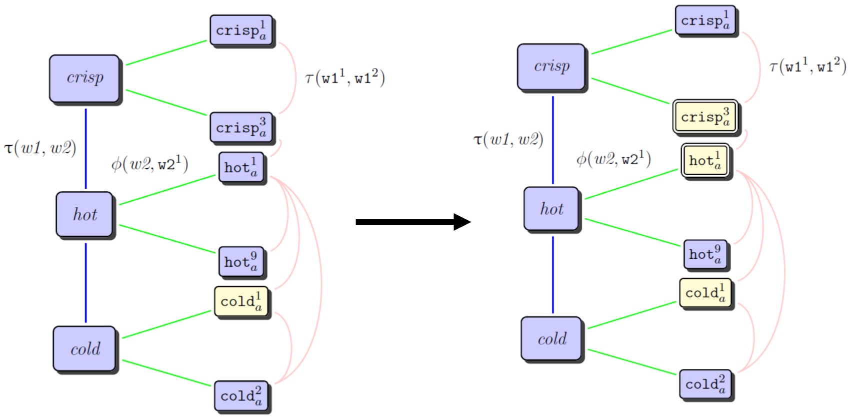


Figure 1: Inferring range of `hasTemperature`. Violet nodes indicate noisy candidate range. Starting with seeds (yellow single outlined), the algorithm enforces that similar nodes have similar labels, and infers range (yellow double outlined). For details on edge weights, see (Tandon et al., 2014a).

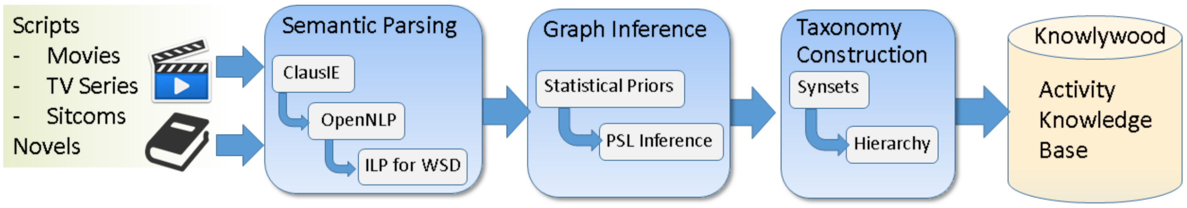


Figure 2: Knowlywood pipeline to extract activity frames.

sitivity and inheritance of assertions (via WordNet's hypernym hierarchy). To estimate visibility, we verify the assertions in images (we call this quasi-visual verification). Quasi-visual verification leverages the best of text-only verification (which is inaccurate due to reporting bias), and visual-only verification (which is inaccurate due to the object detectors' inaccuracies).

Our method generalizes to any relation with finer-grained sub-relations. This specific multimodal approach generalizes to any commonsense relation that has multiple sub-relations and is verifiable in images, e.g., `hasLocation` relation (with sub-relations `hasLocationAbove/Below`, etc.).

3.4 Activity Knowledge

The fourth algorithm (Tandon et al., 2015) is a novel method for a new task to extract and organize semantic frames of human activities, together with their visual content. The Knowlywood pipeline, illustrated in Figure 2, distills such knowledge from raw text, rather than starting with the extractions from Section 2. In particular, we acquire knowledge about human activities from narrative text, focusing in particular on movie scripts, which are structured in terms of scenes, and provide descriptions of scene settings/locations, speakers, etc. Moreover, when scripts come with representative images or time points in the movie, it is possible to align a scene description with the actual visual contents of the movie. The main difficulty, however, is that all this rich contents in movie scripts is merely in textual form – still far from structured KB representation.

Table 1: Semantic Parse Example

Input	WordNet Mapping	VerbNet Mapping	Expected Frame
the man	man#1	Agent . animate	Agent: man#1
begin to shoot	shoot#4	shoot#vn#3	Action: shoot#4
a video	video#1	Patient . solid	Patient:video#1
in	in	PP . in	
the moving bus	bus#1	NP . Location . solid	Location: moving bus#1

Our method considers joint semantic role labeling and word sense disambiguation for parsing these scripts to generate candidate items for the activity frames. In particular, we rely on

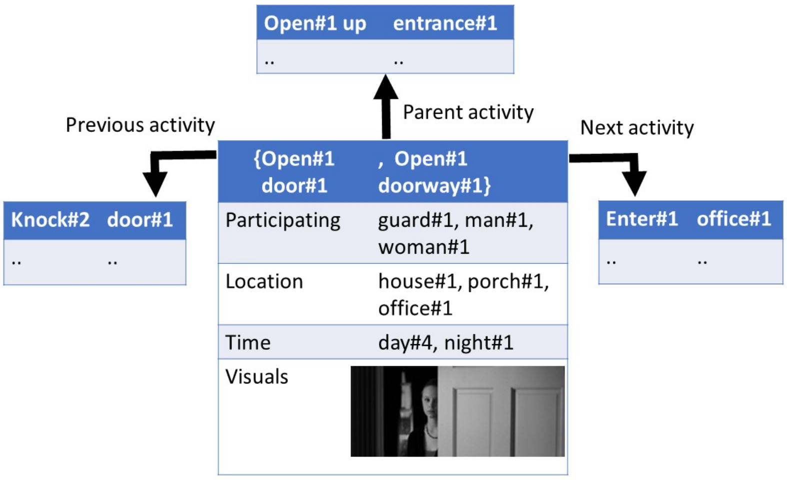


Figure 3: Example of an activity frame

Table 2: Knowlywood Coverage and Precision: manual assessments over a sample of 100 activity frames

Source	#Scenes	#Unique Activities	Parent	Parti.	Prev	Next	Loc.	Time	Avg.
Movie scripts	148,296	244,789	0.87	0.86	0.78	0.85	0.79	0.79	0.84
TV series	886,724	565,394	0.89	0.85	0.81	0.84	0.82	0.84	0.86
Sitcoms	286,266	200,550	0.88	0.85	0.81	0.87	0.81	0.83	0.87
Novels	383,795	137,365	0.84	0.84	0.78	0.88	0.85	0.72	0.84
Crowdsrc.	3,701	9,575	0.82	0.91	0.91	0.87	0.74	0.40	0.86
Knowlywood	1,708,782	964,758	0.87	0.86	0.84	0.85	0.78	0.84	**0.85±0.01**
ConceptNet 5	-	4,757	0.15	0.81	0.92	0.91	0.33	N/A	**0.46±0.02**

a semantic parsing like approach that fills the slot values of such frames (activity type, location, participants, etc.), while also jointly disambiguating the verb and the slot fillers with respect to WordNet. This is achieved by computing strong priors that are fed into an integer linear program. In the example in Table 1, we see how this process also jointly relies on information from VerbNet. For the WordNet verb sense number 2: `shoot#2` (killing), VerbNet provides a role restriction `Agent.animate V Patient.animate PP Instrument.solid`, where `animate` refers to living beings, as opposed to inanimate objects. This allows us to infer that this `shoot#2` sense is not compatible with the argument "the video", which is not `animate`. This way, we can disqualify the incorrect interpretation of "shoot".

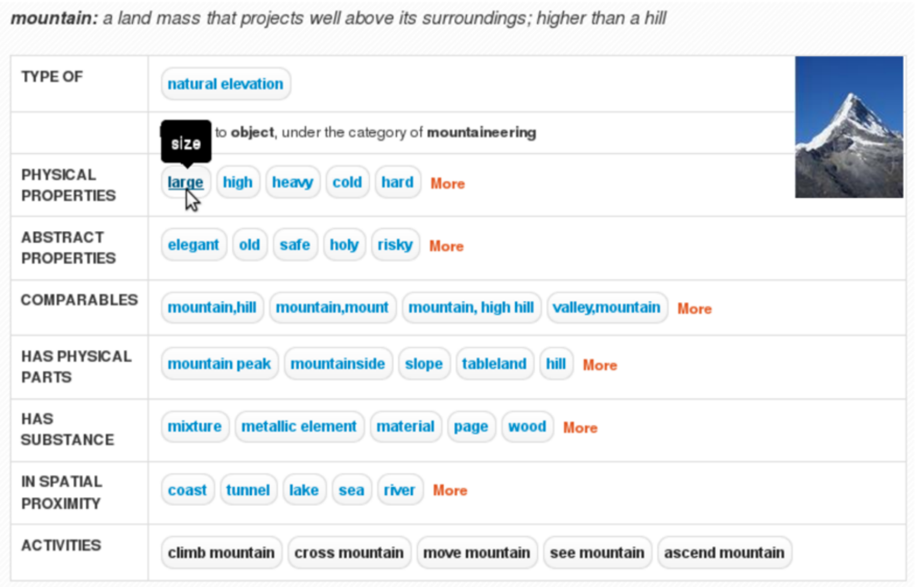
We then perform inference using probabilistic graphical models that can encode joint dependencies among different candidate activity frames. Unlike the previous contribution, this method goes beyond disambiguation of the arguments of an assertion; and, additionally assign roles to these arguments. A final taxonomy construction step groups together similar activity frames and forms a hierarchy of activities. For movie scripts with aligned movie data, we associate the correspond video key frames with our activities. Figure 3 provides an example of the resulting activity frames.

4 Results and Demonstration

Together, these methods have been used to create the WebChild 2.0 KB, which is one of the largest commonsense knowledge bases available, describing over 2 million disambiguated concepts and activities, connected by over 18 million assertions.

Among this data, we highlight the Knowlywood pipeline that produced 964,758 unique activity instances, grouped into 505,788 activity synsets. In addition to the edges mentioned above, we also obtain 581,438 `location`, 71,346 `time`, and 5,196,156 `participant` attribute entries over all activities. This is much larger than other commonsense KBs such as ConceptNet, refer Table 2.

The WebChild 2.0 KB is bigger and richer than any other automatically constructed commonsense KB. It can also be viewed as an extended WordNet (comprising not just words, but also activi-



mountain: *a land mass that projects well above its surroundings; higher than a hill*

TYPE OF	natural elevation
	size to **object**, under the category of **mountaineering**
PHYSICAL PROPERTIES	large · high · heavy · cold · hard · More
ABSTRACT PROPERTIES	elegant · old · safe · holy · risky · More
COMPARABLES	mountain,hill · mountain,mount · mountain, high hill · valley,mountain · More
HAS PHYSICAL PARTS	mountain peak · mountainside · slope · tableland · hill · More
HAS SUBSTANCE	mixture · metallic element · material · page · wood · More
IN SPATIAL PROXIMITY	coast · tunnel · lake · sea · river · More
ACTIVITIES	climb mountain · cross mountain · move mountain · see mountain · ascend mountain

Figure 4: WebChild 2.0 browser results for `mountain`. It presents semantic knowledge for concepts, comparisons, and activities. For more examples, visit `gate.d5.mpi-inf.mpg.de/webchild`

ties and other concepts expressed via multi-word expressions), with an orders of magnitude denser relationship graph (connecting the concepts with novel relations such as comparatives), and with additional multimodal content.

The WebChild 2.0 browser provides a user interface to semantically browse the current commonsense database, combining the knowledge from all of the above algorithms, refer to Figure 4.

5 Conclusion

From a resource perspective, people looking for commonsense knowledge bases had few options available before our construction of the WebChild 2.0 knowledge base. The available alternatives do not offer the same level of size, richness and semantic rigor over multiple modalities. In ongoing work, we are developing improved algorithms to prune noisy extractions, and computing the weights for the inference steps to distill cleaner knowledge.

WebChild 2.0 has already been effective in providing background knowledge to applications such as visual question answering (Wang et al., 2016) and neural relation prediction (Chen et al., 2016). The WebChild 2.0 data is freely downloadable at `http://www.mpi-inf.mpg.de/yago-naga/webchild/`, and browsable at `https://gate.d5.mpi-inf.mpg.de/webchild/`.

References

Jiaqiang Chen, Niket Tandon, Charles Darwis Hariman, and Gerard de Melo. 2016. Webbrain: Joint neural learning of large-scale commonsense knowledge. In *Proceedings of ISWC 2016*.

Catherine Havasi, Robert Speer, and Jason Alonso. 2007. Conceptnet 3: a flexible, multilingual semantic network for common sense knowledge. In *Proceedings of RANLP*.

Partha P. Talukdar and Koby Crammer. 2009. New regularized algorithms for transductive learning. In *Proceedings of ECML/PKDD*.

Niket Tandon, Gerard de Melo, Abir De, and Gerhard Weikum. 2015. Knowlywood: Mining activity knowledge from hollywood narratives. In *Proceedings of CIKM*.

Niket Tandon, Gerard de Melo, Fabian Suchanek, and Gerhard Weikum. 2014a. Webchild: Harvesting and organizing commonsense knowledge from the web. In *Proceedings of WSDM*.

Niket Tandon, Gerard de Melo, and Gerhard Weikum. 2011. Deriving a web-scale commonsense fact database. In *Proceedings of AAAI*.

Niket Tandon, Gerard De Melo, and Gerhard Weikum. 2014b. Acquiring comparative commonsense knowledge from the web. In *Proceedings of AAAI*.

Niket Tandon, Charles Hariman, Jacopo Urbani, Anna Rohrbach, Marcus Rohrbach, and Gerhard Weikum. 2016. Commonsense in parts: Mining part-whole relations from the web and image tags. In *Proceedings of AAAI*.

Peng Wang, Qi Wu, Chunhua Shen, Anton van den Hengel, and Anthony R. Dick. 2016. FVQA: fact-based visual question answering. *CoRR* abs/1606.05433. http://arxiv.org/abs/1606.05433.