From Narrative Text to VerbNet-Based DRSes: System Text2DRS

MS Project Report

Gang Ling

University of Nebraska Omaha

gling@unomaha.edu

Advisor: Yuliya Lierler
May, 2018

Computational linguists have long studied various logic forms for capturing essential semantic information carried by narratives. Among these, discourse representation structure (DRS) form is designed to acquire the entities, entities' properties, events, and event properties. This project report describes a system called Text2DRS that takes English narrative as an input and outputs a DRSes output in Neo-Davidsonian style.

1 Introduction

In this work we understand a narrative text to be a sequence of sentences with action verbs so that the succession of events is given in chronological order. Linguistically, action verbs are words that convey physical or mental actions, such as grab, or go. From the narrative text, readers can reproduce the sequence of events with related information in their minds. This process is not hard for a person but is a challenge for an artificial intelligence (A.I.) system agent to generate the same results automatically. Consider a simple three sentences discourse:

John grabbed the suitcase. (1)
John travelled to the hallway. (2)
Sandra journeyed to the hallway. (3)

From the first sentence, a state-of-the-art A.I. agent can retrieve the direct information such as

grab is an action verb
John is an agent entity of action grab
suitcase is a object entity of action grab

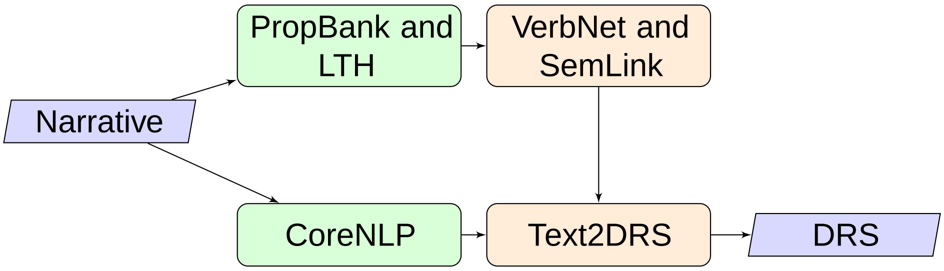

Figure 1: NLP resources

Report Outline Figure 1 presents a schematic architecture of the Text2Drs system. Note that lexical resources as PropBank [Palmer et al., 2005], VerbNet [Kipper-Schuler, 2005, Palmer, 2006], and SemLink [Bonial et al., 2013] form the important components of Text2DRS. Also, several modern natural language processing (NLP) tools such as LTH [Johansson and Nugues, 2007] and CoreNLP [Manning et al., 2014] are part of the system. Also such . We start this report by providing key details on these NLP tools and resources. We then discuss the intuitions behind the discourse representation structures [Kamp and Reyle, 1993] that are essential in understanding the output produced by the Text2DRS tool. In Section 3, we talk about the system architecture of Text2DRS as well as the implementation challenges and solutions. Then, we present the system evaluation results. At last, we conclude with future work discussion in Section 5.

2 Modern NLP Resources

2.1 PropBank

In English, a sentence contains three main components: subjects, verbs, and objects. More importantly, verbs stand for predicates that are used to describe events occurring in our world. These predicates take arguments which are participants of events. By processing verbs from a sentence, an A.I. agent can understand which events happen in a world around the agent.

PropBank [Palmer et al., 2005] is a Verb Lexical Resource that systematizes knowledge about verbs with respect to their predicate-argument or, in another word, event-participant structure. Consider \frame" for verb grab:

id:	grab.01
Semantic roles:	Arg0-PAG: grabber
	Arg1-PPT: entity grabbed

In this frame, Arg0 and Arg1 are called semantic roles, and grabber and entity grabbed are descriptions of these particular semantic roles. We can use the Propbank frames to annotate sentences with verb grab (and others) in a systematic fashion. We can identify event participants and their roles: this task is called \Semantic Role Labeling". For example, a table below presents PropBank semantic role labeling for sentence (1).

John	grabbed	the suitcase
Arg0	grab.01	Arg1

Similarly, we can generate a PorpBank semantic role labeling table for sentences (2) and (3):

John	travelled	to the hallway
Arg0	travel.01	Arg4

Sandra	journeyed	to the hallway
Arg0	journey.01	Arg2

2.2 VerbNet

However, PropBank organizes each predicate individually without grouping similar meaning verbs together. Also, the semantic roles of the predicates are not always consistent across similar verbs. As a result, the A.I. agent needs to include more complex rule handling at each case separately. VerbNet [

to be more suitable in this case. But without the emotional meaning, these three verb classes represent a similar action. Since we are interested in verbs and events in this project, we can pick either one of three verb classes and use it in the nal output. We will discuss this decision-making mechanism in the system architecture section. Right now, let us use the verb class *steal*. The same idea applies to other two sentences in our running example.

Sentence (2):

John	travelled	to the hallway
Arg0	travel.01	Arg4
Theme	run-51.3.2	Location

Sentence (3):

Sandra	journeyed	to the hallway
Arg0	journey.01	Arg2
Theme	run-51.3.2	Location

From the results for sentences (2) and (3), we nd that distinct PropBank predicates *travel.01* and *journey.01*

```
         <role pb-arg="2" vn-theta="Location" />
     </argmap>
</predicate>



    </predicate>
        <predicate lemma="grab">
        <argmap pb-roleset="grab.01" vn-class="10.5-1">
            <role pb-arg="0" vn-theta="Agent" />
        <role pb-arg="1" vn-theta="Theme" />
        </argmap>
    <argmap pb-5(p)1(b-arg="2")-515(vn-theta="Locs3")-51te>
```

event referent and its corresponding VerbNet class. In this example, both verb *travel* and *journey* are mapped to the verb class *run-51.3.2-1*, and verb *grab* is mapped to the verb class *steal-10.5-1* . An eventArgument relation presents information about events. For instance, eventArgument(e2,location,r3) says that entity *r3* (that has a property of being a \hallway") plays a thematic role \location" of event *e2* (that belongs to VerbNet class *run-51.3.2-1*). The eventTime indicates the time order of the occurred events in the narrative. In the action language, we consider event begin at time zero. Therefore, the rst event *e1* (represent verb \grab") starts at \0".

r1 r2 r3 r4 e1 e2 e3
entity(r1) entity(r2) entity(r3) entity(r4)
property(r1, John) property(r2, suitcase)
property(r3, hallway) property(r4, Sandra)
event(e1) event(e2) event(e3)
eventType(e1, steal-10.5-1) eventTime(e1, 0)
eventArgument(e1, agent, r1) eventArgument(e1, theme, r2)
eventType(e2, run-51.3.2-1) eventTime(e2, 1)
eventArgument(e2, theme, r1) eventArgument(e2, location, r3)
eventType(e3, run-51.3.2-1) eventTime(e3, 2)
eventArgument(e3, theme, r4) eventArgument(e3, location, r3)

Table 1: *DRS for the given narrative*

2.5 LTH

In the tool chain of our Text2DRS implementation, the LTH system and Stanford CoreNLP system are used as pre-processing system components. LTH [Johansson and Nugues, 2007] is a semantic role labeler for unrestricted text in English that uses predicates and semantic roles from PropBank. For sentences (1), (2), and (3), LTH produces the following outputs:

```
[A0 John][V (grab.01) grabbed ][A1 the suitcase]         (i)
[A0 John][V (travel.01) travelled][A4 to the hallway]    (ii)
[A0 Sandra][V (journey.01) journeyed][A2 to the hallway] (iii)
```

These outputs are annotated using the semantic roles of the predicates *grab.01*, *travel.01*, and *journey.01* as de ned in their PropBank frames schemas. Recall the listed frame schema of the predicates grab.01 on Section 2.1. The frame schemas of the predicate travel.01 and journey.01 follow:

```
travel.01: the act of moving to
Arg0-PPT: traveller
Arg1-LOC: location or path
Arg2-DIR: start point
Arg4-GOL: destination
```

```
journey.01: the act of moving to
Arg0-PAG: traveller
Arg2-LOC: location or path
```

2.6 CoreNLP

The Stanford CoreNLP system [Manning et al., 2014] provides a set of NLP tools including a coreference resolution system. Text2DRS utilizes the coreference resolution function from CoreNLP system to process a given narrative.

CoreNLP detects four mentioned entities in the discourse: *John, suitcase, hallway*, and *Sandra*. The system also recognizes that the entity *John* in the sentence (2) is the same entity that appears in the sentence (1). Similarly, the noun entity *the hallway* in the sentence (3) is the same noun entity that is used in the sentence (2). This process is called coreference recognition. As a result, CoreNLP identi es four unique entities in this discourse.

3 System's Architecture, Challenges, and Output

As the outcome of this project, we developed and implemented a DRS generating system, named Text2DRS. It contains four main parts: LTH, CoreNLP, SemLink, and itself. In a nutshell, rst LTH and CoreNLP preprocess a given narrative. Then, Text2DRS reads the outputs from these two systems and generates DRS using the mapping data from SemLink.

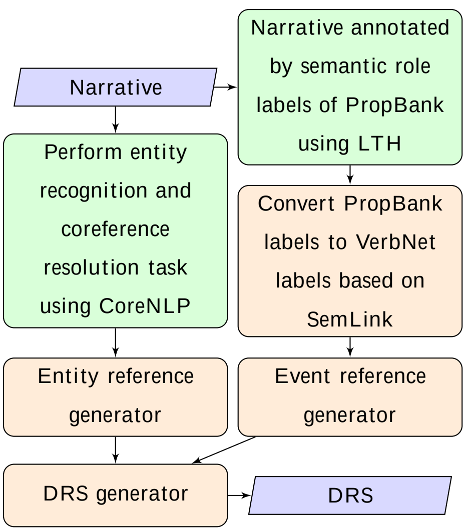

Figure 2: *System architecture*

class number along with the respective thematic roles. However, sometime SemLink maps one PropBank predicate into multiple VerbNet classes. In this case, we pick the first verb class from the data list and use it in the final output. After entity reference generator and event reference generator complete processing of the data, the DRS generator merges the data and outputs the DRS for the given narrative.

Challenges

```
property(r1, "John"). property(r2, "suitcase").
property(r3, "hallway"). property(r4, "Sandra").

event(e1). event(e2). event(e3).

eventType(e1, "10.5-1").
eventType(e2, "51.3.2-1").
eventType(e3, "51.3.2-1").

eventTime(e1, 0). eventTime(e2, 1). eventTime(e3, 2).

eventArgument(e1, "Agent", r1).
eventArgument(e1, "Theme", r2).
eventArgument(e2, "Theme", r1).
eventArgument(e2, "Location", r3).
eventArgument(e3, "Theme", r4).
eventArgument(e3, "Location", r3).
```

4 Evaluation

We evaluate system Text2DRS on sample narratives from two datasets (i) Facebook's bAbI [Weston et al., 2015] collection and (ii) the ROCStories 2017 [Mostafazadeh et al., 2017] collection. We included all considered test cases in the appendix of this paper. Totally, we consider 20 test narratives (ten bAbI cases and ten ROCStories cases) that contain 177 sentences. Following is one of the ten bAbI test cases:

```
John travelled to the hallway.    Mary journeyed to the bathroom.
Daniel went back to the bathroom. John moved to the bedroom.
John went to the hallway.         Sandra journeyed to the kitchen.
Sandra travelled to the hallway.  John went to the garden.
Sandra went back to the bathroom. Sandra moved to the kitchen.
```

Here is an example of the ten ROCStories narrative test case:

```
Tom had a very short temper.
One day a guest made him very angry.
He punched a hole in the wall of his house.
Tom's guest became afraid and left quickly.
Tom sat on his couch filled with regret about his actions.
```

The evaluation is divided into four parts

1. including entity recognition veri cation,

2. entity coreference correctness checking,
3. event recognition verification, and
4. event annotation correctness checking.

4.1 Entity Recognition Verification

In the entity recognition verification, we identify all the entities in the testing cases manually and compare them with entries in generated DRSes.

Entity recognition	Manually identified	Text2DRS identified
Facebook bAbI	264	264
ROCStories 2017	161	161

4.2 Entity Coreference Correctness Checking

In the entity coreference correctness checking, we identify manually entities in each testing case and assign entity referent to the unique entities. Then we compare the numbers of unique entities (entity referents) from the manually created reports with Text2DRS's outputs.

Entity coreference correctness	Manually identified	Text2DRS generates
Facebook bAbI	79	80
ROCStories 2017	103	132

These results show that CoreNLP has an excellent performance of entity coreference recognition when it processes bAbI narratives. The only incorrect case occurred in the Narrative 2 of bAbI (see Section 6). CoreNLP is unable to recognize that two entities *suitcase* from both sentences are the same. CoreNLP drops accuracy when it processes ROCStories narratives. Narrative ROC-04 (Section 6) is an example of a test case where CoreNLP is unable to resolve coreference properly.

4.3 Event Recognition Verification

In the event recognition verification, we repeat the same process as evaluating entity recognition.

Event recognition verification	Manually identified	Text2DRS identified
Facebook bAbI	127	127
ROCStories 2017	91	91

From the result table, we conclude that LTH locates all the events, and Text2DRS can assign event referents to them.

4.4 Event Annotation Correctness Checking

In the event annotation correctness checking, we collect used VerbNet verb-class data entries. We go through each eventArgument condition and compare the annotation with verb class data. The table below summarizes the results.

Event annotation correctness	Text2DRS generates	Incorrect	Accuracy
Facebook bAbI	264	27	89%
ROCStories 2017	199	67	66%

If we nd an incorrect annotation during our evaluation process, we take a further step by looking up the corresponding VerbNetsrl le, LTH outputs, and SemLink mapping entries. In this way, we can categories mistakes. For the case of bAbI narratives, all of the mistakes were due to a single fact: SemLink has a missing mapping entry for predicate *go.01* whereas the verb *go* occurred 27 times among the test cases. Similarly, SemLink has 67 missing mapping entries during the eventArgument generating processes within test cases in ROCStroies.

5 Future Work

One of important future work directions is adding missing mappings into SemLink after we test more narrative cases. The correctness of event annotations heavily depends on the mapping data from the SemLink.

Another essential work is to improve system performance and extend output le format options. Currently, Text2DRS runs LTH and CoreNLP by Java command line, and both systems take time to load their modules. For example, CoreNLP takes 18 seconds on average to setup its pipeline, but it only uses two seconds on average to process an input narrative. We intend to implement server-client based architecture within Text2DRS to keep CoreNLP running as a process in the operating system. The idea applies to the LTH system.

Since LTH is no longer maintained, we intend to incorporate few other state-of-the-art semantic role labelers, such as Neural-dep-srl [Marcheggiani and Titov, 2017] or PathLSTM [Roth and Lapata, 2016] in the future. We anticipate that incorporating a di erent semantic role labeler will be a simple task. Indeed, Text2DRS utilizes the CONLL-2008 standard output of LTH. Provided that more modern mentioned systems produce the same output makes a transition to a new semantic role labeler seamless.

Last but not least, we plan to utilize the Text2DRS system in a larger framework capable of reasoning about events.

Acknowledgements

I would rst like to thank my project advisor Dr. Yuliya Lierler of the Computer Science Department at the University of Nebraska Omaha. I can always nd help from her when I had a question about my research or writing. I would also like to thank Dr. Parvathi Chundi and Dr. Ryan Schuetzler for serving on my project committees. It is a great honor to present my project report to them and gather feedback from them. Finally, I would like to thank the support from my family and my friends. The help from them is immeasurable through my studies and life.

References

Claire Bonial, Kevin Stowe, and Martha Palmer. *SemLink*, 2013. https://verbs.colorado.edu/semlink/ [Accessed: 2017].

Johan Bos. Wide-coverage semantic analysis with boxer. In *Proceedings of the 2008 Conference on Semantics in Text Processing*, STEP '08, pages 277{286, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1626481.1626503.

Richard Johansson and Pierre Nugues. *Language Technology at LTH*, 2007. http://nlp.cs.lth.se/ [Accessed: 2017].

Hans Kamp and Uwe Reyle. *From discourse to logic*, volume 1,2. Kluwer, 1993.

Karin Kipper-Schuler. *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. PhD thesis, University of Pennsylvania, 2005.

Yuliya Lierler, Daniela Inclezan, and Michael Gelfond. Action languages and question answering. In *Proceeding of the 12th International Conference on Computational Se-agniversity o1lor27.7a. 27.7cttN80 Geo1lor27.7a.cnukse6(o.u1lor27.7n)]TJ.y353 Td [(Y7.7cETqcttN8*

Martha Palmer, Daniel Gildea, and Paul Kingsbury. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71{106, March 2005. ISSN 0891-2017. doi: 10.1162/0891201053630264. URL http://dx.doi.org/10.1162/0891201053630264.

Michael Roth and Mirella Lapata. Neural semantic role labelling with dependency path embeddings. In *In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016. URL https://arxiv.org/pdf/1605.07515.pdf.

Daniel moved to the garden.
Mary moved to the hallway.

Daniel grabbed the football.
Mary went to the kitchen.

11. ROC_01:
    Tom had a very short temper.
    One day a guest made him very angry.
    He punched a hole in the wall of his house.
    Tom's guest became afraid and left quickly.
    Tom sat on his couch filled with regret about his actions.

12. ROC_02:
    Melody's parents surprised her with a trip to the big aquarium.
    Melody took a nap during the two hour car ride to the aquarium.
    When they arrived, Melody was energetic and excited.
    At the aquarium Melody saw sharks, tropical fish and many others.
    After five hours at the aquarium, Melody and her family drove home.

13. ROC_03:
    The math teacher announced a pop quiz as class began.
    While some students complained, he began passing out the quiz.
    I took out my pencil and began to work.
    About 5 minutes later, I finished.
    I stood up feeling confident and turned it in.

14. ROC_04:
    Robbie was competing in a cross country meet.
    He was halfway through when his leg cramped up.
    Robbie wasn't sure he could go on.
    He stopped for a minute and stretched his bad leg.
    Robbie began to run again and finished the race in second place.

15. ROC_05:
    When I was 12 years old, my dad got angry and kicked me aggressively.
    Afterward, I became very ill, and tasted something metallic.
    I went to a doctor, and was informed that one of my kidneys was dead.
    Ever since, I've had swelling and hypertension.
    I started taking medications to combat the symptoms at 13.

16. ROC_06:
    David noticed he had put on a lot of weight recently.
    He examined his habits to try and figure out the reason.
    He realized he'd been eating too much fast food lately.
    He stopped going to burger places and started a vegetarian diet.
    After a few weeks, he started to feel much better.

17. ROC_07:
    Soren ran through the airport, pulling her bags behind her.
    The female voice above her announced final boarding to Soren's fligh.
    She yelled for them to wait as she neared her gate, waving her arms.
    The attendant at the desk gave Soren a sad, sympathetic look.
    Nearly out of breath, Soren presented her pass and boarded the plane.

18. ROC_08:
    Karl was a good baseball player in his youth.

As a middle-aged man, he joined an adult baseball team.
Karl is very competitive and will do anything to win.
One day, he slid into second base to beat a throw and hurt his knee.
The injury made Karl realize that he wasn't young anymore.

19. ROC_09:
Ken put a bottle of beer in the freezer.
He heard a popping noise.
He looked in the freezer and saw the bottle had burst.
He didn't want to wait for another beer to get cold.
He drank a warm beer instead.

20. ROC_10:
Justin was terrified of dogs.
His girlfriend had a dog and he wanted to feel comfortable with it.
Justin went to therapy to help him get over his fear.
After a few months of therapy Justin felt better around dogs.
Justin then moved in with his girlfriend and her dog.