

From Discourse Representation Structure to Event Semantics: A Simple Conversion?

Daniel Dakota and Sandra Kübler

Indiana University

Email: {ddakota, skuebler}@indiana.edu

Abstract—Many applications in Natural Language Processing require a semantic analysis of sentences in terms of truth-conditional representations, often with specific desiderata in terms of which information needs to be included in the semantic analysis. However, there are only very few tools that allow such an analysis. We investigate the representations of an automatic analysis pipeline of the C&C parser and Boxer to determine whether Boxer’s analyses in form of Discourse Representation Structure can be successfully converted into a more surface oriented event semantic representation, which will serve as input for a fusion algorithm for fusing hard and soft information. We use a data set of synthetic counter intelligence messages for our investigation. We provide a basic pipeline for conversion and subsequently discuss areas in which ambiguities and differences between the semantic representations present challenges in the conversion process.

I. INTRODUCTION

MANY applications in Natural Language Processing require a semantic analysis of sentences. However, automatic semantic analysis is a field in its infancy in Natural Language Processing. There is work on automatically analyzing semantic role labeling, as evidenced by two shared tasks at the Conference on Natural Language Learning [1], [2] and a special issue of the journal *Computational Linguistics* [3]. But for many downstream applications related to text understanding, semantic roles do not provide enough information. Our current work focuses on fusing soft and hard information, where soft information constitutes natural language. A fusion algorithm accepts information from different sources and provides an integrated, accurate, informative whole. While fusion algorithms for sensor data are advanced and reliable, efforts to include natural language are in their early stages [4]. When language is included, fusion often includes inference mechanisms [5]. In order to be able to integrate language information into a fusion approach, we need to provide the information in a variant of predicate logic, on which inference and fusion algorithms can work.

There are existing approaches to analyzing language into semantic representations based on different syntactic formalisms (cf. e.g., [6] for LFG and [7] for TAG). We focus here on truth-conditional semantics based on Combinatory Categorical Grammar (CCG) [8] since this grammar formalism provides the closest match to our needs in terms of the target predicate logic. CCG relies on combinatory logic, which is equivalent in expressive power to lambda calculus. One approach to parsing CCG is the C&C parser [9], [10], which can be

used in combination with Boxer [11], [12], [13], a module that converts the CCG syntax to semantic representations in the form of Discourse Representation Structure (DRS) [14], [15]. Other CCG-based approaches attempt learning semantic representations from different sources directly (e.g. [16], [17]).

Our target semantic representation is a form of event semantics, which means that neither parser provides us with analyses that are usable directly. Thus, we present work on investigating a rule-based conversion from Discourse Representation Structure as provided by Boxer to our target event semantic representations.

The remainder of this paper is structured as follows: We first provide more details about the conversion task in section II, then we briefly introduce our target semantic representations in section III, focusing on those aspects and distinctions that we target in the conversion. We then describe the analysis pipeline in section IV and discuss cases that can be converted in a rule-based fashion in section V. Finally, we discuss linguistic phenomena that present challenges for a conversion in section VI and conclude with a discussion of approaches to handle those difficulties (section VII).

II. TASK OVERVIEW

Our task is to perform a semantic analysis of sentences in order to use them in a data fusion model for fusing hard and soft information [18], [19]. The fusion model expects an analysis in terms of first order logic and can be extended to a Davidsonian model. In a Davidsonian model, semantics is non-propositional, and references are integrated into the semantic description. References between events are described using event variables.

Since an unlimited truth-conditional analysis of unrestricted sentences is a very challenging task and since we have a very specific task as downstream consumer of our annotations, we have decided to reduce the complexity of the task of semantic analysis by assuming an automatic syntactic simplification of the sentences to be analyzed. In contrast to standard approaches to sentence simplification, our syntactic simplification model (currently under development) will focus on specific syntactic phenomena and will simplify only sentences that display such phenomena. Simplification will be performed by a machine learning module trained on a small set of sentences displaying a specific phenomenon, based on a dependency parse. The simplified sentences will then be reparsed by the CCG parser.

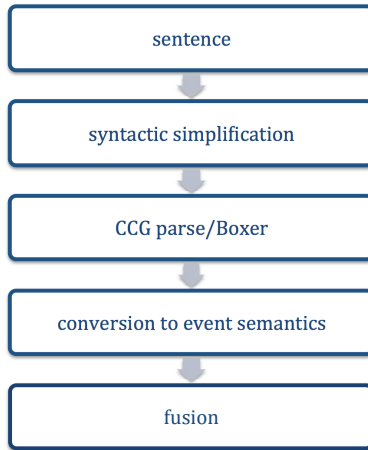


Fig. 1. Proposed Pipeline from the input sentence to information fusion.

The goal is a simplification without loss of information; we currently target coordination, reported speech, and passive sentences. Examples of simplifications are shown in examples (1)–(3). Note that in the case of reported speech, the reporting action is converted into an action and a certainty value (shown in square brackets), both of which can directly be used for information fusion.

- (1) a. The man bought a book and a map.
b. The man bought a book. The man bought a map.
- (2) a. The anonymous caller reported that an attack would happen next week.
b. An attack will happen next week. [action: report; certainty: 0.9]
- (3) a. The man was given an object by a passer-by.
b. The passer-by gave the man an object.

As a consequence of the simplification step, we assume that the simplified sentences will be easier to parse by the CCG parser used in the semantic analysis. The final pipeline of the process is shown in Fig. 1.

III. THE ANNOTATED DATA SET FOR EVENT SEMANTICS

We use the SYNCOIN data set [20] for our experiments. SYNCOIN is a synthetically created set of counter insurgency scenarios in the form of collections of intercepted phone conversations and intelligence reports. The data set was designed to support approaches towards the fusion of hard and soft information.

We have access to truth-conditional semantic annotations of a set of sentences that constitute five “threads”. These sentences were annotated within our project. The annotation scheme is based on Davidsonian event semantics [21], and the annotations are mostly rather surface oriented, in order to allow for robust automatic processing. The annotations are based on the following principles:

TABLE I
ANNOTATION SCHEME

Annotation	Description
x	named entity: person, group, or organization
y	a location
t	a time variable
e	an event variable
z	any other object

Since verbs report actions, they introduce event variables, and their syntactic arguments function as logical arguments in the following order: event, subject, direct object, and indirect object. We show an example sentence and its annotation in (4). Here, the verb *refuse* is analyzed as the 50th event in the thread, and it has two arguments, x_6 referring to *men* and e_2 referring to *attack*. Note that the annotation also states that the agent/subject of the attack are the men (x_6).

- (4) a. The men refuse to reveal operational details of the attack.
b. $\text{refuse}(e_{50}, x_6, e_{51}) \wedge \text{men}(x_6) \wedge$
 $\text{reveal}(e_{51}, x_6, z_{14}) \wedge \text{details}(z_{14}, e_{44}) \wedge$
 $\text{operational}(z_{14}) \wedge \text{attack}(e_{14}, x_6)$

There are two more points worth mentioning concerning the example: 1) Verbs are not the only concepts that introduce events. In the example, the nominalization *attack* is also represented as an event. 2) Both the men and the attack were mentioned previously in the text, which is indicated by shared variables. Thus, we integrate coreference information across sentence boundaries into the annotations.

The annotation scheme uses five types of variables, as shown in Table I. Nouns typically introduce variables of types x or z whereas the type e is typically introduced by verbs and nominalized verbs. Variable types y and z may be introduced by a wider range of parts of speech. Depending upon context, a word can potentially belong to more than one category.

All variables introduced by indefinite noun phrases and verbs are existentially quantified. However, existential quantifiers are not included as part of the semantic annotation. They are assumed to have scope over all sentences in the annotated document. Modifiers are introduced by adjectives, adverbs, and prepositional phrases. Adjectives typically introduce a new predicate, which is a property of objects as seen in (4) in the phrase *operational details*. Adverbs also introduce a new property. However, since they typically modify events, they have an event variable as argument, as seen in (5). Names are introduced by a predicate *named*. Possessive pronouns introduce a modifier which is annotated as a separate predicate as seen in (6). Units of measurements are introduced by a special predicate introducing the unit of measure and the numerical value, see the example in (7).

- (5) a. The car drove fast.
b. $\text{car}(z_1) \wedge \text{drove}(e_1, z_1) \wedge \text{fast}(e_1)$
- (6) a. John’s car departed.

- (7) a. $\text{car}(z1) \wedge \text{possessive}(x1,z1) \wedge \text{named}(x1,\text{John}) \wedge \text{departed}(e1,x1)$
 b. 250 gallon tanks
 b. $\text{tanks}(z1) \wedge \text{size_in_gallon}(z1, 250)$

Temporal phrases normally modify events, but can also introduce dates as seen in (8). Quantification is handled in a surface oriented way, as shown in (9). However, existential quantification over events is introduced by negation (see (10)) and questions.

- (8) a. The man arrived on 01\05\10.
 b. $\text{man}(x1) \wedge \text{arrived}(e1,x1) \wedge \text{on}(e1,x1) \wedge \text{date}(t1,010510)$
- (9) a. Omar Khrayesh visits several bookstores in Adhamiya.
 b. $\text{visit}(e3,x1,y1) \wedge \text{named}(x1, \text{Omar Khrayesh}) \wedge \text{bookstores}(y1) \wedge \text{several}(y1) \wedge \text{in}(y1,y2) \wedge \text{named}(\text{Adhamiya})$
- (10) a. Mallahati did not respond.
 b. $\text{named}(x1, \text{Mallahati}) \wedge \text{not}(\exists e \text{ respond}(e,x1))$

IV. ANALYZING SENTENCES USING DRS

A. Parsing

As discussed above, we utilize Combinatory Categorical Grammar [8] since it provides a clearly defined interface between syntax and truth-conditional semantics, making it ideal for our purposes. We use the C&C parser [9], [10] in combination with Boxer [11], [12], [13]. The parser provides pre-trained models based on CCGbank [22], sections 02-21 and MUC 7 [23]. We parse single sentences of SYNCOIN data into a CCG derivation. These derivations then serve as input for Boxer.

B. Boxer's DRS Analyses

Boxer is a module that uses the CCG derivations produced by the C&C parser to generate semantic representations in the form of Discourse Representation Structures (DRS), which are based on Discourse Representation Theory [15]. The theory assumes that hearers incrementally build a mental representation of a discourse, the DRS, which is a representational and non-compositional semantic representation. A DRS mainly models the referents of a discourse and the conditions that hold. A referent is an entity within the discourse while a condition is a predicate demonstrating properties of the respective referents.

A DRS consists of two parts, describing the referents and the conditions respectively. One important deviation from standard Discourse Representation Theory in Boxer is the use of a Neo-Davidsonian analysis of events and thematic roles [11]. This means that the representations provided by Boxer are close to our target event annotations, but that we need to abstract away from the semantic role representation and the more structured representation of referents.

DRS conditions are either basic or complex¹. Complex

¹A full account of Boxer's DRS representations can be found at <http://svn.ask.it.usyd.edu.au/trac/candc/wiki/DRSs>

x3 x2 x1	e1 s1 x5 x4
.....
(named(x1,café,nam)	+ Theme(e1,x4))
x1 = x3	Actor(e1,x1)
named(x3,internet,nam)	receive(e1)
x1 = x2	of(x4,x5)
named(x2,antar,nam)	computer(x5)
	new(s1)
	Pivot(s1,x5)
	x5 = 10
	delivery(x4)

Fig. 2. Boxer DRS for sentence (11) in box format.

conditions express phenomena such as implication or negation. In this paper, we focus on basic conditions, which express equalities, one-place relations, two-place relations, names, or time expressions. One-place relations are introduced by nouns, verbs, adjectives, and adverbs while two place relations are used to express verb roles and prepositions.

Boxer provides its analyses in two forms: a box representation and the same information in PROLOG style. The DRS structure of the SYNCOIN sentence (11) in box form is shown in Fig. 2. Each box, representing a single DRS, has the referents at the top, and the conditions within the box. The conditions of the second box show that the *café* is considered the actor and *delivery* the theme of the receiving event. The plus sign indicates the merging of the two basic DRS structures into a more complex one.

- (11) The Antar Internet Café received a delivery of 10 new computers.

The second representation that Boxer offers is in PROLOG style, as shown in Fig. 3 (for example (11)). Here, the complex DRS is indicated by the merge operation. Since this representation is more easily processed automatically, we use it in our conversion to event semantics.

There are two primary types of information in the PROLOG representation, individual properties of the token and the semantic information of the tokens in the DRS. These two pieces of information are distinct, particularly as there are often tokens that do not have a semantic role in the DRS. Every token has an identification number indicating the position of the token in the sentence. This is accompanied by the original token, the part-of-speech (POS) tag of the token, the lemma, and information whether it is a named entity.

In regard to the semantic information in Fig. 3, we focus on two different types of conditions, predicates (pred) and relations (rel). Examining $\text{pred}(x4,\text{delivery},n,0)$ more closely, a predicate consists of the referent ($x4$), the lemma (*delivery*), a general POS tag (n), and the word's named entity status (0 indicating false). A relation is represented as $\text{rel}(e1,x1,\text{'Actor'},0)$ consisting of the event ($e1$), the first referent ($x1$), the thematic role of the referent (Actor), and the sense (0 indicating

```

id(1,1).
sem(1,[1001:[tok:'The',pos:'DT',lemma:the,namex:'O'],
1002:[tok:'Antar',pos:'NNP',lemma:'Antar',namex:'O'],
1003:[tok:'Internet',pos:'NNP',lemma:'Internet',namex:'O'],
1004:[tok:'Café',pos:'NNP',lemma:'Café',namex:'O'],
1005:[tok:received,pos:'VBD',lemma:receive,namex:'O'],
1006:[tok:a,pos:'DT',lemma:a,namex:'O'],
1007:[tok:delivery,pos:'NN',lemma:delivery,namex:'O'],
1008:[tok:of,pos:'IN',lemma:of,namex:'O'],
1009:[tok:'10',pos:'CD',lemma:'10',namex:'O'],
1010:[tok:new,pos:'JJ',lemma:new,namex:'O'],
1011:[tok:computers,pos:'NNS',lemma:computer,namex:'O'],
1012:[tok:'.',pos:'.',lemma:'.',namex:'O']],
merge(drs([[:x3,[:x2,[1001]:x1],
[[1004]:named(x1,café,nam,nam),[:eq(x1,x3),
[1003]:named(x3,internet,nam,nam),[:eq(x1,x2),
[1002]:named(x2,antar,nam,nam)]]],drs([[:e1,[:s1,[:x5,
[1006]:x4],[:rel(e1,x4,'Theme',0),[:rel(e1,x1,'Actor',0),
[1005]:pred(e1,receive,v,0),[1008]:rel(x4,x5,of,0),
[1011]:pred(x5,computer,n,0),[1010]:pred(s1,new,a,0),
[:rel(s1,x5,'Pivot',0),[1009]:card(x5,10,eq),
[1007]:pred(x4,delivery,n,0)]])))).

```

Fig. 3. Boxer DRS for sentence (11) in PROLOG representation.

none). The sense of the word will not be further addressed since it is not utilized in the conversion.

V. FROM DRS TO EVENT SEMANTICS

Our goal is to convert Boxer representations to our event semantic annotation scheme as accurately as possible. In the current paper, we first investigate the degree to which the required information is available from the Boxer analysis. A successful conversion can be hampered by several different issues: when information required in semantic representation is not present in Boxer's analyses, when Boxer is not consistent in representing a specific phenomenon, or when the analyses are incorrect. We present a basic methodology for the conversion, highlighting areas in which a successful conversion can be achieved with a high degree of success. We use Boxer given the following settings: PROLOG output format (but for readability, we present example in box format below), using DRS representations, using standard thematic proto-roles, and distinguishing variable types (into events, arguments, and modifiers).

A. Basic Conversion Principles

The PROLOG syntax allows for the ability to scan all aspects of the parse and develop checking mechanisms. The output is systematic in its structure, which allows an easy identification and isolation of referents and relations if they are produced by Boxer. In order to facilitate a closer examination of the conversion, we simplify example (11) to *café received a delivery*. The corresponding part of the DRS is presented in Fig. 4, the event semantic representation in (12).

```

1004:[tok:'Café',pos:'NNP',lemma:'Café',namex:'O'],
1005:[tok:received,pos:'VBD',lemma:receive,namex:'O'],
1007:[tok:delivery,pos:'NN',lemma:delivery,namex:'O'],
      rel(e1,x4,'Theme',0)
      rel(e1,x1,'Actor',0)
[1004]:named(x1,café,nam,nam)
[1005]:pred(e1,receive,v,0)
[1007]:pred(x4,delivery,n,0))

```

Fig. 4. PROLOG representation for *café received a delivery*.

- (12) a. café received a delivery
b. $\text{café}(y1) \wedge \text{received}(e1,y1,z1) \wedge \text{delivery}(z1)$

Relation representations vary in the information they represent, but they always provide information about the event, the referent, and the relation. We use regular expressions to extract all relations in the PROLOG representation, identifying the referents involved, and cross-reference them. In Fig. 4, there are two relations (Actor and Theme) which are captured and then cycled through. Once the referent has been identified, the corresponding predicate is extracted using additional regular expressions in order to extract relevant information: the referent, the lemma, and an indication of a general POS category.

We isolate the thematic role within each relation (in our case Actor and Theme). Based on the specific role, we need to employ specialized conversion strategies since a Pivot, for example, provides more complex information than an Agent. Once a relation type has been identified, the referents are extracted. For the case of Actor, we extract $e1$ and $x1$. In this relation, the first argument, $e1$, refers to the first event introduced in the sentence. The second argument, $x1$, refers to an argument identified by the event. We then subsequently extract information on the corresponding predicate.

As mentioned in Section III, we introduce arguments of events in a specific order: the event, then the subject, then the direct object. This is demonstrated in (12), where the subject is represented by $y1$ and the direct object by $z1$. In general, there are no discrepancies between the ordering of thematic roles and grammatical functions since we assume that passive sentences have been simplified to active ones.

B. Challenges

Our main challenge is to correctly identify and convert the PROLOG variables into our event semantic variables. As presented in Table I, there are five possible variables types in the event semantics. As the PROLOG representation for events is e , these events are easily transferable to our scheme. However, as explained in Section III, verbs are not the only type of concepts that introduce events, and there is no direct correlation between other event types in the Boxer representations and in the event semantics. For example, verbal nominalizations, such as *attack* in example (4) are treated as referents like any other nouns in Boxer. Thus, in order to handle such cases properly, we will need to use additional

x2 x1	x3 e1
.....
(stall(x1) + Actor(e1,x3))	
of(x1,x2)	fire(x3)
market(x2)	Theme(e1,x1)
	damage(e1)

Fig. 5. Passive example.

semantic resources as well as event anaphora detection and resolution strategies.

In addition to event variables, we have four distinct argument variables. Returning to example in (12) and Fig. 4, the subject (*café*) is a named entity with an unknown tag in the PROLOG representation, thus the challenge is to correctly identify that the variable associated with the referent is of type \bar{y} given that it is a location. The direct object (*delivery*) is assigned $z1$ since it is the first non-named entity argument that is not a location. If the parser recognizes named entities, then they can be directly converted to corresponding variables (e.g., *org* \rightarrow x). The difficulty lies in distinguishing between any two non-event labels for given tokens, particularly objects and locations, as although named entities are often detected by the parser while parsing the SYNCOIN data, they are frequently mislabeled, as many of these entities cannot be found in the training data of the parser. Thus, we must distinguish the possible named entities and their associated variables in our conversion (e.g. *Antar café* vs. *weapons cache*). This problem will be further addressed in section VI.

Additionally, the conversion requires multiple checks. One issue concerns the numbering of types of variable, i.e., how many events have been introduced, to correctly assign a number to a newly introduced variable. Another complexity arises from the fact that multiple variable types in the event semantic representations are represented by a single variable in Boxer; thus we need strategies to determine the resulting variable types via heuristics. For example, a referent of Boxer's type x can be of any type shown in Table I.

One discrepancy between the two semantic annotations concerns the thematic roles used by Boxer and the more surface oriented annotations in the event semantics that are based on syntactic arguments rather than thematic roles. The thematic roles generally correspond to syntactic arguments of a specific event and relation. For example, the role Actor is associated with the subject and Theme with a direct object. For many sentences, the order of the arguments is associated with the same sequence as our annotation scheme. However, there is an issue with passive sentences where there is no direct correspondence between roles and grammatical functions, as in Fig. 5 when parsing the passive sentence in (13).

(13) The market stalls were damaged by fire.

Here, strictly associating the Theme with a direct object position, and Actor with the subject, when introducing arguments of an event would result in an incorrect conversion. For such

```
1010:[tok:new,pos:'JJ',lemma:new,namex:'O']
1011:[tok:computers,pos:'NNS',lemma:computer,namex:'O']
      rel(F,G,'Pivot',0)
      [1010]:pred(F,new,a,0)
      [1011]:pred(G,computer,n,0)
```

Fig. 6. PROLOG representation of the modification example in (14).

x1	x2 e1
.....
(explosion(x1) + in(e1,x2))	
	crater(x2)
	of(x2,x3)
	foot(x3)
	x2 = 25
	Actor(e1,x1)
	result(e1)

Fig. 7. DRS representation of the sentence in (15).

cases, we rely on sentence simplification to resolve the issue. Otherwise, we would have to go back to the CCG derivations and extract the syntactic arguments.

Another challenge arises from the representation of nominal modifiers. While the event semantic scheme opts for a surface oriented representation, converting these modifiers into predicates, Boxer uses Pivot roles, which capture direct relationships between referents and modifiers, such as between adjectives and nouns. Returning to the sentence depicted in Fig. 2, the adjective *new* modifies *computers* with the relevant PROLOG representation shown in Fig. 6. In this case, we need to extract the arguments of the Pivot and convert the modifier *new* into a predicate that directly applies to the argument *computers*. The result of this conversion is shown in (14).

(14) computers($z1$) \wedge new($z1$)

Units of measurement are treated similarly. The DRS representation for the sentence in (15) is shown in Fig. 7. We can see that it contains both information that the referent is a number (specified by the cardinality information) and that it modifies the noun *crater* while the measurement *foot* is in an *of* relation to the number.

(15) a. The explosion resulted in a 25 foot crater.
b. crater($z1$) \wedge size_in_foot($z1,25$)

Despite the differences described above, we can convert those concepts directly into event semantics due to the consistent and explicit relationship between nominals and their modifiers.

VI. DISCREPANCIES

The basic conversion principles described above allow for an accurate transformation of many of the basic phenomena in sentences with a high degree of consistency. However, there are also discrepancies between the two representations that are less easily reconciled. We describe here the three

x4 x3	x2 x1
.....
(named(x3,zone,org) + thing(x1)))
x3 = x4	of(x2,x3)
named(x4,green,org)	outside(x1,x2)
.....

Fig. 8. DRS representation of *outside of the Green Zone*.

x5 x4 x2	e1 x3 s1 x1
.....
(named(x5,mandari,org) + Theme(e1,x2)))
named(x4,al-sabah,org)	Actor(e1,x1)
interview(x2)	complete(e1)
.....	of(x2,x3)
x5 C x3	x4 C x3
x4 C x3	force(x1)
bct(s1)	Pivot(s1,x1)
.....

Fig. 9. DRS representation of *BCT forces*.

most important issues, namely the treatment of parsing errors, inconsistencies in the Boxer output, and coreference information.

A. Parsing Errors

While the C&C parser is an accurate parser, there are phenomena that are challenging for the parser, especially where Boxer does not have enough information to make correct decisions. To a certain degree, we anticipate such problems, such as coordination, which are challenging for any parser, and handle them in sentence simplification. Other phenomena, however, are more difficult to address.

1) *Named Entity Referents*: The C&C parser has a named entity recognizer [24], whose analyses are also used by Boxer. Named entities are categorized into various categories including geographical (geo), person (per), organization (org), and nam (unknown). This information provides the basis for determining the correct variable type in the event semantic representations (i.e., assigning x for a name or a person and y for a location). However, given the nature of the SYNCOIN data, there are many names that the parser mislabels, or occasionally fails to recognize. For example, *Green Zone* is correctly identified as a named entity but is consistently labeled as an organization rather than a location (see Fig. 8). In Fig. 9, the organization *BCT forces* is parsed as a referent *force* with a modifier (*BCT*) instead of being treated as a single multi-word named entity. We thus cannot always accept the named entity information provided for the correct categorization between x and y variables in the event semantics.

2) *Recognizing Possessive Relationships and Compounds*: Possessive relationship are represented with an *of* relation. This covers both cases where the possessor is nominal and pronominal. Fig. 10 shows the DRS representation for the sentence in (16). In this example, the pronoun *his* is converted into an *of* relation between *meeting* and a male referent.

x5 x4 x2 x1	e1 x3
.....
(meeting(x5) + for(e1,x5)))
of(x5,x4)	Theme(e1,x3)
male(x4)	Actor(e1,x1)
named(x1,khrayesh,per)	use(e1)
x1 = x2	stand-in(x3)
named(x2,omar,per)
.....

Fig. 10. DRS representation of the example in (16).

x4 x3 x2 x1	e1
.....
(card(x3) + Theme(e1,x3)))
of(x3,x4)	Actor(e1,x1)
business(x4)	accept(e1)
of(x3,x2)
named(x2,khrayesh,org)
male(x1)
.....

Fig. 11. DRS representation of the example in (17).

- (16) a. Omar Khrayesh uses a stand-in for his meeting.
b. Omar_Khrayesh(x1) \wedge use(e1,x1,x2) \wedge
stand_in_for(e2,x2,e3) \wedge meeting(e3,x3,x1)

However, the same relation is also used to represent noun-noun compounds. Fig. 11 shows the DRS analysis of the sentence in (17)². Here, the compound *business card* is analyzed as *card of business*.

- (17) a. He accepts Khrayesh's business card.
b. accept(e1,x1,z1) \wedge business_card(z1) \wedge
named(x2,Khrayesh) \wedge possessive(x2,z1)

This parallel treatment of possessives and compounds in Boxer's DRS introduces the need to distinguish between the two usages since they differ in the event semantic representation: In the event semantics, the compound noun is retained (e.g., *business_card* in (17)) while the possessive pronoun is resolved into a possessive relation (e.g., *possessive*).

B. Inconsistencies

1) *Adverbial Modifiers*: In the DRS produced by Boxer, non-temporal modifiers are predominantly categorized in two different ways, as seen for the sentences in (18) and the DRS presented in Fig. 12 and Fig. 13 respectively.

- (18) a. At approximately 1304hrs he appeared.
b. He is only interested in a prosperous Iraq.

For the first sentence, the adverb *approximately* is analyzed as being in a Pivot relation to the time. For the second

²Note that Boxer provides an option to analyze noun-noun compounds using prepositions [25]. However, this analysis introduces additional variation in the semantics-based relations between nouns, introducing an added level of complexity. For this reason, we choose not to utilize this option.

x1	s1 x2 e1
.....
(male(x1) + at(e1,x2))	
.....	1304hrs(x2)
.....	approximate(s1)
.....	Pivot(s1,x2)
.....	Actor(e1,x1)
.....	appear(e1)

Fig. 12. Example of an adverbial pivot

x1	s3 s2 x2 s1
.....
(male(x1) + only(s3))	
.....	Manner(s1,s3)
.....	in(s1,x2)
.....	named(x2,iraq,geo)
.....	prosperous(s2)
.....	Pivot(s2,x2)
.....	Pivot(s1,x1)
.....	interested(s1)

Fig. 13. Example of an adverb of manner

x2 x1	e1 p1
.....
(male(x2) + Topic(e1,p1))	
soldier(x1)	Recipient(e1,x2)
.....	Actor(e1,x1)
.....	tell(e1)
.....	
.....	x3 s1 e2
.....	p1:
.....	on(e2,x3)
.....	tomorrow(x3)
.....	back(s1)
.....	Manner(e2,s1)
.....	Actor(e2,x2)
.....	come(e2)

Fig. 14. Example of a temporal adverb

sentence, *only* as an adverb of manner. Both indicate that there is a type of relationship to a referent although they are being interpreted differently in terms of semantics. The pivot analysis overlaps with adjectival modifiers. The manner analysis overlaps with phrasal verbs, which will be addressed further in section VI-B3.

2) *Temporal Modifiers*: Time expressions are not consistently or accurately captured in every case [12]. This becomes evident in temporal modifiers. For example, the DRSs for the first two sentences in (19), depicted in Fig. 14 and Fig. 15, show that in the first case, the temporal adverb *tomorrow* is analyzed as being in an *on* relation to the event *come* while in the second case, the adverb *now* directly modifies the event *want*. Now, we could argue that the latter is a consequence of having fronted the adverb. However, if we use the adverb in

x3 x1	e1 p1
.....
(person(x3) + now(e1))	
male(x1)	Topic(e1,p1)
.....	Actor(e1,x1)
.....	want(e1)
.....	
.....	e2 x2
.....	p1:
.....	Recipient(e2,x3)
.....	Theme(e2,x2)
.....	Actor(e2,x1)
.....	give(e2)
.....	money(x2)

Fig. 15. Example of the temporal adverb fronted.

x3 x1	e1 p1
.....
(person(x3) + Topic(e1,p1))	
male(x1)	Actor(e1,x1)
.....	want(e1)
.....	
.....	s1 e2 x2
.....	p1:
.....	now(s1)
.....	Manner(e2,s1)
.....	Recipient(e2,x3)
.....	Theme(e2,x2)
.....	Actor(e2,x1)
.....	give(e2)
.....	money(x2)

Fig. 16. Non-fronted example.

sentence final position, it receives the analysis in Fig. 16, in which it is analyzed as an adverb of manner, giving us a third analysis. Note that both versions of this sentence receive the same analysis in the event semantics, as shown in (19-d).

- (19) a. The soldier told him to come back tomorrow.
b. Now he wants to give me money.
c. He wants to give me money now.
d. $\text{now}(e1) \wedge \text{wants}(e1,x1,\text{give}(e2,x1,x2,z1) \wedge \text{money}(z1))$

Furthermore, the representation of specific dates in the SYNCOIN data is not identified as a time referent by Boxer as seen in Fig. 17 for the sentence in (20).

- (20) A meeting on 04/06/10 will work fine.

In the PROLOG representation, the date reference is marked as being a number, but not a time signature. Considering the importance of time in event semantics, the inconsistency in particular of temporal modifiers makes them a weak point in the conversion.

s1 e1 x2 x1
.....
for(e1,s1)
Pivot(s1,x1)
fine(s1)
Actor(e1,x1)
work(e1)
on(x1,x2)
04/06/10(x2)
meeting(x1)

Fig. 17. Example of a date.

x3 x2	p1 x1
.....
named(x2,square,geo)	of(p1,x2)
x2 = x3	off(p1)
named(x3,antar,geo)	
	x6 x5 x4
	p1:

	x1 = x4
	in(x4,x6)
	house(x6)
	cache(x4)
	of(x4,x5)
	weapon(x5)

Fig. 18. Example of *off* as a preposition.

x1	s1 e1
.....
named(x1,htt,org)	off(s1)
	Manner(e1,s1)
	Theme(e1,x1)
	tip(e1)

Fig. 19. Example of *off* in a phrasal verb.

3) *Prepositions*: In the event semantics, phrasal verbs are analyzed as multi-word expressions. Thus, in order to achieve a correct conversion from Boxer to event semantics, we need to be able to detect phrasal verbs as such. We show an example of a standard prepositional use of *off* and its use in a phrasal verb in (21). Their DRS analyses are shown in Fig. 18 and Fig. 19 respectively.

- (21) a. There is a weapons cache in a house off of Antar Square.
b. HTT was tipped off.

In the DRS analyses, both usages of *off* are categorized similarly, once as a postmodifier, and once as a Manner relation, similar to adverbs. However, there is no indication that the latter is part of a phrasal verb. The corresponding analyses in event semantics are shown in (22).

- (22) a. weapons_cache(y1) \wedge in(y1,y2) \wedge house(y2) \wedge

- off(y2,y3) \wedge named(y3,Antar Square)
b. HTT(x1) \wedge tip_off(e1,x1)

C. Coreference

Another major difference between Boxer's DRS and the event semantic representations is that the DRS is mostly an annotation on the sentential level while the event semantics also annotates discourse relations in form of coreference: Any coreferent entity in a text is referenced by the same variable. An example is shown in (23) where the *BCT patrol* is mentioned in two consecutive sentences, both times identified by variable *x2*, even though the surface representation is different.

- (23) a. BCT patrol approached by man promising to reveal 2 additional weapons cache in Dour'a.
b. BCT reports little value in sites but pays man a small amount of cash
c. approach(e7,x4,x2) \wedge named(x2, BCT_patrol) \wedge man(x4) \wedge promise(e8,x4,e9) \wedge reveal(e9,y3) \wedge additional(y3) \wedge count(y3,2) \wedge weapons_cache(y3) \wedge in(y3,y10) \wedge named(y10,Dour'a)
d. named(x2,BCT) \wedge report(e10,x2,little_value(y3) \wedge sites(y3)) \wedge pay(e11,x2,z2) \wedge cash(z2) \wedge small_amount(z2)

The same also holds for event anaphora, for example in (24), where the *detonation* mentioned in the first sentence and the *attack* in the second sentence share the same event variable *e44*.

- (24) a. Their description was passed to an Iraqi who subsequently apprehended them after a second failed attempt to detonate their satchel charge.
...
b. The men detained for failed attack on 02/05/10 at the Soeudi Café, refuse to reveal operational details of the attack and deny being foreign insurgents.
c. pass(e41,x126,z10,x7) \wedge description(z10) \wedge possessive(x6,z10) \wedge iraqi(x7) \wedge apprehend(e42,x7,x6) \wedge after(e42,e43) \wedge attempt(e43,x6,e44) \wedge fail(e44) \wedge detonate(e44,x6,z11) \wedge satchel_charge(z11) \wedge possessive(x6,z11)
d. refuse(e50,x6,e51) \wedge men(x6) \wedge detain_for(e443,x129,x6,e44) \wedge fail(e44) \wedge attack(e44,x6) \wedge on(e44,t4) \wedge date(t4,020510) \wedge at(e44,y4) \wedge named(y4,Soeudi_Café) \wedge reveal(e51,x6,z14) \wedge details(z14,e44) \wedge operational(z14) \wedge deny(e52,x6,foreign(x6) \wedge insurgents(x6))

Boxer does have an option to perform coreference resolution using binding and accommodation theory to resolve the referents of pronouns and definite noun phrases. However, the

TABLE II
FREQUENCY OF PHENOMENA

Pivots	26
Units of measurement	3
Named entity referents	33
Possessive relationships / compounds	28
Adverbial modifiers	9
Temporal modifiers	6
Prepositions	7

module focuses on high precision, thus “definite descriptions and proper names are only linked to previous discourse referents if there is overlap in the DRS-conditions of the antecedent DRS ...” [26]. Additionally, it does not resolve event anaphora. Since, for our downstream application, the fusion algorithm, recall is extremely important, we will need a full coreference resolution integrated into our conversion procedure. This poses additional problems since especially event anaphora is an understudied process [27], [28].

D. Empirical Overview

We have looked at one of the threads in the SYNCOIN data in order to determine how frequent the phenomena are that we have discussed in the previous two sections. We have parsed those sentences using the C&C parser in combination with Boxer and then have inspected the resulting analyses manually to determine the frequency of the individual phenomena. The manual inspection was necessary since Boxer does not handle certain of those phenomena very well, as discussed in section VI. As a consequence of the necessary manual inspection, we chose the shortest thread, which consists of 21 sentences.

The distribution of phenomena is shown in table II. The numbers show that most of these phenomena occur with moderate frequency, on average in every third sentence. The exception are the named entities and the possessives, which occur on average more than once per sentence. These numbers show very clearly that the phenomena are frequent enough to necessitate a specialized treatment.

VII. FUTURE WORK

One of the major challenges in the conversion from Boxer DRS to event semantics is the underspecification and variance of specific phenomena in Boxer’s DRS analyses. In order to ensure a fully automated high quality conversion, we will need to integrate tools and resources, along with machine learning algorithms to help resolve the ambiguities (e.g., to determine variable types). This includes the utilization of additional semantic resources such as PropBank [29] and full coreference to improve categorization. We will also explore the use of clustering to group frequently mislabeled words with words that most closely resemble their contextual behavior.

SYNCOIN data is full of infrequent words, particularly multi-word expressions of people and places, and given the nature of the data on which the C&C parser’s models were trained, this makes it difficult to predict how an unknown word should be represented. Thus, we will investigate domain adaptation methods for all levels: parsing, named entity

recognition, and DRS analysis. This is particularly necessary to resolve distinctions between proper names of people and proper names of places.

VIII. CONCLUSION

We have presented an investigation into the feasibility of converting from DRS to event semantics, demonstrating that it is a non-trivial task. We have started the conversion process using the PROLOG representation from Boxer to convert basic sentences via regular expressions that identify referents and relations to an event semantic representation. We have also highlighted areas that prove to be problematic in the conversion and require further exploration. We will expand the system beyond basic sentences and incorporate machine learning techniques and coreference resolution to increase the accuracy.

ACKNOWLEDGEMENT

This work is based on research supported by the U.S. Office of Naval Research (ONR) via grant #N00014-10-1-0140.

REFERENCES

- [1] X. Carreras and L. Màrquez, “Introduction to the conll-2004 shared task: Semantic role labeling,” in *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-04)*, Boston, MA, 2004, pp. 89–97.
- [2] X. Carreras and L. Màrquez, “Introduction to the CoNLL-2005 shared task: Semantic role labeling,” in *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-05)*, Ann Arbor, MI, 2005, pp. 152–164.
- [3] L. Màrquez, X. Carreras, K. Litkowski, and S. Stevenson, “Semantic role labeling: An introduction to the special issue,” *Computational Linguistics*, vol. 34, no. 2, pp. 145–159, 2008.
- [4] T. Wickramaratne, K. Premaratne, M. Murthi, M. Scheutz, S. Kübler, and M. Pravia, “Belief theoretic methods for soft and hard data fusion,” in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Prague, Czech Republic, 2011.
- [5] R. Nunez, T. Wickramaratne, K. Premaratne, M. Murthi, S. Kübler, M. Scheutz, and M. Pravia, “Credibility assessment and inference for fusion of hard and soft information,” in *Proceedings of the International Conference on Cross-Cultural Decision Making (HSBC FOCUS)*, San Francisco, CA, 2012.
- [6] A. Cahill, M. McCarthy, J. van Genabith, and A. Way, “Quasi-logical forms from F-structures for the Penn Treebank,” in *Proceedings of the Fifth International Workshop on Computational Semantics*, Tilburg, The Netherlands, 2003.
- [7] C. Gardent and Y. Parmentier, “SemTAG: A platform for specifying Tree Adjoining Grammars and performing TAG-based semantic construction,” in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, 2007, pp. 13–16.
- [8] M. Steedman, *The Syntactic Process*. Cambridge, MA: MIT Press, 2001.
- [9] S. Clark and J. Curran, “Formalism-independent parser evaluation with CCG and DepBank,” in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL)*, Prague, Czech Republic, 2007.
- [10] —, “Wide-coverage efficient statistical parsing with CCG and log-linear models,” *Computational Linguistics*, vol. 33, no. 4, pp. 493–552, 2007.
- [11] J. R. Curran, S. Clark, and J. Bos, “Linguistically motivated large-scale nlp with c&c and boxer,” in *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ser. ACL ’07. Prague, Czech Republic: ACL, 2007, pp. 33–36.
- [12] J. Bos, “Wide-coverage semantic analysis with boxer,” in *Semantics in Text Processing. STEP 2008 Conference Proceedings*, J. Bos and R. Delmonte, Eds. College Publications, 2008, pp. 277–286.

- [13] —, “Open-domain semantic parsing with boxer,” in *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, Vilnius, Lithuania, May 2015, pp. 301–304. [Online]. Available: <http://www.let.rug.nl/bos/pubs/Bos2015NoDaLiDa.pdf>
- [14] H. Kamp, “A theory of truth and semantic representation,” in *Formal Methods in the Study of Language*, J. Groenendijk, T. Janssen, and M. Stokhof, Eds. Amsterdam: Mathematical Centre, 1981, pp. 277–322.
- [15] H. Kamp and U. Reyle, *From Discourse to Logic: An Introduction to Modeltheoretic Semantic of Natural Language, Formal Logic and DRT*. Dordrecht: Kluwer, 1993.
- [16] L. Zettlemoyer and M. Collins, “Learning context-dependent mappings from sentences to logical form,” 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*, Suntec, Singapore, 2009, pp. 976–984.
- [17] Y. Artzi, K. Lee, and L. Zettlemoyer, “Broad-coverage CCG semantic parsing with AMR,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, September 2015, pp. 1699–1710.
- [18] R. Dabarera, R. Nunez, K. Premaratne, and M. Murthi, “Dynamics of belief theoretic agent opinions under bounded confidence,” in *International Conference on Information Fusion (FUSION)*, Salamanca, Spain, 2014.
- [19] R. Nunez, M. Murthi, and K. Premaratne, “Efficient computation of DS-based uncertain logic operations and its application to hard and soft data fusion,” in *International Conference on Information Fusion (FUSION)*, Salamanca, Spain, 2014.
- [20] J. L. Graham, D. L. Hall, and J. Rimland, “A coin-inspired synthetic dataset for qualitative evaluation of hard and soft fusion systems,” in *Proceedings of the 14th International Conference on Information Fusion (FUSION)*. Chicago, Illinois: IEEE, July 2011, pp. 1–8. [Online]. Available: https://www.researchgate.net/profile/David_Hall20/publication/261344900_A_COIN-inspired_synthetic_dataset_for_qualitative_evaluation_of_hard_and_soft_fusion_systems/links/542165450cf203f155c6693c.pdf
- [21] D. Davidson, *Inquiries into Truth and Interpretation*. Oxford University Press, 1984.
- [22] J. Hockenmaier and M. Steedman, “CCGbank: A corpus of CCG derivations and dependency structures extracted from the Penn Treebank,” *Computational Linguistics*, vol. 33, no. 3, pp. 355–396, 2007.
- [23] L. Hirschman and N. Chinchor, “MUC-7 coreference task definition,” 1997, message Understanding Conference. [Online]. Available: http://www-nlpir.nist.gov/related_projects/muc/proceedings/co_task.html
- [24] J. R. Curran and S. Clark, “Language independent ner using a maximum entropy tagger,” in *Proceedings of CoNLL-03*, Edmonton, Canada, 2003, pp. 164–167.
- [25] J. Bos and M. Nissim, “Uncovering noun-noun compound relations by gamification,” in *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, Vilnius, Lithuania, May 2015, pp. 251–255. [Online]. Available: <http://www.let.rug.nl/bos/pubs/BosNissim2015NoDaLiDa.pdf>
- [26] J. Bos, “Implementing the binding and accommodation theory for anaphora resolution and presupposition projection,” *Computational Linguistics*, vol. 29, no. 2, pp. 179–210, 2003.
- [27] S. Pradhan, A. Moschitti, N. Xue, O. Uryupina, and Y. Zhang, “CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes,” in *Proceedings of the Sixteenth Conference on Computational Natural Language Learning (CoNLL 2012)*, Jeju, Korea, 2012.
- [28] S. Pradhan, L. Ramshaw, M. Marcus, M. Palmer, R. Weischedel, and N. Xue, “CoNLL-2011 shared task: Modeling unrestricted coreference in OntoNotes,” in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning (CoNLL 2011)*, Portland, OR, 2011.
- [29] M. Palmer, D. Gildea, and P. Kingsbury, “The proposition bank: An annotated corpus of semantic roles,” *Computational Linguistics*, vol. 31, no. 1, pp. 71–106, 2005.