Becoming Different:
A Language-Driven Formalism for Commonsense Knowledge

James F. Allen[1,2] and Choh Man Teng[1]

[1]Institute for Human and Machine Cognition, Pensacola FL 32502 [2]University of Rochester, Rochester NY 14627
{jallen, cmteng}@ihmc.us

Abstract

In constructing a system for learning commonsense knowledge by reading online resources for word definitions, a key challenge is to develop a formalism rich and expressive enough to capture commonsense concepts expressed in natural language. Derivations based on natural language impose strong requirements on the nature of the representation. Specifically, predicates should correspond to word senses and their argument structures in the language, and complex formulas should be constructed compositionally in a way that parallels the structure of language. To provide a suitable representation framework we need to extend interval temporal logic in several ways, including organizing time around objects rather than predicates, and developing a theory of scales. As a driving example, we analyze core meanings of the verbs *change* and *become* and the adjective *different* and show, after appropriate development of our formalism, how the desired meaning of *change* can be derived from one of its definitions in WordNet: *become different*.

Introduction and Motivation

Many applications of Artificial Intelligence, and natural language processing in particular, are hindered by a lack of extensive commonsense knowledge bases. Vast amount of knowledge is needed to understand language, as well as to plan and reason about the world. Much of it is quite mundane: if you fall asleep you become asleep; you use keys to unlock doors; people don't like pain. While it is everyday ordinary stuff, such knowledge is critical if systems are to achieve human-levels of deep understanding of language.

While there have been some efforts to encode large amounts of commonsense knowledge by hand, e.g., Cyc (Lenat, 1995), SUMO (Niles and Pease, 2001), such efforts barely make a dent in accumulating the knowledge that is needed. Further, such efforts generally are expressed in formal notations using predicates motivated by mathematics rather than attempt to create a close link to the elements of natural language (e.g, word meanings, semantic roles).

Our goal is to create most of the commonsense knowledge base by reading. While recent efforts such as NELL (Carlson *et al.*, 2010) and TextRunner (Yates *et al.*, 2007) have been effective at collecting vast amounts of knowledge about instances (e.g., Chicago is a city) and semantic patterns (e.g., people kill people), the commonsense knowledge we need is definitional in nature to enable necessary entailments: e.g., *kill* means *cause to die*; *murder* means *kill intentionally*; *fall asleep* means *change from awake to asleep*. We are working on building knowledge bases automatically by reading definitions (Allen *et al.*, 2013), starting with the definitions in WordNet (Fellbaum, 1998). The goal of this paper is to describe the formalism we have developed in order to facilitate the construction of effective axioms directly from natural language definitions. This requirement puts strong constraints on the nature of the representation that we note here and will develop further in the paper.

In many ways, this paper has similar goals and motivations as those of Hobbs (2013). We both want to axiomatize core commonsense notions of events. Some of the differences are in the style of the formalism—we start from an explicit interval temporal logic and build from there, whereas Hobbs places eventualities as central and time plays a secondary role. But the most important difference is our emphasis on building a formalism that supports learning the knowledge by reading. Whereas Hobbs does a hand analysis of core verbs, such as *cause* and *have*, and identifies a few core meanings that he argues subsume all the WordNet senses, our goal is to axiomatize automatically most of the WordNet senses directly from their definitions. We would rather have a messy knowledge base that covers as much of the subtleties of language and word senses as possible, rather than developing a more minimal, but more abstract, theory.

We base our formalism on the one developed in Allen & Ferguson (1994), henceforth **AF,** and Allen (1984), in which events are formalized in an interval temporal logic in a way that enables planning and reasoning. AF has reified events, with functional relations capturing semantic roles and arguments. For example, *Jack lifted the ball* (over interval t_1) is represented as

This work was supported in part by NSF grant IIS-1012205, ONR N000141210547 and James S. McDonnell Foundation 220020263.

$\exists e.(\text{LIFT}(e) \wedge (agent(e)=jack_1) \wedge (affected(e)=ball_1) \wedge (time(e)=t_1))$

While this is the underlying logic, as in AF, we usually abbreviate such expressions as $\text{LIFT}(jack_1, ball_1, t_1, e)$ when the specific roles are obvious. When using this abbreviation, predicates might appear to have a varying number of arguments, but this is just because of the abbreviation convention and not a formal part of the logic. The framework also builds from Allen's interval logic of action and time (Allen 1983, 1984), where time periods can be related by Allen's temporal relations. For this paper, we only need the *meets* relation, written $t_1{:}t_2$ and "*during or equal"*, written $t_1 \subseteq t_2$. A *moment* is an interval that has no true subintervals and captures minimally perceptible moments in time. Decomposable periods are often referred to as true intervals. The predicate *Moment* allows us to distinguish moments from true intervals. We also add the strong constraint on our temporal models by asserting that all intervals are constructed out of moments. This can be captured by the simple axiom that every interval contains a moment:

Discrete Time Axiom: $\forall t.\ \exists t' \subseteq t.\ Moment(t')$

Representation and Linguistic Structure

The key driving constraint of this work is that the representational framework should closely parallel linguistic elements and structure. We believe this is essential to enable learning conceptual knowledge by reading definitions. Specifically, we require an equivalence between predicates and functions in the knowledge base and word senses in the language. The word senses correspond 1-1 to the predicates and the arguments to these predicates correspond to the linguistic arguments that the word senses may take. This will allow us to introduce new predicates into the knowledge base in a systematic and straightforward way based on the words used.

The representation of events in AF satisfies this constraint for verbs—the event predicates correspond to verb senses, and the reified events allow the argument functions that correspond directly to a verb's semantic roles. Beyond events though, we need some extensions. First, if we are to maintain the close link between linguistic structure and the representation, we need to reify predicates (e.g., adjective meanings) so that they may serve as arguments to other predicates. To distinguish such predicates from the formal predicates in the logic we will call them **property predicates** (see Table 1). Intuitively, property predicates identify characteristics of the world that can be directly perceived in a moment of time (e.g., at the present moment). For instance, consider the sentence *John's mood changed from happy to sad.* There are three arguments to the event predicate CHANGE: the object undergoing the change (*John's mood*), the prior state (*happy*) and the resulting state (*sad*). By reifying property predicates, we can express this as:

CHANGE(mood(john), Happy, Sad, t, e).

While this is a natural mapping of the sentence meaning, such statements cannot be made in classical first order logic

Construct	*Formal Status*	*Linguistic correlate*	*Notation and Example(s)*	
Formal Predicates	Predicates in the logic	none	Small caps	*TRUEOF*
Event predicates	Predicates in the logic	verb senses	Small caps	*CHANGE*
Property predicates	Terms that denote properties	nouns and adjective senses	Initial caps	*Happy, Dog, ...*
Property functions	Functions that apply to property predicates	comparatives, nominalizations	Start with underbar	*_er, _ness, ...*
Scales	Terms that denote scales	Some nouns (e.g., *size)*	Small caps	*SIZE*
Objects	Terms that denote domain objects	proper names, noun phrases	Lower case	*john, x, father(x)*

Table 1: Notation and Ontological Categories

because predicates, such as *Happy,* cannot serve as arguments to other predicates. While there might be some technical tricks to try to avoid such a generalization of the formalism, we will soon see additional reasons for why the reified predicates are convenient for capturing commonsense knowledge, particularly when representing scales.

We need one more significant change from AF to allow us to stay true to the structures of language. Consider one definition of *change* in WordNet: *Become different*. The meaning of this expression is that an object that changes becomes different from what it was before. We will spend some time defining exactly what this means, but for now just consider that the predicate *Different* needs to apply to the same object *twice*, but at different times. One cannot express such a relation if we can only associate times with predicates or properties, as in AF. Rather, we need a more general logic where terms, rather than the predicates, are temporally qualified. Specifically, we introduce a new function that takes the name of an object and a time and denotes that object over that time, e.g., *x@t* represents "object *x* over time *t*". We refer to these as *temporally situated objects.* Thus, *John is Happy today* is written as

TRUEOF(john@today, Happy)

We need another predicate for binary relations. For example, we would express *I am different today from yesterday* as:

TRUEOF2(me@yesterday, me@today, Different)

Such a proposition cannot be directly expressed using a logic that only attaches time to the predicates.

The notion that objects are temporally situated and properties are not is in stark contrast to standard temporal logics in which objects are atemporal and properties change over time. This view has been discussed in philosophy, going back to before Whitehead (1929). We introduce a predicate EXISTS that defines the temporal range of an object, i.e., when the temporally situated object o@t exists. For instance, if I was born in 1983, then EXISTS(me@1984) and ~EXISTS(me@1982) both hold. Properties only hold on temporally situated objects that exist:

$$\forall t.\ (\sim Exists(o_1@t_1) \supset \forall P, o_2, t_2.\ (\sim \mathrm{TRUEOF}(o_1@t_1, P) \wedge \sim \mathrm{TRUEOF2}(o_1@t_1, o_2@ t_2, P) \wedge \sim \mathrm{TRUEOF2}(o_2@ t_2, o@t_1, P)))$$

Property predicates are homogeneous, which means that if a property holds over some time period I, then it holds over all subintervals of I. We need to define this for both unary and binary predicates:

***Homogeneity Axioms*[1]**

(H1) $\forall o, P, t\ .\ (\mathrm{TRUEOF}(o@t, P) \equiv \forall t' \subseteq t.\ \mathrm{TRUEOF}(o@t', P))$

(H2) $\forall o_1, o_2, P, t_1, t_2\ .\ (\mathrm{TRUEOF2}(o_1@t_1, o_2@t_2, P) \equiv \forall t_1' \subseteq t_1,\ t_2' \subseteq t_2\ .\ \mathrm{TRUEOF2}(o_1@t_1', o_2@t_2', P))$

Note that for binary relations, homogeneity applies to all possible pairs of subintervals associated with the objects. This is a very strong constraint but necessary because there is no constraint on how the time periods t_1 and t_2 relate to each other.

As a final observation, note that we have two types of negation, and we use the notation in AF. *Weak negation*, e.g, $\sim \mathrm{TRUEOF}(b@t,\ Clear)$, simply states that $\mathrm{TRUEOF}(b@t,\ Clear)$ does not hold —i.e., it is not the case that b is clear over the entire time interval t, although it might be true over a subpart of t. *Strong negation*, in contrast, uses a negation function on property predicates, which we write as $\mathrm{TRUEOF}(b@t,\ \neg Clear)$. As in AF, we have an axiom defining strong negation:

$$\forall o, P, t.\ (\mathrm{TRUEOF}(o@t, \neg P) \equiv \forall t' \subseteq t.\ \sim \mathrm{TRUEOF}(o@t', P))$$

Note that a direct corollary of this axiom is that strong negation and weak negation are equivalent for moments:

$$\forall o, P, t.\ \ (Moment(t) \supset (\mathrm{TRUEOF}(o@t, \neg P) \equiv \sim \mathrm{TRUEOF}(o@t, P)))$$

Also, we get that for any moment either P or $\neg P$ holds. This can be extended to $\mathrm{TRUEOF2}$ in the obvious way.

There is one more important constraint on the logic that we need that was captured in AF's *discrete variation axiom schema.* This constraint prevents the possibility of properties changing truth values infinitely often within an interval. The philosophical underpinnings of this issue have been discussed in for example (Hamblin, 1972).

Discrete Variation Axiom

$$\forall o, P, t.\ (\sim \mathrm{TRUEOF}(o@t, P) \equiv \exists\, m \subseteq t.\ \ (Moment(m) \wedge \mathrm{TRUEOF}(o@m, \neg P)))$$

With this in hand, we can then prove some useful theorems about strong negation:

Negation Inverse Theorems

(N1) $\forall o, P, t.\ (\mathrm{TRUEOF}(o@t, \neg\neg P) \equiv \mathrm{TRUEOF}(o@t, P))$

(N2) $\forall o_1, o_2, P, t_1, t_2.\ (\mathrm{TRUEOF2}(o_1@t_1, o_2@t_2, \neg\neg P) \equiv \mathrm{TRUEOF2}(o_1@t_1,\ o_2@t_2,\ P))$

In the rest of the paper we will develop these ideas further by examining how we might define three related words: *change, become* and *different*. We chose these three because they are closely related in their definitions in WordNet. We can explore the adequacy of our formalism by examining how well their definitions capture the intuitive senses of the words. Specifically, we examine a key definition of *change* in WordNet, namely *become different*. A basic desideratum of our formalism is that the definitions of *become* and *different* should combine compositionally to capture what it means to *change*. If we can accomplish this, we will have some initial confidence that we have created a suitable groundwork for acquiring, on a large scale, commonsense knowledge by reading definitions automatically.

A First Attempt to Define Change

Intuitively we might define *CHANGE* and *BECOME* as follows. A *CHANGE* event e, involving an object o over time t, from property P_1 to property P_2, occurs when there are two time intervals t_1 and t_2, such that P_1 is true of o immediately before t (over t_1) and P_2 is true of o immediately after t (over t_2).

$$\forall o, P_1, P_2, t, e.\ (\mathrm{CHANGE}(o@t, P_1, P_2, e) \equiv \exists\, t_1,t_2.\ (t_1{:}t{:}t_2 \wedge \mathrm{TRUEOF}(o@t_1,P_1) \wedge \mathrm{TRUEOF}(o@t_2,P_2)))$$

Similarly, a *BECOME* event e, involving an object o over time t, to property P, might be defined as follows:

$$\forall o, P, t, e.\ (\mathrm{BECOME}(o@t, P, e) \equiv \exists t_1,t_2.\ (t_1{:}t{:}t_2 \wedge \mathrm{TRUEOF}(o@t_1,\neg P) \wedge \mathrm{TRUEOF}(o@t_2,P)))$$

The two events are clearly related in some way. We would like whenever a *CHANGE* event obtains, a corresponding *BECOME* event obtains: $\mathrm{CHANGE}(o@t,\ P_1,\ P_2,\ e_1) \supset \exists\ e_2.\ \mathrm{BECOME}(o@t, P_2, e_2)$, or roughly, whenever o changes from P_1 to P_2, we also have o becomes P_2. However, $\mathrm{CHANGE}(o@t,\ P_1,\ P_2,\ e_1)$ only gives us, with appropriate instantiations, $\mathrm{TRUEOF}(o@t_2,\ P_2)$ but not $\mathrm{TRUEOF}(o@t_1,\ \neg P_2)$ as is needed by the *BECOME* event. The two predicates P_1 and P_2 in *CHANGE* are currently not constrained by any relation.

Hobbs addresses this issue in his definition of *CHANGE* by requiring that P_1 and P_2 must be contradictory, but this constraint is too strong. For example, it will not allow us to have a *CHANGE* event of an object changing from being small to being tiny. After this change, we are tiny but at the same time we are still small. To account for this subtlety, we introduce a predicate combination function, "*P but not Q*", which might be realized in English as *small but not tiny*. We write it as $P \backslash Q$, where P and Q are property predicates, and define it as:

$$\forall o, P, Q, t.\ (\mathrm{TRUEOF}(o@t, P\backslash Q) \equiv \mathrm{TRUEOF}(o@t, P) \wedge \mathrm{TRUEOF}(o@t, \neg Q))$$

That is, $P\backslash Q$ is true of o whenever P but not Q is true of o.

[1] Note that homogeneity only applies to properties that can be true over a moment. Thus, an expression such as "grew more than 5 inches" cannot be captured with a property predicate as it can only be true over certain intervals. We do not have the space to discuss such predicates here, and they are not important to the content of this paper.

Now we can reformulate the *CHANGE* predicate:

$$\forall o, P_1, P_2, t, e.\ (\mathit{CHANGE}(o@t, P_1, P_2, e) \equiv \exists t_1,t_2.(t_1{:}t{:}t_2 \wedge \mathit{TRUEOF}(o@t_1,P_1\backslash P_2) \wedge \mathit{TRUEOF}(o@t_2,P_2)))$$

Now we are able to express a change of an object from being small to being tiny: small but not tiny is true of the object before the change, while tiny is true of the object after the change.

One can verify that this formulation also applies to predicates that *are* inconsistent, as in $\mathit{CHANGE}(light_1@t, Red, Green, e_1)$, and that with this definition, $\mathit{CHANGE}(o@t, P_1, P_2, e_1) \supset \exists e_2.\ \mathit{BECOME}(o@t, P_2, e_2)$ holds.

We have skirted one of the most crucial aspects of *CHANGE* in our discussion so far. The relationship between the pairs of predicates that can legitimately occupy the P_1 and P_2 slots in a change event is still under-constrained. The formulation above allows, for example, a change of an object from being red to being small. While this might be acceptable from a logical point of view, it does not capture intuitions in language about change. To tackle this problem we need to develop a theory of predicate relatedness, which is closely associated with the notion of scales, discussed next.

A Theory of Scales

There are three common types of scales, classified according to the kind of relation between the elements on the scale: interval (e.g. temperature as Celsius degrees), ordinal (e.g. edibility as {raw, ripe, rotten}) and the degenerate abstract scale in which no relations need exist between the values on the scale (e.g. occupations as {Farmer, Chef, ...}).

A **scale** consists of a partially ordered set of values with an associated function from objects over time intervals to sets of values on the scale, and predicates corresponding to sets of values on the scale. Typically, names of scales (and scale functions) stem from nouns and scale value predicates stem from adjectives in natural language. For example, $height(o@t)$ maps a temporally situated object to the set of values on the *HEIGHT* scale that the object takes over period t. A predicate such as *Tall,* corresponding to an adjective of the same name, is true of objects whose height values over t is in the upper range of the *HEIGHT* scale. We will define scale value predicates more formally shortly.

In this example, the values in *HEIGHT* are fully ordered (it is an interval scale), and the predicates *Tall* and *Short* are captured by convex subsets of *HEIGHT* values. In general, however, the scale values need not be fully ordered, and the scale predicates need not be convex (e.g. the predicate denoting "not of medium height", which corresponds to "tall or short"). Note that an object o may take different values on a scale sc over a given time interval t. Thus, $sc(o@t)$ is a *set* of values. For the special case of moments, $sc(o@m)$ is a singleton for any moment m. The following relates the two:

$$\forall sc, o, t.\ (sc(o@t) = \cup\, sc(o@m){:}\ m \subseteq t \wedge Moment(m))$$

If sc is not applicable to $o@t$, then $sc(o@t)$ is empty, for example, $mood(rock_1@t) = \varnothing$.

We extend the ordering relation over values on a scale to subsets of values on a scale: Subset S_1 is less than subset S_2 iff every value in S_1 is less than every value in S_2.

We can then define predicates for comparing temporally situated objects. Defining equality on a scale is more complex than one might think. We cannot use simple equality over the scale value sets (that is, have an axiom $\mathit{TRUEOF2}(o_1@t_1, o_2@t_2, ScaleEqual(sc)) \equiv (sc(o_1@t_1) = sc(o_2@t_2))$ (*)) as there might be subintervals where they were not equal. For example, my car accelerates from 0 to 60 mph while your car decelerates from 60 to 0 mph. Our speeds take on the same set of values over the entire mentioned time interval(s), but our speeds over any subintervals are rarely equal. Thus (*) violates the homogeneity requirement. We therefore adopt the following stronger definition of *ScaleEqual* with explicit mention of the subintervals.

$$\forall sc, o_1, o_2, t_1, t_2.\ (\mathit{TRUEOF2}(o_1@t_1, o_2@t_2, ScaleEqual(sc)) \equiv \forall t_1' \subseteq t_1, t_2' \subseteq t_2.\ (sc(o_1@t_1') = sc(o_2@t_2')))$$

This consideration does not apply to the less-than comparison between two temporally situated objects, but we include the subinterval specification for uniformity:

$$\forall sc, o_1, o_2, t_1, t_2.\ (\mathit{TRUEOF2}(o_1@t_1, o_2@t_2, ScaleLessThan(sc)) \equiv \forall t_1' \subseteq t_1, t_2' \subseteq t_2.\ (sc(o_1@t_1') < sc(o_2@t_2')))$$

As a consequence, if $\mathit{TRUEOF2}(o_1@t_1, o_2@t_2, ScaleEqual(sc))$, then both $sc(o_1@t_1)$ and $sc(o_2@t_2)$ map to an identical singleton, since some subintervals are moments. If $\mathit{TRUEOF2}(o_1@t_1, o_2@t_2, ScaleLessThan(sc))$, then $sc(o_1@t_1)$ and $sc(o_2@ t_2)$ are disjoint.

Now, let us explore some of the nuances of these definitions in a few examples.

Example 1 Growing up I was always the same height as my sister at the same age (so I got all her hand-me-down clothes).

$$\forall t_1, t_2.\ (\mathit{TRUEOF2}(me@t_1, sister@t_2, ScaleEqual(\mathit{AGE})) \supset \mathit{TRUEOF2}(me@t_1, sister@t_2, ScaleEqual(\mathit{HEIGHT})))$$

Here, I at age 3 was as tall as my sister at age 3; I at age 4 was as tall as my sister at age 4; etc.

Example 2 I have always been the same height as my sister (so she can't put things up on the shelves I can't reach... and neither can I).

$$\forall t.\ (Moment(t) \wedge Exists(me@t) \wedge Exists(sister@t) \supset \mathit{TRUEOF2}(me@t, sister@t, ScaleEqual(\mathit{HEIGHT})))$$

Here, I was as tall as my sister on March 25, 1992; on December 31, 1999; etc. It is not the case that $\mathit{TRUEOF2}(me@t, sister@t, ScaleEqual(\mathit{HEIGHT}))$ for arbitrary time intervals t, since this assertion is only true if my height and my sister's height are identical and constant over the entire t. We thus need to circumscribe the length of the time interval under consideration: moments are suitable as our heights cannot change within a moment. These moments should further be constrained to be drawn from only when both my sister and I are in existence so that it makes sense to talk about our heights.

We will make use of the following lemma in a later section. It states that if two temporally situated objects are not equal on a scale, then they have disjoint values on that scale.

Lemma 1 *(¬ScaleEqual)*

$\forall sc, o_1, o_2, t_1, t_2.\ (\mathit{TRUEOF2}(o_1@t_1, o_2@t_2, \neg ScaleEqual(sc))$
$\quad \supset (sc(o_1@t_1) \cap sc(o_2@t_2) = \varnothing))$

Scale Value Predicates

As discussed earlier scale value predicates typically correspond to adjectives in natural language and they denote (often but not necessarily convex) subsets of values on the scale. They have additional properties with respect to their scales. We use

$$\mathit{SCALEVALPRED}(sc, Vp, +/-/\phi)$$

to denote that the predicate *Vp* is a predicate with a positive (+), negative (-) or neutral (ϕ) orientation on scale *sc*. (The function of the orientation $+/-/\phi$ are needed for handling comparatives and superlatives, but we have no room to discuss these here in this paper.) Two examples are *SCALEVALPRED(TEMPERATURE, Cold, -)* and *SCALEVALPRED(MOOD, Happy, +)*. As a convenient abbreviation, we define

$$\mathit{SCALEPRED}(sc, Vp) \equiv \exists x.\ \mathit{SCALEVALPRED}(sc, Vp, x)$$

when the orientation of the scale value predicate is not important. Note also that most adjectives are ambiguous as to which scales they refer to. We differentiate them by adding the scale they refer to as a subscript. For example, the adjective *deep* has at least three senses, captured by three predicates: *SCALEPRED(DEPTH,* $\mathrm{Deep}_{\mathrm{DEPTH}}$), *SCALEPRED(PROFUNDITY,* $\mathrm{Deep}_{\mathrm{PROFUNDITY}}$) and *SCALEPRED(INTENSITY,* $\mathrm{Deep}_{\mathrm{INTENSITY}}$). We will sometimes omit the subscript to Vp_{sc} when the associated scale *sc* is clear from the context. Otherwise if scale value predicates are used without the subscript, then they remain ambiguous between all their senses. In other words,

$\forall o, t, Vp.\ (\mathit{TRUEOF}(o@t, Vp) \wedge \exists sc.\ \mathit{SCALEPRED}(sc, Vp)$
$\quad \equiv \exists sc'.\ (\mathit{SCALEPRED}(sc', Vp) \wedge \mathit{TRUEOF}(o@t, Vp_{sc'})))$

An object may have more than one scale value predicate holding on a scale. For example, an object may be both small and tiny at the same time.. We may also have composite scale value predicates, for example, $\mathit{TRUEOF}(o@t, Small_{\mathrm{SIZE}} \setminus Tiny_{\mathrm{SIZE}})$ would indicate that an object is small but not tiny. In general, we may define a scale value predicate for any "intuitive" subset of values on a scale.

Let us define a function for the scale values corresponding to a scale value predicate:

$$\forall sc, Vp.\ (scaleVal(sc, Vp) = \begin{cases} \cup sc(o@t): \mathit{TRUEOF}(o@t, Vp) & \text{if } \mathit{SCALEPRED}(sc, Vp)) \\ \varnothing & \text{otherwise}) \end{cases}$$

Note in this definition, if *Vp* is not on the scale *sc*, then $scaleVal(sc, Vp) = \varnothing$. We then have the following:

$\forall sc, Vp, o, t.\ (\mathit{SCALEPRED}(sc, Vp) \wedge sc(o@t) = \varnothing$
$\quad \supset (\mathit{TRUEOF}(o@t, Vp) \equiv sc(o@t) \subseteq scaleVal(sc, Vp)))$

and we can show

$\forall sc, Vp, o, t.\ (\mathit{SCALEPRED}(sc, Vp)$
$\quad \supset (\mathit{TRUEOF}(o@t, \neg Vp) \equiv sc(o@t) \cap scaleVal(sc, Vp) = \varnothing))$

If *Vp* is a scale value predicate on scale *sc*, that is, $\mathit{SCALEVALPRED}(sc, Vp, +/-/\phi)$, so is $\neg Vp$ but with reverse orientation: $\mathit{SCALEVALPRED}(sc, \neg Vp, -/+/\phi)$. Also if Vp_1 and Vp_2 are scale value predicates on a scale *sc*, so is $Vp_1 \setminus Vp_2$ with the same orientation as Vp_1.

Note that Lemma 1 can be stated in terms of scale value predicates.

Corollary 2 *(¬ScaleEqual)*

$\forall sc, o_1, o_2, t_1, t_2.\ (\mathit{TRUEOF2}(o_1@t_1, o_2@t_2, \neg ScaleEqual(sc))$
$\quad \supset \exists Vp.\ (\mathit{SCALEPRED}(sc, Vp) \wedge \mathit{TRUEOF}(o_1@t_1, Vp)$
$\qquad \wedge \mathit{TRUEOF}(o_2@t_2, \neg Vp)))$

Besides expressing scale values as adjectives, natural language also describes quantitive scale values in terms of units and quantities. We will not go into details here, but to accommodate different unit systems on the same scale, especially for interval scales, we introduce a mechanism to produce scale value predicates by composing units and quantities, written as *<unit, quantity>*. For example, "an object *o* is worth 10 dollars at time *t*" can be expressed by $\mathit{TRUEOF}(o@t, \langle dollar, 10\rangle_{\mathit{WORTH}})$. With arithmetic operations we can define conversion rates between different units, for example,

$\forall o, t, x.\ (\mathit{TRUEOF}(o@t, \langle inch, x\rangle_{\mathrm{LENGTH}})$
$\qquad \equiv \mathit{TRUEOF}(o@t, \langle cm, 2.54x\rangle_{\mathrm{LENGTH}}))$.

Using Scales: Revisiting *CHANGE* and *BECOME*

Our previous formulation of *CHANGE* allows an object to change from being red to being small. This prompted a detour to develop a theory of scales. Now let us reconsider *CHANGE*, using scales to solve the original problem.

A *CHANGE* event can only occur when the "from" and "to" predicates involved are related by a common scale. The previous definition can be extended with the additional requirement that the two predicates exist on the same scale:

$\forall o, Vp_1, Vp_2, t, e.\ (\mathit{CHANGE}(o@t, Vp_1, Vp_2, e)$
$\quad \equiv \exists t_1, t_2, sc.\ (t_1{:}t{:}t_2 \wedge \mathit{SCALEPRED}(sc, Vp_1) \wedge \mathit{SCALEPRED}(sc, Vp_2)$
$\qquad \wedge \mathit{TRUEOF}(o@t_1, Vp_1 \setminus Vp_2) \wedge \mathit{TRUEOF}(o@t_2, Vp_2)))$

With this definition, it is not permissible to change from being red to being small, because there is no natural scale with both *Red* and *Small* as its scale value predicates. But $\mathit{CHANGE}(light_1@t, Red, Green, e_1)$ is permissible since both *Red* and *Green* reside on the *COLOR* scale.

Now consider defining *BECOME,* as in the example "The light becomes green", which can be formulated as

$\mathit{BECOME}(light_1@t, Green, e_1)$,
where $\mathit{SCALEVALPRED}(\mathit{COLOR}, Green, \phi)$.

In this example, the object may have any value immediately before the *BECOME* event as long as that value maps to a

Sense	*Final Axiomatization*
Change	$\forall\, o, Vp_1, Vp_2, t, e\, .\, (\textit{CHANGE}(o@t, Vp_1, Vp_2, e)$ $\equiv \exists\, t_1, t_2, sc.\ (t_1{:}t{:}t_2\ .\ \textit{SCALEPRED}(sc, Vp_1) \wedge \textit{SCALEPRED}(sc, Vp_2) \wedge \textit{TRUEOF}(o@t_1, Vp_1 \setminus Vp_2) \wedge \textit{TRUEOF}(o@t_2, Vp_2)))$
Become (with a unary predicate)	$\forall\, o, Vp, t, e\ \exists\, sc.\, (\textit{BECOME}(o@t, Vp_{sc}, e) \wedge \textit{SCALEPRED}(sc, Vp_{sc})$ $\equiv \exists\, t_1, t_2.\, (t_1{:}t{:}t_2 \wedge \textit{TrueOf}(o@t_2, Vp_{sc}) \wedge \textit{TrueOf2}(o@t_1, o@t_2, \neg \textit{ScaleEqual}(sc))))$
Become (with a binary relation)	$\forall\, o, P, t, e\ .\, (\textit{BECOME}(o@t, P, e) \wedge \textit{ARITY}(P, 2)$ $\equiv \exists\, t_1, t_2.\, (t_1{:}t{:}t_2 \wedge \textit{TRUEOF2}(o@t_1, o@t_2, P)))$
Different	$\forall\, o_1, o_2, t_1, t_2\ .\, (\textit{TRUEOF2}(o_1@t_1, o_2@t_2, \textit{Different})$ $\equiv \exists\, sc\ .\ \textit{TRUEOF2}(o_1@t_1, o_2@t_2, \neg \textit{ScaleEqual}(sc)))$

Table 2: The Final Definitions

scale value predicate on the same scale as the one *Green* belongs to (in this case the scale COLOR) and is different from *Green*. After the BECOME event the object is *Green*. In other words,

$$\textit{BECOME}(light_1@t, Green, e_1) \equiv \exists t_1, t_2.\ (t_1{:}t{:}t_2 \wedge \textit{TRUEOF}(light_1@t_2, Green_{COLOR}) \wedge \textit{TRUEOF2}(light_1@t_1, light_1@t_2, \neg \textit{ScaleEqual}(COLOR)))$$

This suggests the following formulation of BECOME:

$$\forall o,Vp,t,e\ \exists sc.\ (\textit{BECOME}(o@t, Vp_{sc}, e) \wedge \textit{SCALEPRED}(sc, Vp_{sc})) \equiv \exists t_1, t_2.\ (t_1{:}t{:}t_2 \wedge \textit{TRUEOF}(o@t_2, Vp_{sc}) \wedge \textit{TRUEOF2}(o@t_1, o@t_2, \neg \textit{ScaleEqual}(sc))))$$

So far, so good. But this formulation does not handle cases involving binary predicates, as in *become different.* To capture an intuitive sense of *different*, we appeal to the formulation of scales again. We say that two objects x and y are different if there is some scale on which the objects have different values. We do this because it seems incoherent to say o_1 is different from o_2 because o_1 weighs 5 pounds and o_2 is green. Rather, the two objects must have different properties on a common scale, i.e.,

$$\forall\, o_1, o_2, t_1, t_2.\ (\textit{TRUEOF2}(o_1@t_1, o_2@t_2, \textit{Different}) \equiv \exists\, sc.\ \textit{TRUEOF2}(o_1@t_1, o_2@t_2, \neg \textit{ScaleEqual}(sc)))$$

We introduce a new *formal predicate* that indicates the arity of scale predicates. *Different* has arity 2:

$$\textit{ARITY}(\textit{Different}, 2)$$

With this in hand we have an axiom for BECOME with binary predicates:

$$\forall o, P, t, e.\ (\textit{BECOME}(o@t, P, e) \wedge \textit{ARITY}(P, 2) \equiv \exists t_1, t_2.\ (t_1{:}t{:}t_2 \wedge \textit{TRUEOF2}(o@t_1, o@t_2, P)))$$

Note that this axiom would also apply to comparative adjectives, such as *become greener*, for all comparative predicates are binary predicates.

Automatically Deriving the Meaning of *Change*

Our final hand-built definitions are summarized in Table 2. We now have developed all the formalism needed to prove that the full meaning of *change* can be derived from its definition: *become different*. We have the following equivalence theorem.

Theorem 3 (CHANGE ≡ BECOME Different)

$$\forall\, o, t.\ (\exists\, Vp_1, Vp_2, e_1.\ \textit{CHANGE}(o@t, Vp_1, Vp_2, e_1) \equiv \exists\, e_2.\ \textit{BECOME}(o@t, \textit{Different}, e_2))$$

Thus, for every CHANGE event, there is a corresponding BECOME *Different* event, and vice versa.

In this case, we showed that we can derive the definition of CHANGE automatically from its natural language definition by composing BECOME and *Different.* However, this will not always be the case. The composite definition will often overlap the hand-crafted definition considerably, but will not be quite equivalent. Natural language definitions are typically underspecified and vague, emphasizing the necessary but not the sufficient conditions. Even though we may not always obtain equivalence, we have shown that this is a promising approach for learning key properties of the commonsense meanings of words by reading and composing natural language definitions automatically.

Conclusion

We have presented the first steps in developing a framework for formalizing commonsense knowledge, especially about verbs and adjectives. Unlike the work of Hobbs and colleagues, we have been strongly motivated by the need to have a formalism that parallels linguistic structures and thus allows commonsense knowledge to be derived automatically by reading definitions. This requirement entailed some strong constraints on the logical framework in which knowledge is expressed and led us to particular formalizations of certain underlying conceptualizations such as scales that are essential elements of the commonsense notions of change and being different. While we only addressed the formalization of a small number of concepts here, we already have laid a key piece of groundwork for automatically creating knowledge by reading—as the concepts of *change* and *become* are fundamental to a large number of verbs in English. With such a machinery, we endeavor to keep the number of concepts that need to be hand-axiomatized to a minimum, while learning the rest by composing definitions automatically.

Preliminary work on automatically building a knowledge base is described in (Allen *et al.,* 2013). We show there that we can derive knowledge capturing event hierarchies and causal relations. The work here provides the underlying temporal logic that will "activate" this knowledge, suggesting the potential for automatically building large scale commonsense knowledge bases that have broad coverage of word senses (e.g., all of WordNet).

Proofs of the theorems are included in the longer version of this paper, available online.

References

Allen, J. F. (1983) Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832-843.

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

Allen, J. F. and Ferguson, G. (1994) Actions and events in interval temporal logic. *Journal of Logic and Computation*, 4(5): 531-579.

Allen, J. F., 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*.

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E. R., Jr. and Mitchell, T. M. (2010) Toward an architecture for never-ending language learning. In *Proceedings of AAAI,* 1306-1313, AAAI Press.

Fellbaum, C. (1998, ed.) *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press.

Hobbs, J. R. (2013) http://www.isi.edu/~hobbs/csk.html

Hamblin, C. L. (1972) Instants and intervals. In J. T. Fraser, F. C. Haber, and G. H. Muller (ed), *The Study of Time*. Springer-Verlag, New York.

Lenat, D. B. (1995) Cyc: A large-scale investment in knowledge infrastructure. *The Communications of the ACM*, 38(11):33-38.

Niles, I. and Pease, A. (2001) Towards a standard upper ontology. In *Proc. 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001)*, Chris Welty and Barry Smith, eds, Ogunquit, Maine.

Yates, A., Banko, M., Broadhead, M., Cafarella, M., Etzioni, O. and Soderland, S. (2007) TextRunner: Open information extraction from the web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*.

Whitehead, A.N. (1929). *Process and Reality*, Macmillan, New York.