Annotating Lexically Entailed Subevents for Textual Inference Tasks

Seohyun Im and **James Pustejovsky**
Computer Science Department, Brandeis University
Waltham, MA 02453
ish97, jamesp@cs.brandeis.edu

Abstract

This paper presents a procedure for constructing an *Event Structure Lexicon* (ESL), a resource which represents the lexically-entailed subevents in text as a support for textual inference tasks. The ESL is used as a resource for a subevent markup algorithm, called SUBEVITA, which annotates event implicatures on top of TimeML-based extraction algorithms. Such a resource can be used independently within the RTE task and other linguistic reasoning applications. Finally, we present experimental results of the classification for building the ESL of motion verbs in English.

1. Introduction

The goal of this research is to construct a lexical resource, called an Event Structure Lexicon (ESL), and develop an algorithm for additional markup of subevents on top of TimeML-based procedures (Pustejovsky et al. 2003a; Verhagen and Pustejovsky 2008), which we call SUBEVITA (**SubE**vents **I**n **T**ext **A**nalyzer), in order to support event implicature-based inferencing. Event Implicature here is defined as the lexical entailment or presupposition based on the Event Structure of event-denoting expressions, which is composed of pre-state, process, and result state (post-state)[1].

As results of the recent RTE tasks demonstrate (Bar-Haim et al. 2006; Giampiccolo and Magnini 2007), the amount of lexical knowledge a system is able to exploit is most important in the performance of a deep entailment system. However, the RTE systems which use the existing lexical resources do not show significantly better results than simple lexical overlap (Burchardt et al. 2008; Pazienza, Pennacchiotti, and Zanzotto 2006). One of the reasons is the dearth of knowledge about event-related entailment (e.g. *kill* ⟶ *die*). Out of the existing lexical resources, WordNet and FrameNet have no knowledge about event-related entailments. VerbNet has a SEMANTICS frame

[1]Textual Entailment in RTE challenges is defined as a directional relationship between a pair of text fragments ((Dagan, Glickman, and Magnini 2006). As a reviewer pointed out, we recognize that First Order Logic (FOL) generally is not expressive enough for Natural Language semantics but for computational linguistic reasoning tasks such as RTE, formalizations for less powerful than FOL have been employed.

for event structure repersentation but it is inconsistent and incomplete (Zaenen, Condoravdi, and Bobrow 2008).

As a solution to this problem, we have developed the ESL so that RTE systems can recognize the entailment between text and hypothesis as in the following.

(1) **Text:** *The Clark County medical examiner's office said* <u>*the man who was killed*</u> *was 33 years old.*
Hypothesis: *The Clark County medical examiner's office put* <u>*the dead man*</u>*'s age at 33.*

Understanding the semantic relationship between this pair of expressions requires the recognition of the entailment between *kill* and *dead*. The verb *kill* has several entailments in the following sentence:

(2) *Oswald* **killed** *Kennedy November 22, 1963.*
   a. *Kennedy* <u>died</u> *November 22, 1963.*
   b. *Kennedy* <u>was dead</u> **after** *November 22, 1963.*
   c. *Kennedy* <u>was alive</u> **before** *November 22, 1963.*

All event implicatures in (a-c) above are related to the lexically encoded event structure of *kill*. The *killing* causes *dying*; *be_dead* in (b) is a result state (post-state) of the event; and the state *be_alive* in (c) is a pre-state of the *killing* event be carried out.

In addition to the above RTE task, the ESL can support various NLP applications such as QA, temporal and spatial reasoning with the help of TimeML, TimeBank (Pustejovsky et al. 2003b), and SpatialML (MITRE 2007), and identification of event antecedents in co-reference tasks (cf. Im and Pustejovsky (2009) for more detail)[2].

In this paper, we present the procedure of constructing the ESL. In the following section, we describe the procedure of automating the construction of an ESL entry. Section 3 briefly discuss the SUBEVITA. Finally, we show the result of classification of motion verbs in text with a Maxent classifier as a part of the ESL entry in section 4.

[2]Im and Pustejovsky (2009) present an initial architecture for constructing an ESL. Here we expand upon this discussion, focusing on a specific example from motion verbs, while also presenting new results from the event classification experiments.

2. Building Subevent Structures

In this section, we introduce a semi-automated procedure for constructing a lexicon of event-based implicatures (ESL) using a combination of corpora and lexical resources. Our assumption for development of the ESL is that (i) verb occurrences are classified into verb classes, and (ii) each verb class has its own proper event structure frame. The classification process consists of the three steps: event type (aspectual class), verb class, and subclass. For each verb occurrence in text, building the ESL involves the following steps:

1. Identify the "event type in context" (the contextualized Aktionsart);

2. Assign the appropriate subevent structure frame associated with this subclass;

3. Paraphrase the predicates associated with each subevent;

4. Assemble resulting information as a structured object for each verb into ESL.

Event Type Identification. The first task is "Event Type Identification," which is to identify the aspectual class of each verb (Vendler 1967; Dowty 1979; Pustejovsky 1995) as it occurs in context in text. Recognizing the event type of a verb in context is difficult (Klavans and Chodorow 1992). Recently, however, Zarcone and Lenci (2008) demonstrated that robust event type classification is possible.

We adopt context-dependent event type identification as proposed in Zarcone and Lenci (2008). The event type of a verb occurrence is determined by the complex interaction among different features such as the verb's argument structure, its aspect, the definiteness, and plurality of its arguments, frequency and genericity marking, and so on (Zarcone and Lenci 2008). For example, the progressive aspect cancels the result state of a lexically-marked accomplishment (transition) event and thus changes its event type to a process (e.g. *build*: transition; *be_building*: process). Verb occurrences in a corpus are manually annotated with their proper event types. Then Maximum Entropy classifiers are applied and trained on the corpus. The event type consists of: process, state, and transition.

Verb class and Subclass. Once its event type is identified, the verb occurrence is classified into verb class and then subclass. The verb classes and subclasses are based on the Brandeis Semantic Ontology, BSO (Pustejovsky et al. 2006). The upper level class is composed of: process, state, change_of_location, change_of_possession, and change_of_state. All classes except for state have their corresponding causation verb classes. Each of these may have subclasses.

Event Structure Frame Assignment. Each of the subclasses has its own proper event structure frame. For example, the event structure of the subclass to_goal is as follows: se1: pre-state: not_be_at (x, y); se2: process: pred-ing (x); se3: post-state: be_at (x, y). We assume a model of event structure frame as presented in Generative Lexicon, GL (Pustejovsky 1995). The event structure frame in GL is a representation associated with a verb where predicative content is decomposed into subevents and their temporal ordering, along with headedness.

Paraphrasing. The event structure frame of the subclass cause_to_go_out_of_existence (e.g. *kill*) is shown below:

(3) event structure frame of cause_to_go_out_of_existence subclass
se1: pre-state: not_be_pred-pp (y)
se2: process: pred-ing (x,y)
se3: process: being_pred-pp (y)
se4: post-state: be_pred-pp(y)

The verb *kill* substitutes for the position of *pred* as in (4):

(4) event structure of kill
se1: pre-state: not_be_killed (y)
se2: process: killing (x)
se3: process: being_killed (y)
se4: post-state: be_killed (y)

Paraphrasing is required to derive the event structure with different predicates in (5) from (4).

(5) subevents of kill: different predicates
se1: pre-state: be_alive (y)
se2: process: killing (x)
se3: process: dying (y)
se4: post-state: be_dead (y)

After the assignment of an event structure frame to a verb, we compile paraphrases for the predicates associated with each subevent in the event structure. For this step, we utilize the lexical resources of WordNet and Extended WordNet and clustering technique.

For paraphrasing, we distinguish between *closed domain* and *open domain*. The former contains predicates falling into semantic classes with generally well-defined predications associated with the subevents. This includes, for example, the verb classes change_of_location and change_of_possession. For instance, the verb *drive* as a change_of_location verb generates the closed domain ESL entry shown below.

(6) drive in *John drove to Boston*
se1: pre-state: not_be_in (x,y)
se2: process: driving (x)
se3: post-state: be_in (x,y)

Open domain predicates include verbs in the large change_of_state verb class, where there are few if any general predications associated with subevents in the event structure as in (3). After a verb is identified with a particular open domain verb class, paraphrases are generated for each subevent in the event structure frame with the help of various resources such as WordNet.

The last step involves compiling the extracted event structure frames of verb occurrences into the ESL. Table 1 shows

the ESL of the verb *arrive*, compared with the semantic frame of the verb in VerbNet. As we see in the next section, SUBEVITA uses the ESL as a lexical resource for markup of SUBEVENT tags.

3. Annotating Text with ESL

Using the ESL as a reference library, a subevent annotation algorithm called SUBEVITA is now able to annotate an EVENT-tagged corpus such as TimeBank with SUBEVENT tags to represent the event structure frames of EVENT-tagged expressions. SUBEVITA takes text that has been processed by a temporal parsing systems such as TTK (Verhagen and Pustejovsky 2008), with EVENT and TIMEX3 tags explicitly annotated, and generates the appropriate subevent tags for each event. We can think of SUBEVENT tagging as a general, domain-independent meta-data enrichment of text, which can be exploited by diverse NLP applications, such as RTE, QA, and other such tasks. We will not elaborate on SUBEVITA here, but the output of this process is illustrated below, with a text fragment containing the verb *arrive*.

(7) **Today**$_{t1}$ *King Hussein of Jordan* **arrived**$_{e2}$ *in Washington.*

<TIMEX3 tid="t1" type="DATE" value="1989-03"> **today** </TIMEX3> < EVENT eid="e2" class="OCCURRENCE" tense="PAST" aspect="NONE" polarity="POS> **arrived** </EVENT>

Evita annotates *arrived* with an EVENT tag and assigns the appropriate attribute-value pairs. According to its ESL entry, the verb *arrive* has three subevents and thus SUBEVITA inserts three SUBEVENT tags as meta-data markup, based on the ESL in table 1.

<SUBEVENT seid="se1" partOf="e2" />
<SUBEVENT seid="se2" partOf="e2" />
<SUBEVENT seid="se3" partOf="e2" />

SUBEVITA connects the appropriate arguments of the verb in text with SUBEVENTs via ARGLINK tags (Pustejovsky, Littman, and Saurí 2006). The resulting meta-data annotation of this text now enables the inferencing capabilities mentioned in section 1. That is, entailments referring to the subevent implicatures of the movement can now be addressed, by virtue of the explicit representation of these events in the annotation through the ESL.

4. Classification of Motion Verbs for ESL

As described in section 2, the initial process of building the ESL involves a series of classification tasks: event_type, verb_class, and subclass. Once a verb occurrence is classified into a specific subclass, its proper event structure frame is assigned automatically. Then for open domain verb classes, a paraphrasing step is added. In this section, we show the classification experiment with motion verbs.

Data. For classification of motion verbs, we collected texts (about 40k words) from traveler's blog (www.travelblog.org) and chose all motion verbs (total 39 verbs and 1657 occurrences). The verbs are: APPEAR,

ARRIVE, BOARD, CARRY, CLIMB, COME, CROSS, CYCLE, DEPART, DIVE DRIVE, ENTER, ESCAPE, EXPLORE, FALL, FLY, GET, HANG, HEAD, HOP, JUMP, LAND, LEAVE, LOAD, PARK, PASS, PEDAL, PROCEED, PROGRESS, PUT, REACH, RETURN, RIDE, ROLL, RUN, TAKE, TRAVEL, WALK, and WANDER. The occurrences of the motion verbs in the texts are used as a test set. Because nouns are not considered in this research, we excluded nominal use of the verbs such as gerundive nominals and nominalizations.

Classification was performed with Maximum Entropy Classifiers[3], following Zarcone and Lenci (2008). For training, we got 4449 occurrences of the 39 motion verbs from British National Corpus and one of the authors annotated mannually with their event type, verb class, and subclass. Then, a maxent classifier is applied to and trained on the data.

Verb classes. Event type is simplified into a three-way distinction of process, state, and transition, because their basic event structure frames are the same: pre-state, process, and post-state.

After event type classification, the verb occurrence is classified into one of the upper level verb classes: process, state, change_of_possession, change_of_state, change_of_location. Finally, if its verb class is change_of_location, the verb occurrence is classified into one of the subclasses and assigned its event structure frame according to subclass. The subclass includes only to_goal, from_source and from_source_to_goal, paying attention to pre-state and post-state. The others are classified into change_of_location for now. The subclasses of motion verbs are being developed in the broader context of modeling motion in language (Pustejovsky and Moszkowicz 2008). The process verb class is just inherited to subclass.

Feature selection. The features for classification are mannually selected, based on the result of parsing with Stanford dependecy parser. The features for Event Type classifications of motion verbs are like:

(8) **Event Type Features**
   a. presence of locative PP, directional PP, or particles
   b. semantic class of subjects, direct objects, prepositional objects
   c. presence of locative adverbials
   d. aspect (progressive, perfective)
   e. presence of complement clauses
   f. voice (passive)
   g. part_of_speech (participles)

For event type classification of motion verbs, presence of prepositions, particles, and adverbials which entail change_of_location or state (e.g. to, from, into, onto, on, across, off, out, in, home, etc.) is the most important feature. Motion verbs are lexically classified into manner_of_motion (e.g. *walk, run, jump*) or change_of_location (e.g. *arrive, leave, come*). However, they change their verb classes in context. Consider the following example:

(9) a. John **ran** fast.

---

[3]We used the Carafe classifier developed by Ben Wellner.

ESL		VerbNet	
VERB	**ARRIVE**	VERB	**ARRIVE**
CLASS	change_of_location	CLASS	escape-51.1-2
SUB_CLASS	to_goal		
EVENT_TYPE	transition		
SUBEVENT	se1: pre-state: not_be_in (x,y) se2: process: arriving(x,y) se3: post-state: be_in (x,y)	SEMANTICS	motion(during(e)) location(end(e), theme, oblique)
TEMP_ORDER	se2 ENDS se1 se3 MEETS se2		
SENTENCE	*John arrived in Boston.*	EXAMPLE	*He arrived in U.S.*

Table 1: *arrive* in ESL vs. VerbNet

b. John **ran** into the store.
c. John **is running** into the store.

The verb *run* in (9a) is an activity verb which has only a process subevent (*running(John)*). On the other hand, it changes its verb class into a change_of_location class with a preposition as in (9b). Hence, the disambiguation of motion verbs is mainly dependent on their adjunct composition with prepositional phrases and particles. While, progressive_aspect cancels the result state of transition as shown in (9c).

Second, Semantic classes of arguments are decided with a reference to classes of Brandeis Semantic Ontology. For example, if the subject of a sentence belongs to the representational object class in BSO, the verb in the sentence is a state event type as below:

(10) a. The report **explore** the desirability of transferring sewage loadings either within the catchment or by diversion to the Edinburgh sewerage system. (state)
b. We **explored** the town. (process)

If the subject in a sentence represents location or path as in 11, the verb denotes a state.

(11) a. John **crossed** the street to come to school.
(change_of_location)
b. The bridge **crosses** the river. (state)

Third, a motion verb such as *appear* has the event type of state, if it has a complement clause.

(12) a. John **appeared**. (change_of_location)
b. It **appears** that John passed the exam. (state)

After the Event Type of a verb occurrence in text is determined, we use the result of the classification for verb class distinction. The verbs of transition event type are classified into one of change_of_location, change_of_state, and change_of_possession. For classification of motion verbs, we used features to distinguish change_of_location verbs from the others. The features for verb class distinction are given below:

(13) **Verb Class Features**
a. event type

b. semantic class of prepositional and direct objects and subjects

If prepositional objects or direct objects of verb occurrences are locative expressions, they belong to change_of_location verb class.

(14) a. John **ran** into the store. (change_of_location)
b. The resolution **ran** into much more opposition. (change_of_state)

Finally, the features for subclass are shown below:

(15) **Subclass Features**
a. event type
b. verb class
c. kinds of prepositions, particles, and adverbs

Specific prepositions or particles help determine the subclasses of change_of_location verbs.

The Result. The MaxEnt classification for motion verbs demonstrated a very high accuracy: event type - 97%; verb class - 87%; subclass - 93%. Table 2 presents the statistics of classifying the motion verbs in the context in the motion corpus.

M_Verbs		Precision	Recall	F-meature	Accuracy
ETYPE	S	1.00	0.72	**0.84**	0.97
	P	0.96	0.91	**0.93**	
	T	0.98	0.99	**0.99**	
VCLASS	S	1.00	1.00	**1.00**	0.87
	P	1.00	1.00	**1.00**	
	COP	0.14	0.01	**0.02**	
	COS	0.89	0.81	**0.85**	
	COL	0.81	0.94	**0.87**	
SCLASS	S	1.00	1.00	**1.00**	0.93
	P	1.00	1.00	**1.00**	
	COL	0.92	0.27	**0.42**	
	FS	0.93	0.97	**0.95**	
	TG	0.98	0.98	**0.98**	
	FSTG	0.48	0.57	**0.52**	
	COS	0.88	1.00	**0.94**	

Table 2: Statistics of Motion Verb Classification

Each of the verb occurrences in each subclass gets its proper event structure frame automatically. The event

structure frames are illustrated below.

(16) process
    se1: process: pred-ing (x)

(17) from_source
    se1: pre-state: be_at (x,y)
    se2: process: pred-ing (x)
    se3: post-state: not_be_at (x,y)

(18) to_goal
    se1: pre-state: not_be_at (x,y)
    se2: process: pred-ing (x)
    se3: post-state: be_at (x,y)

(19) from_source_to_goal
    se1: pre-state: be_at (x,y)
    se2: pre-state: not_be_at (x,z)
    se3: process: pred-ing (x)
    se4: post-state: not_be_at (x,y)
    se5: post-state: be_at (x,z)

It should be pointed out that we do not treat negation, quantification (e.g. *always, often, all, etc.*), or modality (e.g. *can, must, neceassarily, possibly*) as features that determine the event structure, since they do not change the classification itself. They do, of course, affect the modality and truth value of the propositional content of the subevents.

(20) a. John **ran** <u>into the store</u>.
    b. John **didn't** run <u>into the store</u>

Negation in (20b) negates the entire event, but this is an independent semantic computation, done on the sentence level when modality information from EVITA is taken advantage of.

5. Conclusion

In this paper, we presented a procedure for automating the construction of an Event Structure Lexicon (ESL) that can be used as a lexical resource for textual inference tasks such as RTE and other NLP applications. The ESL is used as a resource for a subevent markup algorithm, called SUBEVITA, which creates a subevent-annotated corpus when embedded within the TimeML-based TARSQI Toolkit. Such a resource can be used independently within the RTE task and other language reasoning applications.

The present work is obviously programmatic and is still in development. As a first step, we ran classification experiments for motion verbs in a corpus created from travelers' blogs. The result shows higher accuracy and F-measure. Although the verb set is relatively small, it indicates that our system is on the right track. Our goal is to cover all of Levin's verb classes with the ESL. Some of the risks and uncertainties in the above technique include: overgeneration of paraphrases for each subevent predicate; and misclassificaiton of the verb class, due to lexical ambiguity. These are matters we hope to address in the future.

References

Bar-Haim, R.; Dagan, I.; Dolan, B.; Ferro, L.; Giampiccolo, D.; Magnini, B.; and Szpektor, I. 2006. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*.

Burchardt, A.; Pennacchiiotti, M.; Thater, S.; and Pinkal, M. 2008. Assessing the impact of frame semantics on textual entailment. *Journal of Natural Language Engineering*. Special Issue on Textual Entailment.

Dagan, I.; Glickman, O.; and Magnini, B. 2006. *The PASCAL Recognising Textual Entailment Challenge*, volume 3944 of *Machine Learning Challanges*. Springer. Lecture Notes in Computer Science.

Dowty, D. 1979. *Word Meaning and Montague Grammar*. Dordrecht, The Netherlands: Kluwer.

Giampiccolo, D., and Magnini, B. 2007. The third pascal recognizing textual entailment challenge. In *Proceedings of the Workshop on Textual Entailment and Paraphrasing*, 1–9. Prague: Association for Computational Linguistics.

Im, S., and Pustejovsky, J. 2009. Annotating event implicatures for textual inference tasks. In Rumshisky, A., and Calzolari, N., eds., *Proceedings of the 5th International Conference on Generative Approaches to the Lexicon*.

Klavans, J., and Chodorow, L. 1992. Degrees of stativity: The lexical representation of verb aspect. In *Proceedings of the 14th International Conference on Computational Linguistics (COLING)*.

MITRE. 2007. Spatialml: Annotation scheme for marking spatial expressions in natural language. http://sourceforge.net/projects/spatialml/.

Pazienza, M. T.; Pennacchiotti, M.; and Zanzotto, F. M. 2006. Mixing wordnet, verbnet and propbank for studying verb relations. In *Proceedings of the 5th Language Resource and Evaluation Conference*.

Pustejovsky, J., and Moszkowicz, J. L. 2008. Integrating motion predicate classes with spatial and temporal annotations. In *COLING 2008: Companion volume - Posters and Demonstrations*, 95–98.

Pustejovsky, J.; Castaño, J.; Ingria, R.; Saurí, R.; Gaizauskas, R.; Setzer, A.; and Katz, G. 2003a. Timeml: Robust specification of event and temporal expressions in text. In *Proceedings of IWCS-5, Fifth International Workshop on Computational Semantics*.

Pustejovsky, J.; Hanks, P.; Saurí, R.; See, A.; Gaizauskas, R.; Setzer, A.; Radev, D.; Sundheim, B.; Day, D.; Ferro, L.; and Lazo, M. 2003b. The timebank corpus. In *Proceedings of Corpus Linguistics*, 647–656.

Pustejovsky, J.; Havasi, C.; Littman, J.; Rumshisky, A.; and Verhagen, M. 2006. Towards a generative lexical resource: The brandeis semantic ontology. In *Proceedings LREC 2006*.

Pustejovsky, J.; Littman, J.; and Saurí, R. 2006. Argument structure in timeml. In *Dagstuhl Seminar Proceedings: Annotating, Extracting, and Reasoning about Time and Events*.

Pustejovsky, J. 1995. *The Generative Lexicon*. USA: The MIT Press.

Vendler, Z. 1967. *Linguistics in Philosophy*. Ithaca, NY: Cornell University Press.

Verhagen, M., and Pustejovsky, J. 2008. Temporal processing with the tarsqi toolkit. In *In Proceedings of Coling 2008: Companion Volume-Posters and Demonstractions*, 189–192.

Zaenen, A.; Condoravdi, C.; and Bobrow, D. 2008. The encoding of lexical implications in verbnet. In *Proceedings of LREC 2008*.

Zarcone, A., and Lenci, A. 2008. Computational models of event type classification in context. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*. Marrakech, Morocco: European Language Resources Association (ELRA).