# Learning New Relations from Concept Ontologies Derived from Definitions

**Jansen Orfan** and **James Allen**
Department of Computer Science, University of Rochester
Rochester, New York 14627 USA

## Abstract

Systems that build general knowledge bases from concept definitions mostly focus on knowledge extraction techniques on a per-definition basis. But, definitions rely on subtext and other definitions to concisely encode a concept's meaning. We present a probabilistic inference process where we systematically augment knowledge extracted from several WordNet glosses with subtext and then infer likely states of the world. From those states we learn new semantic relations among properties, states, and events. We show that our system learns more relations than one without subtext and verify this knowledge using human evaluators.

## Introduction

Word definitions contain information useful for building an ontology of conceptual common sense knowledge. (Allen et al. 2013) shows that causal relations between word senses can be automatically extracted from logical forms (LF) of WordNet glosses with high precision. However, this approach was limited to deriving relations directly from the gloss LFs or through basic logical inferences over the resulting knowledge base (KB).

Glosses are meant primarily for human readers, so they tend to be vague and leave a lot of knowledge unstated. For example, the definition for Stay_up is, "not go to bed"; if taken literally, a person can stay up while sleeping because they are not currently going to bed. The reliance of definitions on subtext and other definitions limits what can be done with logical inference because the LF of a gloss is not guaranteed to provide sufficient information about a concept.

In this paper we explore the possibility of learning more semantic relations from axioms in a KB built from LFs of WordNet glosses. We concern ourselves with the temporal relationships among word senses representing properties (e.g., Asleep), states (e.g., Sleeping), and the events that affect those states and properties (e.g., Wake_up). We augment previously learned concept knowledge with assumptions about the subtext of their source definitions. We then use the resulting KB to build a Markov Logic Network (MLN) and generate scenarios - artificially created sequences of events and states. Sets of scenarios provide a means to estimate the likelihood of certain semantic relationships between concepts that logical inference would miss. Relationships with a high likelihood can then be added to the original KB.



## Related Work

(Jiang, Lowd, and Dou 2012) uses MLNs to improve the accuracy of knowledge extracted by NELL (Carlson et al. 2010). By considering the joint probability of several facts they can identify and remove those that are unlikely. Although NELL differs in approach from (Allen et al. 2013), the goal of (Jiang, Lowd, and Dou 2012) is similar to ours: improve existing KBs through probabilistic reasoning.

(Beltagy et al. 2013) performs semantic textual similarity and textual entailment by representing LFs in an MLN to infer likely entailments of the text. Similarly, we also use LFs to create an MLN and find likely entailments.

## Modeling Scenarios

Scenarios are probabilistic sequences of events, states, and property ascriptions inferred from a set of initial assertions (see Figure 1). They are used to find correlations among temporal concepts and are not meant for general inference; so, our model is considerably simpler than the LFs found in (Allen et al. 2013). We use only three concepts representing events, properties/states, and time.

- Event - telic processes that affect properties (Wake_up)
- Property - properties, states, and atelic processes that objects may be ascribed (Asleep, Sleeping)
- Time - a discrete span of time; a time step

We use a temporal theory inspired by (Allen 1984); albeit much less expressive. The biggest departure from (Allen 1984) is that time steps cannot overlap - they can only be

| P= | | | |
|---|---|---|---|
| | **1.0** | ¬hold(Sleep, 1) | hold(Sleep, 2) |
| | **.9** | | ¬hold(Stay_up, 2) |
| | **.8** | occur(Go_to_bed, 1) | ¬hold(Awake, 2) |

Figure 1: Scenario inferred by asserting: ¬hold(Sleep,1) and hold(Sleep,2).

consecutive. Events occur during a single time step; the starting conditions of the event are true at the same time the event occurs and the effects of the event are true at the next time step. Unlike (Beltagy et al. 2013), we further simplify our model by not representing objects - that is, something that causes or is affected by an event or has a property. Instead we consider all properties to belong to a single unrepresented object and all events to affect it.

The following predicates are used to describe a scenario.

- occur(Event,Time) - an event occurs at a certain time. By convention, when an event takes place at time $t$ its preconditions are true at $t$ and its effects are true at $t$+1
- hold(Property, Time) - something has a property or is in a state at a certain time
- meets(Time,Time) - two time steps are consecutive

Finally, we make the assumption that properties do not spontaneously change (i.e., an event must have caused the change):
hold($p$,$t$) $\wedge$ ¬hold($p$,$t$+1) $\vee$ ¬hold($p$,$t$) $\wedge$ hold($p$,$t$+1)$\rightarrow \exists e$. occur($e$, $t$).

### Adding Subtext

We now look at two ways we add some of the subtext of a definitions to our axioms. The first way is a simple heuristic regarding concepts that entail the prevention or otherwise non-occurrence of an event - e.g. Stay_up - "not go to bed" or Keep_up - "prevent from going to bed". We assume that if something is kept up then it must be possible to go to bed; i.e. it entails the conditions necessary for going to bed: "not sleeping". Generally, if something prevents an event, $e$, then it entails whatever holds at the start of $e$.

We also tried to add subtext by treating definitions as possibly sufficient conditions of the concept they define. Concepts entail their definition; however, always assuming definitions entail their concept is problematic. In some cases it is desirable, like in Sleeping - "the state of being asleep" we can safely assume that anything that is in "the state of being asleep" is Sleeping. However, we cannot do that in the case of Unconscious - "not conscious". An object can arguably be both not Conscious and not Unconscious - e.g. being dead or otherwise non-living. We chose to use an MLN because it allows us to add uncertainty to the converse of a definition via weighted soft constraints. When creating a hard axiom from a definition we also create a soft converse axiom with some weight, [w]. In effect, we weaken our belief in the converse axioms and stronger contrary evidence can overcome the rule. With this model we can describe a scenario where a generic actor goes from sleeping to not sleeping and then infer the rest of the scenario(see Figure 1).

- Hard Necessary Conditions:
  - hold(Sleeping, t) $\rightarrow$ hold(Asleep, t).
  - occur(Awaken, t) $\rightarrow$ hold(Sleeping, t) $\wedge$ ¬hold(Sleeping, t+1).
- Soft Sufficient Conditions:
  - [w] hold(Asleep, t) $\rightarrow$ hold(Sleeping,t)
  - [w] hold(Sleeping, t) $\wedge$ ¬hold(Sleeping, t+1) $\rightarrow$ occur(Awaken, t)

## Inferring Semantic Relationships

We intend to infer three types of semantic relations from sets of scenarios created from definitions; the relations are among the most important to common sense about events and states.

- entailsP($p$,$p'$): $\forall t$. if holds($p$,$t$) then holds($p'$,$t$)
- preCondition($e$,$p$): $\forall t$. if occurs($e$,$t$) then holds($p$,$t$)
- postCondition($e$,$p$): $\forall t$. if occurs($e$,$t$) then holds($p$,$t$+1)

We use if/then statements and estimate each relation by, P(conclusion | premise)·P(¬premise | ¬conclusion) - instead of P(¬premise∨conclusion). Doing so lowers the chance we draw conclusions from weak evidence by reducing the baseline probability of a relationship between two concepts from p=.5 to p=.25. Furthermore, this fits our intuition that most properties and events are likely unrelated.

Instead of collecting observations from corpora or some other real-world source, we generate basic combinations of asserted conditions and then use our model to infer the likely state of the scenario. For each property, $p$, we generate sets of assertions for scenarios of a certain length by permuting all possible sequences of $p$ holding and $p$ not holding. Generating scenarios from events differs slightly in that we do not allow an event to occur twice consecutively as that would cause a contradiction in most cases. To remain tractable, we limit the maximum time steps in the scenarios and only use a subset of interrelated axioms in our MLN. For example, given a property, $p$, and a max length of two we create six scenarios: {¬hold($p$,1)}, {hold($p$,1)}, {¬hold($p$,1), ¬hold($p$,2)}, {¬hold($p$,1), hold($p$,2)},{hold($p$,1), ¬hold($p$,2)},{hold($p$,1), hold($p$,2)}

From these sets of scenarios we estimate the probabilities of the relations introduced above. Scenarios best characterize only the concepts used in their original assertions. This means that Figure 1 only characterizes the relationships involving Sleep. To properly estimate the probability of R($c1$, $c2$) we only consider the scenarios that were generated from both $c1$ and $c2$. More formally we estimate the probability of a relation like so:

Let $R$ be a binary relation and $\mathrm{P}_C(R(x, y))$ give the probability of $R(x, y)$ estimated using all of the scenarios generated from concept $C$.

$$\mathrm{P}(R(x,y)) = \frac{\mathrm{P}_x(R(x,y)) + \mathrm{P}_y(R(x,y))}{2}$$

## Experiment

We created a small micro-theory of sleep from hand selected concepts in WordNet related to Sleep. We then hand converted the LFs of the concepts' glosses into several axioms and chose 2.0 for the weight of the soft axioms. Example hard axioms are shown below, soft axioms can be inferred.

- occur(Wake_up,t)$\rightarrow$ hold(Sleeping,t) $\wedge$ ¬hold(Sleeping,t+1).
- hold(Sleeping,t)$\rightarrow$ hold(Asleep,t).
- hold(Sleep,t)$\rightarrow$ ¬hold(Conscious,t) $\wedge$ hold(Rest,t).

| **Question** | **Property** |
|---|---|
| If something is sleeping$_x$ then it is ___ asleep$_y$. | entailsP($x$,$y$) |
| If something wakes up$_x$ then it is ___ sleeping$_y$ immediately before. | preCondition($x$,$y$) |
| If something awakens$_x$ then it is ___ unconscious$_y$ immediately after. | postCondition($x$,$y$) |

Table 1: Sample questions with corresponding property.

| **Response** | **System Criteria** | **Value** |
|---|---|---|
| definitely | $.8 \leq$ P(R($x$,$y$)) | 5 |
| likely | $.6 \leq$ P(R($x$,$y$)) $< .8$ | 4 |
| likely not | $.6 \leq$ P(R($x$,$\neg y$)) $< .8$ | 2 |
| definitely not | $.8 \leq$ P(R($x$,$\neg y$)) | 1 |
| maybe | $otherwise$ | 3 |

Table 2: Possible responses, conditions for the system to respond that way, and the values assigned for evaluation.

- hold(Awake,t)→ hold(Conscious,t) ∧ ¬hold(Sleep,t).
- hold(Kip,t)→ hold(Asleep,t).
- occur(Go_to_bed,t)→ ¬hold(Kip,t) ∧ hold(Kip,t+1).
- occur(Knock_out,t)→
¬(hold(Unconscious,t) ∨ hold(Senseless,t)) ∧
(hold(Unconscious,t+1) ∨ hold(Senseless,t+1)).
- hold(Unconscious,t)→ hold(Conscious,t).
- hold(Stay_up,t)→ ¬occur(Go_to_Bed,t) ∧ ¬hold(Kip,t).

For each concept we generated several scenarios of lengths one, two, and three and then used Alchemy (Kok et al. 2005) to infer the likely state based on our model. We then estimated the probability of each of the three semantic relations - entailsP, preCondition, and postCondition. Figure 2 shows some of the knowledge inferred about Sleep.

To evaluate the approach, we created surveys with three types of questions each corresponding to one of the three relations (see Table 1). Each question asks the respondent to fill in a blank in the sentence with one of five responses. The respondent was given glosses for both concepts but was asked to use their own knowledge when answering.

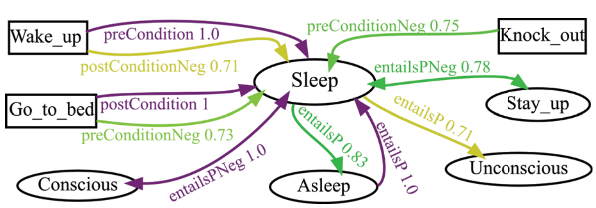


Figure 2: Some inferred relationships and their probabilities. Events are boxes and properties are ovals. A relation ending in "Neg" indicates that the sink is negated. As in postCondition(Wake_up,¬Sleep)

To test the effectiveness of both subtext heuristics introduced above, we created another model using only hard axioms derived from the LFs without subtext added. Table 2 shows the possible answers and the criteria for both models to give that response. For human comparison, we created two surveys, each had 25 unique questions which were chosen equally among the set of system responses as well as the set of question types, i.e., all of the system responses are uniformly distributed as are the question types. The first survey had six respondents while the second had only five.

To compare system responses to human responses we assigned each possible response a value (see Table 2) and for each question we took the average value of all of the responses. The human answers were fairly consistent: the average standard deviation for each question being about .67. We count a system response as correct only if the difference between it and the average human response is $< 1$.

Table 3 compares the average human response with results from the model using the subtext heuristics with the model using only the axioms derived literally from LFs. Accuracy is simply the number of correct answers divided by the number of answers. Precision only considers the questions that the system did not respond, "maybe".

The heuristic system has a high accuracy compared to the LF-Only system. The LF-Only system gives more false negatives by answering "Maybe" 17 times more often. Among just those 17 questions, the heuristic system's precision was .76 (13 out of 17 were correct). Meaning, the heuristic system added more knowledge albeit with lower precision.

## Conclusion

While the results reported here are encouraging there is much to be done before such a system can generate a KB from all of WordNet. At the forefront is the creation of an automated way of converting LFs to axioms. Recent experiments show that this likely can be accomplished by defining patterns in the LF graph and the relationships they represent; like, "x stops y" : changeFrom(x,y).Furthermore, a WordNet scale MLN is unlikely to be tractable. We have to devise a way to only use knowledge relevant to whichever concept we want to learn about. (Jiang, Lowd, and Dou 2012) addressed a similar issue by breaking their KB into different domains. We will likely follow suit and build microtheories about a concept by looking at nearby concepts in the LF graph. Completing both these tasks will allow us to run much larger experiments and further improve our model.

## Acknowledgements


This work was supported in part by the Nuance Foundation and the Office of Naval Research (Grant #N000141110417).


| **Heuristic** | | **LF-Only** | |
|---|---|---|---|
| Acc. | Prec. | Acc. | Prec. |
| $\frac{34}{50} = .68$ | $\frac{31}{37} = .84$ | $\frac{20}{50} = .4$ | $\frac{17}{20} = .85$ |

Table 3: Results with and without heuristic subtext axioms

## References

Allen, J.; de Beaumont, W.; Galescu, L.; Orfan, J.; Swift, M.; and Teng, C. M. 2013. Automatically Deriving Event Ontologies for a Commonsense Knowledge Base. In *Proceedings of the International Conference for Computational Semantics*.

Allen, J. F. 1984. Towards a general theory of action and time. *Artificial Intelligence* 23(2):123–154.

Beltagy, I.; Chau, C.; Boleda, G.; Garrette, D.; Erk, K.; and Mooney, R. 2013. Montague Meets Markov: Deep Semantics with Probabilistic Logical Form. *Joint Conference on Lexical and Computational Semantics (*SEM)* 11–21.

Carlson, A.; Betteridge, J.; Kisiel, B.; and Settles, B. 2010. Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010)*.

Jiang, S.; Lowd, D.; and Dou, D. 2012. Learning to Refine an Automatically Extracted Knowledge Base Using Markov Logic. *2012 IEEE 12th International Conference on Data Mining* 912–917.

Kok, S.; Singla, P.; Richardson, M.; Domingos, P.; Sumner, M.; Poon, H.; and Lowd, D. 2005. The Alchemy system for statistical relational AI. *University of Washington, Seattle*.