

# Million-scale Derivation of Semantic Relations from a Manually Constructed Predicate Taxonomy

Motoki Sano\* Kentaro Torisawa† Julien Kloetzer‡ Chikara Hashimoto §  
István Varga¶ Jong-Hoon Oh||

\*†‡§|| National Institute of Information and Communications Technology, Kyoto, 619-0289, Japan

¶ NEC Knowledge Discovery Research Laboratories, Kanagawa, 211-8666, Japan

{\*msano, †torisawa, ‡julien, §ch, ||rovellia}@nict.go.jp, ¶vistvan@az.jp.nec.com

## Abstract

We manually created a semantic taxonomy called *Phased Predicate Template Taxonomy (PPTT)* that covers 12,023 predicate templates (i.e., predicates with one argument slot like “*rescue X*”) and derived from it various semantic relations between these templates on a million-instance scale (70%-80% precision level). The derived relations include entailment (e.g., *rescue X*  $\supset$  *X is alive*), happens-before (e.g., *buy X*  $\Rightarrow$  *drink X*), and a novel relation type *anomalous obstruction* (e.g., *X is sold out*  $\rightsquigarrow$  *cannot buy X*). Such derivation became possible thanks to PPTT’s design and the use of statistical methods.

## 1 Introduction

Databases of various semantic relations between natural language expressions are indispensable knowledge for many NLP applications. For instance, entailment relations are crucial in information extraction and QA (Dagan et al., 2009; Weisman et al., 2012; Berant et al., 2012; Turney and Mohammad, 2014). Temporal relations such as happens-before (Chklovski and Pantel, 2004b; Regneri et al., 2010) are important for enhancing deep semantic processing. A problem, however, is that it is difficult to acquire those relations with a broad coverage. Although many sophisticated machine learning techniques have been applied to various kinds of corpora for this task (Szpektor et al., 2007; Chambers and Jurafsky, 2008; Hashimoto et al., 2009; Chambers and Jurafsky, 2009; Hashimoto et al., 2012; Talukdar et al., 2012; Kloetzer et al., 2013), no satisfactory coverage has been achieved, probably due to data sparseness in the input data. In this work we take a completely different approach: we manually construct a semantic lexicon called *Phased Predicate Template Taxonomy (PPTT)*, and derive various types of semantic relations on a large-scale by using it. Our target language is Japanese, but examples are given in English for simplicity throughout this paper.

PPTT is a taxonomy of predicate templates (predicates with one argument slot like *rescue X*, “Template” hereafter) that classifies templates according to *phases of story* concerning an entity denoted by *X*. In the story, or the “life” of the entity *X*, *X* can be *anticipated*, *created*, then *execute* its function and finally it may *collapse* and become *deficient*. Anticipation, creation, execution, collapse, deficiency of *X* can be seen as such *phases of story* concerning *X*, and PPTT classifies templates into 41 semantic classes each of which corresponds to a phase. In other words, PPTT provides a way to describe the stories of various entities that constitute this world, and we believe that PPTT (partly) reflects how we understand the world and its entities. Accordingly, PPTT can also provide a way to derive various semantic knowledge about this world such as the happens-before relation between events involving an entity, e.g., since the creation phase usually occurs before the execution phase, *invent X* (creation phase) is likely to happen-before *use X* (execution phase). In addition, entailment relations can be derived: since the creation phase of an object *X* *must* have occurred if *X* is in its execution phase, it implies that *use X* is likely to entail *invent X*.

In addition, there are ups and downs in stories; some entities suffer setbacks in their stories. PPTT describes such “ups and downs” by means of a recently proposed semantic polarity, *excitation* (Hashimoto

This work is licenced under a Creative Commons Attribution 4.0 International License. Page numbers and proceedings footer are added by the organizers. License details: <http://creativecommons.org/licenses/by/4.0/>

et al., 2012). Excitation classifies templates into *excitatory*, *inhibitory*, and *neutral*; an excitatory template like *install X* and *buy X* indicates that the main function, effect, purpose or role of the entity referred to by the *X* of the template is *activated*, *enhanced*, or *prepared*,<sup>1</sup> while an inhibitory template like *uninstall X* and *X is canceled* roughly indicates that it is *deactivated* or *suppressed*. Neutral templates are neither excitatory nor inhibitory (e.g., *consider X*). Roughly speaking, an excitatory template expresses the events that contribute to *turn on* the function of *X*, while an inhibitory template expresses the events that contribute to *turn off* or *not to turn on* the function of *X*. Then, in PPTT, excitatory and inhibitory respectively correspond to “ups” and “downs” in the story of *X*. The phases in PPTT are marked according to these ups and downs. Accordingly, PPTT can derive many antonymous contradiction pairs like *install X*  $\Leftrightarrow$  *uninstall X*, as Hashimoto et al. did, though we omit the detail for space limitation. Moreover, PPTT can derive a huge volume of *anomalous obstruction*, a contradiction-like novel semantic relation that we propose in this paper, like *X is canceled*  $\rightsquigarrow$  *(cannot) buy X* and *X is sold out*  $\rightsquigarrow$  *(cannot) buy X*, which indicate that if *X* is canceled or sold out, you cannot buy *X*. Anomalous obstruction should be used for Why-type QA (Oh et al., 2013), as well as a novel system that warns a user who wants to buy a commercial product that the product is started to be *sold out* or *canceled* in various e-commerce sites without any application-specific coding.

As suggested, a story has a temporal order between its phases, which we call the *canonical temporal order*. In addition, some phases in a story would *enable* or *necessitate* another phase in the same story to occur. In PPTT, these relations are embodied in various temporal-semantic links between phases. Note that each link between two phases does not *guarantee* that every possible pair of templates taken from the two phases has such semantic relations; it just indicates that there exists such *tendencies*. Despite the absence of the guarantee, PPTT’s links enable a million-scale derivation of semantic relations with the help of distributional similarity. In existing resources such as WordNet (Fellbaum, 1998), the links are assumed to be 100% correct, but it would be hard to have such absolutely correct links in a million-scale. Hence, we believe that our *approximate* links are more useful for a large-scale relation derivation.

Note that the goal of our PPTT project is to derive a wide range of semantic relations on a large scale, rather than to complete a comprehensive template taxonomy. As such, PPTT lacks some templates as described in later sections. Nevertheless, we believe that our design brings much more good than harm, since we could generate various semantic relations on a million scale thanks to PPTT. Our experimental results show that we can derive 4.4 million happens-before relation instances with 79.5% precision, 0.5 million entailment relation instances with 70.0% precision, and one million anomalous obstruction relation instances with 73.5% precision. Constructing the PPTT taxonomy requires a manual labor cost, which amounted to three man-months in our case; however, we believe that this cost is *lower* than the cost for developing highly-precise automatic acquisition methods for all of happens-before, entailment, contradiction, and anomalous obstruction relations.

We plan to release PPTT and the derived relation instances after the manual annotation of the derived instances to the NLP community.

## 2 Related Works

PPTT might resemble other semantic lexicons created in the long history of NLP (Levin, 1993; Kipper et al., 2006; Fellbaum, 1998; Bond et al., 2009; Fillmore, 1976; Baker et al., 1998; Halliday, 1985; Pustejovsky et al., 2003; Pustcasu and Mititelu, 2008; Bejar et al., 1991; Jurgens et al., 2012). PPTT is different in that it primarily aims at deriving various types of semantic relations on a large scale exploiting the notion of the *phase of story*, rather than being a comprehensive taxonomy like those existing semantic lexicons. As a result, PPTT can derive more varieties of semantic relations between templates than any one of those existing lexicons. From **WordNet** (Fellbaum, 1998; Bond et al., 2009), we can derive entailment and contradiction relations using synsets and synset-links that represent relations such as ‘troponym’, ‘antonym’ and ‘entailment’. However, happens-before and anomalous obstruction relations

<sup>1</sup>The above definition is slightly different from the original one in Hashimoto et al. (2012). We inserted the verb “prepared” into the original definition. This clarifies that various preparation processes for *X*, such as *buy X*, can be regarded as excitatory templates. We also assume that such templates as *X exists* and *have X*, which mean little more than just existence, are regarded as excitatory templates in PPTT based on the assumption that *existence* can be regarded as *preparation* for the function of *X*.

cannot be derived from it, since there is no information on temporal ordering except that on causality. From **VerbNet** (Levin, 1993; Kipper et al., 2006), the hyponymy/synonymy type of entailment relations may be derived using templates in the same verb classes constructed based on shared syntactic behavior, possibly with the help of statistical methods. However, the other types of relations that can be derived from PPTT cannot be derived from VerbNet, since there is no link representing relationships between the verb classes. **FrameNet** (Fillmore, 1976; Baker et al., 1998) was used to derive hyponymy/synonymy types of entailment (Coyne and Rambow, 2009; Aharon et al., 2010) using information such as a Frame-to-frame relation ‘Inheritance’ (*is-a* relation). In addition, happens-before relations can be derived using ‘Precedes’ (*Later-Earlier* relations). However, since it does not contain semantic constraints like *enablement* and *necessity* that PPTT contains, it is not trivial to derive presupposition type of entailment or anomalous obstruction instances from it. **TimeML** (Pustejovsky et al., 2003; Puscasu and Mititelu, 2008) contains various temporal information and can be used to derive *context-dependent* happens-before relations such as the relation between “leaves” and “will not hear” in the sentence “If Graham leaves today, he will not hear Sabine” through TLINK (Pustejovsky et al., 2003) annotated manually; thus, it is difficult to derive *context-independent* relations from it, while they can be derived from PPTT. Besides, since it covers only temporal information, it is difficult to derive other types of relations from it. From **Bejar et al.’s semantic relation taxonomy of lexical pairs** (Bejar et al., 1991; Jurgens et al., 2012), using semantic relation categories such as “act: act attribute” (e.g., creep:slow), lexical entailment relations were extracted (Turney and Mohammad, 2014). However, it is not trivial to derive happens-before or anomalous obstruction relations from it since it does not contain information on temporal sequences between verbs.

Furthermore, our work differs from automatic methods for extracting temporal or causal relations (Szpektor et al., 2007; Chambers and Jurafsky, 2008; Chambers and Jurafsky, 2009; Talukdar et al., 2012; Hashimoto et al., 2012; Hashimoto et al., 2014) in that our method does not require that target pairs co-occur in a document, unlike the previous methods. Hence, our method is likely to be immune to data sparseness. We could actually derive a wide range of relation instances that were rarely written in documents because they were too commonsensical (e.g., *X is constructed* happens-before *sew (something) at X*). Needless to say, such commonsensical knowledge is often needed to develop intelligent systems.

### 3 PPTT Design

In PPTT, templates are organized hierarchically into three levels. In each level, there are classes that correspond to phases of stories, which we call Level-0 (L0), Level-1 (L1), and Level-2 (L2) classes. Each template belongs to only one class at each level. In the following, we describe each level.

#### 3.1 L0-Classes and L0-Links

First we divided the *entire story* concerning an entity *X* into five phases: *non-existence*, *existence*, *functioning*, *non-existence to existence transition* and *existence to non-existence transition*. Then we created the five L0-classes listed below, each of which corresponds to one of these five phases.

**Non-existence Class** The class of templates that do not entail the existence of *X*, e.g., *plan X*.<sup>2</sup>

**Existence Class** The class of templates that entail *X*’s existence but does not imply the execution of its main function or the achievement of its objectives, e.g., *buy X*, *X exists*.

**Functioning Class** The class of templates that imply the execution of *X*’s main function or the achievement of its objectives, e.g., *use X*, *eat X*.

**Non-existence to Existence Transition Class (NET Class)** The class of templates that express the transition from a situation in which *X* does not exist to a situation in which it exists, e.g., *manufacture X*.

<sup>2</sup>One might think the definition of the Non-existence Class should be “the templates that DO entail *X*’s NON-EXISTENCE”. We did not use such a definition because it would overlook many templates that are consistent with *X*’s NON-EXISTENCE but DO NOT entail *X*’s NON-EXISTENCE, like *order X*.

**Existence to Non-existence Transition Class (ENT Class)** The class of templates that express the transition from a situation in which  $X$  exists to a situation in which it does not exist, e.g., *dismantle X*.

As mentioned in the introduction, we assume a canonical temporal order among L0-classes. For instance, templates in the **NET class** (e.g., *manufacture X*) should refer to events that usually happen before those events referred to by templates in the **Existence class** (e.g., *buy X*), **Functioning class** (e.g., *use X*) and **ENT class** (e.g., *dismantle X*). We enumerated such temporal restrictions, each of which is represented by a link in Figure 1, which we call *L0-links* and used them for deriving relations. Note that we did not set any L0-link between the **Existence class** and the **Functioning class** because the events described by them may happen in various orders or have temporal overlap. For example,  $X$  exists should have temporal overlap with *use X*.

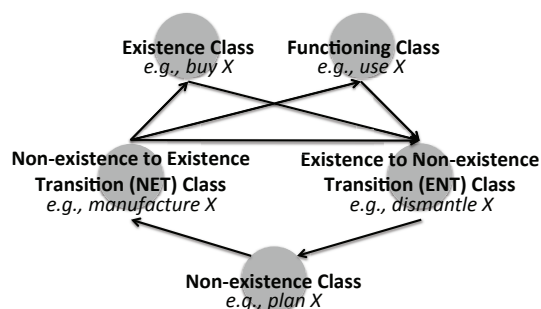


Figure 1: L0-links among L0-classes.

Of course, such *metaphysical* notions as the canonical temporal order and the phases must have many complications and exceptions. First, many templates that have the neutral excitation polarity (Hashimoto et al., 2012) did not seem to follow the canonical temporal order among L0-classes. For instance, since the neutral template *think about X* does not entail the existence of  $X$ , it belongs to the **Non-existence class** but one can consider  $X$  while  $X$  exists or while it is functioning or even after it is collapsed and violate canonical temporal ordering. For this reason, we excluded neutral templates from PPTT and will deal with them in a different framework as a future work. In addition, although we did not assume a temporal order between the **Existence class** and the **Functioning class**, some templates in these classes have a happens-before relation as special cases (e.g., *buy X* in the **Existence class** happens before *eat X* in the **Functioning class**). The proposed L0-links also cause problems. For instance, *order X* (**Non-Existence class**) may not always happen before *create X* (**NET class**) even though the L0-links indicate a happens-before relation between their classes. We dealt as far as possible with such cases in level 2 with L2-classes, which are finer than L0-classes. Nonetheless, we stress that the overall plausibility of the canonical temporal order among L0-classes was experimentally confirmed through the derivation of happens-before relations only using L0-links. Note that the design of the L0-classes was inspired by the Generative Lexicon (Pustejovsky, 1998) and Aristotle’s *Entelecheia* (Aristotle, 1987).

### 3.2 L1-Classes

Next, we divided some L0-classes into L1-classes using the excitation polarity (Hashimoto et al., 2012) to introduce “ups and downs” to PPTT, which enables to capture semantic inconsistencies between templates (e.g., *install X*  $\leftrightarrow$  *uninstall X*) and negative interaction between the events referred to by the templates in PPTT (e.g., *X is canceled*  $\rightarrow$  (*cannot*) *hold X*). Excitation was originally proposed for recognizing contradictions and causal relations between templates and then was successfully applied to other deep semantic processing (Oh et al., 2013; Varga et al., 2013; Kloetzer et al., 2013; Hashimoto et al., 2014).

L0-class	Excitation	
	Excitatory	Inhibitory
<b>Non-existence class</b>	POTENTIAL class e.g., <i>plan X</i>	FORECLOSING class e.g., <i>prevent X</i>
<b>Existence class</b>	ENABLING class e.g., <i>buy X</i>	INCOMMODE class e.g., <i>weaken X</i>
<b>Functioning class</b>	ACTUALIZING class e.g., <i>X functions</i>	DISORDERING class e.g., <i>X loses</i>
<b>NET class</b>	GENERATING class e.g., <i>X is born</i>	N/A
<b>ENT class</b>	N/A	CORRUPTING class e.g., <i>destroy X</i>

Table 1: L1-classes.

As shown in Table 1, we divided each of three L0-classes (**Non-existence class**, **Existence class** and **Functioning class**) into two L1-classes, each of which corresponds to excitatory and inhibitory. Since the transition to an existence situation can be interpreted as an enhancement of an entity’s function, we assumed that all the templates in the NET classes are excitatory because they express a transition of entity  $X$  from a non-existence situation to an existence situation. Similarly, we assume all the templates from the ENT class are inhibitory. Also, L1-classes do not have specific links between them beside the L0-links from their parent classes.

### 3.3 L2-Classes and L2-Links

Finally, we divided L1-classes into 41 L2-classes. Specifically, we first roughly grouped together semantically similar templates from the same L1-class and identified the common semantic properties among them. Note that in the rough grouping, we classified templates so that the resulting groups fit into fine-grained phases in the story concerning  $X$ .

After this initial grouping, we classified all the templates into the L2-classes that are listed in Table 3 alongside the classification criteria and the number of templates in each class. As the classification criteria, we used the identified common semantic properties among members of each class. Note that some L2-classes can be regarded as a subset of another L2-class. For instance, the PROHIBIT L2-class can be seen as a subset of the PREVENTION L2-class. When a template meets the classification criteria of both a subset class and its superset class, we classified it into the subset class.

We also made links called L2-links between the L2-classes. The motivation behind this is to capture finer temporal-semantic constraints that could not be specified at Level-0 and Level-1 as well as to capture the temporal-semantic constraints inside a single L0 or L1-class. For example, the temporal order between *buy X* and *eat X* is encoded in a L2-link between the ACQUISITION and EXECUTION L2-classes, while there is no L0-link between the Existence L0-class (class of *buy X*) and the Functioning L0-class (class of *eat X*). This exemplifies that the L2- and L0-links complement each other.

Each L2-link has one of the six types of temporal-semantic links that are summarized in Table 2 with the number of links of each type. The link types were designed to capture how the events referred to by the templates in a class *affect* the *occurrence or non-occurrence* of the events referred to by the templates in a class in the *past, present, or future*.  $C_1$  and  $C_2$  being two L2-classes,  $C_1$ ’s effect on the occurrence or non-occurrence of  $C_2$  is represented by Positive (+) and Negative (−) links, respectively, while  $C_1$ ’s effect on the past, present, or future phase of  $X$  expressed by  $C_2$  is represented by Past, Present, and Future links, respectively. For instance, the Past<sup>+</sup> link from the ABANDONMENT class to the ACQUISITION class indicates that a template from the ACQUISITION class (e.g., *obtain X*) *must occur* before a template from the ABANDONMENT class (e.g., *get rid of X*), and the Future<sup>−</sup> link from the PROHIBIT class to the EXECUTION class indicates that templates from the PROHIBIT class (e.g., *ban X*) *disable* templates from the EXECUTION class (e.g., *utilize X*). Notice that L2-links represent such semantic constraints as *enablement* and *necessity* in addition to temporal order, and they are useful for deriving various kinds of semantic relations including entailment and anomalous obstruction, as shown in a later section. The first author of this paper hand-labeled the links between every combination of L2-class pairs by considering the name of the classes and a few example templates in each.

	Positive	Negative
<b>Past</b>	If $C_1$ occurred, $C_2$ must have occurred. e.g., FORGETTING $\xrightarrow{Past^+}$ RECOGNITION; $X$ is forgotten $\xrightarrow{Past^+}$ $X$ is recognized (55 links)	If $C_1$ occurred, $C_2$ COULD NOT have occurred. e.g., CREATION $\xrightarrow{Past^-}$ PREVENTION; $X$ is generated $\xrightarrow{Past^-}$ $X$ is prevented (438 links)
<b>Present</b>	While $C_1$ is taking place, $C_2$ must be taking place. e.g., INITIATION $\xrightarrow{Present^+}$ BEING; $X$ is started $\xrightarrow{Present^+}$ $X$ exists (73 links)	While $C_1$ is taking place, $C_2$ CANNOT take place. e.g., ENHANCEMENT $\xrightarrow{Present^-}$ DEGRADATION; $X$ is enhanced $\xrightarrow{Present^-}$ $X$ is deteriorated (496 links)
<b>Future</b>	$C_1$ enables $C_2$ to occur. e.g., PREPARATION $\xrightarrow{Future^+}$ EXECUTION; $X$ is customized $\xrightarrow{Future^+}$ $X$ is executed (90 links)	$C_1$ DISABLES $C_2$ to occur. e.g., DEFICIENCY $\xrightarrow{Future^-}$ PROVISION; $X$ does not exist $\xrightarrow{Future^-}$ $X$ is provided (210 links)

Table 2: Types and numbers of L2-links in PPTT. Link direction is  $C_1 \rightarrow C_2$ .

<b>Non-existence L0-class: Potential L1-class (578) / Foreclosing L1-class (178)</b>	
DESIRE	entails that <i>X</i> is desired but unlike PLANNING or DEMAND, it does not entail that <i>X</i> is planned or requested, e.g., <i>desire X, want X</i> (48).
PLANNING	entails that <i>X</i> is planned but does not entail that <i>X</i> is requested. Unlike DEMAND, it does not assume that a person other than the Planner will carry out <i>X</i> , e.g., <i>plan X, conspire X</i> (72).
DEMAND	entails that <i>X</i> is requested. Unlike PLANNING, it assumes that a person other than the Demander will carry out <i>X</i> , e.g., <i>order X</i> (252).
APPROVAL	entails that <i>X</i> is approved or permitted and that there was a plan or a demand before approving, e.g., <i>permit X, accept X</i> (80).
FEAR	entails that <i>X</i> is expected and that <i>X</i> is a source of anxiety or fear, e.g., <i>fear X, worry about X</i> (13).
ANTICIPATION	entails that <i>X</i> is expected but unlike FEAR, does not entail that <i>X</i> is a source of anxiety or fear, e.g., <i>forecast X, predict X</i> (24).
SEARCH	entails that <i>X</i> is searched for but unlike DESIRE or DEMAND, does not entail that <i>X</i> is desired or requested, e.g., <i>search for X</i> (89).
PREVENTION	entails that <i>X</i> is prevented. Unlike CANCELATION, it does not entail that there was a plan or a demand before preventing, e.g., <i>preclude X</i> (54).
CANCELATION	entails that <i>X</i> is canceled and that there was a plan or demand before canceling, e.g., <i>cancel X, give up X</i> (34).
PROHIBIT	entails that <i>X</i> is prohibited. <i>X</i> 's right or ability to be generated or used is taken away, e.g., <i>ban X, forbid X</i> (39).
POSTPONE	entails that <i>X</i> is postponed, e.g., <i>postpone X, defer X</i> (15).
DEFICIENCY	entails that <i>X</i> does not exist but does not entail that it is prevented, canceled, prohibited, or postponed, as in other L2-classes of Foreclosing L1-class. e.g., <i>lack X, X is absent</i> (36).
<b>NET L0-class: Generating L1-class (596)</b>	
SYMBOLIZATION	entails that <i>X</i> transits from non-existence to existence as a kind of (semiotic) representation, e.g., <i>write X, compose (music) X</i> (13).
CREATION	entails that <i>X</i> transits from non-existence to existence. Unlike SYMBOLIZATION, <i>X</i> is not limited to a semiotic representation, and unlike TRANSFORMATION, it focuses less on transformation from another entity. <i>generate X, cause X</i> (509).
TRANSFORMATION	entails that <i>X</i> transits from non-existence to existence as a result of transformation. Unlike CREATION, it focuses on the transformation from another entity, e.g., <i>turn into X</i> (74).
<b>ENT L0-class: Corrupting L1-class (622)</b>	
COLLAPSE	entails that <i>X</i> transits from existence to non-existence by dying, being eliminated, or being destroyed. Unlike CONVERSION, it focuses less on transformation, e.g., <i>destroy X, kill X</i> (588).
CONVERSION	entails that <i>X</i> transits from existence to non-existence by transforming <i>X</i> into an another entity, e.g., <i>turned from X, changed from X</i> (34).
<b>Existence L0-class: Enabling L1-class (3,536) / Incommodate L1-class (1,355)</b>	
RECOGNITION	entails that <i>X</i> is recognized or sensed, e.g., <i>find X, feel X</i> (308).
SELECTION	entails that <i>X</i> is selected, e.g., <i>appoint X, choose X</i> (139).
ENCOUNTER	entails that <i>X</i> emerges as a result of transportation, e.g., <i>send X, X arrives</i> (407).
ACQUISITION	entails that <i>X</i> is obtained and possessed, e.g., <i>buy X, catch X</i> (482).
PROVISION	entails that <i>X</i> is handed to be possessed, e.g., <i>sell X, render X</i> (422).
ENHANCEMENT	entails that <i>X</i> is extended, improved, or supported, e.g., <i>increase X, help X</i> (880).
PREPARATION	entails that <i>X</i> is arranged, connected, or qualified in preparation to execute its function, e.g., <i>cook X, install X</i> (822).
BEING	entails that <i>X</i> is existing or living but does not entail that <i>X</i> is recognized, selected, encountered, acquired, enhanced, or prepared, as in other L2-classes of the Enabling L1-class, e.g., <i>X exists, X lives</i> (76).
UNRECOGNIZING	entails that <i>X</i> is not recognized or sensed but unlike FORGETTING, does not entail that <i>X</i> was previously recognized, e.g., <i>overlook X</i> (8).
FORGETTING	entails that <i>X</i> is forgotten and that <i>X</i> was once recognized, e.g., <i>forget X, lose memory of X</i> (8).
UNSELECTING	entails that <i>X</i> is not selected, e.g., <i>alternate X, reject X</i> (46).
SEPARATION	entails that <i>X</i> is left or separated as a result of transportation, e.g., <i>X leaves, send X away</i> (114).
ABANDONMENT	entails that <i>X</i> is not possessed as a result of being thrown away, e.g., <i>throw X away, renounce X</i> (58).
DEPRIVATION	entails that <i>X</i> was taken away without the permission of a possessor, e.g., <i>steal X, take X away</i> (102).
DEGRADATION	entails that <i>X</i> is reduced, deteriorated, or interrupted, e.g., <i>X is weakened, attack X</i> (908).
UNPREPARED	entails that <i>X</i> is unprepared, disconnected, or unqualified, e.g., <i>X is uninstalled, X is disconnected</i> (111).
<b>Functioning L0-class: Actualizing L1-class (4,460) / Disordering L1-class (698)</b>	
EXECUTION	entails that the function of <i>X</i> is executed but unlike WORKING, does not entail that <i>X</i> successfully satisfies its function, e.g., <i>ignite X</i> (966).
WORKING	entails that the function of <i>X</i> is carried out and that <i>X</i> successfully satisfies its function, e.g., <i>X functions, cleaned by X</i> (3,106).
INITIATION	entails that <i>X</i> is started or continued, e.g., <i>start X, open X</i> (185).
SUCCESS	entails that <i>X</i> accomplished its goal and the result of the execution of its function is evaluated positively, e.g., <i>accomplish X, X wins</i> (203).
SUSPENSION	entails that the function of <i>X</i> is suspended but unlike FINISHING, does not entail that its function is terminated, e.g., <i>suspend X</i> (133).
DYSFUNCTION	entails that the function of <i>X</i> is executed but <i>X</i> is performing poorly, e.g., <i>X is sluggish, bored by X</i> (196).
FINISHING	entails that <i>X</i> is terminated, e.g., <i>end X, finish X</i> . (110).
FAILURE	entails that <i>X</i> fails to accomplish its goal and the result of the execution of its function is evaluated negatively, e.g., <i>X is defeated</i> (259).

Table 3: PPTT classes. The number in parentheses indicates the number of templates in PPTT.

Note that the existence of an L2-link does not guarantee that the semantic properties specified by it hold for all the possible template pairs taken from the class pair it connects. The cost of hand-labelling the links with such guarantees is prohibitively high because we would have to check all of the template combinations. We empirically evaluated the validity of the links in our experiments below although this is not a *direct* evaluation since the relations we derived are different from the ones given to the links.

## 4 Construction of PPTT and Relation Derivation

Using the automatic acquisition method proposed by Hashimoto et al. (2012), we collected 10,825 candidates of excitatory/inhibitory templates from a 600-million-page web corpus (hereafter, *WCorpus*). Hashimoto et al.’s method constructs a network of templates based on their co-occurrence in sentences with a small number of seed templates of which excitation polarity are assigned manually, and infers the polarity of all the templates in the network by a constraint solver based on the spin model (Takamura et al., 2005). Then, we added the 20,000 most frequent templates in the corpus that could not be extracted automatically for a total of 30,825 templates.

Three human annotators (not the authors) judged the polarity of the templates, and we included the excitatory and the inhibitory templates but excluded the neutral templates in PPTT due to the reason discussed in Section 3.1. We also excluded templates whose variable  $X$  is the subject of a transitive verb. This is because the subject position is often occupied by living things, and since the *functions/objectives* of such subjects seem difficult to identify, it is often difficult to judge whether such templates should be classified into the **Functioning** class or another. After applying these two restrictions, the first author classified the remaining 12,023 templates in PPTT.

In this work, we derived happens-before, entailment and anomalous obstruction relations among templates from PPTT. The target data is the set of all the template pairs such that a noun exists with which both templates of the pair co-occur at least 100 times in *WCorpus*. We denote this set of the template pairs by  $TP100$ , and all the relation derivations pick up template pairs as relation instances from it. This is because in our preliminary experiments, we found that the relation instance candidates taken from outside of  $TP100$  had much lower precision. The relation derivation itself is quite simple and consists of the following two steps.

**Step 1** Select L0-links or types of L2-links that are expected to represent a target semantic relation (e.g.,  $Present^+$  links are expected to represent entailment, since they represent the relations between classes where “While  $C_1$  is taking place,  $C_2$  must be taking place”.) and extract all the class pairs connected by the selected links (e.g., INITIATION L2-class  $Present^+ \rightarrow$  BEING L2-class). Enumerate all the template pairs from the intersection between  $TP100$  and the extracted class pairs (e.g.,  $X$  is started  $Present^+ \rightarrow$   $X$  exists).

**Step 2** If necessary, rank the relation instance candidates that are extracted in Step1 by distributional similarity scores between the templates that compose the candidates, computed with *WCorpus*.

## 5 Experiments

This section reports our experiments on semantic relation derivation. Derived relation instances were marked by three human annotators (not the authors) who voted to break ties. Unless stated otherwise, we asked them to mark a template pair as negative if they found any noun that can be placed in both templates’ argument slots and makes the template pair a negative sample for the target relation, and positive otherwise.

### 5.1 Happens-Before Relation

Following Regneri et al. (2010), we assumed template<sub>1</sub> ( $T_1$ ) has a *happens-before relation* with template<sub>2</sub> ( $T_2$ ) iff one event expressed by  $T_1$  normally happens before another expressed by  $T_2$ , provided that both events occur. Below are our four methods to derive happens-before relation instances, each of which uses different links. Note that we did not use distributional similarity in this experiment.

- H1** uses the 55 pairs of L2-classes connected by L2-link  $Past^+$ , meaning that a template in a class *must occur before* another.
- H2** uses the 90 pairs of L2-classes connected by L2-link  $Future^+$ , i.e., a template in a class often *enables* another to occur.
- H3** uses the 474 pairs of L2-classes connected by one of the seven L0-links in Figure 1, i.e., the canonical temporal order links.
- All** is the union of **H1-H3** results.

We prepared two baselines; **HB-Ptn** is a pattern-based method based on Chklovski and Pantel (2004a). It extracts template pairs in *TP100* that were connected in *WCorpus* by one of manually collected 73 conjunctives expressing temporal order, such as *after* and *before*, and which either shared the same argument or the second template was filled by the pronouns *it*, *this*, or *that*. **Random** is a random sampling from *TP100*.

Three annotators annotated 200 random samples from each method’s output. Fleiss’ kappa was .56 (moderate agreement). The results of their majority vote are summarized in Table 4. The recall was estimated against the number of positive samples in *TP100* based on the precision of **Random**. The precision of all of our four methods is reasonably high for such a difficult task, and the number of relations derived by **All** reached about 4.4 million. The recall of **All** exceeds 65%, which we believe is quite high. **HB-Ptn** suffered from low recall, probably due to the data sparseness in *WCorpus*. Table 5 shows examples of the derived happens-before relations alongside L2-classes of the templates, the L2-links between the classes and the original Japanese templates. The acquired relations included many unexpected but correct happens-before relations, like *compose (a piece of music) X  $\supset$  relax by X*.

Actually, it is difficult to fairly compare our work and previous works on temporal relation acquisition, due to differences in language, the data used, and the methodologies. Nonetheless, our result with 79.5% precision is at least five times larger than the English data released by Chambers et al. ([cs.stanford.edu/people/nc/schemas](http://cs.stanford.edu/people/nc/schemas)), which contains around 870,000 “before” relation candidates and happens-before database in the VerbOcean (Chklovski and Pantel, 2004a) that covers 4,205 relations. Considering our method is completely different from theirs, we believe that our contribution is valuable.

Setting/Method	Precision (%)	# of Pairs	Recall (%)
<b>H1</b>	<b>83.5</b>	1,113,280	18.0
<b>H2</b>	70.5	1,524,557	20.8
<b>H3</b>	67.0	3,837,116	49.7
<b>All</b>	79.5	4,387,781	<b>67.5</b>
<b>HB-Ptn</b>	53.0	32,288	0.3
<b>Random</b>	18.0	28,717,454	100.0

Table 4: Happens-before derivation performance.

<i>boil X <math>\Rightarrow</math> eat X</i>	
PREPARATION Class	$\xrightarrow{Future^+}$ EXECUTION Class
X wo niru $\Rightarrow$ X wo taberu	
<i>compose (a piece of music) X <math>\Rightarrow</math> relax by X</i>	
SYMBOLIZATION Class	$\xleftarrow{Past^+}$ WORKING Class
X wo sakkyoku-suru $\Rightarrow$ X de rirakkusu-suru	

Table 5: Examples of happens-before relation.

## 5.2 Entailment Relation

Below are our proposed methods to derive entailment relations.

**Present+.DIFF** extracts the 32 class pairs that are composed of *DIFFERENT* L2-classes and are connected by the  $Present^+$  links, meaning that a template in a class *must occur simultaneously with* another template in another class, and ranks all the possible template pairs taken from each class pair using Hashimoto et al.’s (2009) conditional probability based similarity measure for entailment recognition.

**Present+.SAME** extracts the 41 class pairs that are composed of the *SAME* L2-classes and are connected with the  $Present^+$  links, and ranks all the template pairs from each class pair using Hashimoto et al.’s similarity.

**Past+** extracts the 55 pairs of L2-classes that are connected with the  $Past^+$  links, meaning that a template in a class *must occur before* another, and ranks all the template pairs from each class pair using Hashimoto et al.’s similarity.



**Baseline-HAS** is our baseline which is our implementation of Hashimoto et al. (2009) for entailment recognition; it ranks all the template pairs in *TP100* by Hashimoto et al.’s score. Our methods can be seen as the restrictions of the output of the baseline method using the extracted PPTT’s class pairs.

Three annotators hand-labeled 500 random samples from the top 100,000 template pairs for each method. The kappa was .59 (moderate agreement), and the results of their majority vote are presented in Figure 2. Table 6 shows examples of Proposed methods’ outputs. The restriction of the class pairs in our method contributed to much higher precision than using the state-of-the-art method alone.

Since the precision of **Past+** is quite high for the top 100,000 pairs, we annotated an additional 500 random samples from the top 500,000 pairs. According to this annotation, the top 408,610 pairs had 70% precision, implying that after merging all the top pairs extracted

by **Present+.DIFF**, **Present+.SAME** and **Past+** whose precisions exceeded 70%, we had 0.49 million entailment pairs with 70% precision. With **Baseline-HAS**, we derived only 24,000 with the same precision. Also, the Japanese WordNet (v.1.1) covers only 2.4% of the pairs in the manually annotated positive samples from our proposed methods through the ‘synsets’ or any ‘synlinks’. We analyzed 200 samples from the positive samples not covered by WordNet and found that 49.5% are the hyponymy type (e.g., *boil X*  $\supset$  *heat X*), 39.0% are the backward presupposition type (e.g., *complete X*  $\supset$  *start X*), and 11.5% are the synonymy type (e.g., *X passes away*  $\supset$  *X dies*). This seems to imply that our methods are better at deriving all types of entailment, while WordNet might be effective for only the synonymy type. In addition, by analyzing all the positive samples, we confirmed that the different types of entailment pairs were derived with different L2-links; 88.1% of the positive samples from *Present+.DIFF* and *Present+.SAME* require that two events referred to by the two templates occur with temporal overlap (e.g., *equip X*  $\supset$  *X exists*, i.e. *X* is equipped while *X* exists), while 96.7% of those from *Past+* were the backward presupposition type, in which an event entails another event that happened before it. This shows that the L2-links were useful for deriving various fine-grained types of entailment.

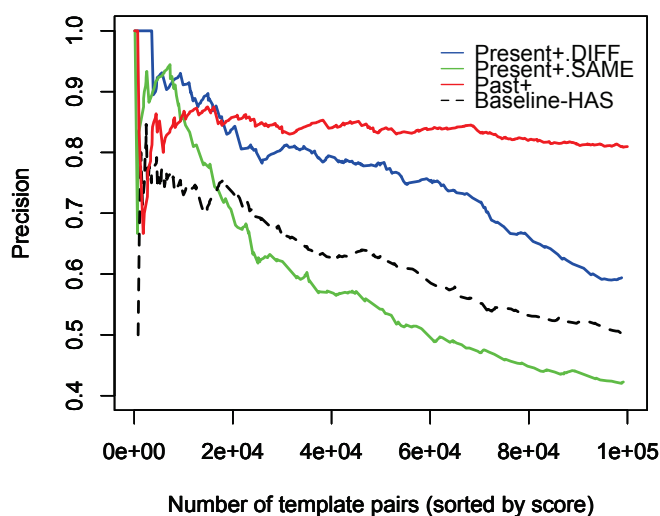


Figure 2: Entailment derivation performance.

<i>get X</i> $\supset$ <i>X exists</i> ( <i>X wo nyuushu-suru</i> $\supset$ <i>X ga sonzai-suru</i> ) ACQUISITION Class $\xrightarrow{Present^+}$ BEING Class
<i>evolve into X</i> $\supset$ <i>change into X</i> ( <i>X ni shinka-suru</i> $\supset$ <i>X ni kawaru</i> ) TRANSFORMATION Class $\xrightarrow{Present^+}$ TRANSFORMATION Class
<i>close (a shop) X</i> $\supset$ <i>make X</i> ( <i>X wo heiten-suru</i> $\supset$ <i>X wo tsukuru</i> ) FINISHING Class $\xrightarrow{Past^+}$ CREATION Class

Table 6: Examples of entailment.

### 5.3 Anomalous Obstruction Relation

We assumed that template<sub>1</sub> (*T*<sub>1</sub>) like *X is sold out* has an *anomalous obstruction relation* with template<sub>2</sub> (*T*<sub>2</sub>) like *buy X* (denoted as *X is sold out*  $\leadsto$  (cannot) *buy X*) iff: (A) the event expressed by *T*<sub>1</sub> prevents the event expressed by *T*<sub>2</sub> from occurring; (B) *T*<sub>1</sub> expresses an event that should not happen if everything about the variable *X* goes as expected; and (C) *T*<sub>2</sub> expresses another event in which the function of *X* is executed, enhanced, or prepared. We derived anomalous obstructions, by generating all of the possible template pairs from the 88 L2-class pairs connected by Future<sup>-</sup> L2-links. These indicate that the events expressed by the templates in the first class of a pair *disable* the events expressed by the templates in the second class. Also, to confirm that the templates of the first class in a pair express an unexpected event,

we required the *disabler* class to have the inhibitory polarity and the *disabled* class to be excitatory. Otherwise, we would obtain such pairs as INITIATION $\rightsquigarrow$ PLANNING (e.g., *start X $\rightsquigarrow$ schedule X*), which indeed express the *prevention* relation (Barker and Szpakowicz, 1995), i.e., “*scheduling X would not occur after starting X*,” which is different from anomalous obstruction.

Three annotators annotated 200 random samples for each method, and the results of their majority vote are summarized in Table 7, where **Random** refers to a random baseline using *TP100*. The recall was estimated using the number of positive samples provided by **Random**. The kappa was .60 (moderate agreement). 73.5% precision, 26.4% recall against the positive samples in *TP100*, and more than one million outputs of our proposed method are reasonably high/large results for this difficult task. Table 8 shows examples of **Proposed**’s outputs. “(cannot)” was attached to *disabled* templates for readability.

Setting/Method	Precision	# of Pairs	Recall
<b>Proposed</b>	<b>73.5</b>	1,081,405	26.4
<b>Random</b>	10.5	28,717,454	100.0

Table 7: Performance of anomalous obstruction derivation.

<i>prohibit X<math>\rightsquigarrow</math>(cannot) exhibit X</i>	PROHIBIT Class	$\xrightarrow{Future^-}$	EXECUTION Class
<i>X wo kinshi-suru<math>\rightsquigarrow</math>X wo kookai-suru</i>			
<i>break X<math>\rightsquigarrow</math>(cannot) utilize X</i>	COLLAPSE CLASS	$\xrightarrow{Future^-}$	EXECUTION CLASS
<i>X wo kowasu<math>\rightsquigarrow</math>X wo riyo-suru</i>			

Table 8: Examples of anomalous obstruction.

## 6 Conclusion

In this work, we manually constructed a Phased Predicate Template Taxonomy (PPTT), which is a network of semantically coherent classes of templates and derived semantic relations including entailment from it in a million-instance scale. Future work will extend PPTT to cover non-excitatory/non-inhibitory templates and generate richer structural knowledge similar to *full-fledged* scripts (Schank and Abelson, 1977) and narrative schemas (Chambers and Jurafsky, 2011).

## Acknowledgements

We would like to thank three anonymous reviewers for many useful comments and advices on the manuscript of this paper.

## References

- Roni Ben Aharon, Idan Szpektor, and Ido Dagan. 2010. Generating entailment rules from framenet. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort ’10, pages 241–246, Stroudsburg, PA, USA. ACL.
- Aristotle. 1987. *De Anima (Translated by Hugh Lawson-Tancred)*. Penguin Classics, London.
- Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL ’98, pages 86–90, Stroudsburg, PA, USA. ACL.
- Ken Barker and Stan Szpakowicz. 1995. Interactive semantic analysis of clause level relationships. In *Proceedings of PACLING ’95*, Brisbane.
- I.I. Bejar, R. Chaffin, and S.E. Embretson. 1991. *Cognitive and Psychometric Analysis of Analogical Problem Solving*. Springer-Verlag, New York.
- Jonathan Berant, Ido Dagan, and Jacob Goldberger. 2012. Learning entailment relations by global graph structure optimization. *Comput. Linguist.*, 38(1):73–111, March.
- Francis Bond, Hitoshi Isahara, Sanae Fujita, Kiyotaka Uchimoto, Takayuki Kuribayashi, and Kyoko Kanzaki. 2009. Enhancing the japanese wordnet. In *Proceedings of the 7th Workshop on Asian Language Resources*, ALR7, pages 1–8, Stroudsburg, PA, USA. ACL.
- Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised learning of narrative event chains. In *Proceedings of ACL-08: HLT*, pages 789–797, Columbus, Ohio, June. ACL.

- Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 602–610, Stroudsburg, PA, USA. ACL.
- Nathanael Chambers and Dan Jurafsky. 2011. Template-based information extraction without the templates. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 976–986, Portland, Oregon, USA, June. ACL.
- Timothy Chklovski and Patrick Pantel. 2004a. Path analysis for refining verb relations. In *In Proceedings of KDD Workshop on Link Analysis and Group Detection (LinkKDD-04)*, Seattle, WA.
- Timothy Chklovski and Patrick Pantel. 2004b. Verbocean: Mining the web for fine-grained semantic verb relations. In Dekang Lin and Dekai Wu, editors, *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, EMNLP '04, pages 33–40, Barcelona, Spain, July. ACL.
- Robert Coyne and Owen Rambow. 2009. Lexpar: A freely available english paraphrase lexicon automatically extracted from framenet. In *Proceedings of the Third IEEE International Conference on Semantic Computing*.
- Ido Dagan, Bill Dolan, Bernardo Magnini, and Dan Roth. 2009. Recognizing textual entailment: Rational, evaluation and approaches. *Natural Language Engineering*, 15(4):i–xvii.
- Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.
- Charles J. Fillmore. 1976. Frame semantics and the nature of language. *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, 280(1):20–32.
- Michael A.K. Halliday. 1985. *An Introduction to Functional Grammar*. Arnold, 1st edition.
- Chikara Hashimoto, Kentaro Torisawa, Kow Kuroda, Stijn De Saeger, Masaki Murata, and Jun'ichi Kazama. 2009. Large-scale verb entailment acquisition from the Web. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1172–1181, Singapore, August. ACL.
- Chikara Hashimoto, Kentaro Torisawa, Stijn De Saeger, Jong-Hoon Oh, and Jun'ichi Kazama. 2012. Excitatory or inhibitory: a new semantic orientation extracts contradiction and causality from the web. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 619–630, Stroudsburg, PA, USA. ACL.
- Chikara Hashimoto, Kentaro Torisawa, Julien Kloetzer, Motoki Sano, István Varga, Jong-Hoon Oh, and Yutaka Kidawara. 2014. Toward future scenario generation: Extracting event causality exploiting semantic relation, context, and association features. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, USA, June. Association for Computational Linguistics.
- David A. Jurgens, Peter D. Turney, Saif M. Mohammad, and Keith J. Holyoak. 2012. Semeval-2012 task 2: Measuring degrees of relational similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 356–364, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. 2006. Extending verbnet with novel verb classes. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, pages 731–738, Genoa, Italy, June.
- Julien Kloetzer, Stijn De Saeger, Kentaro Torisawa, Chikara Hashimoto, Jong-Hoon Oh, Motoki Sano, and Kiyonori Ohtake. 2013. Two-stage method for large-scale acquisition of contradiction pattern pairs using entailment. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 693–703, Seattle, Washington, USA, October. ACL.
- Beth Levin. 1993. *English Verb Classes and Alternations*. The University of Chicago Press, Chicago and London.
- Jong-Hoon Oh, Kentaro Torisawa, Chikara Hashimoto, Motoki Sano, Stijn De Saeger, and Kiyonori Ohtake. 2013. Why-question answering using intra- and inter-sentential causal relations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1733–1743, Sofia, Bulgaria, August. ACL.

- Georgiana Puscasu and Verginica Barbu Mititelu. 2008. Annotation of wordnet verbs with timeml event classes. In Bente Maegaard Joseph Mariani Jan Odijk Stelios Piperidis Daniel Tapias Nicoletta Calzolari (Conference Chair), Khalid Choukri, editor, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, may. European Language Resources Association (ELRA). <http://www.lrec-conf.org/proceedings/lrec2008/>.
- James Pustejovsky, Jos Castao, Robert Ingria, Roser Saur, Robert Gaizauskas, Andrea Setzer, and Graham Katz. 2003. Timeml: Robust specification of event and temporal expressions in text. In *Fifth International Workshop on Computational Semantics (IWCS-5)*.
- James Pustejovsky. 1998. *The Generative Lexicon*. MIT Press, Cambridge.
- Michaela Regneri, Alexander Koller, and Manfred Pinkal. 2010. Learning script knowledge with web experiments. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 979–988, Uppsala, Sweden, July. ACL.
- Roger C. Schank and Robert P. Abelson. 1977. *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*. L. Erlbaum, Hillsdale, NJ.
- Idan Szpektor, Eyal Shnarch, and Ido Dagan. 2007. Instance-based evaluation of entailment rule acquisition. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 456–463, Prague, Czech Republic, June. ACL.
- Hiroya Takamura, Takashi Inui, and Manabu Okumura. 2005. Extracting semantic orientations of words using spin model. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 133–140, Ann Arbor, Michigan, June. Association for Computational Linguistics.
- Partha Pratim Talukdar, Derry Wijaya, and Tom Mitchell. 2012. Acquiring temporal constraints between relations. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12*, pages 992–1001, New York, NY, USA. ACM.
- P. D. Turney and S. M. Mohammad. 2014. Experiments with three approaches to recognizing lexical entailment. *Natural Language Engineering*, FirstView:1–40, 5.
- István Varga, Motoki Sano, Kentaro Torisawa, Chikara Hashimoto, Kiyonori Ohtake, Takao Kawai, Jong-Hoon Oh, and Stijn De Saeger. 2013. Aid is out there: Looking for help from tweets during a large scale disaster. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1619–1629, Sofia, Bulgaria, August. ACL.
- Hila Weisman, Jonathan Berant, Idan Szpektor, and Ido Dagan. 2012. Learning verb inference rules from linguistically-motivated evidence. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 194–204, Jeju Island, Korea, July. ACL.