From Event to Story Understanding

by

Nasrin Mostafazadeh

Submitted in Partial Fulfillment

of the

Requirements for the Degree

Doctor of Philosophy

Supervised by

Professor James F. Allen

Department of Computer Science
Arts, Sciences and Engineering
Edmund A. Hajim School of Engineering and Applied Sciences

University of Rochester
Rochester, New York

2017

Table of Contents

Biographical Sketch

Nasrin Mostafazadeh was born in Tabriz, Iran in 1990. She received her Bachelor of Science's degree from the Computer Engineering department at Sharif University of Technology in 2012 where she was a member of Natural Language Processing lab. She started her PhD studies at the University of Rochester under Professor James F. Allen working on natural language understanding. She received her Master of Science degree in Computer Science in 2014 from University of Rochester.She spent summer 2014 at Google NYC working on Sentiment Analysis. Nasrin spent summer 2015, spring 2016, and summer 2016 at Microsoft Research working on vision and language and language generation. Nasrin's PhD research resulted in the following publications:

Mostafazadeh, Nasrin, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016a. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of NAACL HLT*. Association for Computational Linguistics, San Diego, California

Mostafazadeh, Nasrin, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. 2017. Lsdsem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51. Association for Computational Linguistics, Valencia, Spain

Mostafazadeh, Nasrin, Lucy Vanderwende, Wen-tau Yih, Pushmeet Kohli, and James Allen. 2016d. Story cloze evaluator: Vector space representation evaluation by predicting what happens next. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 24–29. Association for Computational Linguistics, Berlin, Germany

Mostafazadeh, Nasrin, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. 2016c. Generating natural questions about an image. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1802–1813. Association for Computational Linguistics, Berlin, Germany

Mostafazadeh, Nasrin, Alyson Grealish, Nathanael Chambers, James Allen, and Lucy Vanderwende. 2016b. Caters: Causal and temporal relation scheme for semantic annotation of event structures. In *Proceedings of the Fourth Workshop on Events*, pages 51–61. Association for Computational Linguistics, San Diego, California

Huang, Ting-Hao (Kenneth), Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, C. Lawrence Zitnick, Devi Parikh, Lucy Vanderwende, Michel Galley, and Margaret Mitchell. 2016. Visual storytelling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1233–1239. Association for Computational Linguistics, San Diego, California

Ferraro, Francis, Nasrin Mostafazadeh, Ting-Hao Huang, Lucy Vanderwende, Jacob Devlin, Michel Galley, and Margaret Mitchell. 2015. A survey of current datasets for vision and language research. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 207–213. Association for Computational Linguistics, Lisbon, Portugal

Llorens, Hector, Nathanael Chambers, Naushad UzZaman, Nasrin Mostafazadeh, James Allen, and James Pustejovsky. 2015. Semeval-2015 task 5: Qa tempeval - evaluating temporal information understanding with question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 792–800. Association for Computational Linguistics, Denver, Colorado

Mostafazadeh, Nasrin and James F. Allen. 2015. Learning semantically rich event inference rules using definition of verbs. In *Computational Linguistics and Intelligent Text Processing - 16th International Conference, CICLing 2015, Cairo, Egypt, April 14-20, 2015, Proceedings, Part I*, pages 402–416

Abstract

Building systems that have natural language understanding capabilities has been one of the oldest and the most challenging pursuits in AI. In this thesis we present our research on modeling language in terms of 'events' and how they interact with each other in time, mainly in the domain of stories.

Deep language understanding, which enables inference and commonsense reasoning, requires systems that have large amounts of knowledge which would enable them to connect surface language to the concepts of the world. A part of our work concerns developing approaches for learning semantically rich knowledge bases on events. First, we present an approach to automatically acquire conceptual knowledge about events in the form of inference rules, which can enable commonsense reasoning. We show that the acquired knowledge is precise and informative which can be employed in different NLP tasks.

Learning stereotypical structure of related events, in the form of narrative structures or scripts, has been one of the major goals in AI. The research on narrative understanding has been hindered by the lack of a proper evaluation framework. We address this problem by introducing a new framework for evaluating story understanding and script learning: the 'Story Cloze Test (SCT). In this test, the system is posed with a short four-sentence narrative context along with two alternative endings to the story, and is tasked with choosing the right ending. Along with the SCT, We have worked on developing the ROCStories corpus of about 100K commonsense short stories, which enables building models for story understanding and story generation. We present various models and baselines for tackling the task of SCT and show that human can perform with an accuracy of 100%.

One prerequisite for understanding and proper modeling of events and their interactions is to develop a comprehensive semantic framework for representing their variety of relations. We introduce 'Causal and Temporal Relation Scheme (CaTeRS)' which is a rich semantic representation for event structures, with an emphasis on the domain of stories. The impact of the SCT and the ROCStories project goes beyond this thesis, where numerous teams and individuals across academia and industry have been using the evaluation framework and the dataset for a variety of purposes. We hope that the methods and the resources presented in this thesis will spur further research on building systems that can effectively model eventful context, understand, and generate logically-sound stories.

Contributors and Funding Sources

This material is based upon work supported by the Office of Naval Research awards SRI International Awards #25546 & 000678, DOD/NAVY/ONR Award #N00014-11-1-0417. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of above named organizations. The committee for this thesis consists of Professor James Allen (Advisor) from the Computer Science Department, Professor Lenhart Schubert from the Computer Science Department, Professor Daniel Gildea from the Computer Science Department, Professor Henry Kautz from the Computer Science Department and Professor Gregory Calrson from the Linguistics Department.

List of Tables

List of Figures

1 Introduction

(1) **Linda** woke up this morning. While getting ready to go to work, **she** slipped on the bathroom floor and dislocated **her** shoulder. **She** was in deep pain and had to call for paramedics. They shortly arrived at **Linda's** apartment and took **her** to the emergency room.

In order for the system to deeply understand this story, i.e., to make further inferences, it has to do the following:

- Extract all the events, e.g., 'slipped [on the floor]', 'dislocated [her shoulder]', 'called [for paramedics]')
- Extract all the actors in the story and connect them to their corresponding events by following the coreference chains (shown in boldface and underline in the above text)
- Infer the semantic relation between the events, e.g., 'slipped [on the floor]' *caused* 'dislocated [her shoulder]' and 'dislocated [her shoulder]' *caused* 'called [for paramedics]'

Now imagine the system is posed with the following alternative endings to the above story:

- **She** received a proper treatment and felt much better.
- **She** refused to get any treatment.

It is clear that in order for a system to choose the right ending it has to not only deeply understand the story 1, but also perform reasoning and use its commonsense knowledge to find out which of the two events 'received [treatment]' or 'refused [to get treatment]' can temporally and causally follow the given context, all of which are very challenging tasks in NLP.

This thesis describes research on modeling events and their semantic relations, with a major focus on acquiring commonsense knowledge and story understanding. In Chapter 2, we provide the background required for understanding events and the existing natural language inference frameworks. Deep language understanding, which enables inference, requires systems that have large amounts of knowledge enabling them to connect natural language to the concepts of the world. In Chapter 3, we present a novel attempt to automatically acquire conceptual knowledge about events in the form of inference rules by reading verb definitions. We learn semantically rich inference rules which can be actively chained together in order to provide deeper understanding of conceptual events. We also show that the acquired knowledge is precise and informative which can be potentially employed in different NLP tasks which require language understanding.

Representation and learning of commonsense knowledge is one of the foundational problems in the quest to enable deep language understanding. This issue is particularly challenging for understanding casual and correlational relationships between events.

While this topic has received a lot of interest in the NLP community, research has been hindered by the lack of a proper evaluation framework. In Chapter 4, we address this problem with a new framework for evaluating story understanding and script learning: the ‘Story Cloze Test. This test requires a system to choose the correct ending to a four-sentence story. We created a new corpus of about 100k five-sentence commonsense stories, ROCStories, to enable this evaluation. This corpus is unique in two ways: (1) it captures a rich set of causal and temporal commonsense relations between daily events, and (2) it is a high quality collection of everyday life stories that can also be used for story generation. Experimental evaluation shows that a host of baselines and state-of-the-art models based on shallow language understanding struggle to achieve a high score on the Story Cloze Test. We discuss these implications for script and story learning, and offer suggestions for deeper language understanding.

Learning commonsense causal and temporal relation between events is one of the major steps towards deeper language understanding. This is even more crucial for understanding stories and script learning. A prerequisite for learning scripts is a semantic framework which enables capturing rich event structures. In Chapter 5, we introduce a novel semantic framework, called Causal and Temporal Relation Scheme (CaTeRS), which is unique in simultaneously capturing a comprehensive set of temporal and causal relations between events. By annotating a total of 1,600 sentences in the context of 320 five-sentence short stories, we demonstrate that these stories are indeed full of causal and temporal relations. Furthermore, we show that the CaTeRS annotation scheme enables high inter-annotator agreement for broad-coverage event entity annotation and moderate agreement on semantic link annotation.

Last but not least, in Chapter 6, we present our analysis on the first shared task on the Story Cloze Test as the challenge and outline our vision for the forthcoming challenges. Finally, we conclude this work and sketch our future work in Chapter 7.

2 Background

2.1 What Is an Event?

The notion of 'Event' is shared among many areas within NLP, each of which define it differently. Here we introduce the main definition and categorization for events, which is eventually used as a basis for what we mean by 'Event' in the upcoming Chapter.

Events as Verbs and Verb Nominalizations

Perhaps the simplest definition for an event is the following:

(2) An event is a verb together with its set of arguments.

Definition 2 can be simply extended to include verb nominalizations, such as 'explosion' which is nominalization of the verb 'explode'. Verbs denote events that take place in time, hence it is crucial to differentiate the verbs according to the way their corresponding events take place in time. This is mainly addressed by studying aspectual classes of verbs.

Aspect has been traditionally known to refer to “different ways of viewing the internal temporal organization of a situation”[Comrie, 1976]. That is, ‘aspects’ mainly distinguish between different ways of viewing the internal temporal structure of the same situation (event, process, or state). This is unlike the situation-external temporal organization, which is represented by tenses. The main aspectual distinction is made between ‘States’ vs. ‘Events’ [Levin, 2000], defined as follows:

- Events: involve **“change”**. This class includes result verbs and events which are denoted by a manner, .e.g., jog, walk, eat, arrive, reach, wake up, dress.
- States: do not involve **“change”**: This class includes verbs which mainly denote states of the world, e.g., love, believe, be mad, know.

‘Events’ are classified into ‘Durative’ vs. ‘Punctual’, each of which are sub-divided as follows:

- Durative: Having an interval as the duration. This class is divided into the following classes:
    - Activities: These are verbs denoting events that have duration, but do not have an inherent temporal endpoint. For example, the verbs in the sentences ‘Sam swam’, ‘Raj wiped the table’, and ‘Mike drank the milk’ denote activities.
    - Accomplishments: These are verbs denoting events that have duration and an inherent temporal endpoint. The endpoint is a result state, also called ‘culmination’ or ‘telos’. For example, the verbs in the sentences ‘Sam drew a painting’, ‘Jennifer removed the glass’, and ‘Eileen emptied the trash’ are accomplishments.

- Punctual: instantaneous or non-durative
  - Semelfactives: This class resembles activities, i.e., these verbs describe events that are instantaneous and they have no end result state that necessarily follows. For example, the verbs in the sentences ‘Cam hit the ball’, ‘She hopped’, and ‘He winked at her’ are Semelfactives.
  - Achievements: There are verbs that denote the event that describe the moment at which there is a transition to a result state. Hence, they resemble to accomplishments in being defined by an end result state. For example, the verbs in the sentences ‘The room exploded’, ‘Melissa broke the glasses’, and ‘The guests arrived late’ are achievements.

Events Beyond Verbs

Event is mainly used as a term referring to any situation that can happen, occur, or hold. More formally, an event can be defined as follows:

(3) An event is any **situation** (including a process or state) that happens, occurs, or holds to be true or false during some time point or time interval.

According to this definition, events may be lexicalized by tensed or un-tensed verbs, nominalizations, adjectives, predicative clauses, or prepositional phrases. This view on events is mainly implemented by TimeML project [Pustejovsky et al., 2003], which will be described in Section 5.4.1.

2.1.1 Identifying and Extracting Events in NLP

Given the huge amounts of unstructured textual data available on the web, automatic information extraction (IE) becomes a crucial necessity. The information extracted by an IE system can be useful for distinctive NLP applications, e.g., question answering, textual entailment, and summarization among others. One of the sub-tasks of IE is event extraction and identification. Event Identification is counted as a non-trivial task. This is because of the fact that event triggers in text can be lexicalized in many different ways in sentences. Moreover, the attributes describing an event usually span multiple sentences.

2.1.2 Identifying The Semantic Relations Between Events

A more challenging problem than event extraction is identification of the semantic relation that holds between events. Events take place in time, hence temporal relation between events is the most crucial relation to study.

Temporal Relation

Time is the main notion for anchoring the changes of the world triggered by events in order. Of course having temporal understanding and temporal reasoning capabilities is crucial for many NLP applications such as question answering, text summarization and many others. For Instance, answering questions like “Did Einstein die before the Many Worlds theory was proposed?” requires accurate understanding and identification of the two events in question and finding their temporal relation. Moreover, text summarization approaches also rely heavily on identifying key events and their chronological order in the given text. Over the years the problems of temporal analysis and reasoning in natural language have been addressed via different approaches. The main one is the theory for representing actions and time [Allen, 1984], known as Allen’s Interval Algebra, which mainly introduces a set of 13 distinct, exhaustive, and qualitative temporal relations[1] that can hold between two time intervals.

Based on Interval Algebra, a new markup language for annotating events and temporal expressions in natural language was proposed [Pustejovsky et al., 2003], called TimeML. It is designed to address problems in event and temporal expression markup. It covers two major tags: 'EVENT' and 'TIMEX3'. EVENT tag is used to annotate elements in a text that represent events such as 'kill' and 'crash'. TIMEX is mainly used to annotate explicit temporal expressions, such as times, dates and durations. TimeML also takes into account 'Link tags'. These tags encode the semantic relations that exist between the temporal elements annotated in a document. The main link tag is TLINK, which stands for 'Temporal Link' which represents a temporal relationship between two events, according to which they can be ordered in time. TimeML has been accepted as an ISO standard. TimeBank corpora have been annotated as an illustration corpus for TimeML specification since 2002. TimeBank corpora have been serving as training data for learning systems which can extract temporal information from text.

The TempEval challenge is a framework (with variety of focused shared tasks) for evaluating systems that automatically annotate texts in TimeML format. TempEval was first created in the context of the SemEval 2007 workshop and was renewed in 2010, 2013 and 2015. In general, annotating in TimeML format is a nontrivial task given: (1) the diversity of events and time expressions (2) the complexity of identifying the type of temporal relations among events and (3) major problems in NLP such as ambiguity, coreference, and ellipsis. The results of TempEval-3 [UzZaman et al., 2012] show that the performance of extracting TimeML events and time expressions is good (80% F-score), however, the state-of-the-art performance on the temporal relation extraction is about 36% F-score [Bethard, 2013] which signals the complexity of the relation extraction

[1] This set includes *before, after, meets, met by, overlaps, overlapped by, starts, started by, finishes, finished by, contains, during, equals.*

task.

Causal Relation

Research on the extraction of event relations has concerned mainly the temporal relation and time-wise ordering of events. A more complex semantic relationship between events is causality. Causality is one of the main semantic relationships between events where an event (CAUSE) results in another event (EFFECT) to happen or hold.

It is clear that identifying the causal relation between events is crucial for numerous applications. Predicting occurrence of future events is the major benefit of causal analysis, which can itself help risk analysis and disaster decision making. There is an obvious connection between causal relation and temporal relation: by definition, the CAUSE event happens 'BEFORE' the EFFECT event. Hence, predicting causality between two events also requires/results in a temporal prediction. In the next Section we define causality in natural language and give an overview on the existing approaches and resources concerning causality in NLP community.

2.2 Causation in Natural Language

It is challenging to define causality in natural language. Causation, as commonly understood as a notion for understanding the world in the philosophy and psychology is not fully predicated in natural language [Neeleman and Koot, 2012]. There have been several attempts in the field of psychology for modeling causality, e.g., the counterfactual model [Lewis, 1973] and the probabilistic contrast model [Cheng and Novick, 1992]. Mainly, Leonard Talmy's seminal work in the field of cognitive linguistics models the world in terms of entities interacting with respect to force [Talmy, 1988], hence introducing force dynamics in language and cognition.

Talmy's effort on organizing meaning in language by systematic application of force concepts to them [Talmy, 2003] shows the prevalence of force-dynamics way of thinking. "Force dynamics" mainly involves semantic category of how entities interact with respect to force. This semantic category includes concepts such as the employment of force, resistance to force and the overcoming of resistance, blockage of a force, removal of blockage, and etc. Force dynamics provides a generalization over the traditional linguistic understanding of causation by categorizing causation into 'letting', 'helping', 'hindering' and etc. Wolff and Song [Wolff and Song, 2003] base their theory of causal verbs on force dynamics. Wolff continues [Wolff, 2007] proposing that causation includes three main types of *causal concepts*: **'Cause', 'Enable'** and **'Prevent'**. These three causal concepts are lexicalized through distinctive types of verbs [Wolff and Song, 2003] which are as follows:

- Cause-type verbs: e.g. cause, start, prompt, force.
- Enable-type verbs: e.g. allow, permit, enable, help.
- Prevent-type verbs: e.g. block, prevent, hinder, restrain.

Given that Wolff's model accounts for different ways that causal concepts are lexicalized in language, we mainly base our upcoming framework in Chapter 5 on this.

2.2.1 Causation in Natural Language Processing

As mentioned earlier, causality is a notion that has been widely studied in psychology, philosophy, and logic. However, precise modeling and representation of causality in NLP applications is still an open issue. Causality is one of the major semantic relations between events, which has not been covered as elaborately as temporal (e.g., before, after, etc) relations in the NLP community. However, knowing the causal relation between events plays a significant role in understanding the world and finding coherence between various scenarios happening under specific circumstances. As an example, 67% of Winograd Schemas (Section 2.3.2) are causal, i.e., the two clauses are connected via a causal connective such as 'because'.

A formal definition of causality in lexical semantics can be found in [Hobbs, 2005]. Hobbs introduces the notion of "causal complex", which refers to some collection of eventualities (events or states) for which holding or happening entails the happening of effect. A few specific phenomena related to causality have been investigated in NLP community throughout the years. In the rest of this Section we will describe various works concerning causality in NLP community. Before that, we point out the regular categorization for the scope of the works on causality, which is as follows:

- 'Explicit' or 'Implicit': The expression of the causal relation in text could be Explicit, i.e., a clear causal signal (e.g., *because*) appears in the text. Otherwise, the causal relation is implicit in the text. Different systems could account for either or both of these classes.

- 'Inter-sentential' or 'Intra-sentential': Inter-sentential causal relation extraction systems attempt to find causal relations between different events at document level. However, intra-sentential systems focus on finding causal relation between events at sentence-level.

Causal Relation Learning Using Penn Discourse Tree Bank (PDTB)

One of the major works which includes causality is the effort on annotating causal discourse relations in Penn Discourse Tree Bank (PDTB) corpus [Prasad et al., 2008]. The PDTB corpus [Prasad et al., 2008] provides manual annotations for the argument structure of both 'Explicit' and 'Implicit' connectives, the senses[2], attributes, and arguments of connectives. This corpus annotates semantic relations holding between exactly two Abstract Objects (called Arg1 and Arg2), expressed either explicitly via lexical items or implicitly via adjacency in discourse. The 'Explicit' connectives are identified from the following three grammatical classes:

- Subordinating conjunctions: e.g., because, when.
- Coordinating conjunctions: e.g., and, or.
- Discourse adverbials : e.g., however, otherwise.

For instance consider the following example:

(4) Some have raised their cash positions to record levels.[Implicit =BECAUSE] High cash positions help buffer a fund when the market falls.

Example 4 showcases an implicit causal relation between 'raising cash positions to record levels' and 'high cash positions helping to buffer a fund'. As this example shows, there is no Explicit connective in the text to express that this relation holds.

PDTB contains 7,448 instances of causal relations. Since its initial release, this corpus has been leveraged for various linguistic and computational purposes, such as text summarization, language generation, automatic discourse analysis, and many other

[2] For instance, the connective 'since' has three difference senses: purely causal, purely temporal, and both temporal and causal.

tasks that can benefit from knowing discourse structure relations [Prasad et al., 2005; Verberne et al., 2007; Ramesh et al., 2012].

There have been some works on event causality identification some of which have benefited from PDTB. Do et al. [Do et al., 2011] attempted to detect causality between verb-verb, verb-noun, and noun-noun event pairs. They measured the causality by computing Point-wise Mutual Information (PMI) between the cause and the effect together with incorporating discourse information, specifically the connective types extracted from the PDTB. They achieved F-score of 46.9% evaluating on a dataset of 20 news articles.

Another recent work [Riaz and Girju, 2013] attempts to identify causal relations between verbal events. Their approach mainly involves using unambiguous discourse markers such as ‘because’ and ‘but’ to collect training examples of cause and non-cause event pairs. The outcome of their approach is a knowledge base of three causal relations between verbs: strongly causal, ambiguous and strongly non-causal.

Classification of Semantic Relations between Nominals

Another effort which includes the notion of causality is the ongoing work on identifying the semantic relation between nominals. A series of SemEval tasks are designed to provide a framework for evaluating classification of semantic relations between nominals in a given sentence. SemEval-2007 (SemEval-1) Task 4 on ‘Classification of Semantic Relations between Nominals’ [Girju et al., 2007] provides a corpus (separated into training and test) of nominal relation annotations including total 220 cause-effect relations. They define ‘nominal’ to be simple nouns, noun compounds and multi-word expressions taking noun role in a sentence. This task provided binary-label per each of the seven semantic relations of the dataset for each entry in the dataset. Competing on this dataset, the top system for cause-effect relation classification trained an SVM classifier on several lexico-semantic features such as the word stems and POS sequences between the nominals.

Successor to SemEval-1 Task 4 was SemEval-2010 task 8 on ‘Multi-way classification of semantic relations between pairs of nominal’ [Hendrickx et al., 2010], which required the label to be chosen out of ten possible relations. The final dataset includes total 1,331 cause-effect pairs. An example cause-effect pair in their dataset is as follows:

(5) Those cancers were caused by radiation exposures.

This dataset has been used for training classifiers on nominal relation learning [Davidov and Rappoport, 2008].

Joint Temporal and Causal Relation Classification

Another line of work considered annotating temporal and causal relations in parallel [Steven Bethard and Martin, 2008]. They annotated a dataset of 1,000 conjoined-event temporal-causal relations, collected from Wall Street Journal corpus. Each event pair was annotated manually with both temporal (BEFORE, AFTER, NO-REL) and causal relations (CAUSE, NO-REL). For example, following is a sentence and its corresponding annotation in their dataset:

(6) Fuel tanks had leaked and contaminated the soil.
* (leaked BEFORE contaminated)
* (leaked CAUSED contaminated).

They use 697 event pairs for training a classification model for causal relations, which results in 37.4% F-score. In this dataset there is no distinction between various types of causal relation. For instance in the sentence ‘We recognized the problem and took care of it’, they annotate a ‘cause’ relation between the events ‘recognize’ and ‘took’, which is indeed an ‘enable’ relation.

Using this corpus, a new method for discovering causal relations between events [Rink et al., 2010] was proposed. In order to determine the causality relation between two events, they build a graph representation of the sentence augmented with its lexical, syntactic, and semantic information. Then they automatically extract multiple graph patterns which can potentially determine the causality between two events in a sentence. Then they use the graph patterns mapped to the graph representation of the two events at hand for training a binary causal classifier. Their approach achieves 57.9% F-score, which shows 15% improvement as compared with the state-of-the-art systems.

Annotation Framework to Capture Causality

A recent work [Mirza and Tonelli, 2014] has proposed a TimeML-style [Pustejovsky et al., 2003] annotation standard for capturing causal relation between events, which could potentially lead towards benchmarking causal relation extraction. They mainly introduce ‘CLINK’ and ‘C-SIGNAL’, analogous to ‘TLINK’ and ‘SIGNAL’ in TimeML, to be added to the existing TimeML link tags. ‘SIGNAL’ tags in TimeML mainly annotate temporal prepositions, connectives, and subordinators. Similarly, C-SIGNALs are used to annotate sentence elements which signal a causal relation, such as the following:

- Causal uses of prepositions: such as *because of, as a result of, due to.*

- Conjunctions: such as *because, since, so that.*

- Adverbial connectors: such as *so, therefore, thus.*

- Clause-integrated expressions: such as *the reason why, the result is, that is why.*

Under this framework, Mirza et al [Mirza and Tonelli, 2014] annotates causal links and signals (CLINK and C-SIGNAL) in TempEval-3 TimeBank. They annotated total of 171 C-SIGNALs and 318 CLINKs. Using this TimeBank, they train a causal signal and link extractor between given event pairs. For C-SINGAL extraction they achieve 91.03% Precision, 41.76% Recall and 57.26% F-score using a SVM classifier. For CLINK (being provided with gold C-SIGNALs) they achieve 74.67%, 35.22%, and 47.86% on Precision, Recall and F-score respectively.

Causal Relation Extraction Using Patterns

As reported earlier in Sections 2.2.1 and 2.2.1, extracting causal relations between events in text is a hard task. One simplification to causal extraction task is to simply focus on 'Explicit' and 'Intra-sentential' expressions of causal relations and use patterns to detect causality. Blanco et al. [Blanco et al., 2008] defined a list of lexico-syntactic patterns (pairs of phrases with a causative relator in between) that refer to causal relations with additional semantic constraints. Their approach first identifies the syntactic patterns that may encode a causation, then they train a decision tree to decide whether or not a pattern instance is a valid causation. They end up using only one pattern, which is the most effective one: *[VP rel C], [rel C, VP]*, which encode a verb phrase followed by a relator and a clause, and its reverse version. Here relators are causal signals *as, after, because and since*. They build a gold-dataset, labeling sentences of TREC5 corpus with $cause$ or $\neg cause$. Evaluating their approach on the test set, they achieve F-Score of 91% on determining $\neg cause$ and 89% on $cause$.

Predicting Future Events by Mining Causality

One of the main works that aims at predicting future events using the notion of causality is the work on ‘Learning to Predict from Textual Data’ [Radinsky et al., 2012]. This work mines 150 years of New York Times articles to predict future events. They posit that most of the information regarding an event can be found in the news headlines, which are usually more structured pieces of text. They develop an extraction method to identify the headlines which have cause and effect events. They mainly use grammatical patterns such as usage of causality connectors to pinpoint cause and effect. Some of their causal connectors are as follows:

- Causal Connectives: *because, as, and after*.
- Causal Prepositions: *due to and because of*.
- Periphrastic Causatives: *cause and lead to*.

They’ve extracted cause-effect event pairs from a sub-set of NYT corpus with accuracy of 78%.

For generalization, the model uses a vast number of world knowledge ontologies. After extracting cause-event pairs, they propose the following two-stage algorithm for predicting the effects of a given event:

1. Learning Phase
    - Generalize Events
    - Causality Prediction Rule Generation
2. Prediction Phase

- Finding Similar Generalized Events
- Application of Causality Prediction Rule

They present various experiments for evaluating different stages of this algorithm. As an example, presenting the event “6.1 magnitude aftershock earthquake hits Haiti” to their system yields the following highest matching predictions, ordered according to their point-wise mutual concept information score:

- X people will be dead.
- X people will be missing.
- X magnitude aftershock earthquake will strike island near Haiti.
- Earthquake will turn to United States Virgin Islands.

One limitation of their approach is their reliance on only positive examples which results in over-generalization. For instance, for the event “Lightning killed 5 people” they predict the effect “lightning goes to jail”.

2.3 Natural Language Inference Frameworks

Natural language understanding (NLU) is a subfield of natural language processing that deals with machine reading comprehension. The goal of an NLU system is to interpret an input text into an unambiguous formal language which basically represents the system’s perception of the world. One of the clearest example of NLU is Natural Language Inference (NLI), which is the ability of a system to reason about the truth of a given statement based on some given premise, i.e., infer the truth value.

In this chapter we introduce the challenges and frameworks within NLP community which involve language understanding and inference, all of which clearly require large amounts of knowledge for making meaningful progress.

2.3.1 Recognizing Textual Entailment (RTE)

Starting from 2004, researchers have worked on the task of Recognizing Textual Entailment (RTE), which is designed to integrate various efforts on textual inference as an essential problem in natural language understanding. RTE is the longest running challenge task in NLP community which involves entailment and inference in natural language. The subsequent challenges are all more of less sub-tasks of the RTE framework. Given two text fragments, the RTE task is to decide whether the meaning of one piece of text can be inferred from another [Dagan et al., 2005]. A more formal definition is as follows:

- **Definition:** A directional relationship between pairs of text fragments T (Text) and H (Hypothesis) is called 'entailment' if T entails H, i.e., the meaning of H can be inferred from the meaning of T with high probability, according to a typical interpretation by people.

The above definition is based on common human language understanding and common background knowledge [Sammons et al., 2012]. Figure 2.1 shows an example a text together with a hypothesis. In this figure, according to definition of entailment, Hypothesis 1 is entailed from the Text and Hypothesis 3 is not. The specification of the RTE task requires the Text to be an inherent part of the reasoning for inferring the truth of the Hypothesis: while background knowledge may augment the inference, it may not replace the text. i.e., if the Hypothesis is true by itself (say as a general fact

Text: On 18 April 1955, Aortic aneurism killed Albert Einstein. This is when blood vessels gather in the aorta stretching out this part of the heart.

Hypothesis 1: A health issue caused Einstein to die.

Hypothesis 2: The Bell Inequalities were not presented while Einstein was alive.

Hypothesis 3: Einstein was executed by Nazi Germany.

Figure 2.1: Sample Text and Hypothesis for RTE Task

or knowledge), and the Text does not play any inherent role in its truth value, we do not say that Text entails the Hypothesis. For instance, in figure 2.1, Hypothesis 2 is actually true, but we cannot infer its truth from the Text, so we say that Text does not entail Hypothesis 2.

Entailment in Linguistics

Classical semantic entailment in the linguistics literature specifies that a text T entails another text H, if in every circumstance that T is true, H is true [Hierchia and McConnell-Ginet, 2001] – which is related to our definition of textual entailment. However, normally the earlier presented standard definition of entailment allows for practical cases in which the entailment is highly possible, rather than absolutely certain, i.e., we do not address relatively delicate logical issues as typical in classical linguistic literature.

2.3.2 Winograd Challenge

In June 2012, a chatbot named Eugene Goostman[3] won the Turing test contest, in which it successfully convinced 29% of its judges that it was human. This win faced criticism, since it more revealed how easy it is to 'fool' some humans throughout a short conversation, than how intelligent the system actually is. This motivated some alternatives or modifications to Turing Test to enforce human-level of intelligence.

The Winograd Schema Challenge (WSC) is a test of artificial intelligence proposed by Hector Levesque [Levesque, 2011] in 2011. Winograd Challenge is mainly designed to be an improvement on the Turing test [4]. Like Turing test, it involves responding to typed English sentences. Unlike Turning test, since there is no conversation, the issue of fooling or deceiving the evaluator into believing the subject is human is eliminated. In order to enforce actual language understanding the test is carefully designed so that having full access to a large amounts of textual data might not be enough.

The Winograd Challenge questions have very specific structure and are called Winograd Schemas (WS). Each schema is a small reading comprehension test involving a binary question. The following example, a variant of which was first presented by Terry Winograd [Winograd, 1972], illustrates WS:

(7) The town councilors refused to give the demonstrators a permit because they feared (advocated) violence. Who feared (advocated) violence?

- Answer 0: the town councilors
- Answer 1: the angry demonstrators

[3] Goostman was presented as a 13-year-old Ukrainian boy which resulted in more forgiveness for its grammatical errors or lack of general knowledge.

[4] Whether or not Winograd Schema Challenge is an appropriate alternative to Turing test (which actually involves dialogue and unconstrained conversation rather than a binary-choice question) is controversial issue in research community.

here the special word is ‘feared’ together with its alternative which is ‘advocated’. The correct answer to the question is flipped going from the special word to its alternative. Following are the characteristics of a valid WS:

Property 1 Two entities or sets of entities, not necessarily people or sentient beings, are mentioned in the sentences by noun phrases.

Property 2 A pronoun or possessive adjective is used to reference one of the parties (of the right sort so it can refer to either party).

Property 3 The question involves determining the referent of the pronoun.

Property 4 There is a *special word* that is mentioned in the sentence and possibly the question. When replaced with an alternate word, the answer changes although the question still makes sense (e.g., in the above examples, big can be changed to small; feared can be changed to advocated.)

To put it in a nutshell, a WS is a pair of sentences that are different in only one or two words (special word) and involve a pronoun resolution that is resolved in opposite ways in the two sentences. To make the challenge require world knowledge and reasoning, Levesque proposes the following constraints on WS:

Constraint 1 Easily disambiguated by the human, preferably without even noticing the underlying resolution ambiguity.

Constraint 2 Not solvable by simple coreference resolution techniques such as selectional restrictions.

Constraint 3 Is Google-proof, i.e., mining large amounts of textual corpora with obvious statistical measures does not suffice for disambiguating these correctly.

As the setup of this task suggests, this is an easy task for a subject that has basic 'understanding' of natural language; however, a hard task for any subject that can only intelligently guess the correct answer without any deep understanding of language and world phenomena. Levesque posits that "with a very high probability", any subject that can resolve correctly a series of WS "is thinking in the full-bodied sense we usually reserve for people" [Levesque, 2011]. Therefore WSC seems to be a powerful methodology for tracking progress in automating commonsense reasoning. Altogether, WSC is a promising framework to work on for fostering the research on deep language understanding and getting closer to solving the core problem of AI: building a truly intelligent machine.

2.3.3 Approaches for Tackling Winograd Schema Challenge

It is important to note that the traditional approaches of coreference resolution are not useful for solving Winograd Schemas (WS). For instance, linguistic constraints such as syntactic ones (e.g., binding constraints) or constraints concerning agreement in number, gender, and semantic class will not be helpful, since both the entities mentioned by the noun phrases are compatible with the pronoun in question.

It is obvious that addressing the WS questions requires significant amounts of background knowledge, together with an inference backbone that can make use of the knowledge for resolving the referent of the pronoun in question. Given the recency of WSC, there are only two major publications tackling a variant of this framework. Davis hosts an online collection [5] of 133 Winograd schemas which meet all the properties and constrains listed in previous Subsection.

[5] https://www.cs.nyu.edu/davise/papers/OldSchemas.xml

Resolving Complex Cases of Definite Pronouns

Rahman & Ng [2012] present the first published work tackling Winograd Schemas. They present the test framework of twin-sentence coreference resolution (the same setting as WS) as a complex case of pronoun resolution. They gathered a dataset containing 1,886 sentences [6] of such complex pronoun resolution problems. Following is an example instance in their dataset:

(8) Professors give a lot of advice to students, because they are caring.

- Answer 0: Professors
- Answer 1: Students

(9) Professors give a lot of advice to students, but they rarely care.

- Answer 0: Professors
- Answer 1: Students

As the above example shows, their dataset does not observe Property 4, i.e., their twin sentences are not different in only one or two words necessarily. Mainly, the connective is not necessarily shared between the twin sentences. Moreover, their dataset does not guarantee Constraint 3, i.e., many of the instances in this dataset can directly benefit from search engine hits and are not Google-proof. However, as the results show, this dataset is still hard enough to require certain amount of deep language understanding.

As an indication to the difficulty of the instances in this dataset (Rahman & Ng dataset hereinafter), it is useful to note that the a state-of- the-art coreference resolution system [Chang et al., 2013] achieves precision of 53.26% on it.

[6] http://www.hlt.utdallas.edu/~vince/data/emnlp12/train-emnlp12.txt

Rahman & Ng propose a ranking algorithm for finding the correct antecedent for the target pronoun given an instance of their test dataset. They train their model on 70% of their hand-annotated twin sentences. The features they use are as follows: *Narrative Chains [Chambers and Jurafsky, 2008], Google, FrameNet, Heuristic Polarity, Learned Polarity, Connective-Based Relation, Semantic Compatibility and Lexical Features*. They achieve precision of 73.05% on the test set. The major contributing feature for their model was reported to be lexical features, Google, and Narrative Chains.

Solving Hard Coreference Problems by Predicate Schema

The most recent work proposes a solution involving a representation called Predicate Schema for the knowledge required to address complex coreference problems [Peng et al., 2015]. The Predicate Schemas are instantiated with automatically acquired knowledge, and is then turned into constraints that are part of an Integer Linear Programming formulation for resolving the coreference. More specifically, they study two types of Predicate Schemas: one predicate with its subject and object, providing information on the subject and object preferences of a given predicate; the other specifying two predicates with a shared argument (either subject or object), hence specifies role preferences of one predicate. For instantiating the Predicate Schemas they use the statistics they have acquired from multiple resources such as Gigaword corpus, Wikipedia, Web Queries and polarity information.

Testing their system under Rahman & Ng's dataset, they achieve precision of 76.41% which is 3.36% improvement over Rahman & Ng's approach.

2.3.4 Choice of Plausible Alternatives (COPA)

The Choice of Plausible Alternatives task was proposed in 2011 [Roemmele et al., 2011]. It provides an evaluation metric for the progress made in common-sense causal reasoning. They wrote thousand English questions, each of which gives a premise and two plausible causes or effects, where the correct answer is the alternative that is 'more plausible' than the other. Following is an example instance of COPA task, asking for the most plausible 'effect' of the given sentence:

(10) Premise: I poured water on my sleeping friend.

- Alternative Effect 0: My friend awoke.
- Alternative Effect 1: My friend snored.

The best performing baseline on his framework with accuracy of 58.3% uses pointwise mutual information (PMI) between keywords of the premise and each alternative, drawn from a large corpus of personal story blogs [Gordon et al., 2011]. According to this method, keyword PMI outperformed a complicated approach that detected temporally related clauses using a parser trained on rhetorical discourse theory (RST) corpora. Also they found out that a personal story (narrative) corpus generated from Weblogs resulted in better performance than a corpus they had acquired from Project Gutenberg.

COPA was also a SemEval 2012 task [Gordon et al., 2012]. The only competing system [Goodwin et al., 2012] trained a SVM classifier for labeling the pair of premise and alternative as either causal or not. The features they used included POS and dependency parse labels, event extractions, polarity comparison and mutual information between bigrams. Their classifier achieved only 62% accuracy.

2.4 Available Inferential Knowledge Resources

Deep language understanding, which enables inference, requires systems that have large amounts of knowledge enabling them to connect natural language to the concepts of the world. As mentioned in a few parts of the previous chapter, having access to vast amount of inferential knowledge is crucial for many Natural Language Understanding (NLU) tasks and frameworks.

Obviously any kind of knowledge may be useful for the goals of NLU: knowledge about lexical meaning of the words, about a specific domain in question, about laws of nature, or about the specificity of the surrounding world. However, throughout this work we mainly focus on conceptual and common-sense knowledge which we are interested in. In this Chapter we provide an overview on the major available knowledge resources.

2.4.1 Hand-built and Community-powered Knowledge Bases

WordNet

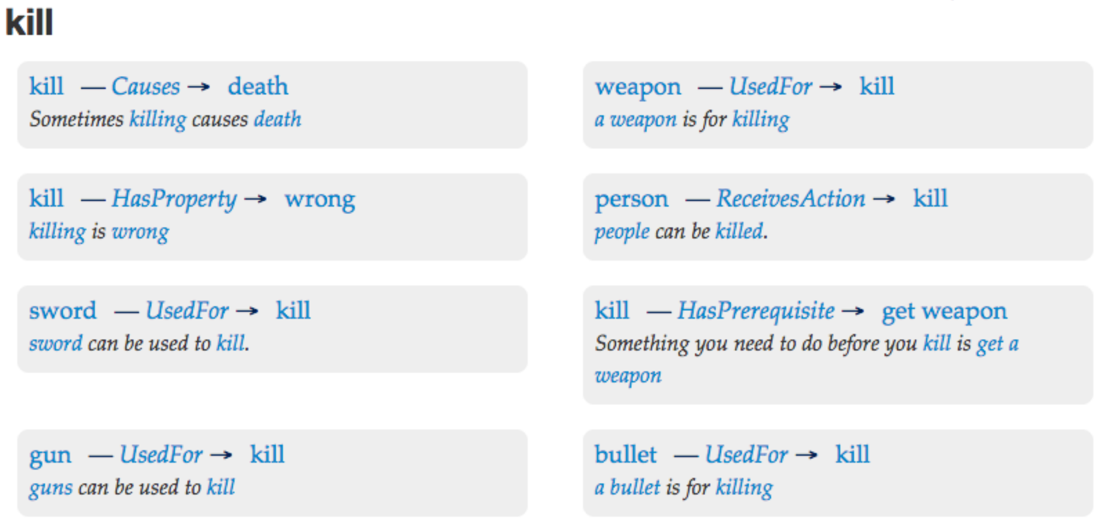
WordNet [Miller., 1995] is the most widely used lexical resource in NLP community. WordNet is basically a lexical database in English. At its core WordNet has (1) synsets, which is the way similar concepts of language are clustered together; (2) hypernymy and hyponymy links which provide a hierarchical structuring for lexical entries. WordNet also has many more additional relationships between entries, e.g., part-of ('engine' is a part of a 'car'), causes ('revive' causes 'come-to'), entails ('oversleeping' entails 'sleeping'), antonyms ('cold' is antonym of 'hot'), etc. WordNet is a comprehensive ontology with about 120,000 concepts, providing various levels of knowledge regarding different concepts, however, WordNet does not provide common facts about the world or what to expect from various situations.

ConceptNet

ConceptNet [Liu and Singh, 2004] is a semantic network that is designed to give common sense knowledge (concept) to machine. ConceptNet is the successor to OpenMind Common Sense (OMCS) project [Singh et al., 2002]. OMCS was a project at MIT Media Lab with the goal of building a large commonsense knowledge base by contributions of thousands of people on the Web. They gathers millions of facts, rules, stories, and descriptions using a variety of elicitation methods [Singh, 2001] on the web. Following are example items that they collected:

- Every person is younger than the person's mother
- A butcher is unlikely to be a vegetarian
- People do not like being repeatedly interrupted
- If you hold a knife by its blade then it may cut you
- If you drop paper into a flame then it will burn

ConceptNet is based on OMCS corpus. About fifty extraction rules are used for mapping OMCS's English sentences into ConceptNet's binary relations between concepts in the underlying semantic network. Figure 2.2 shows an example entry in ConcepNet, for the concept 'kill'. As you can see in this Figure, one of the semantic relations is 'Causes'.



kill

kill — *Causes* → death
Sometimes *killing* causes *death*

weapon — *UsedFor* → kill
a weapon is for *killing*

kill — *HasProperty* → wrong
killing is *wrong*

person — *ReceivesAction* → kill
people can be *killed*.

sword — *UsedFor* → kill
sword can be used to *kill*.

kill — *HasPrerequisite* → get weapon
Something you need to do before you *kill* is *get a weapon*

gun — *UsedFor* → kill
guns can be used to *kill*

bullet — *UsedFor* → kill
a bullet is for *killing*

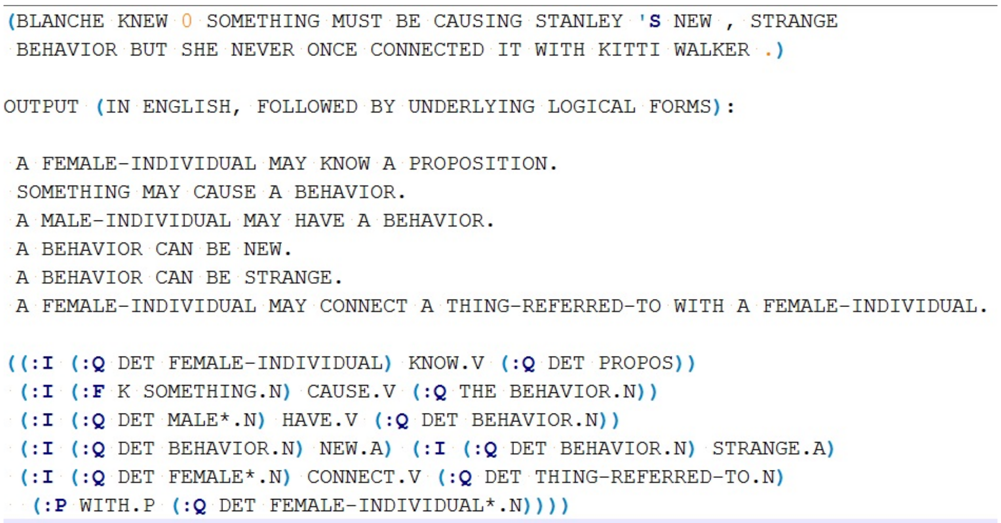
Figure 2.2: A part of the entry for the English concept ‘kill’ in ConcepNet 5.0.

2.4.2 Automatic Knowledge Acquisition Approaches

Hand-engineering knowledge is rather expensive and time-consuming. An ideal alternative is to automatically acquire

2.4.3 KNowledge EXtraction from Text: KNEXT

KNEXT [Schubert, 2002; Durme and Schubert, 2008] is a tool for extracting general world knowledge from large collections of text. The core idea is to syntactically parse each sentence with a Treebank-trained parser (e.g., Charniak) and then compositionally apply the interpretive rules in order to compute logical forms. The results are quantificationally underspecified Episodic Logic formulas (with quantifiers such as ‘most’, ‘few’, ‘many’), which are verbalized in English as possibilistic claims. An example of the knowledge discovered by KNEXT is ‘Persons may want to get rid of a dictator’, or ‘A person usually has a name’. In evaluating this approach [Durme and Schubert, 2008], it was observed that propositions found at least twice were judged more accept-



```
(BLANCHE KNEW 0 SOMETHING MUST BE CAUSING STANLEY 'S NEW , STRANGE
 BEHAVIOR BUT SHE NEVER ONCE CONNECTED IT WITH KITTI WALKER .)

OUTPUT (IN ENGLISH, FOLLOWED BY UNDERLYING LOGICAL FORMS):

 A FEMALE-INDIVIDUAL MAY KNOW A PROPOSITION.
 SOMETHING MAY CAUSE A BEHAVIOR.
 A MALE-INDIVIDUAL MAY HAVE A BEHAVIOR.
 A BEHAVIOR CAN BE NEW.
 A BEHAVIOR CAN BE STRANGE.
 A FEMALE-INDIVIDUAL MAY CONNECT A THING-REFERRED-TO WITH A FEMALE-INDIVIDUAL.

((:I (:Q DET FEMALE-INDIVIDUAL) KNOW.V (:Q DET PROPOS))
 (:I (:F K SOMETHING.N) CAUSE.V (:Q THE BEHAVIOR.N))
 (:I (:Q DET MALE*.N) HAVE.V (:Q DET BEHAVIOR.N))
 (:I (:Q DET BEHAVIOR.N) NEW.A) (:I (:Q DET BEHAVIOR.N) STRANGE.A)
 (:I (:Q DET FEMALE*.N) CONNECT.V (:Q DET THING-REFERRED-TO.N)
   (:P WITH.P (:Q DET FEMALE-INDIVIDUAL*.N))))
```

Figure 2.3: An example output of KNEXT system for a sentence from the Brown corpus.

able than those extracted only once. Figure 2.3 shows an example output of KNEXT system for a sentence from the Brown corpus.

Gordon's Lore system [Gordon, 2014] strengthens and partially disambiguates many of the KNEXT propositions to quantified ones such as "All or most elm trees permanently have some branch as a part", where these are in a logical form suitable for inference.

Open Information Extraction from the Web: TextRunner

TextRunner [Banko et al., 2007] is an unsupervised, single-pass extraction technique from the Web, where no relation names are required for input. It is a tool for extracting explicitly stated information as tuples of normalized text fragments. Its outputs are mainly about specific individuals and generic claims. To each instance of the relation, they assign a probability of being correct, using the number of distinct sentences from which a tuple was extracted. Figure 2.4 shows an example of extracted tuple using

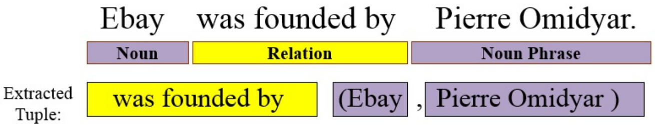


Figure 2.4: An example of a tuple extracted using TextRunner.

TextRunner.

Distributional Learning of Rules

The line of research based on Distributional Learning of Rules is based on Zelig Harriss distributional hypothesis [Harris, 1985] which states that words that occur in similar contexts have similar meanings. Lin and Pantel [Lin and Pantel, 2001] adapted this idea and applied it on sentence fragments, which resulted in Discovery of Inference Rules from Text (DIRT). Their algorithm can be summarized as follows:

1. Gather a large collection of dependency parses of sentences.

2. Identify the basic noun phrases in each sentence and extract the paths. For example, the path $find \rightarrow V : obj : N \rightarrow solution \rightarrow N : to : N$ corresponds to 'X finds solution to Y'.

3. Collect all paths that connect similar nouns. This is according to the "extended distributional hypothesis", i.e., if two paths tend to occur in similar contexts, the meanings of the paths tend to be similar. Define path similarity in terms of co-occurrence counts with various slot fillers.

For example, some of the top similar paths to "x finds a solution to Y" are: 'X tackles Y', 'X resolves Y', and 'Y is solved by X'. There are a lot of other attempts at acquiring inference rules. Intuitively such rules can be very effective for RTE task.

However, in the RTE-3 challenge as an instance, only a small number of systems used DIRT as a resource, and the evaluation results did not show any important contribution of DIRT to their final result. This is mainly because of low accuracy (about 50%) of the rules generated by DIRT method, and also its limited applicability which is at the surface level of language. We will discuss this issue with more details in Section 3.4.2.

It is important to note that many Text-Hypothesis pairs, which require some kind of lexical or world knowledge, mostly need complicated inference methods for reasoning, and the ability to combine different pieces of knowledge together. All these issues make the task of 'using knowledge bases effectively in RTE' a challenge which is yet to overcome. Next chapter introduces a novel approach on learning conceptual knowledge on events, which could enhance the inference capabilities of RTE systems.

Script-like Knowledge about Events

As humans, we mainly find correlation between daily events by having experienced them or just knowing the existing correlation as a common-sense knowledge. For instance, we know that a person normally goes to school, then attends a college and afterward gets a job. The question is how to provide such knowledge regarding events to machines. Perhaps one of the main contributions to Artificial Intelligence was Marvin Minsky's work on the notion of frames as *"a data-structure representing a stereotyped situation"* [Minsky, 1975]. Minsky's work included characterization of different kinds of frames, some corresponding to the case frames introduced by Fillmore in linguistics [Fillmore, 1968]. Fillmore characterized the Case Frames as "a small abstract 'scene' or 'situation', which helps in understanding the semantic structure of the verbs necessary for understanding the properties of the underlying schematized scenes".

The most notable next work was on story understanding [Schank and Abelson, 1977] which coined the term ‘script’ (comparable to Minsky’s frames) referring to the knowledge structures representing the sequences of events. Their most famous script is the Restaurant Script, which includes the events (entering, sitting down, asking for menus, choosing meals, etc.), the participants (Customer, Waiter, Chef, Tables, etc.), and the preconditions, event ordering, and results of the actions.

Manually built scripts were used in the 70’s and 80’s as the major knowledge base enabling inference for NLP tasks which required deep semantic knowledge. In general, structured sequences of events together with their participants have been called ‘scripts’ (Schankian scripts) or ‘Fillmorean frames’. In order to make inferences, the events and their participant slots can be instantiated in a particular context or situation. It is obvious that NLU applications can hugely benefit from the rich inferential capabilities that structured knowledge about events provides. Text summarization, co-reference resolution and question answering, among others, can use script-like knowledge bases for improving their performances.

Learning Narrative Event Chains

Given that developing hand-built scripts is extremely time-consuming and non-scalable, there is a serious need for automatically induced scripts. The main work tackling this issue is Nate Chambers and Daniel Jurafsky’s work on ‘Narrative Chains’. The narrative chains can be seen as a generalization of case frames with which one can find a coherent structured narrative in some piece of text. More specifically, Chamber’s narrative chains are partially ordered sets of events all sharing a common actor, called ‘protagonist’. Perhaps the major characteristic of narrative chains is the idea of using co-referring arguments as an evidence of an underlying narrative relation. They learn these chains in an unsupervised manner. Figure 2.5 depicts an example chain learned

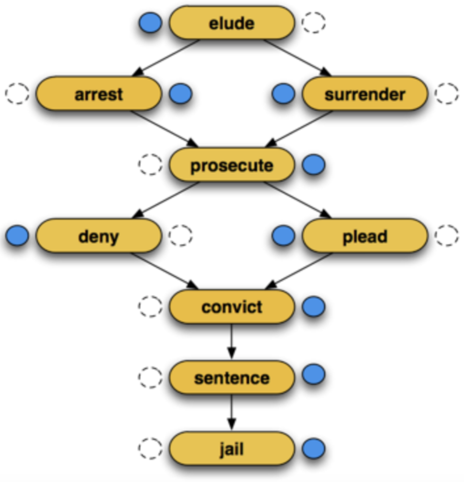


Figure 2.5: An example Narrative Chain learned by Chambers and Jurafsky's 2008 work. The blue dots are the protagonist shared among the events.

by their method.

They accomplish this by the following steps:

1. Narrative event induction : In order to find a measure of how related two events are in a narrative structure, they use an unsupervised distributional method, which is summarized as follows:
   (a) Dependency parse GigaWord corpus.
   (b) Run OpenNLP coreference resolver to cluster entity mentions sharing an argument.
   (c) Count pairs of verbs which have co-referring arguments and computer their pointwise mutual information (PMI) to measure their relatedness.
2. Temporal ordering of events: After knowing what events are related, a temporal classifier is trained to partially order the related events.

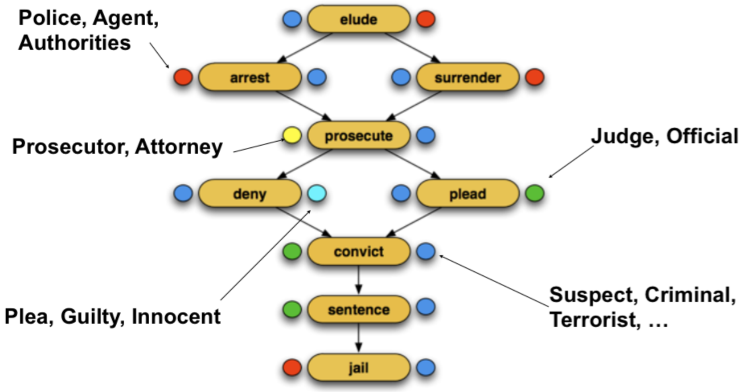


Figure 2.6: An example Narrative Schema learned by Chambers and Jurafsky's 2009 work.

3. Structured selection: At last, they should prune the event space into sets of narrative chains, which is done by clustering events based on their similarity metric.

In their model, a narrative event, called a 'narrative slot', is a tuple of an event (simply a verb) and its participants (arguments), represented as type dependencies. The permissible set of type dependencies are *subject, object, preposition*.

Learning Narrative Schemas

A subsequent work is on learning Narrative Schemas [Chambers and Jurafsky, 2009]. A Narrative Schema is a 'set' of typed narrative chains, as opposed to only one narrative chain. They have mainly addressed two shortcomings of the earlier work:

- Representing the 'types' of the arguments with sets such as 'Police, Agent, Cop', to fill in argument slots.
- Making judgements based on all argument slots, not only the protagonist.

They learn the pair of co-occurring events in an unsupervised manner as before, but using a new similarity metric based on argument types. Then they learn Schemas by pruning the event space based on maximizing the score of all chains participating in one schema. Figure 2.6 shows an example Narrative Schema learned by their method. Comparing this Figure to Figure 2.5 reveals the contribution of this method.

3 Learning Semantically Rich Event Inference Rules Using Definition of Verbs

3.1 Introduction

Systems performing NLP tasks such as Question Answering (QA), Recognizing Textual Entailment (RTE) and reading comprehension depend on extensive language understanding techniques to function. Deep language understanding enables an intelligent agent to construct a coherent representation of the scene intended to be conveyed through natural language utterances, connecting natural language to the concepts of the world. Developing a deep understanding system requires large amounts of conceptual and common-sense understanding of the world. As an example, consider a QA system which is given the question in Figure 3.1. One pre-requisite for answering this question is to semantically understand and interpret both query and the snippet. Figure 3.1 shows a generic semantic interpretation of the question and the snippet with grey labels. Throughout this chapter we use the verbal semantic roles[1] as distinguished by TRIPS

[1] `http://trips.ihmc.us/parser/LFDocumentation.pdf`

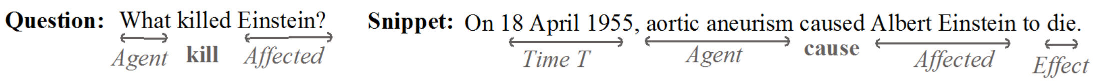

Question: What killed Einstein?
Agent
kill
Affected
Snippet: On 18 April 1955, aortic aneurism caused Albert Einstein to die.
Time T
Agent
cause
Affected
Effect

Figure 3.1: Example question and its corresponding relevant information posed to a question answering system

system [Mehdi H. Manshadi, 2008].

After the semantic interpretation, the system understands that it should look for a *kill* event with Einstein as the *affected* person. However, the system does not see any explicit connection between the event in the question and the event presented in the snippet. Now let us provide the system with the following piece of knowledge in the form of an inference rule about the event *kill*:

$$(X_{agent}\ \textbf{kills}\ Y_{affected}) \xrightarrow{entails} (X_{agent}\ \textbf{causes}\ Y_{affected}\ \textit{to die}) \tag{3.1}$$

By having access to such an inference rule, the system will know that ‘killing’ entails ‘cause to die’, where explicitly the ‘killer’ causes the ‘affected’ to die. One can imagine many more complex pieces of knowledge presented in the form of inference rules, each of which can provide a new clue for a system which requires language understanding. It is obvious that a system should have various natural language processing capabilities in order to successfully answer questions, however, here we focus on the bottleneck of conceptual knowledge on events.

As the earlier example shows, having conceptual knowledge about the events in the form of semantically rich inference rules – such as knowing what happens to the participants before and after it occurs or the consequences of the event – can play a major role in language understanding in different NLP applications. We believe that an effective conceptual knowledge should provide semantic reasoning capabilities, with semantic roles and sense disambiguation. In this chapter, we introduce a novel attempt to automatically learn a semantically rich knowledge base for events, which provides

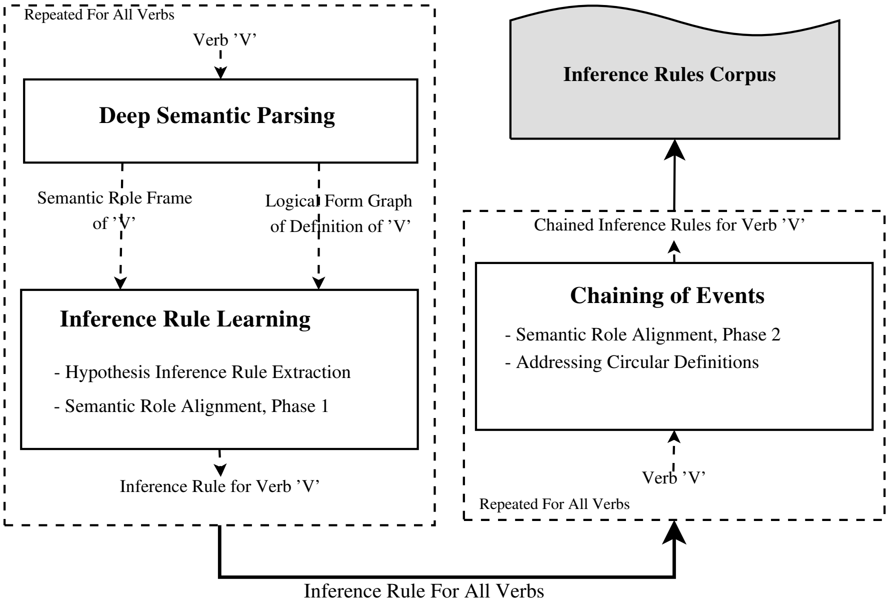

Repeated For All Verbs
Verb 'V'
Deep Semantic Parsing
Semantic Role Frame of 'V'
Logical Form Graph of Definition of 'V'
Inference Rule Learning
- Hypothesis Inference Rule Extraction
- Semantic Role Alignment, Phase 1
Inference Rule for Verb 'V'
Inference Rules Corpus
Chained Inference Rules for Verb 'V'
Chaining of Events
- Semantic Role Alignment, Phase 2
- Addressing Circular Definitions
Verb 'V'
Repeated For All Verbs
Inference Rule For All Verbs

Figure 3.2: The phases of our approach

high precision inference rules aligned by their semantic roles. We propose to learn the knowledge base by automatically processing large amounts of definitional knowledge about verbs, using their WordNet [Miller., 1995] word sense definitions (glosses). We accomplish this by deep semantic parsing of glosses, automatic extraction of inference rules, unsupervised alignment of semantic role labels, and chaining inference rules together until hitting a 'core' concept.

The phases of our approach are shown in Figure 3.2. We will provide details about each of these phases in Sections 3.3-3.5. The main outcome of our approach is the Inference Rules corpus which could be used for different language understanding tasks. In Section 3.6 we show that our semantic role alignment methodology is a promising way for acquiring precise and semantically rich inference rules. Moreover, we show that the inference rules acquired by our approach have higher precision than any other related work. Although we use WordNet here, our approach is applicable to any other definitional resources.

3.2 Related Work

Early research has shown that definitions in online resources (such as dictionaries and lexicons) contain the type of knowledge that systems can benefit from for conceptual understanding of the world [Ide and Vronis., 1994]. More specifically, WordNet's glosses have substantial world knowledge that could leverage semantic interpretation of text [Clark et al., 2008]. Some earlier works [Moldovan and Rus, 2001; Clark et al., 2008] have tackled the problem of encoding WordNet glosses as axioms in first-order logic. These works use syntactically processed glosses for extracting the logical information (e.g., they map the NP in a subject position to an *agent* role), and successfully incorporate these axioms in QA and RTE tasks [Moldovan et al., 2007; Clark et al., 2008]. However, their syntactic representation limits the functionality of semantic representation. Semantically rich logical representations (as opposed to syntactic ones) are proven to perform better on textual similarity and understanding tasks [Blanco and Moldovan, 2013].

The recent work on Multilingual eXtended WordNet [Erekhinskaya et al., 2014] attempts to semantically parse the glosses which is promising. The work on deriving event ontologies [Allen et al., 2013] using WordNet glosses best addresses the shortcomings of semantic interpretation in previous works. It tries to build complex concepts compositionally using OWL-DL description logic and enables reasoning to derive the best classification of knowledge. However, their work mainly derives ontological information, whereas our work extracts full axioms in the form of inference rules. Also, as shown in Section 3.4, we use a more simple approach for expressing our inference rules (axioms), which enables semantic role alignment (a novel task introduced in this chapter), resulting in a more precise, accurate, and easily usable inference rules for other NLP tasks. The earlier works on predicate-argument alignment have been mainly focused on finding lexical similarity and overlaps between pairs of sentences [Wolfe et al., 2013] which is different from aligning the semantic roles of not necessarily similar predicates as we do.

The main relevant work is on automatic acquisition of inference rules. Inference rules, e.g., ‘someone$_x$ commutes $\rightarrow$ someone$_x$ changes positions’, are very useful for tasks such as QA and RTE. The predominant approach, DIRT [Lin and Pantel, 2001], is based on distributional similarity, where two templates (such as ‘X murder Y’ and ‘X kill Y’) are deemed semantically similar if their argument vectors are similar. This similarity measure results in weak (and often incorrect) entailments [Melamud et al., 2013], but results in huge datasets. Among the 12 million DIRT inference rules only about about 50% seem correct and reasonable [Melamud et al., 2013]. One instance of an incorrect rule is ‘X entered Y $\rightarrow$ X left Y’, which captures temporal relation between two predicates, and is incorrect as an entailment. Some later works have attempted to make the inference rules more precise by using lexical expansions for argument vectors [Melamud et al., 2013]. However, their approaches still tend to produce many incorrect or too general entailments, such as ‘Y is hijacked in X $\rightarrow$ Y crashes in X’, which is the result of reporting bias which means there have been many reported hijacking events which have resulted in crashes, but hijacking does not entail crash necessarily. All of the earlier inference rule acquisition approaches mostly use predicates with two arguments, which can result in limited and less-accurate application of rules for textual understanding tasks; however, our approach covers many complex predicate structures with various number of arguments and inter-connected predicates.

VerbOcean [Chklovski and Pantel, 2004] is another related work, which identifies verb entailment through instantiation of some manually constructed patterns. This idea led to more precise rules, but weak coverage since verbs do not co-occur often with patterns. In Section 3.6 we show that our approach outperforms VerbOcean by about 17% in precision and 30% in recall. Another recent work on acquiring inference rules is the work on learning verb inference rules from linguistically-motivated evidence [Weisman et al., 2012]. This work argues that although most of the works on learning inference rules are using distributional similarity, they utilize information from various textual scopes ranging from verb co-occurrence within a sentence to a document, as well as corpus statistics, which results in richer set of linguistically motivated features in their supervised classification framework. Although they outperform some earlier methods, their method is still limited to verb to verb entailment without typed entities and semantic roles, which could make their rules less effective in actual language understanding tasks. In Section 3.6 we show that our approach results in a more accurate verb inference rules, outperforming this work by about 10%. More importantly, our approach attempts to produce semantically rich inference rules, i.e, sense-disambiguated predicates with all of the necessary semantic roles, which is far beyond the simple inference rules produced by this work.

Furthermore, paraphrases can be viewed as bidirectional inference rules. The works on automatic derivation of paraphrase databases [Dolan et al., 2004; Quirk et al., 2004] share some of the shortcomings of the works on acquiring inference rules. Mostly the paraphrase sets with the highest precision contain too general/trivial paraphrase rules [Ganitkevitch et al., 2013] such as ‘higher than 90% ↔ higher than 90 per cent’ or ‘and its relationship ↔ and its link’. Inherently, definitions provide non-trivial pieces of information, so our set of high precision inference rules hardly contains such rules. Unlike these works, we rely on reading definitions instead of web-scale free texts which gives us higher precision of non-trivial inference rules, however, results in a smaller set of rules. By incorporating and linking various definitional resources, one can increase the size of the inference rules yielded by our approach.

Furthermore, paraphrases can be viewed as bidirectional inference rules. There have been many attempts on automatically deriving paraphrase resources such as the MSR Paraphrase and Phrase Table [Dolan et al., 2004; Quirk et al., 2004]. Among the works which extract paraphrases using bilingual pivoting technique, PPDB [Ganitkevitch et al., 2013] extracts lexical, phrasal, and syntactic paraphrases. One strength of PPDB is its pruning method which provides a metric for distinguishing between weak and strong paraphrases based on a paraphrase pair's monolingual distributional similarity score. However, the smallest size of PPDB (containing pairs with highest precision) mainly contains too general or too simple paraphrases, such as 'higher than 90 % ↔ higher than 90 per cent' or 'and its relationship ↔ and its link'.

3.3 Deep Semantic Parsing of Definitions

As the first phase of our approach, we need to have deep semantic understanding of the verb definitions. Here we use the TRIPS broad-coverage semantic parser[2] which produces state-of-the-art logical form (LF) from natural language text [Mehdi H. Manshadi, 2008]. TRIPS provides an essential processing boost beyond other off-the-shelf applications, mainly sense disambiguated and semantically rich deep structures. The approaches presented in this chapter can be applied to any other wide-coverage semantic parsers, such as Boxer system [Bos, 2008].

Many glosses are complex, often highly elliptical and hard to parse. For example, 'kill.v.1'[3] is defined as 'cause to die' which does not explicitly mention the subject or

[2] `http://trips.ihmc.us/parser/cgi/parse`

[3] We represent WordNet words sense disambiguated using their part of speech and sense number. So 'kill.v.1' is the first sense of the verb 'kill'.

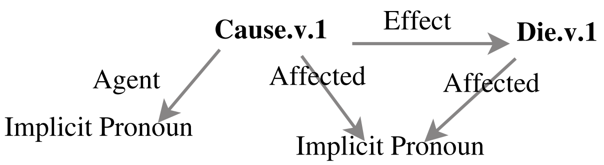


Figure 3.3: Logical form produced by TRIPS for 'kill.v.1'

the object of the sentence. Another example is 'love.v.1' which has the gloss 'have a great affection or liking for', where the object of the sentence is missing. TRIPS recovers such missing information, producing a parse such as 'something causes something to die' [Allen et al., 2013] for the gloss of 'kill.v.1'. The output of this phase is the semantic role frame[4] (semframe) for each verb synset together with LF graph of its gloss. For instance, the semframe of the verb *kill.v.1* is given as $\{agent_{ont:person.n.1}$, $affected_{ont:organism.n.1}\}$. Figure 3.3 shows the simplified LF graph of the gloss of 'kill.v.1'.

3.4 Inference Rule Learning

In the second phase of our approach we aim to extract semantically rich inference rules for all verb synsets.

3.4.1 Hypothesis Inference Rule Extraction

Hypothesis inference rules are preliminary rules which are extracted using the two outputs of the deep semantic parsing phase. A hypothesis rule is an axiom with Left Hand Side (LHS) and Right Hand Side (RHS), each consisted of predicates where LHS log-

[4] Semantic role frame is called to the set of semantic roles associated with a verb.

ically entails RHS. There is always one predicate on the LHS, but there could be more than one predicates on the RHS (one of which is the root predicate, marked with '*'). LHS predicate comes from the semframe of the verb and the RHS predicates come from the LF graph of the verb's definition. We define a predicate to be either a verb or verb nominalization, as they inherently have the potential to occur at some time point as events. Here we stick to a very simple logical representation of axioms in the form of inference rules, which enables easy incorporation of our knowledge base in various systems. For instance, the following is the hypothesis rule that we deterministically extract for the verb 'kill.v.1':

$$\begin{aligned}(\mathbf{kill}.\mathrm{v}.\mathbf{1}\ X_{agent}\ Y_{affected}) \Rightarrow\ & (\mathbf{cause}.\mathrm{v}.\mathbf{1}\ A_{agent}\ B_{affected}\ C_{effect})^{*} \\ & \wedge (\mathbf{die}.\mathrm{v}.\mathbf{1}_{C}\ B_{affected})\end{aligned} \tag{3.2}$$

where each predicate is enclosed within parenthesis which has some arguments (semantic roles) realized with either variables or constants. As you can see, a predicate itself can be an argument of some other predicate, e.g., in the earlier example 'die.v.1' (reified with variable *C*) is the *effect* role of 'cause.v.1'. We call the set of hypothesis inference rules for all the WordNet verb sysnsets $corpus_{hypothesis}$. Now the question is whether a hypothesis rule is usable as an inference rule. The answer is that we often do not get a LF graph with all of the roles recognized correctly; and even if we do, more importantly we still do not know which role on the LHS corresponds to a role on RHS. This issue motivates 'semantic role alignment'.

3.4.2 Semantic Role Alignment, Phase 1

We want to know whether or not it is always the case that the *agent* role in the LHS of a rule maps to the *agent* role in the RHS and they should have the same realization. What happens to the *agent* role of LHS in case there is no *agent* role on the RHS? We call the problem of mapping the roles of the LHS to the roles of RHS 'Semantic Role Alignment' (SRA). The machine translation (MT) community has established an extensive literature on word alignment [Brown et al., 1993; Och and Ney, 2003], where translating 'she came' into French sentence 'elle est venue' requires an alignment between 'she' and 'elle', and between 'came' and 'est venue'. We believe that MT alignment approaches are suitable for the SRA task because of the following reasons:

- The semantic roles on the LHS and RHS tend to have semantic equivalence. So it is intrinsically the case that there is a (partial) mapping from roles on the RHS to the roles on the LHS.
- As opposed to the kind of inference rules learned based on distributional similarity [Harris, 1985] (to be discussed in Section 3.2), here the semantic content of LHS should not diverge substantially from RHS given the fact that RHS is basically defining LHS.
- MT alignment models are typically trained in an unsupervised manner, depending on sentence-aligned parallel corpora. For our task large volumes of training data are lacking, so an unsupervised training (to be explained in this section) is the most suitable approach.
- As it will be discussed in Section3.6, unsupervised aligners (which find hidden structures in data) can actually account for some frequent parsing errors in our system, which is very promising.

We model the SRA problem as a maximum bipartite matching problem: for each inference rule, we define n_{lhs} as the set of nodes such as l_i, each of which corresponds to a role in LHS; n_{rhs} is another set of nodes such as r_j, each of which corresponds to a role in RHS. Each pair of nodes (l_i, r_j) has an edge connecting them, which is weighted with the plausibility of the alignment of that pair. An alignment function a is defined as follows:

$$a : n_{lhs} \rightarrow n_{rhs} \cup \{null\}$$

which is a function mapping each role $\in n_{lhs}$ to a role $\in n_{rhs}$ or a null symbol, similar to IBM-style machine translation model [Brown et al., 1993]. Here, mapping a LHS role to *null* means that the role should be 'inserted' in the RHS. Then the SRA problem is considered as a maximum weighted matching problem where the best alignment for the inference rule is the highest scoring a^*, under the constraint of 'one-to-one' matching, which is defined as follows:

$$a^* = arg\,\underset{a}{max}\{score(n_{lhs}, a, n_{rhs})\}$$

$$score(n_{lhs}, a, n_{rhs}) = \sum_{\substack{l_i \in n_{lhs} \\ r_j \in n_{rhs}}} score(l_i, a, r_j)$$

$$score(l_i, a, r_j) = log(Pr(l_i, a|r_j))$$

The training of the probability of aligning a role on LHS to a role on RHS, $Pr(l_i, a|r_j)$, is accomplished using the Expectation Maximization (EM) algorithm [Brown et al., 1993]. In the E-step the expected counts for each role pair (l_i, r_j) are calculated and in M-step we normalize and maximize. We mainly estimate the so-called translation probability parameter $t(r|l)$ [Brown et al., 1993].

In order to prepare the data for performing the alignment explained above, we should firstly build an appropriate parallel corpus. Our idea is to build a corpus of LHS roles parallel with RHS roles from the set of hypothesis inference rules for all verbs in $corpus_{hypothesis}$. One issue to consider is that the rules with multi-predicate RHS cannot have a two-sided mapping. Among all 13,249 hypothesis inference rules

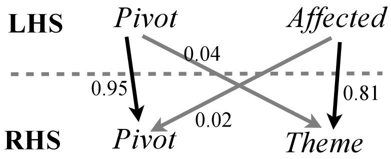


Figure 3.4: The bipartite matching graph for alignment of inference rule 3.3

that we generate, 649 one of them have two RHS predicates and only 10 of them have three RHS predicates. As the first step, we remove all the rules with multi-predicate RHS – about 0.4% of all the rules. With the remaining rules, we build a corpus of all LHS roles parallel with the RHS roles. We call this $corpus_{unary}$. Then we apply the alignment algorithm explained earlier to this corpus for learning the model parameters. Using the learned parameters, for each hypothesis inference rule we find the maximum weighted alignment. As an example, consider the verb *digest.v.3* which is defined as 'to tolerate something or somebody unpleasant'. The hypothesis inference rule produced for this verb is as follows:

$$(\mathbf{digest}.\mathbf{v}.\mathbf{3}\ X_{pivot}\ Y_{affected}) \Rightarrow (\mathbf{tolerate}.\mathbf{v}.4\ A_{pivot}\ B_{theme})^{*} \tag{3.3}$$

Figure 3.4 shows the bipartite matching graph for this inference rule. The maximum weighted matching is shown by the dark edges. As a result of maximum weighted matching, the aligned inference rule for 'digest.v.3' is the following:

$$(\mathbf{digest}.\mathbf{v}.\mathbf{3}\ X_{pivot}\ Y_{affected}) \Rightarrow (\mathbf{tolerate}.\mathbf{v}.4\ X_{pivot}\ Y_{affected})^{*} \tag{3.4}$$

We evaluate the outcome of this experiment, called '$phase1_{unary}$', in Section 3.6.

Our approach for SRA of the inference rules with multi-predicate RHS is linguistically motivated by the fact that the root predicate captures the core semantic meaning of the LHS. In short, our approach is as follows:

- Step 1: Discard the non-root RHS predicate and find the maximum weighted matching between LHS and the root RHS[5].

- Step 2: Make a set of nodes from the LHS roles which are matched to NULL in Step 1. Use this set as a new LHS, then find the maximum weighted matching to the non-root predicate.

This approach can generalized as a recursive SRA for rules which have more than two RHS predicates. The results of this experiment, named '$phase1_{bin}$', can be reviewed in Section 3.6. Applying this alignment approach the inference rule (3.2) results in the following aligned rule:

$$\begin{aligned}&(\mathbf{kill}.\mathbf{v}.\mathbf{1}\ X_{agent}\ Y_{affected})\\ &\quad\Rightarrow(\mathbf{cause}.\mathbf{v}.\mathbf{1}\ X_{agent}\ Y_{affected}\ C_{effect})^{*}\wedge(\mathbf{die}.\mathbf{v}.\mathbf{1}_{C}\ Y_{affected})\end{aligned} \tag{3.5}$$

At the end of this phase, we will have an aligned and ready to use level-1 inference rule generated for each WordNet verb synset. We call this collection $corpus_{level-1-rules}$. We associate a score with each inference rule, which is its normalized weighted matching score.

3.5 Chaining of Events

Given the inference rule for each verb from the previous phase, we want to expand our understanding of each event by chaining verbs together. For example, consider a QA system that has encountered the sentence "the boy skinned his knee when he fell" and wants to know more about the concept of 'skinning' by looking up the verb 'skin.v.2'

[5] It is evident that a roles which is realized with the reification of another predicate, as with the *effect* role in (3.2), does not take part in the alignment problem.

in our knowledge base. The ideal information that we would like to be able to get by forward chaining of the level-1 inference rules is as follows:

$\mathcal{X}$ **skin.v.2** $\mathcal{Y}$:

$\xrightarrow{means}$ $\mathcal{X}$ **bruise.v.1** the [skin] of $\mathcal{Y}$

$\xrightarrow{means}$ $\mathcal{X}$ **injure.v.1** [the underlying soft tissue] of the [skin] of $\mathcal{Y}$

$\xrightarrow{means}$ $\mathcal{X}$ **cause.v.1** [harm] to the [underlying soft tissue] of the [skin] of $\mathcal{Y}$

Obtaining the above chaining requires yet another phase of role alignment, going from each level to the next one and expanding each predicate on the RHS.

3.5.1 Semantic Role Alignment, Phase 2

Consider the inference rule (3.5) which we obtained in the previous section. For the expansion of 'die.v.1' on the RHS, we will use its inference rule which is as follows:

$$(\mathbf{die.v.1}\ X_{agent}) \Rightarrow (\mathbf{lose.v.1}\ X_{agent}\ bodily\_attributes_{theme}) \tag{3.6}$$

As you can see the semframe of 'die.v.1' in inference rule (3.6) does not match its semframe in inference rule (3.5). There are many cases similar to this one and the reason is that semantic parsing and sense disambiguation are not perfect and are error prone. Moreover, verbs can have different semframes in different contexts. Here we perform semantic role alignment phase 2, using a similar method to '$phase1_{unary}$'. This time we build a corpus of all LHS definitions parallel with any of their usages in the entire $corpus_{level-1-rules}$. We call this new corpus $corpus_{def-use}$. EM can find hidden error patterns here as well as actual semantic alignment patterns. The results of this experiment named '$phase2_{EM}$' can be reviewed in Section 3.6.

After finding the maximum weighted matching using the trained parameters, we get a new inference rule proper for continuing the forward chaining on 'kill.v.1' which is as follows:

$$(\mathbf{die}.\mathrm{v}.\mathbf{1}\ X_{affected}) \Rightarrow (\mathbf{lose}.\mathrm{v}.\mathbf{1}\ X_{affected}\ bodily\_attributes_{theme}) \quad (3.7)$$

We have also obtained a probabilistic distribution on semframes for each synset given context, using the parsed glosses. We used this statistics together with EM alignment for favoring a specific semframe over another, resulting in a higher precision alignment in phase 2. The results of this experiment named '$phase2_{EM+}$' is reported in section 3.6.

The second phase of semantic role alignment results in high precision chaining and on average we get 10 new inference rules with high matching score after three levels of chaining – which increases the size of our inference rules corpus by an order of magnitude. For instance consider the verb 'kill.v.14', which is defined as 'to cause to cease operating'. This verb has three RHS predicates: cause, cease and operate and could have 2^3 different expansions just for the first level.

3.5.2 Addressing Circular Definitions

Usually there are some circular definitions for words in definitional resources including WordNet [Allen et al., 2011; Ide and Vronis., 1994]. For example, the synset 'cause.v.1' is defined as 'cause.v.1 to happen.v.1' which is an immediate circulation. There have been some preliminary strategies [Allen et al., 2011] for breaking the definition cycles. Those findings show that some cycles can be resolved by selecting an alternative sense for the cyclical definition or simplifying the definitions. However, there are some key cycles which cannot be broken in this manner because there is essentially no specific simpler definition for some concepts, e.g., 'cause'. This is an essential problem with machine understanding, because machines have no direct experience with the world, which could have enabled them understand what a natural concept means.

This issue brings up an important psycholinguistic research, where it is believed that human lexicon is a complicated web of semantically related nodes instead of a one-to-one mapping of concepts to the words [Levary et al., 2012]. According to earlier work, dictionaries have a set of highly interconnected nodes from which all other words can be defined [Picard et al., 2009]. To our knowledge, there has not been any research on finding core concepts on WordNet verbs, using the graph theory experiments. Continuing the work on building dictionary graphs [Levary et al., 2012], we built a graph where directed links are drawn from a word to the words in its definition. In this graph, we found the strongly connected components using Tarjan's algorithm [Tarjan, 1972], which resulted in a set of 56 strongly connected components with size bigger than 1, which included 158 verb synsets. Other definitional paths of WordNet verbs converge to this set quickly, which we call the core verbs. Our idea is to stop forward chaining of definitions (avoiding circulation trap) when we hit a core verb. The main core concepts that we have identified are as follows: cause, make, be, do, stop, start, begin, end, have, prevent, enable, disable.

After chaining of events and SRA phase 2, we obtain our final corpus of Inference Rules, containing new rules derived from chaining started from level-1 rules and going up to higher levels. We assign a score to each inference rule in different levels of the final corpus, which is the sum of normalized weighted matching scores divided to the number of levels.

3.6 Evaluation and Results

We have conducted two focused experiments for evaluating the two major contributions of our approach.

Semantic Role Alignment: We attempted to build a gold-standard corpus on semantic role alignment. For annotators, we used seven linguistics experts who had no relation to the work and three researchers who were involved. For each individual annotator, we randomly sampled 100 hypothesis inference rules from the $corpus_{hypothesis}$, and asked them to perform the role alignment for the given hypothesis rule[6]. The role alignment task was either of the following actions towards each RHS role:

- *Substitute*: substitute the role with one of the LHS roles (also decide about the realization value). This action corresponds to a role matching from LHS to RHS.
- *Delete:* Completely remove the role.

Moreover, they had the option of performing *Add* action, which involves adding a new role on RHS. This action corresponds to matching a LHS role with NULL.

The annotators were also asked to assign a confidence score (out of three) to the resultant aligned inference rule. This score takes into account the cases in which there is really no good alignment, and the annotator feels that his/her best possible alignment is not good at all[7]. We used this gold standard for computing precision scores for the SRA Phase 1 methods: $phase1_{base}$, $phase1_{unary}$, and $phase1_{bin}$. As it is critical to get the exact output, we used strict evaluation with no partial credit. We performed the same procedure on $corpus_{def-use}$, and built a gold-standard for evaluating the precision of SRA Phase methods: $phase2_{base}$, $phase2_{EM}$, and $phase2^{+}_{EM+}$). Both $phase1_{base}$ and $phase2_{base}$ are baselines which deterministically align LHS roles with the same RHS roles[8]. The results of these experiments reporting precision and average confidence score is presented in Table 3.1. In this table, Sc_{err} is the average annotator confidence

[6] We presented each rule together with some example usages of the synset, to give the annotators the context [Szpektor et al., 2007].

[7] This mostly happens for vague definitions or essential parsing errors.

[8] For 78% of all verb synsets we could find an exact name-based role match going from LHS to RHS.

Method	$phase1_{base}$	$phase1_{unary}$	$phase1_{bin}$	$phase2_{base}$	$phase2_{EM}$	$phase2_{EM+}$
Precision	51%	**90%**	**87%**	10%	72%	79%
Sc_{corr}	3.0	2.7	2.5	3.0	2.28	2.32
Sc_{err}	2.5	2.3	1.2	2.7	1.3	1.4

Table 3.1: Semantic role alignment evaluation results

score on incorrect alignments and Sc_{corr} is the average annotator confidence score on correct alignments of the corresponding method.

The results show that the alignment using EM performs very well, providing promising framework for the task of semantic role alignment. The points that the system has missed are mostly for 'Delete' actions (91% of the time) of the annotators. System prefers not to delete any piece of information from the RHS, as it might be necessary for next chaining levels. However, there are some roles on the RHS which are artifacts of bad parse or inconsistent definitions, which annotator can pinpoint but the system cannot. Parsing artifacts are quite easy to be corrected by human, so the average confidence score on those errors is high, which has resulted in pretty high Sc_{err}.

The results obtained for $phase1_{bin}$ show that the alignment on binary rules (which initially seemed more complex) performs as well as the alignment on unary rules. Our observations show that this is because of the fact that many of binary rules are composed of a core predicate such as 'cause', 'stop', or 'do' which all have a recurring usage pattern, making the unsupervised alignment more successful. The baseline $phase1_{base}$ performs mediocre as a simple alignment method for phase1. Phase 2 alignment is always more complicated than phase 1. The baseline $phase2_{base}$ performs very poorly because verbs are mostly used (in context) with different semframe as compared with the semframe they are defined with (out of the context). The method $phase2_{EM}$ performs good, but is not enough for handling complicated alignments in def-use cases.

The low Sc_{err} for phase 2 methods indicate the complexity of alignment task at this phase. $Phase2_{EM+}$ outperforms $phase2_{EM}$, which is mainly because it better predicts the cases of occasional bad parsing.

Inference Rules Corpus: To our knowledge, none of the earlier works on acquiring inference rules (details in Section 3.2) have inference rules with complex and semantically rich semantic roles and sense disambiguation as we do. Hence, in order to compare our inference rules to earlier works we simplify our inference rules dataset, removing all the sense tags and semantic roles. Here we use the most recent manually created verb inference rules dataset [Weisman et al., 2012], hereafter, test-set. This test-set is created by randomly sampling 50 common verbs in the Reuters corpus, and is then randomly paired with 20 most similar verbs according to the Lin similarity measure (Lin, 1998). This dataset includes 812 verb pairs, which are manually annotated by the authors as representing a valid entailment rule or not. They have used rule-based approach for annotation of entailment, where a rule $v1 \rightarrow v2$ is annotated ‘yes’ if the annotator could think of plausible contexts under which the rule holds [Szpektor et al., 2004]. In this dataset 225 verb pairs are labeled as entailing and 587 verb pairs were labeled as non-entailing. Although this dataset is not very rich, it is a good testbed for comparing our inference rules against the state-of-the-art work on verb inference rules. Table 3.2 shows the results of the following methods:

- $Semantic - Rules_{simplified}$ is our simplified approach: given our final Inference Rules corpus (containing rules up to three levels of chaining or until hitting a core concept), simplify the rules by removing all the semantic roles, all the sense tags, and introduce a new rule for each of the predicates of a multi-predicate RHS. Given a pair (v_1, v_2) from the test-set, if the entailment $v_1 \rightarrow v_2$ exists in the simplified corpus, classify the pair as ‘yes’.

Method	Precision	Recall	F1-Score
$Semantic - Rules_{simplified}$	**50.0%**	45.1%	0.47
$Supervised_{linguistically-motivated}$	40.2%	71.0%	0.51
$VerbOcean_{KB}$	33.1%	14.8%	0.2
$Random$	27.9%	28.8%	0.28

Table 3.2: Evaluation results on hand annotated verb entailment pairs test-set

- $Supervised_{linguistically-motivated}$ is the work on supervised learning of verb inference rules from linguistically-motivated evidence [Weisman et al., 2012].
- $VerbOcean_{KB}$ is the method that classifies a given pair as 'yes' if the pair appears in the strength relation in the VerbOcean knowledge-base [Chklovski and Pantel, 2004].
- $Random$ is the method that randomly classifies a pair as 'yes' with a probability 27.7%, proportional to the number of 'yes' instances in the test-set against the number of 'no' instances.

As the results show, our simplified method outperforms the best method by 10% in precision. This reveals that the accuracy of our inference rules is high and our approach is capable of acquiring more precise verb inferences than the other methods. As expected, our coverage is lower than the $Supervised$ method, which is due to the fact that we acquire our rules by reading verb definition and not by mining significantly large web-scale corpora, resulting in a smaller-scale dataset. However, our recall outperforms the $VerbOcean$ method and has also a competing F-1 score compared with the $Supervised$ method. Of course for a successful usage of a knowledge base in an application, accuracy is crucial and coverage can be mitigated by using various kinds of precise knowledge bases. A large but noisy and unreliable knowledge base will be of little use in reasoning.

Analyzing the pairs that we have miss-classified as 'yes', there are many pairs which do not seem to be correctly annotated as ‘no’ in the test-set, such as (reveal, disclose) and (require, demand), where we argue that according to rule-based approach one can indeed think of a reasonable context under which $reveal \rightarrow disclose$ and $require \rightarrow demand$ hold. Another example is the pair (stop, prevent), which we classify as ‘yes’ in the context of the sixth sense of the verb ‘stop’, but is classified as ‘no’ in the test-set as the verbs in the test-set are not sense-disambiguated and do not have any context. Overall, our simplified approach proves to be competent with other works and also outperforms the state-of-the-art in precision, which is very promising.

3.7 Conclusion

We presented a novel attempt to automatically build a conceptual knowledge about events in the form of inference rules, which can serve as a semantically rich knowledge base useful for various language understanding tasks. We accomplish this by deep semantic parsing of glosses, inference rule learning enhanced by semantic role alignment, and chaining of the events. The evaluation results show that our semantic role alignment technique is very promising and our inference rules are precise and informative pieces of knowledge. We have shown that learning inference rules by reading definitional resources can result in high accuracy and inherently non-trivial pieces of knowledge. In order to expand the coverage of our knowledge base, we are planning to apply our approach to other dictionaries. Moreover, we are looking into improving our semantic role alignment techniques for chaining of events, which can potentially result in more accurate inference rules. Our future goal is to experiment employing our definitional knowledge in QA and Reading Comprehension Tests.

One of the downsides of learning and representing knowledge about events in the form of Inference Rules is that you cannot model the interaction between various events as the rules are not interconnected. In Chapters 4 and 5 we will propose different methods for representing and learning a rich causal and temporal network of events, the result of which can significantly improve the inference tasks requiring causal understanding of events.

4 Commonsense Story Understanding

4.1 Introduction

Story understanding is an extremely challenging task in natural language understanding with a long-running history in AI [Charniak, 1972; Winograd, 1972; Turner, 1994; Schubert and Hwang, 2000]. Recently, there has been a renewed interest in story and narrative understanding based on progress made in core NLP tasks. This ranges from generic story telling models to building systems which can compose meaningful stories in collaboration with humans [Swanson and Gordon, 2008]. Perhaps the biggest challenge of story understanding (and story generation) is having commonsense knowledge for the interpretation of narrative events. The question is how to provide commonsense knowledge regarding daily events to machines.

A large body of work in story understanding has focused on learning scripts [Schank and Abelson, 1977]. Scripts represent structured knowledge about stereotypical event sequences together with their participants. It is evident that various NLP applications (text summarization, co-reference resolution, question answering, etc.) can hugely benefit from the rich inferential capabilities that structured knowledge about events can provide. Given that developing hand-built scripts is extremely time-consuming, there is a serious need for automatically induced scripts. Most relevant to this issue is work on unsupervised learning of 'narrative chains' [Chambers and Jurafsky, 2008] and event schemas [Chambers and Jurafsky, 2009; Balasubramanian et al., 2013; Cheung et al., 2013; Nguyen et al., 2015]. The first requirement of any learner is to decide on a corpus to drive the learning process. We are foremost interested in a resource that is full of temporal and causal relations between events because causality is a central component of coherency. Personal stories from daily weblogs are good sources of commonsense causal information [Gordon and Swanson, 2009; Manshadi et al., 2008], but teasing out useful information from noisy blog entries is a problem of its own. Consider the following snippet from ICWSM 2011 Spinn3r Dataset of Weblog entries [Burton et al., 2009]:

> "I had an interesting day in the studio today. It was so interesting that I took pictures along the way to describe it to you. Sometimes I like to read an autobiography/biography to discover how someone got from there to here.....how they started, how they traveled in mind and spirit, what made them who they are now. Well, today, my work was a little like that."

This text is full of discourse complexities. A host of challenging language understanding tasks are required to get at the commonsense knowledge embedded within such text snippets. What is needed is a simplified version of these narratives. This chapter introduces a new corpus of such short commonsense stories. With careful prompt design and multiple phases of quality control, we collected 50k high quality five-sentence stories that are full of stereotypical causal and temporal relations between events. The corpus not only serves as a resource for learning commonsense narrative schemas, but is also suitable for training story generation models. We describe this corpus in detail in Section 4.3.

This new corpus also addresses a problem facing script learning over the past few years. Despite the attention scripts have received, progress has been inhibited by the lack of a systematic evaluation framework. A commonly used evaluation is the 'Narrative Cloze Test' [Chambers and Jurafsky, 2008] in which a system predicts a held-out event (a verb and its arguments) given a set of observed events. For example, the following is one such test with a missing event: {X threw, pulled X, told X, ???, X completed}[1]. As is often the case, several works now optimize to this specific test, achieving higher scores with shallow techniques. This is problematic because the models often are not learning commonsense knowledge, but rather how to beat the shallow test.

This chapter thus introduces a new evaluation framework called the Story Cloze Test. Instead of predicting an event, the system is tasked with choosing an entire sentence to complete the given story. We collected 3,742 doubly verified Story Cloze Test cases. The test is described in detail in Section 4.4.

Finally, this chapter proposes several models, including the most recent state-of-the-art approaches for the narrative cloze test, for tackling the Story Cloze Test. The results strongly suggest that achieving better than random or constant-choose performance requires richer semantic representation of events together with deeper levels of modeling the semantic space of stories. We believe that switching to the Story Cloze Test as the empirical evaluation framework for story understanding and script learning can help direct the field to a new direction of deeper language understanding.

[1] Narrative cloze tests were not meant to be human solvable.

4.2 Related Work

Several lines of research have recently focused on learning narrative/event representations. Chambers and Jurafsky first proposed narrative chains [Chambers and Jurafsky, 2008] as a partially ordered set of narrative events that share a common actor called the 'protagonist'. A narrative event is a tuple of an event (a verb) and its participants represented as typed dependencies. Several expansions have since been proposed, including narrative schemas [Chambers and Jurafsky, 2009], script sequences [Regneri et al., 2010], and relgrams [Balasubramanian et al., 2013]. Formal probabilistic models have also been proposed to learn event schemas and frames [Cheung et al., 2013; Bamman et al., 2013; Chambers, 2013; Nguyen et al., 2015]. These are trained on smaller corpora and focus less on large-scale learning. A major shortcoming so far is that these models are mainly trained on news articles. Little knowledge about everyday life events are learned.

Several groups have directly addressed script learning by focusing exclusively on the narrative cloze test. Jans et al. [Jans et al., 2012] redefined the test to be a text ordered sequence of events, whereas the original did not rely on text order [Chambers and Jurafsky, 2008]. Since then, others have shown language-modeling techniques perform well [Pichotta and Mooney, 2014a; Rudinger et al., 2015]. This chapter shows that these approaches struggle on the richer Story Cloze evaluation.

There has also been renewed attention toward natural language comprehension and commonsense reasoning [Levesque, 2011; Roemmele et al., 2011; Bowman et al., 2015]. There are a few recent frameworks for evaluating language comprehension [Hermann et al., 2015; Weston et al., 2015], including the MCTest [Richardson et al., 2013] as a notable one. Their framework also involves story comprehension, however, their stories are mostly fictional, on average 212 words, and geared toward children in grades

1-4. Some progress has been made in story understanding by limiting the task to the specific domains and question types. This includes research on understanding newswire involving terrorism scripts [Mueller, 2002], stories about people in a restaurant where a reasonable number of questions about time and space can be answered [Mueller, 2007], and generating stories from fairy tales [McIntyre and Lapata, 2009]. Finally, there is a rich body of work on story plot generation and creative or artistic story telling [Méndez et al., 2014; Riedl and León, 2008]. This chapter is unique to these in its corpus of short, simple stories with a wide variety of commonsense events. We show these to be useful for learning, but also for enabling a rich evaluation framework for narrative understanding.

4.3 ROCStories: A Corpus of Short Commonsense Stories

We aimed to build a corpus with two goals in mind:

1. The corpus contains a *variety* of commonsense causal and temporal relations between everyday events. This enables learning narrative structure across a range of events, as opposed to a single domain or genre.

2. The corpus is a high quality collection of non-fictional daily short life stories, which can be used for training rich coherent story-telling models.

In order to narrow down our focus, we carefully define a narrative or story as follows: ‘A narrative or story is anything which is told in the form of a causally (logically) linked set of events involving some shared characters’. The classic definition of a story requires having a plot, (e.g., a character following a goal and facing obstacles), however, here we are not concerned with how entertaining or dramatic the stories are. Instead, we are concerned with the essence of actually being a logically meaningful story. We follow the notion of ‘storiness’ [Forster, 1927; Bailey, 1999], which is described as “the expectations and questions that a reader may have as the story develops”, where expectations are ‘common-sense logical inferences’ made by the imagined reader of the story.

We propose to satisfy our two goals by asking hundreds of workers on Amazon Mechanical Turk (AMT) to write novel five-sentence stories. The five-sentence length gives enough context to the story without allowing room for sidetracks about less important or irrelevant information in the story. In this Section we describe the details about how we collected this corpus, and provide statistical analysis.

4.3.1 Data Collection Methodology

Crowdsourcing this corpus makes the data collection scalable and adds to the diversity of stories. We tested numerous pilots with varying prompts and instructions. We manually checked the submitted stories in each pilot and counted the number of submissions which did not have our desired level of coherency or were specifically fictional or offensive. Three people participated in this task and they iterated over the ratings until everyone agreed with the next pilot’s prompt design. We achieved the best results when we let the workers write about anything they have in mind, as opposed to mandating a pre-specified topic. The final crowdsourcing prompt can be found in supplementary material.

The key property that we had enforced in our final prompt was the following: the story should read like a coherent story, with a specific *beginning* and *ending*, where *something happens* in between. This constraint resulted in many causal and temporal

✗	The little puppy thought he was a great basketball player. He challenged the kitten to a friendly game. The kitten agreed. Kitten started to practice really hard. Eventually the kitten beat the puppy by 40 points.
✓	Bill thought he was a great basketball player. He challenged Sam to a friendly game. Sam agreed. Sam started to practice really hard. Eventually Sam beat Bill by 40 points.
✗	I am happy with my life. I have been kind. I have been successful. I work out. Why not be happy when you can?
✗	The city is full of people and offers a lot of things to do. One of my favorite things is going to the outdoor concerts. I also like visiting the different restaurants and museums. There is always something exciting to do in the city.
✓	The Smith family went to the family beach house every summer. They loved the beach house a lot. Unfortunately there was a bad hurricane once. Their beach house was washed away. Now they lament the loss of their beach house every summer.
✗	Miley was in middle school. ~~She lived in an apartment~~. Once Miley made a mistake and cheated in one of her exams. She tried to hide the truth from her parents. After her parents found out, they grounded her for a month.
✓	Miley was in middle school. She usually got good grades in school . Once Miley made a mistake and cheated in one of her exams. She tried to hide the truth from her parents. After her parents found out, they grounded her for a month.

Table 4.1: Examples of good and bad stories provided to the crowd-sourced workers. Each row emphasizes one of the three properties that each story should satisfy: (1) being realistic, (2) having clear beginning and ending, and (3) not stating anything irrelevant to the story.

X *challenge* Y - Y *agree* play — Y *practice* —— Y *beat* X

Figure 4.1: An example narrative chain with characters X and Y.

links between events. Table 4.1 shows the examples we provided to the workers for instructing them about the constraints. We set a limit of 70 characters to the length of each sentence. This prevented multi-part sentences that include unnecessary details. The workers were also asked to provide a title that best describes their story. Last but not least, we instructed the workers not to use quotations in their sentences and avoid using slang or informal language.

Collecting high quality stories with these constraints gives us a rich collection of commonsense stories which are full of stereotypical inter-event relations. For example, from the good story in first row of Table 4.1, one can extract the narrative chain represented in Figure 4.1. Developing a better semantic representation for narrative chains which can capture rich inter-event relations in these stories is a topic of future work.

Quality Control: One issue with crowdsourcing is how to instruct non-expert workers. This task is a type of creative writing, and is trickier than classification and tagging tasks. In order to ensure we get qualified workers, we designed a qualification test on AMT in which the workers had to judge whether or not a given story (total five stories) is an acceptable one. We used five carefully selected stories to be a part of the qualification test. This not only eliminates any potential spammers on AMT, but also provides us with a pool of creative story writers. Furthermore, we qualitatively browsed through the submissions and gave the workers detailed feedback before approving their submissions. We often bonused our top workers, encouraging them to write new stories on a daily basis.

Statistics: Figure 4.2 shows the distribution of number of tokens of different sentence positions. The first sentence tends to be shorter, as it usually introduces characters or sets the scene, and the fifth sentence is longer, providing more detailed conclusions to the story. Table 4.2 summarizes the statistics of our crowdsourcing effort. Figure 4.3 shows the distribution of the most frequent 50 events in the corpus. Here we count event as any hyponym of 'event' or 'process' in WordNet [Miller., 1995]. The top two events, 'go' and 'get', each comprise less than 2% of all the events, which illustrates the rich diversity of the corpus.

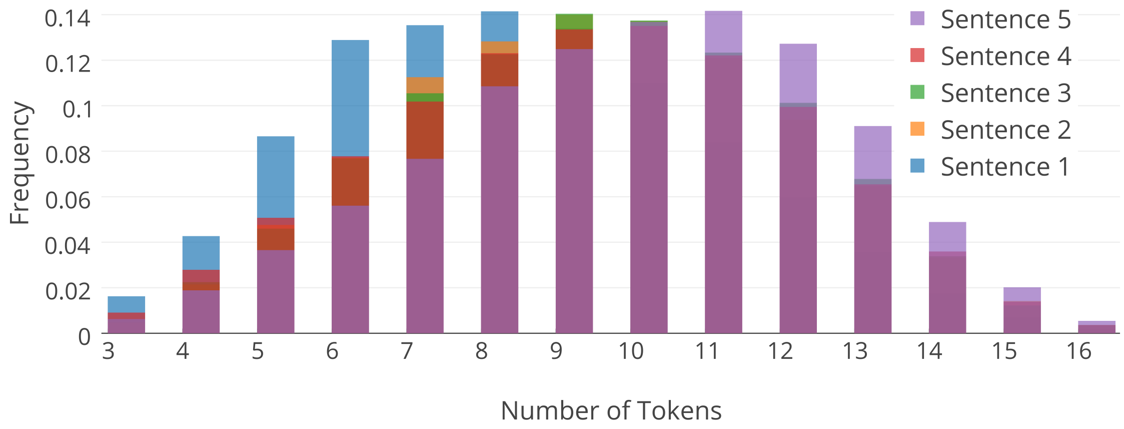


Figure 4.2: Number of tokens in each sentence position.

# submitted stories	49,895
# approved stories	**49,255**
# workers participated	932
Average # stories by one worker	52.84
Max # stories written by one worker	3,057
Average work time among workers (minute)	4.80
Median work time among workers (minute)	2.16
Average payment per story (cents)	26

Table 4.2: Crowdsourcing worker statistics.

Figure 4.4 visualizes the n-gram distribution of our story titles, where each radial path indicates an n-gram sequence. For this analysis we set n=5, where the mean num-

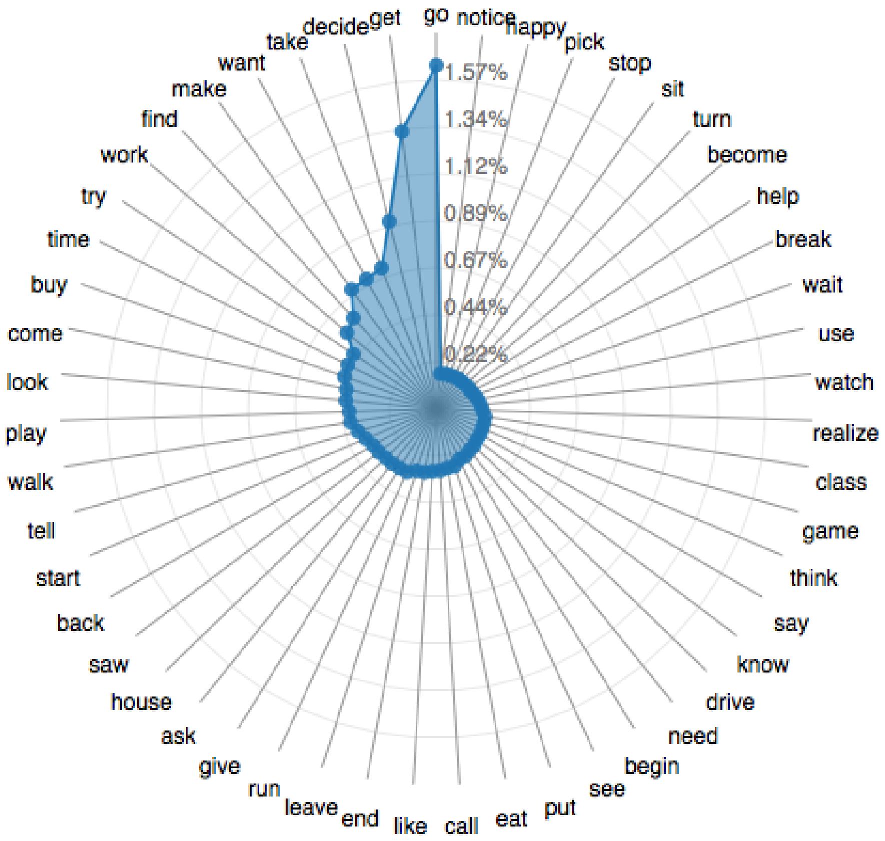


Figure 4.3: Distribution of top 50 events in our corpus.

ber of tokens in titles is 9.8 and median is 10. The ‘end’ token distinguishes the actual ending of a title from five-gram cut-off. This figure demonstrates the range of topics that our workers have written about. The full circle reflects on 100% of the title n-grams and the n-gram paths in the faded 3/4 of the circle comprise less than 0.1% of the n-grams. This further demonstrates that the range of topics covered by our corpus is quite diverse. A full dynamic visualization of these n-grams can be found here: `http://goo.gl/Qhg60B`.

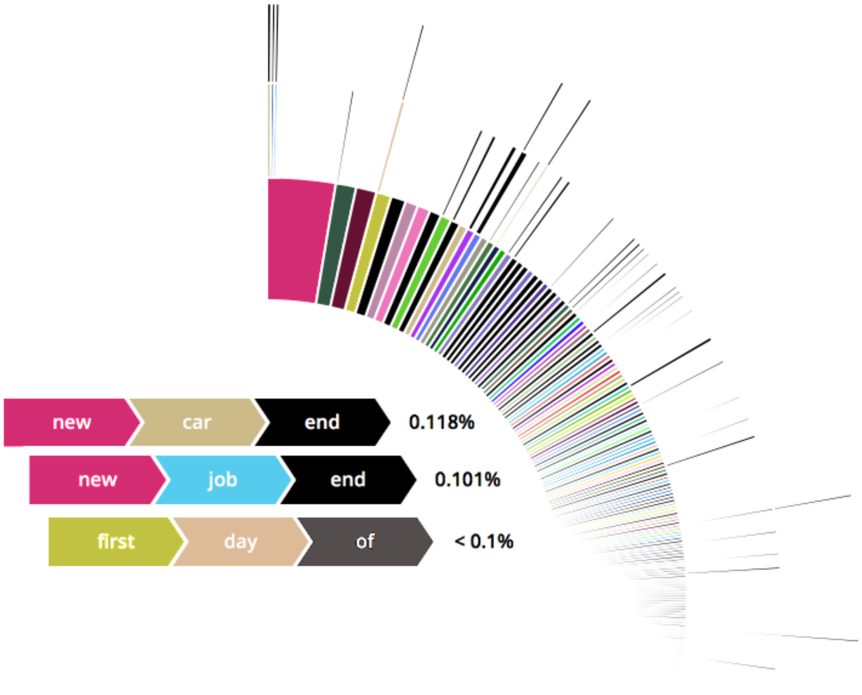


Figure 4.4: N-gram distribution of story titles.

4.3.2 Corpus Release

The corpus is publicly available to the community and can be accessed through `http://cs.rochester.edu/nlp/rocstories`, which will be grown even further over the coming years. Given the quality control pipeline and the creativity required from workers, data collection goes slowly.

We are also making available semantic parses of these stories. Since these stories are not newswire, off-the-shelf syntactic and shallow semantic parsers for event extraction often fail on the language. To address this issue, we customized search parameters and added a few lexical entries[2] to TRIPS broad-coverage semantic parser[3], optimizing its performance on our corpus. TRIPS parser [Allen et al., 2008] produces state-of-the-

[2] For example, new informal verbs such as ‘vape’ or ‘vlog’ have been added to the lexicon of this semantic parser.

[3] `http://trips.ihmc.us/parser/cgi/step`

art logical forms for input stories, providing sense disambiguated and ontology-typed rich deep structures which enables event extraction together with semantic roles and coreference chains throughout the five sentences.

	Good-Stories$_{50}$	*Random-Stories*$_{50}$
% perfectly ordered, taking majority ordering for each of the 50 stories	100	86
% all sentences perfectly ordered, out of 250 orderings	95.2	82.4
% $\leq$ 1 sentences misplaced, rest flow correctly, out of 250 orderings	98.0	96.0
% correct placements of each position, 1 to 5	**98.8**, 97.6, 96, 96, **98.8**	**95.6**, 86, 86.8, 91.2, **96.8**

Table 4.3: Results from the human temporal shuffling experiment.

4.3.3 Temporal Analysis

Being able to temporally order events in the stories is a pre-requisite for complete narrative understanding. Temporal analysis of the events in our short commonsensical stories is an important topic of further research on its own. In this Section, we summarize two of our analyses regarding the nature of temporal ordering of events in our corpus.

Shuffling Experiment: An open question in any text genre is how text order is related to temporal order. Do the sentences follow the real-world temporal order of events? This experiment shuffles the stories and asks AMT workers to arrange them back to a coherent story. This can shed light on the correlation between the original position of the sentences and the position when another human rearranges them in a commonsensically meaningful way. We set up this experiment as follows: we sampled two sets of 50 stories from our corpus: *Good-Stories*$_{50}$ and *Random-Stories*$_{50}$. *Good-*

Stories$_{50}$[4] is sampled from a set of stories written by top workers who have shown shown consistent quality throughout their submissions. *Random-Stories*$_{50}$[5] is a random sampling from all the stories in the corpus. Then we randomly shuffled the sentences in each story and asked five crowd workers on AMT to rearrange the sentences.

Table 4.3 summarizes the results of this experiment. The first row shows the result of ordering if we take the absolute majority ordering of the five crowd workers as the final ordering. The second row shows the result of ordering if we consider each of the 250 (50 stories x 5 workers ordering each one) ordering cases independently. As shown, the good stories are perfectly ordered with very high accuracy. It is important to note that this specific set rarely had any linguistic adverbials such as 'first', 'then', etc. to help human infer the ordering, so the main factors at play are the following: (1) the commonsensical temporal and causal relation between events (narrative schemas), e.g., human knows that first someone loses a phone then starts searching; (2) the natural way of narrating a story which starts with introducing the characters and concludes the story at the end. The role of the latter factor is quantified in the misplacement rate of each position reported in Table 4.3, where the first and last sentences are more often correctly placed than others. The high precision of ordering in sentences 2 up to 4 further verifies the richness of our corpus in terms of logical relation between events.

TimeML Annotation: TimeML-driven analysis of these stories can give us finer-grained insight about temporal aspect of the events in this corpus. We performed a simplified TimeML-driven [Pustejovsky et al., 2003] expert annotation of a sample of 20 stories[6]. Among all the temporal links (TLINK) annotated, 62% were 'before' and 10% were 'simultaneous'. We were interested to know if the actual text order mirrors

[4] This set can be found here: `https://goo.gl/VTnJ9s`

[5] This set can be found here: `https://goo.gl/pgm2KR`

[6] The annotation is available: `http://goo.gl/7qdNsb`

Context	**Right Ending**	**Wrong Ending**
Tom and Sheryl have been together for two years. One day, they went to a carnival together. He won her several stuffed bears, and bought her funnel cakes. When they reached the Ferris wheel, he got down on one knee.	Tom asked Sheryl to marry him.	He wiped mud off of his boot.
Karen was assigned a roommate her first year of college. Her roommate asked her to go to a nearby city for a concert. Karen agreed happily. The show was absolutely exhilarating.	Karen became good friends with her roommate.	Karen hated her roommate.
Jim got his first credit card in college. He didn't have a job so he bought everything on his card. After he graduated he amounted a $10,000 debt. Jim realized that he was foolish to spend so much money.	Jim decided to devise a plan for repayment.	Jim decided to open another credit card.

Table 4.4: Three example Story Cloze Test cases, completed by our crowd workers.

# cases collected	13,500
# workers participated	282
Average # cases written by one worker	47.8
Max # cases written by one worker	1461
Average payment per test case (cents)	10
Size of the final set (verified by human)	**3,744**

Table 4.5: Statistics for crowd-sourcing Story Cloze Test instances.

real-world order of events. We found that sentence order matches TimeML order 55% of the time.

4.4 Story Cloze Test

As described earlier in the introduction, the common evaluation framework for script learning is the ‘Narrative Cloze Test’ [Chambers and Jurafsky, 2008], where a system generates a ranked list of guesses for a missing event, given some observed events. The original goal of this test was to provide a comparative measure to evaluate narrative knowledge. However, gradually, the community started optimizing towards the performance on the test itself, achieving higher scores without demonstrating narrative knowledge learning. For instance, generating the ranked list according to the event’s corpus frequency (e.g., always predicting ‘X said’) was shown to be an extremely strong baseline [Pichotta and Mooney, 2014b]. Originally, narrative cloze test chains were extracted by hand and verified as gold chains. However, the cloze test chains used in all of the most recent works are not human verified as gold.

It is evident that there is a need for a more systematic automatic evaluation framework which is more in line with the original deeper script/story understanding goals. It is important to note that reordering of temporally shuffled stories (Section 4.3.3) can serve as a framework to evaluate a system’s story understanding. However, reordering can be achieved to a degree by using various surface features such as adverbials, so this cannot be a foolproof story understanding evaluation framework. Our ROCStories corpus enables a brand new framework for evaluating story understanding, called the *‘Story Cloze Test’*.

4.4.1 Story Cloze Test

The cloze task [Taylor, 1953] is used to evaluate a human (or a system) for language understanding by deleting a random word from a sentence and having a human fill in the blank. We introduce ‘Story Cloze Test’, in which a system is given a four-sentence ‘context’ and two alternative endings to the story, called ‘right ending’ and ‘wrong ending’. Hence, in this test the fifth sentence is blank. Then the system’s task is to choose the right ending. The ‘right ending’ can be viewed as ‘entailing’ hypothesis in a classic Recognizing Textual Entailment (RTE) framework [Giampiccolo et al., 2007], and ‘wrong’ ending can be seen as the ’contradicting’ hypothesis. Table 4.4 shows three example Story Cloze Test cases.

Story Cloze Test will serve as a generic story understanding evaluation framework, also applicable to evaluation of story generation models (for instance by computing the log-likelihoods assigned to the two ending alternatives by the story generation model), which does not necessarily imply requirement for explicit narrative knowledge learning. However, it is safe to say that any model that performs well on Story Cloze Test is demonstrating some level of deeper story understanding.

4.4.2 Data Collection Methodology

We randomly sampled 13,500 stories from ROCStories Corpus and presented only the first four sentences of each to AMT workers. For each story, a worker was asked to write a ‘right ending’ and a ‘wrong ending’. The workers were prompted to satisfy two conditions: (1) the sentence should follow up the story by sharing at least one of the characters of the story, and (2) the sentence should be entirely realistic and sensible when read in isolation. These conditions make sure that the Story Cloze Test cases are not trivial. More details on this setup is described in the supplementary material.

Quality Control: The accuracy of the Story Cloze Test can play a crucial role in directing the research community in the right trajectory. We implemented the following two-step quality control:

1. Qualification Test: We designed a qualification test for this task, where the workers had to choose whether or not a given 'right ending' and 'wrong ending' satisfy our constraints. At this stage we collected 13,500 cloze test cases.

2. Human Verification: In order to further validate the cloze test cases, we compiled the 13,500 Story Cloze Test cases into $2 \times 13,500 = 27,000$ full five-sentence stories. Then for each story we asked three crowd workers to verify whether or not the given sequence of five sentences makes sense as a meaningful and coherent story, rating within {-1, 0, 1}. Then we filtered cloze test cases which had 'right ending' with all ratings 1 and 'wrong ending' with all ratings 0. This process ensures that there are no boundary cases of 'right ending' and 'wrong ending'. This resulted in final 3,742 test cases, which was randomly divided into validation and test Story Cloze Test sets. We also made sure to remove the original stories used in the validation and test set from our ROCStories Corpus.

Statistics: Table 6.1 summarizes the statistics of our crowdsourcing effort. The Story Cloze Test sets can also be accessed through our website.

4.5 Story Cloze Test Models

In this Section we demonstrate that Story Cloze Test cannot be easily tackled by using shallow techniques, without actually understanding the underlying narrative. Following other natural language inference frameworks such as RTE, we evaluate system performance according to basic accuracy measure, which is defined as $\frac{\#correct}{\#test\ cases}$. We present the following baselines and models for tackling Story Cloze Test. All of the models are tested on the validation and test Story Cloze sets, where only the validation set could be used for any tuning purposes.

1. Frequency: Ideally, the Story Cloze Test cases should not be answerable without the context. For example, if for some context the two alternatives are 'He was mad after he won'[7] and 'He was cheerful after he won', the first alternative is simply less probable in real world than the other one. This baseline chooses the alternative with higher search engine[8] hits of the main event (verb) together with its semantic roles (e.g., 'I*poison*flowers' vs 'I*nourish*flowers'). We extract the main verb and its corresponding roles using TRIPS semantic parser.

2. N-gram Overlap: Simply chooses the alternative which shares more n-grams with the context. We compute Smoothed-BLEU [Lin and Och, 2004] score for measuring up to four-gram overlap of an alternative and the context.

3. GenSim: Average Word2Vec: Choose the hypothesis with closer average word2vec [Mikolov et al., 2013] embedding to the average word2vec embedding of the context. This is basically an enhanced word overlap baseline, which accounts for semantic similarity.

4. Sentiment-Full: Choose the hypothesis that matches the average sentiment of the context. We use the state-of-the-art sentiment analysis model [Manning et al., 2014] which assigns a numerical value from 1 to 5 to a sentence.

5. Sentiment-Last: Choose the hypothesis that matches the sentiment of the last context sentence.

6. Skip-thoughts Model: This model uses Skip-thoughts' Sentence2Vec embedding

[7] Given our prompt that the 'wrong ending' sentences should make sense in isolation, such cases should be rare in our dataset.

[8] `https://developers.google.com/custom-search/`

[Kiros et al., 2015] which models the semantic space of novels. This model is trained on the ‘BookCorpus’ [Zhu et al., 2015] (containing 16 different genres) of over 11,000 books. We use the skip-thoughts embedding of the alternatives and contexts for making decision the same way as with GenSim model.

7. Narrative Chains-AP: Implements the standard approach to learning chains of narrative events based on Chambers and Jurafsky [2008]. An event is represented as a verb and a typed dependency (e.g., the *subject* of *runs*). We computed the PMI between all event pairs in the Associated Press (AP) portion of the English Gigaword Corpus that occur at least 2 times. We run coreference over the given story, and choose the hypothesis whose coreferring entity has the highest average PMI score with the entity’s chain in the story. If no entity corefers in both hypotheses, it randomly chooses one of the hypotheses.

8. Narrative Chains-Stories: The same model as above, but trained on ROCStories.

9. Deep Structured Semantic Model (DSSM): This model [Huang et al., 2013] is trained to project the four-sentences context and the fifth sentence into the same vector space. It consists of two separate deep neural networks for learning jointly the embedding of the four-sentences context and the fifth sentence, respectively. As suggested in Huang et al. [2013], the input of the DSSM is based on context-dependent characters, e.g., the distribution count of letter-trigrams in the context and in the fifth sentence, respectively. The hyper parameters of the DSSM is determined on the validation set, while the model’s parameters are trained on the ROCStories corpus. In our experiment, each of the two neural networks in the DSSM has two layers: the dimension of the hidden layer is 1000, and the dimension of the embedding vector is 300. At runtime, this model picks the candidate with the largest cosine similarity between its vector representation and the context’s vector representation.

The results of evaluating these models on the Story Cloze validation and test sets are shown in Table 4.6. The constant-choose-first (51%) and human performance (100%) is also provided for comparison. Note that these sets were doubly verified by human, hence it does not have any boundary cases, resulting in 100% human performance. The DSSM model achieves the highest accuracy, but only 7.2 points higher than constant-choose-first. Error analysis on the narrative chains model shows why this and other event-based language models are not sufficient for the task: often, the final sentences of our stories contain complex events beyond the main verb, such as 'Bill was highly unprepared' or 'He had to go to a homeless shelter'. Event language models only look at the verb and syntactic relation like 'was-object' and 'go-to'. In that sense, going to a homeless shelter is the same as going to the beach. This suggests the requirement of having richer semantic representation for events in narratives. Our proposed Story Cloze Test offers a new challenge to the community.

	Constant-choose-first	Frequency	N-gram-overlap	GenSim	Sentiment-Full	Sentiment-Last	Skip-thoughts	Narrative-Chains-AP	Narrative-Chains-Stories	DSSM	Human
Validation Set	0.514	0.506	0.477	0.545	0.489	0.514	0.536	0.472	0.510	0.614	1.0
Test Set	0.513	0.520	0.494	0.539	0.492	0.522	0.552	0.478	0.494	**0.595**	1.0

Table 4.6: The accuracy of various models on The Story Cloze validation and test sets.

4.6 Discussion

There are three core contributions in this chapter: (1) a new corpus of commonsense stories, called ROCStories, (2) a new evaluation framework to evaluate script/story learners, called Story Cloze Test, and (3) a host of first approaches to tackle this new test framework. ROCStories Corpus is the first crowdsourced corpus of its kind for the community. We have released about 50k stories, as well as validation and test sets for Story Cloze Test. This dataset will eventually grow to 100k stories, which will be released through our website. In order to continue making meaningful progress on this task, although it is possible to keep increasing the size of the training data, we expect the community to develop models that will learn to generalize to unseen commonsense concepts and situations.

The Story Cloze Test proved to be a challenge to all of the models we tested. We believe it will serve as an effective evaluation for both story understanding and script knowledge learners. We encourage the community to benchmark their progress by reporting their results on Story Cloze test set. Compared to the previous Narrative Cloze Test, we found that one of the early models for that task actually performs worse than random guessing. We can conclude that Narrative Cloze test spurred interest in script learning, however, it ultimately does not evaluate deeper knowledge and language understanding.

5 CaTeRS: Causal and Temporal Relation Scheme for Semantic Annotation of Event Structures

5.1 Introduction

Understanding events and their relations in natural language has become increasingly important for various NLP tasks. Most notably, story understanding [Charniak, 1972; Winograd, 1972; Turner, 1994; Schubert and Hwang, 2000] which is an extremely challenging task in natural language understanding, is highly dependent on understanding events and their relations. Recently, we have witnessed a renewed interest in story and narrative understanding based on the progress made in core NLP tasks.

Perhaps the biggest challenge of story understanding (and story generation) is having commonsense knowledge for the interpretation of narrative events. This commonsense knowledge can be best represented as scripts. Scripts present structured knowledge about stereotypical event sequences together with their participants. A well known script is the Restaurant Script, which includes the events {Entering, Sitting down, Asking for menus, Choosing meals, etc.}, and the participants {Customer, Waiter, Chef, Tables, etc.}. A large body of work in story understanding has focused on learning scripts [Schank and Abelson, 1977]. Given that developing hand-built scripts is extremely time-consuming, there is a serious need for automatically induced scripts [Chambers and Jurafsky, 2008, 2009; Balasubramanian et al., 2013; Cheung et al., 2013; Nguyen et al., 2015]. It is evident that various NLU applications (text summarization, co-reference resolution and question answering, among others) can benefit from the rich inferential capabilities that structured knowledge about events can provide.

The first step for any script learner is to decide on a corpus to drive the learning process. The most recent resource for this purpose is a corpus of short commonsense stories, called ROCStories [Mostafazadeh et al., 2016a], which is a corpus of 50,000 short commonsense everyday stories [1]. This corpus contains high quality[2] five-sentence stories that are full of stereotypical causal and temporal relations between events, making them a perfect resource for learning narrative schemas.

One of the prerequisites for learning scripts from these stories is to extract events and find inter-event semantic relations. Earlier work [Chambers and Jurafsky, 2008, 2009; Pichotta and Mooney, 2014a; Rudinger et al., 2015] defines verbs as events and uses TimeML-based [Pustejovsky et al., 2003] learning for temporal ordering of events. This clearly has many shortcomings, including, but not limited to (1) not capturing a wide range of non-verbal events such as 'earthquake', (2) not capturing a more comprehensive set of semantic relations between events such as causality, which is a core relation in stories.

In this chapter we formally define a new comprehensive semantic framework for

[1] These stories can be found here: `http://cs.rochester.edu/nlp/rocstories`

[2] Each of these stories have the following major characteristics: is realistic, has a clear beginning and ending where something happens in between, does not include anything irrelevant to the core story.

capturing stereotypical event-event temporal and causal relations in commonsense stories, the details of which can be found in Sections 5.3-5.5. Using this semantic framework we annotated 320 stories sampled from ROCStories to extract inter-event semantic structures. Our inter-annotator agreement analysis, presented in Section 5.6 shows that this framework enables high event entity annotation agreement and promising inter-event relation annotation agreement. We believe that our semantic framework better suits the goals of the task of script learning and story understanding, which can potentially enable learning richer and more accurate scripts. Although this work focuses on stories, the CaTeRS annotation framework for capturing inter-event relations can be applied to other genres.

5.2 Related Work

One of the most recent temporal annotation schemas is Temporal Histories of Your Medical Event (THYME) [Styler et al., 2014]. This annotation guideline was devised for the purpose of establishing timelines in clinical narratives, i.e. the free text portions contained in electronic health records. In their work, they combine the TimeML annotation schema with Allen Interval Algebra, identifying the five temporal relations BEFORE, OVERLAP, BEGINS-ON, ENDS-ON, and CONTAINS. Of note is that they adopt the notion of narrative containers [Pustejovsky and Stubbs, 2011], which are time slices in which events can take place, such as DOCTIME (time of the report) and before DOCTIME. As such, the THYME guideline focuses on ordering events with respect to specific time intervals, while in our work, we are only focused on the relation between two events, without concern for ordering. Their simplification of temporal links is similar to ours, however, our reasoning for simplification takes into account the existence of causality, which is not captured by THYME.

Causality is a notion that has been widely studied in psychology, philosophy, and logic. However, precise modeling and representation of causality in NLP applications is still an open issue. A formal definition of causality in lexical semantics can be found in [Hobbs, 2005]. Hobbs introduces the notion of “causal complex”, which refers to some collection of eventualities (events or states) for which holding or happening entails the happening of effect. In part, our annotation work is motivated to learn what the causal complexes are for a given event or state. The Penn Discourse Tree Bank (PDTB) corpus [Prasad et al., 2008] addresses the annotation of causal relations, annotating semantic relations that hold between exactly two Abstract Objects (called Arg1 and Arg2), expressed either explicitly via lexical items or implicitly via adjacency in discourse. In this chapter, we present a semantic framework that captures both explicit and implicit causality, but no constraint of adjacency is imposed, allowing commonsense causality to be captured at the inter-sentential level (story).

Another line of work annotates temporal and causal relations in parallel [Steven Bethard and Martin, 2008]. Bethard et al. annotated a dataset of 1,000 conjoined-event temporal-causal relations, collected from Wall Street Journal corpus. Each event pair was annotated manually with both temporal (BEFORE, AFTER, NO-REL) and causal relations (CAUSE, NO-REL). For example, sentence 11 is an entry in their dataset. This dataset makes no distinction between various types of causal relation.

(11) Fuel tanks had leaked and contaminated the soil.
- (leaked BEFORE contaminated)
- (leaked CAUSED contaminated).

A recent work [Mirza and Tonelli, 2014] has proposed a TimeML-style annotation standard for capturing causal relations between events. They mainly introduce ‘CLINK’, analogous to ‘TLINK’ in TimeML, to be added to the existing TimeML link tags. Under this framework, Mirza et al [Mirza and Tonelli, 2014] annotates 318 CLINKs in TempEval-3 TimeBank. They only annotate explicit causal relations signaled by linguistic markers, such as {because of, as a result of, due to, so, therefore, thus}. Another relevant work is Richer Event Descriptions (RED) [Ikuta et al., 2014], which combines event coreference and THYME annotations, and also introduces cause-effect annotation in adjacent sentences to achieve a richer semantic representation of events and their relations. RED also distinguishes between 'PRECONDITION' and 'CAUSE', similarly to our 'ENABLE' and 'CAUSE' relations. These can be in the context of BEFORE or OVERLAP, but they do not include PREVENT and CAUSE-TO-END. Our set of comprehensive 9 causal relations distinguishes between various temporal implications, not covered by any of the related work.

5.3 Event Entities

Our semantic framework captures the set of event entities and their pairwise semantic relations, which together form an inter-connected network of events. In this Section we define event entities and discuss their annotation process.

5.3.1 Definition

As discussed earlier in the Introduction of this thesis, event is mainly used as a term referring to any situation that can happen, occur, or hold. The definition and detection of events has been a topic of interest in various NLP applications. However, there is still no consensus regarding the span of events and how they should be annotated. There has been some good progress in domain-specific annotation of events, e.g., recent Clinical TempEval task [Bethard, 2013] and THYME annotation scheme [Styler et al., 2014], however, the detection of events in broad-coverage natural language has been an ongoing endeavor in the field.

One of the existing definitions for event is provided in the TimeML annotation schema [Pustejovsky et al., 2003]:

> "An event is any situation (including a process or state) that happens, occurs, or holds to be true or false during some time point (punctual) or time interval (durative)."

According to this definition, adjectival states such as 'on board' are also annotated as events. As we are focused on the task of narrative structure learning, we want to capture anything that 'happens and occurs', but not including holds. Formally, we therefore define an event as follows:

> "An event is any situation (including a process or state) that happens or occurs either instantaneously (punctual) or during a period of time (durative)."

In order to make the event annotation less subjective, we specifically define an event to be any lexical entry under any of the following ontology types in the TRIPS ontology[3] [Allen et al., 2008]:

- Event-of-state: e.g., have, lack.
- Event-of-change: e.g., kill, delay, eat.
- Event-type: e.g., effort, procedure, turmoil, mess, fire.

[3] http://www.cs.rochester.edu/research/trips/lexicon/browse-ont-lex-ajax.html

- Physical-condition: all medical disorders and conditions, e.g., cancer, heart attack, stroke, etc.
- Occurring: e.g., happen, occur.
- Natural-phenomenon: e.g., earthquake, tsunami.

This ontology has one of the richest event hierarchies, which perfectly serves our purpose of broad-coverage event extraction.

5.3.2 How to annotate events?

After pinpointing an event entity according to the formal definition presented earlier, one should annotate the event by selecting the corresponding span in the sentence. Here, we define the event span to be the head of the main phrase which includes the complete event. For example, in the sentence 'I [climbed] the tree', we annotate 'climb', the head of the verb phrase, while the complete event is '[climb] tree' . This implies that only main events (and not their dependents) are annotated. Annotating the head word enables us to delegate the decision about adding the dependent children to a post-process, which can be tuned per application.

Moreover, no verbs which take the role of an auxiliary verb are annotated as events. For instance, in the sentence 'She had to [change] her jeans' the main event is 'change'. For multi-word verbs such as 'turn out or 'find out', the entire span should be selected as the event. The annotators can consult lexicons such as WordNet [Miller., 1995] for distinguishing multi-word verbs from verbs with prepositional phrases adjuncts.

The Case for Embedded Events

Another important controversial issue is what to do with embedded events, where one event takes another event as its core argument (if neither of the verbs are auxiliary). For instance, consider the following example:

(12) Sam [wanted]$_{e1}$ to [avoid]$_{e2}$ any trouble, so he [drove]$_{e3}$ slowly.

According to our event entity definition, there are three events in example 12, $e1$ and $e2$ and $e3$, all of which should be annotated. However, more complicated is the case of a main event in an embedded event construction which signals any of the semantic relations in the annotation scheme. Consider the sentence in example 13. In this sentence, there are also three events according to our definition of event entities, where (cause (die)) is an embedded event construction. In this case the verb 'cause' simply signals a causal relation between $e1$ and $e3$, which will be captured by our existing semantic relation (to be described in Section 5.4), and so we do not annotate the verb 'cause' as an event.

Likewise, the sentence in example 14 showcases another embedded event construction, (cause (explosion)), so the event 'cause' should not be annotated.

(13) The [explosion]$_{e1}$ caused him to [die]$_{e3}$.

(14) The [fire]$_{e1}$ caused an [explosion]$_{e2}$.

The same rule applies to the synonyms of these verbs in addition to other verbs that signal a temporal or causal relation, including but not limited to {start, begin, end, prevent, stop, trigger}, which hereinafter we call 'aspectual verbs'[4]. It is important to note that the above rule for aspectual verbs can be applied only to embedded event

[4] These verbs are the same as aspectual events characterized by TimeML, which include 'INITIATES', 'CULMINATES', 'TERMINATES', 'CONTINUES' and 'REINITIATES'.

constructions and may be overridden. Consider example 15. In this example, it is clear that the event ‘prevent’ plays a key semantic role in the sentence and should be annotated as an event since it is the only viable event that can be semantically connected to other events such as $e3$.

(15) John [prevented]$_{e1}$ the vase from [falling]$_{e2}$ off the table, I was [relieved]$_{e3}$.

5.3.3 The Case for Copulas

A copula is a verb which links the subject of a sentence with a predicate, such as the verb ‘is’ which links ‘this suit’ to the predicate ‘dark blue’ in the sentence ‘This suit is dark blue’. Many such constructions assign a state to something or someone which holds true for some duration. The question is what to specify as the event entity in such sentences. According to our definitions, an adjective such as ‘blue’ is not an event (that is, it does not occur or happen), but after many rounds of pilot annotations, we concluded that annotating the predicate adjective or predicate nominal best captures the core semantic information. Thus, the sentences 16-17 will be annotated as follows:

(16) He was really [hungry]$_{e1}$.

(17) He [ate]$_{e1}$ a juicy burger.

It is important to emphasize that annotating states such as ‘hungry’ as an event is only done in the case of copulas, and, for example in sentence 17, the adjective ‘juicy’ will not be annotated as an event.

Our annotation of light verb constructions (e.g., do, make, have) is consistent with the annotation of copulas and auxiliary verbs. Whenever the semantic contribution of the verb is minimal and the non-verb element of the construction is an event in the TRIPS ontology, we annotate the non-verb element as the event. Thus, we annotate the

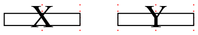
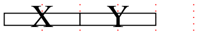
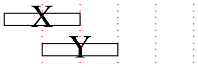
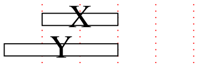
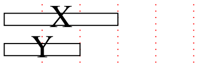
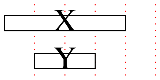
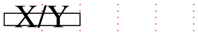
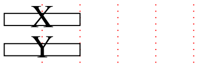
Allen	Visualization	Allen - Inverse	TimeML	THYME	CaTeRS
X Before Y		Y After X	Before	Before	Before
X Meets Y		Y Is Met X	IBefore	-	-
X Overlaps Y		Y Is overlapped by X	-	Overlaps	Overlaps
X Finishes Y		Y Is finished by X	Ends	Ends-on	-
X Starts Y		Y Is started by X	Begins	Begins-on	-
X Contain Y		Y During X	During	Contains	Contains
X Equals Y		-	Identity	Identity	Identity
-		-	Simultaneous	-	-

Table 5.1: The correspondence of temporal relation sets of different annotation frameworks.

noun predicate ’offer’ in the sentence ‘Yesterday, John made an offer to buy the house for 350,000’, similarly to the way Abstract Meaning Representation (AMR) drops the light verb and promotes the noun predicate [Banarescu et al., 2013]. This annotation is also close to the PropBank annotation of copulas and light verbs [Bonial et al., 2014], where they annotate the noun predicate and predicate adjective as the event; however, PropBank includes an explicit marking of the verb as either a light verb or a copula verb.

5.4 The Semantic Relations Between Event Entities

A more challenging problem than event entity detection is the identification of the semantic relation that holds between events. Events take place in time, hence temporal relations between events are crucial to study. Furthermore, causality plays a crucial role in establishing semantic relation between events, specifically in stories. In this Section, we provide details on both temporal and causal semantic relations.

5.4.1 Temporal Relation

Time is the main notion for anchoring the changes of the world triggered by sequences of events. Of course having temporal understanding and temporal reasoning capabilities is crucial for many NLP applications such as question answering, text summarization and many others. Throughout the years the issue of temporal analysis and reasoning in natural language has been addressed via different approaches. Allen's Interval Algebra [Allen, 1984] is one theory for representing actions and introduces a set of 13 distinct, exhaustive, and qualitative temporal relations that can hold between two time intervals. The first three columns of Table 5.1 list these 13 temporal relations together with their visualization, which includes 6 main relations and their corresponding inverses –together with the 'equal' relation which does not have an inverse.

Based on Interval Algebra, a new markup language for annotating events and temporal expressions in natural language was proposed [Pustejovsky et al., 2003], named TimeML. This schema is designed to address problems in event and temporal expression markup. It covers two major tags: 'EVENT' and 'TIMEX3'. The EVENT tag is used to annotate elements in a text that represent events such as 'kill' and 'crash'. TIMEX is mainly used to annotate explicit temporal expressions, such as times, dates and durations. One of the major features introduced in TimeML was the LINK tag. These tags encode the semantic relations that exist between the temporal elements annotated in a document. The most notable LINK is TLINK: a Temporal Link representing the temporal relationship between entities (events and time expressions). This link not only encodes a relation between two entities, but also makes a partial ordering between events. There are 14 TLINK relations in TimeML, adding ‘simultaneous’ to the list of temporal links between events. The fourth column of Table 5.1 shows the correspondence between Allen relations and TimeML TLINKs. Furthermore, in the fifth column we include the THYME annotation schema (to be discussed in Section 5.2).

We propose a new set of temporal relations for capturing event-event relations. Our final set of temporal relations are shown in the sixth column of Table 5.1[5]. As compared with TimeML, we drop the relations ‘simultaneous’, ‘begins’ and ‘ends’. ‘Simultaneous’ was not a part of the original Allen relations. Generally it is hard to be certain about two events occurring exactly during the same time span, starting together and ending together. Indeed, the majority of events which are presumed ‘simultaneous’ in TimeML annotated corpora are either (1) EVENT-TIMEX relations which are not event-event relations, or (2) wrongly annotated and should be the ‘overlapping’ relation, e.g., in the following sentence from TimeBank corpus the correct relation for the two events $e1$ and $e2$ should be ‘overlap’:

She [listened]$_{e1}$ to music while [driving]$_{e2}$.

We acknowledge that having ’simultaneous’ can make the annotation framework more comprehensive and may apply in few certain cases of punctual events, however, such cases are very rare in our corpus, and in the interest of a more compact and less am-

[5] Since a main temporal relation and its inverse have a reflexive relation, the annotation is carried out only on the main temporal relation.

biguous annotation, we did not include it. THYME also dropped ‘simultaneous’ for similar reasons.

As for the ‘begins’ and ‘ends’, our multiple pilot studies, indicated that these relations are more accurately captured by one of our causal relations (next subsection) or the relation ‘overlaps’. We believe that our simplified set of 4 temporal relations can be used for any future broad-coverage inter-event temporal relation annotation. We also drop the temporal relation ‘IBefore’, given that this relation usually reflects on causal relation between two events which will be captured by our causal links.

5.4.2 Causal Relation

Research on the extraction of event relations has concerned mainly the temporal relation and time-wise ordering of events. A more complex semantic relationship between events is causality. Causality is one of the main semantic relationships between events where an event (CAUSE) results in another event (EFFECT) to happen or hold. It is clear that identifying the causal relation between events is crucial for numerous applications, including story understanding. Predicting occurrence of future events is the major benefit of causal analysis, which can itself help risk analysis and disaster decision making. There is an obvious connection between causal relation and temporal relation: by definition, the CAUSE event starts ‘BEFORE’ the EFFECT event. Hence, predicting causality between two events also requires/results in a temporal prediction.

It is challenging to define causality in natural language. Causation, as commonly understood as a notion for understanding the world in the philosophy and psychology, is not fully predicated in natural language [Neeleman and Koot, 2012]. There have been several attempts in the field of psychology for modeling causality, e.g., the counterfactual model [Lewis, 1973] and the probabilistic contrast model [Cheng and Novick, 1992]. Leonard Talmy's seminal work [Talmy, 1988] in the field of cognitive linguistics models the world in terms of semantic categories of how entities interact with respect to force (Force Dynamics). As a reminder of some ideas discussed in the Introduction of this thesis, these semantic categories include concepts such as the employment of force, resistance to force and the overcoming of resistance, blockage of a force, removal of b¿ here is a reiteration of Talmy's 'force dynamics tockage, and etc. Force dynamics provides a generalization over the traditional linguistic understanding of causation by categorizing causation into 'letting', 'helping', 'hindering' and etc. Wolff and Song [Wolff and Song, 2003] base their theory of causal verbs on force dynamics. Wolff proposes [Wolff, 2007] that causation includes three main types of *causal concepts*: **'Cause', 'Enable'** and **'Prevent'**. These three causal concepts are lexicalized through distinctive types of verbs [Wolff and Song, 2003] which are as follows:

- Cause-type verbs: e.g. cause, start, prompt, force.
- Enable-type verbs: e.g. allow, permit, enable, help.
- Prevent-type verbs: e.g. block, prevent, hinder, restrain.

Wolff's model accounts for various ways that causal concepts are lexicalized in language and we base our annotation framework on this model. However, we will be looking at causal relation between events more from a **'commonsense reasoning'** perspective than linguistic markers. We define cause, enable and prevent for commonsense co-occurrence of events, inspired by mental model theory of causality [Khemlani et al., 2014], as follows:

- A $Cause$ B: In the context, If A occurs, B most probably occurs as a result.
- A $Enable$ B: In the context, If A does not occur, B most probably does not occur (not enabled to occur).

- A $Prevent$ B: In the context, If A occurs, B most probably does not occur as a result.

where *In the context* refers to the underlying context in which A and B occur, such as a story. This definition is in line with the definition of CAUSE and PRECONDITION presented in the RED annotation guidelines [Ikuta et al., 2014] (to be discussed in Section 5.2).

In order to better understand the notion of commonsense causality, consider the sentences 18-20.

(18) Harry [fell]$_{e1}$ and [skinned]$_{e2}$ his knee.

(19) Karla [earned]$_{e1}$ more money and finally [bought]$_{e2}$ a house.

(20) It was [raining]$_{e1}$ so hard that it prevented me from [going]$_{e2}$[6] to the school.

In the above three sentences, the relation between $e1$ and $e2$ is ‘cause’, ‘enable’ and ‘prevent’ in order.

It is important to note that our scope of lexical semantic causality only captures events causing events [Davidson, 1967], however, capturing individuals as cause [Croft, 1991] is a another possible extension.

Temporal Implications of Causality

The definition of causality implies that when A Causes B, then A should start before B in order to have triggered it. It is important to note that for durative events the temporal implication of causality is mainly about the start of the causal event, which should be before the start of the event which is caused. Consider the following example:

[6] As discussed earlier, here the embedded event construction is (prevent (going)) where only the event ‘going’ will be annotated.

(21) The [fire]$_{e1}$ [burned down]$_{e2}$ the house.

In this example there is a 'cause' relation between $e1$ and $e2$, where 'fire' clearly does not finish before starting of the 'burn' event. So the temporal relation between the two events is 'overlaps'. Here we conclude that when 'A cause/enable/prevent B', we know as a fact that As start is before Bs start, but there is no restriction on their relative ending. This implies that a cause relation can have any of the two temporal relations: *before* and *overlaps*.

All the earlier examples of causal concepts we explored involved an event causing another event to happen or to start (in case of a durative event). However, there are examples of causality which involve not starting but ending an ongoing event. Consider the sentence 22. In order to capture causality relations between pairs of events such as $e1$ and $e2$, we introduce a *Cause-to-end* relation, which can have one of the three temporal implications: *before*, *overlaps*, and *during*. Hence, in sentence 22, the relation between $e1$ and $e2$ will be *Cause-to-end (overlap)*.

(22) The [famine]$_{e1}$ ended the [war]$_{e2}$.

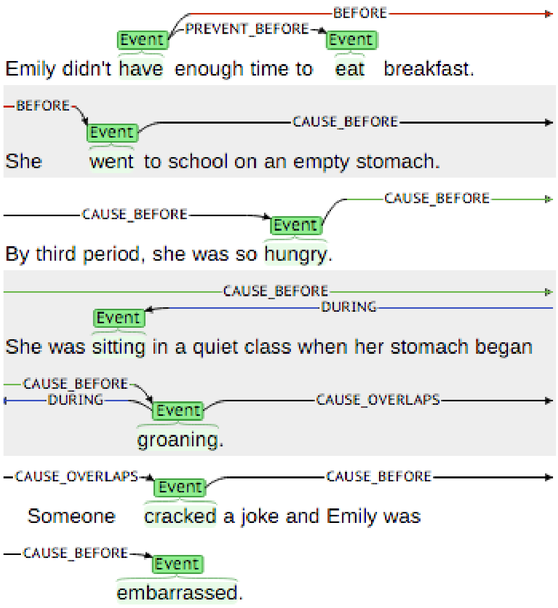
5.4.3 How to annotate semantic relations between events?

In summary, the disjunctive[7] set of 13 semantic relations between events in our annotation framework are as follows:

– **9 causal relations:** Including 'cause (before/overlaps)', 'enable (before/overlaps)', 'prevent (before/overlaps)', 'cause-to-end (before/overlaps/during)'

– **4 temporal relations:** Including 'Before', 'Overlaps', 'Contains', 'Identity'.

[7] This implies that only one of these semantic relations can be selected per each event pair.



Emily didn't have enough time to eat breakfast.
She went to school on an empty stomach.
By third period, she was so hungry.
She was sitting in a quiet class when her stomach began groaning.
Someone cracked a joke and Emily was embarrassed.

Figure 5.1: Semantic annotation of a sample story.

The semantic relation annotation between two events should start with deciding about any causal relations and then, if there was not any causal relation, proceed to choosing any existing temporal relation.

5.5 Annotating at Story level

It has been shown [Bittar et al., 2012] that temporal annotation can be most properly carried out by taking into account the full context for sentences, as opposed to TimeML, which is a surface-based annotation. The scope and goal of this chapter very well aligns with this observation. We carry out the annotation at the story level, meaning that we annotate inter-event relations across the five sentences of a story. It suffices to do the event-event relation specification minimally given the transitivity of temporal relations. For example for three consecutive events $e1$ $e2$ $e3$ one should only annotate the 'before' relation between $e1$ and $e2$, and $e2$ and $e3$, since the 'before' relation between $e1$ and $e3$

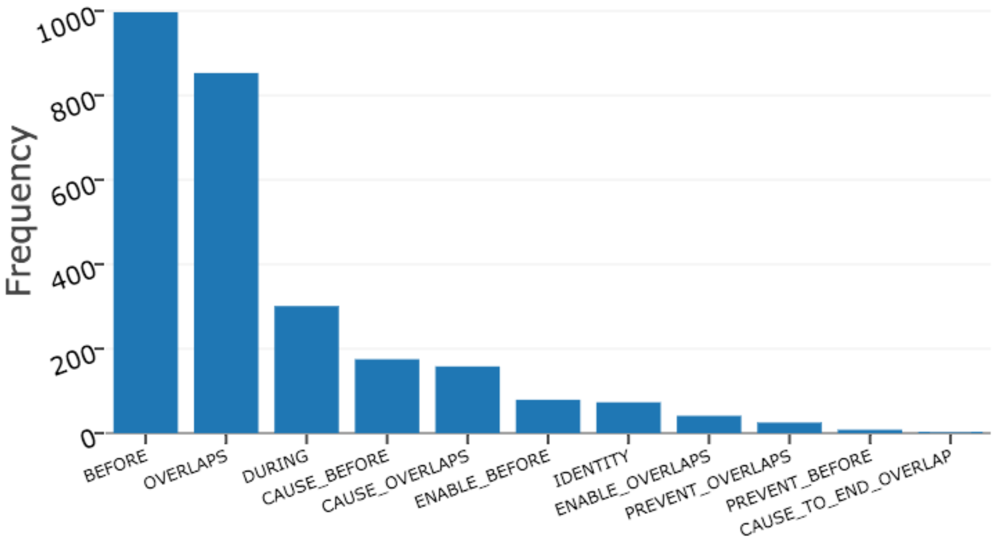


Figure 5.2: Frequency of semantic links in our dataset.

can be inferred from the other two relations. The event-event relations can be from/to any sentence in the story. It is important to emphasize here that the goal of annotation is to capture commonsensical relations between events, so the annotation effort should be driven by intuitive commonsensical relation one can pinpoint throughout a story.

Consider an example of a fully annotated story shown in Figure 5.1. As you can see, all of the semantic annotations reflect commonsense relation between events given the underlying story, e.g., ‘cracked a joke’ *cause-before* ‘embarrassed’.

5.6 Annotated Dataset Analysis

We randomly sampled 320 stories (1,600 sentences) from the ROCStories Corpus. This set covers a variety of everyday stories, with titles ranging from ‘got a new phone’ to ‘Left at the altar’. We provided our main expert annotator with the annotation guidelines, with the task of annotating each of the 320 stories at story level. The annotation task was set up on Brat tool[8]. On average, annotation time per story was 11 minutes. These annotations can be found through `http://cs.rochester.edu/nlp/rocstories/CaTeRS/`.

5.6.1 Statistics

Overall, we have annotated 2,708 event entities and 2,715 semantic relations. Figure 5.2 depicts the distribution of different semantic relations in our annotation set. For this Figure we have removed any semantic relations with frequency less than 5. As you can see, the temporal relation *before* is the most common relation, which reflects on the natural reporting of the sequence of events throughout a story. There are overall 488 various causal links, the most frequent of which is *cause-before*. Given these statistics, it is clear that capturing causality along with temporal aspects of stories is crucial.

Our annotations also enable some deeper analysis of the narrative flow of the stories. One major question is if the text order of events appearing in consecutive sentences mirrors the real-world order of events. Although the real-world order of events is more complex than just a sequence of *before* relations, we can simplify our set of semantic links to make an approximation: we count the number of links (from any[9] type) which connect an event entity appearing in position X to an event entity appearing in position $X-i$. We found that temporal order does not match the text order 23% of the time. This reinforces the quality of the narrative flow in the ROCStories corpus. Moreover, 23% is statistically significant enough to motivate the requirement for temporal ordering models for these stories.

[8] `http://brat.nlplab.org/`

[9] This is based on the fact that any relation such as ‘A enable-before B’ or ‘A overlaps B’ can be naively approximated to ‘A before B’.

5.6.2 Inter-annotator Agreement

In order to compute inter-annotator agreement on our annotation framework, we shared one batch of 20 stories between four expert annotators. Our annotation task consists of two subtasks: (1) entity span selection: choosing non-overlapping event entity spans for each story, (2) semantic structure annotation: building a directed graph (most commonly connected) on top of the entity spans.

Agreement on Event Entities

Given that there are no prefixed set of event entity spans for straight-forward computation of inter-annotator agreement, we do the following: among all the annotators, we aggregate the spans of the annotated event entity as the annotation object [Artstein and Poesio, 2008]. Then, if there exists a span which is not annotated by one of the coders (annotators) it will be labeled as 'NONE' for its category. The agreement according to Fleiss's Kappa $\kappa = 0.91$, which shows substantial agreement on event entity annotation. Although direct comparison of κ values is not possible, as a point of reference, the event span annotation of the most recent clinical TempEval [Bethard et al., 2015] was 0.81.

Agreement on Semantic Links

Decisions on semantic links are dependent on two things (1) decisions on event entities; (2) the decision about the other links. Hence, the task of annotating event structures is in general a hard task. In order to relax the dependency on event entities, we fix the set of entities to be the ones that all annotators have agreed on. Following the other discourse structure annotation tasks such as Rhetorical Structure Theory (RST), we aggregate all the relations captured by all annotators as the annotation object, then labeling 'NONE' as the category for coders who have not captured this relation. The agreement according to Fleiss's Kappa $\kappa = 0.49$ without applying basic closure and $\kappa = 0.51$ with closure[10], which shows moderate agreement. For reference, the agreement on semantic link annotation in the most recent clinical TempEval was 0.44 without closure and 0.47 with closure.

5.7 Conclusion

In this chapter we introduced a novel framework for semantic annotation of event-event relations in commonsense stories, called CaTeRS. We annotated 1,600 sentences throughout 320 short stories sampled from ROCStories corpus, capturing 2,708 event entities and 2,715 semantic relations, including 13 various types of causal and temporal relations. This annotation scheme is unique in capturing both temporal and causal inter-event relations. We show that our annotation scheme enables high inter-annotator agreement for event entity annotation. This is due to our clear definition of events which (1) is linked to lexical entries in TRIPS ontology, removing problems caused by each annotator devising his or her own notion of event, (2) captures only the head of the underlying phrase.

Our inter-annotator analysis of semantic link annotation shows moderate agreement, competitive with earlier temporal annotation schemas. We are planning to improve the agreement on semantic links even further by setting up two-stage expert annotation process, where we can pair annotators for resolving disagreements and modifying the annotation guideline. A comprehensive study of temporal and causal closure in our framework is a future work.

[10] Temporal closure [Gerevini et al., 1995] is a reasoning for deriving explicit relations to implicit relations, applying rules such as transitivity.

We believe that our semantic framework for temporal and causal annotation of stories can better model the event structures required for script and narrative structure learning. Although this work focuses on stories, our annotation framework for capturing inter-event relations can be applicable to other genres. It is important to note that this framework is not intended to be a comprehensive analysis of all temporal and causal aspects of text, but rather we focus on those temporal and causal links that support learning stereotypical narrative structures. As with any annotation framework, this framework will keep evolving over time, the updates of which can be followed through `http://cs.rochester.edu/nlp/rocstories/CaTeRS/`.

6 The Analysis of the First Story Cloze Shared Task

6.1 Introduction

Building systems that can understand stories or can compose meaningful stories has been a long-standing ambition in natural language understanding [Charniak, 1972; Winograd, 1972; Turner, 1994; Schubert and Hwang, 2000]. Perhaps the biggest challenge of story understanding is having commonsense knowledge for comprehending the underlying narrative structure. However, rich semantic modeling of the text's content involving words, sentences, and even discourse is crucially important. The workshop on Linking Lexical, Sentential and Discourse-level Semantics (LSDSem)[1] is committed to encouraging computational models and techniques which involve multiple levels of semantics.

The LSDSem'17 shared task is the Story Cloze Test. As presented in Chapter 4, in this test, the system reads a four-sentence story along with two alternative endings. It is then tasked with choosing the correct ending. As presented in Chapter 4, the existing models were only slightly better than random performance and more powerful models

[1] http://www.coli.uni-saarland.de/~mroth/LSDSem/

will require richer modeling of the semantic space of stories.

Given the wide gap between human (100%) and state-of-the-art system (58.5%) performance, the time was ripe to hold the first shared task on SCT. In this chapter, we present a summary on the first organized shared task on SCT with eight participating systems. The submitted approaches to this non-blind challenge ranged from simple rule-based methods, to linear classifiers and end-to-end neural models, to hybrid models that leverage a variety of features on different levels of linguistic analysis. The highest performing system achieves an accuracy of 75.2%, which substantially improves the previously established state-of-the-art. We hope that our findings and discussions can help reshape upcoming evaluations and shared tasks involving story understanding.

6.2 Shared Task Setup

For the shared task, we provided the same dataset as created in Chapter 6, which consists of a development and test set each containing 1,871 stories with two alternative endings. At this stage, we used this already existing non-blind dataset with established baselines to build up momentum for researching the task. This dataset can be accessed through `http://cs.rochester.edu/nlp/rocstories/`.

As the training data, we released an extended set of ROCStories[2], called ROCStories Winter 2017. We followed the same crowdsourcing setup described in Chapter 6. Table 6.2 provides three example stories in this dataset. As these examples show, these are complete stories and do not come with a wrong ending[3]. Although we provided the additional ROCStories, the participants were encouraged to use or construct any

[2] The extended ROCStories dataset can be accessed via `http://cs.rochester.edu/nlp/rocstories/`.

[3] The ROCStories corpus can be used for a variety of applications ranging from story generation to script learning.

training data of their choice. Overall, the participants were provided three datasets with the statistics listed in Table 6.1.

ROCStories (training data)	**98,159**
Story Cloze validation set, Spring 2016	**1,871**
Story Cloze test set, Spring 2016	**1,871**

Table 6.1: The size of the provided shared task datasets.

Story Title	**S**tory
The Hurricane	Morgan and her family lived in Florida. They heard a hurricane was coming. They decided to evacuate to a relative's house. They arrived and learned from the news that it was a terrible storm. They felt lucky they had evacuated when they did.
Marco Votes For President	Marco was excited to be a registered voter. He thought long and hard about who to vote for. Finally he had decided on his favorite candidate. He placed his vote for that candidate. Marco was proud that he had finally voted.
Spaghetti Sauce	Tina made spaghetti for her boyfriend. It took a lot of work, but she was very proud. Her boyfriend ate the whole plate and said it was good. Tina tried it herself, and realized it was disgusting. She was touched that he pretended it was good to spare her feelings.

Table 6.2: Example ROCStories instances from the Winter 2017 release.

We evaluate the systems in terms of accuracy, which we measure as $\frac{\#correct}{\#test\ cases}$. Any other details regarding our shared task can be accessed via our shared task page `http://cs.rochester.edu/nlp/rocstories/LSDSem17/`.

6.3 Submissions

The Shared Task was conducted through CodaLab competitions[4]. We received a total of 18 registrations, out of which eight teams participated: four teams from the US, three teams from Germany and one team from India.

In the following, we provide short paragraphs summarizing our baseline and approaches of the submissions.

msap (University of Washington). Linear classifier based on language modeling probabilities of the entire story, and linguistic features of only the ending sentences [Schwartz et al., 2017]. These ending "style" features include sentence length as well as word and character n-gram in each candidate ending (independent of story). These style features have been shown useful in other tasks such as age, gender, or native language detection.

cogcomp (University of Illinois). Linear classification system that measures a story's coherence based on the sequence of events, emotional trajectory, and plot consistency. This model takes into account frame-based and sentiment-based language modeling probabilities as well as a topical consistency score.

acoli (Goethe University Frankfurt am Main) and **tbmihaylov (Heidelberg University).** Two resource-lean approaches that only make use of pretrained word representations and compositions thereof [Schenk and Chiarcos, 2017; Mihaylov and Frank, 2017]. Composition functions are learned as part of a feed-forward and LSTM neural networks, respectively.

[4] The Story Cloze CodaLab page can be accessed here: `https://competitions.codalab.org/competitions/15333`

Rank	CodaLab Id	Model	ROCStories	Pre-trained Embeddings	Other Resources	Accuracy
1	**msap**	Logistic regression	Spring 2016, Winter 2017	–	NLTK Tokenizer, Spacy POS tagger	**0.752**
2	**cogcomp**	Logistic regression	Spring 2016, Winter 2017	Word2Vec	UIUC NLP pipeline, FrameNet, two sentiment lexicons	0.744
3	**tbmihaylov**	LSTM	–	Word2Vec	–	0.728
4	**ukp**	BiLSTM	Spring 2016, Winter 2017	GloVe	Stanford CoreNLP, DKPro TC	0.717
5	**acoli**	SVM	–	GloVe, Word2Vec	–	0.700
6	**roemmele**	RNN	Spring 2016, Winter 2017	Skip-Thought	–	0.672
7	**mflor**	Rule-based	–	–	VADER sentiment lexicon, Gigaword corpus PMI scores	0.621
8	**Pranav_Goel**	Logistic regression	Spring 2016, Winter 2017	Word2Vec	VADER sentiment lexicon, SICK data set	0.604
9	**ROCNLP (baseline)**	DSSM	Spring 2016, Winter 2017	–	–	0.595

Table 6.3: Overview of models and resources used by the participating teams. For each team only their best performing system on the Spring 2016 Test Set is included, as submitted to CodaLab. Human is reported to perform at 100%.

ukp (Technical University of Darmstadt). Combination of a neural network-based (Bi-LSTM) classifier and a traditional feature-rich approach [Bugert et al., 2017]. Linguistic features include aspects of sentiment, negation, pronominalization and n-gram overlap between the story and possible endings.

roemmele (University of Southern California). Binary classifier based on a recurrent neural network that operates over (sentence-level) Skip-thought embeddings [Roemmele et al., 2017]. For training, different data augmentation methods are explored.

mflor (Educational Testing Service). Rule-based combination of two systems that score possible endings in terms of how well they lexically cohere with and fit the sentiment of the given story [Flor and Somasundaran, 2017]. Sentiment is given priority, and the model backs off to lexical coherence based on pointwise mutual information scores.

Pranav_Goel (IIT Varanasi). Ensemble model that takes into account scores from two systems that measure overlap in sentiment and sentence similarity between the story and the two possible endings [Goel and Singh, 2017].

ROCNLP (baseline) Two feed-forward neural networks trained jointly on ROCStories to project the four-sentences context and the right fifth sentence into the same vector space. This model is called Deep Structured Semantic Model (DSSM) [Huang et al., 2013] and had outperformed all the other baselines reported in Chapter 6.

6.4 Results

An overview of the models and the resources used in each participating system, along with their quantative results, is given in Table 6.3. Given that the DSSM model was previously trained on about 50K ROCStories, we retrained this model on our full dataset of 98,159 stories. We include the results of this model under ROCNLP in Table 6.3. With accuracy values in a range from 60% to 75.2%, we observe that all teams outperform the baseline model. The best result in this shared task has been achieved by **msap**, the participating team from the University of Washington.

6.5 Discussion

We briefly highlight some observations regarding modeling choices and results.

Embeddings. All but two teams made use of pretrained embeddings for words or sentences. **tbmihaylov** [Mihaylov and Frank, 2017] experimented with various pretrained embeddings in their resource-lean model and found that the choice of embeddings has a considerable impact on model accuracy. Interestingly, the best participating team used no pretrained embeddings at all.

Neural networks. The six highest scoring models all include neural network architectures in one way or another. While the teams ranked 3–6 attempt to utilize hidden layers directly for prediction, the top two teams use the output of neural language models to generate different combinations of features. Further, while the third place team's best model was an LSTM, their logistic regression classifier with Word2Vec-based features achieved similar performance. The combination of different neural features (including non-neural ones) appears to have made the difference in the top system's ablation tests.

Sentiment. Three teams report concurrently that a sentiment model alone can achieve 60–65% accuracy but performance seems to vary dependent on implementation details. This is notable in that the sentiment baseline which chose the ending with a matching sentiment to the context (presented in Chapter 6) did not achieve accuracy above random chance. One difference is that these more successful approaches used sentiment lexicons to score words and sentences, whereas in Chapter 6 we used the automatic sentiment classifier in Stanford's CoreNLP. Finally, **mflor** [Flor and Somasundaran, 2017] analyzed the Story Cloze Test Validation (Spring 2016) set and found that 78% of the stories have sentiment bearing words in the first sentences and in at least one possible ending. Evaluating on that subset showed increased performance, further suggesting that sentiment is an important factor in alternate ending prediction.

Stylistic Features on Endings. One of the models proposed by **msap** [Schwartz et al., 2017] ignored the entire story, building features only from the ending sentences. They trained a linear classifier on the right and wrong ending sentences adopting style features that have been shown useful in other tasks such as gender or native language detection. This model achieved remarkably good performance at 72.4%, indicating that there are characteristics inherent to right/wrong endings independent of story reasoning. It is not clear whether these results generalize to novel story ending predictions, beyond the particular Spring 2016 sets. Whether this model captures an artifact of the test set creation, or it indicates general features about how stories are ended must remain for future investigation.

Negative results. Some papers describe additional experiments with features and methods that are not part of the submitted system, because their inclusion resulted in sub-optimal performance. For example, **Pranav_Goel** [Goel and Singh, 2017] discuss additional similarity measures based on doc2vec sentence representations [Le and Mikolov, 2014]; **tbmihaylov** [Mihaylov and Frank, 2017] experiment with ConceptNet Numberbatch embeddings [Speer and Chin, 2016]; and **mflor** [Flor and Somasundaran, 2017] showcase results with alternative sentiment dictionaries such as MPQA [Wilson et al., 2005].

6.6 Conclusions

All participants in the Story Cloze shared task of LSDSem outperformed the previously published best result of 58.5%, and the new state-of-the-art accuracy dramatically increased to 75.2% with the help of a well-designed RNNLM and unique stylistic features on the ending sentences.

One of the main takeaways from the 8 submissions is that the detection of correct ending sentences requires a variety of different reasoners. It appears from both results and post-analysis that sentiment is one factor in correct detection. However, it is also clear that coherence is critical, as the systems with language models all observed increases in prediction accuracy. Beyond these, the best performing system showed that there are stylistic features isolated in the ending sentences, suggesting yet another area of further investigation for the next phases of this task.

As the first shared task on SCT, we decided not to hold a blind challenge. For the future blind challenges, the question is how robust are the presented approaches to novel test cases and how well can they generalize out of the scope of the current evaluation sets. We speculate that the models which use generic language understanding and semantic cohesion criteria rather than relying on certain intricacies of the testing corpora can generalize more successfully, which should be carefully assessed in future.

Although this shared task was successful at setting a new state-of-the-art for SCT, clearly, there is still a long way towards achieving human-level performance of 100% on even the current test set. We are encouraged by the high level of participation in the LSDSem 2017 shared task, and hope the new models and results encourage further research in story understanding. Our findings can help direct the creation of the next SCT datasets towards enforcing deeper story understanding.

7 Conclusion

Understanding and modeling events and their semantic relations plays a crucial role in language understanding. To this end, in Chapter 3 we presented a new model for learning semantically rich event inference rules. Modeling the interaction between events and understanding their causal and temporal relations is a crucial aspect of story understanding and narrative structure learning. The lack of a comprehensive resource of everyday stories along with the lack of an evaluation framework for benchmarking the progress had been hindering the progress in narrative understanding. In order to overcome these issues, in Chapter 4, we introduced the ROCStories corpus of 100K short stories along with the Story Cloze Test as the new evaluation framework. We analyze the outcome of the first shared task on the Story Cloze Test in Chapter 6.

Story Cloze Test provides a great benchmark for automatic evaluation of story understanding and narrative structure learning. Beyond this thesis, the ROCStories[1] project, as a whole, has been widely used in the community for various research purposes. The clear next step in building systems that can understand stories is to go beyond classification as an end goal and focus on story telling and story generation.

[1] http://cs.rochester.edu/nlp/rocstories/

After all, any classification task is prone to being engineered without delivering on the premise of learning commonsense knowledge or performing reasoning. Hence, building systems that can generate logically sound stories is the next step. A major issue in story generation (as a task with a diverse set of possible generations) is automatic evaluation, where the existing word overlap metrics[2] are not adequate. We believe that the Story Cloze Test can be adapted to serve as a benchmark for evaluating story generation as well. The first step towards establishing such a benchmark is to do correlation analysis between any automatic metric and human judgment.

[2] Machine Translation metrics such as BLEU [Clark and Harrison, 2009] or METEOR [Denkowski and Lavie, 2014].

Bibliography

Allen, James, Will De Beaumont, Nate Blaylock, Lucian Galescu George Ferguson, Jansen Orfan, Mary Swift, and Choh Man Teng. 2011. Acquiring commonsense knowledge for a cognitive agent. In *Proceedings of the AAAI Fall Symposium Series: Advances in Cognitive Systems*.

Allen, James, Will De Beaumont, Lucian Galescu, Jansen Orfan, Mary Swift, and Choh Man Teng. 2013. Automatically deriving event ontologies for a commonsense knowledge base. In *IWCS*.

Allen, James F. 1984. Towards a general theory of action and time. *Artif. Intell.*, 23(2):123–154.

Allen, James F., Mary Swift, and Will de Beaumont. 2008. Deep semantic analysis of text. In *Proceedings of the 2008 Conference on Semantics in Text Processing*, STEP '08, pages 343–354. Association for Computational Linguistics, Stroudsburg, PA, USA.

Artstein, Ron and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Comput. Linguist.*, 34(4):555–596.

Bailey, Paul. 1999. Searching for storiness: Story-generation from a reader's perspective. In *AAAI Fall Symposium on Narrative Intelligence*.

Balasubramanian, Niranjan, Stephen Soderland, Oren Etzioni Mausam, and Oren Etzioni. 2013. Generating coherent event schemas at scale. In *EMNLP*, pages 1721–1731.

Bamman, David, Brendan OConnor, and Noah Smith. 2013. Learning latent personas of film characters. ACL.

Banarescu, Laura, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking.

Banko, Michele, Michael J Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *IN IJCAI*, pages 2670–2676.

Bethard, Steven. 2013. Cleartk-timeml: A minimalist approach to tempeval 2013. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 10–14. Association for Computational Linguistics, Atlanta, Georgia, USA.

Bethard, Steven, Leon Derczynski, Guergana Savova, James Pustejovsky, and Marc Verhagen. 2015. Semeval-2015 task 6: Clinical tempeval. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 806–814. Association for Computational Linguistics, Denver, Colorado.

Bittar, Andr, Caroline Hagge, Vronique Moriceau, Xavier Tannier, and Charles Teissdre. 2012. Temporal annotation: A proposal for guidelines and an experiment with inter-annotator agreement. In Nicoletta Calzolari (Conference Chair), Khalid

Choukri, Thierry Declerck, Mehmet Uur Doan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. European Language Resources Association (ELRA), Istanbul, Turkey.

Blanco, Eduardo, Núria Castell, and Dan I. Moldovan. 2008. Causal relation extraction. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*.

Blanco, Eduardo and Dan Moldovan. 2013. A semantically enhanced approach to determine textual similarity. In *Proceedings of EMNLP*, pages 1235–1245. ACL, Seattle, Washington, USA.

Bonial, Claire, Julia Bonn, Kathryn Conger, Jena D. Hwang, and Martha Palmer. 2014. Propbank: Semantics of new predicate types. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland.

Bos, Johan. 2008. Wide-coverage semantic analysis with boxer. In Johan Bos and Rodolfo Delmonte, editors, *Semantics in Text Processing. STEP 2008 Conference Proceedings*, Research in Computational Semantics, pages 277–286. College Publications.

Bowman, Samuel R, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. Learning natural language inference from a large annotated corpus. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Pro-*

cessing, pages 632–642. Association for Computational Linguistics, Stroudsburg, PA.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19:263–311.

Bugert, Michael, Yevgeniy Puzikov, Andreas Rckl, Judith Eckle-Kohler, Teresa Martin, Eugenio Martnez-Cmara, Daniil Sorokin, Maxime Peyrard, and Iryna Gurevych. 2017. LSDSem 2017: Exploring data generation methods for the story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics (LSDSem)*. Association for Computational Linguistics, Valencia, Spain.

Burton, K., A. Java, , and I. Soboroff. 2009. The icwsm 2009 spinn3r dataset. In *In Proceedings of the Third Annual Conference on Weblogs and Social Media (ICWSM 2009)*. San Jose, CA.

Chambers, Nathanael. 2013. Event schema induction with a probabilistic entity-driven model. In *EMNLP*, volume 13, pages 1797–1807.

Chambers, Nathanael and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 602–610. Association for Computational Linguistics, Stroudsburg, PA, USA.

Chambers, Nathanael and Daniel Jurafsky. 2008. Unsupervised learning of narrative event chains. In Kathleen McKeown, Johanna D. Moore, Simone Teufel, James Allan, and Sadaoki Furui, editors, *ACL*, pages 789–797. The Association for Computer Linguistics.

Chang, K.-W., R. Samdani, and D. Roth. 2013. A constrained latent variable model for coreference resolution. In *EMNLP*.

Charniak, Eugene. 1972. Toward a model of children's story comprehension.

Cheng, Patricia W. and Laura R. Novick. 1992. Covariation in natural causal induction. *Psychological Review*, 99(2):365382.

Cheung, Jackie, Hoifung Poon, and Lucy Vanderwende. 2013. Probabilistic frame induction. In *ACL*.

Chklovski, Timothy and Patrick Pantel. 2004. *Proceedings of the EMNLP*, chapter VerbOcean: Mining the Web for Fine-Grained Semantic Verb Relations.

Clark, P., C. Fellbaum, J. R. Hobbs, P. Harrison, W. R. Murray, and J. Thompson. 2008. Augmenting wordnet for deep understanding of text. In *In Semantics in Text Processing*.

Clark, Peter and Phil Harrison. 2009. An inference-based approach to recognizing entailment. In *TAC*, pages 63–72.

Comrie, Bernard. 1976. *Aspect*. Cambridge University Press, Cambridge.

Croft, William. 1991. *Syntactic categories and grammatical relations : the cognitive organization of information / William Croft*. University of Chicago Press Chicago.

Dagan, Ido, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*.

Davidov, Dmitry and Ari Rappoport. 2008. Classification of semantic relationships between nominals using pattern clusters. Association for Computational Linguistics.

Davidson, Donald. 1967. Causal relations. *Journal of Philosophy*, 64(21):691–703.

Denkowski, Michael and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*.

Do, Quang Xuan, Yee Seng Chan, and Dan Roth. 2011. Minimally supervised event causality identification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 294–303. Association for Computational Linguistics, Stroudsburg, PA, USA.

Dolan, Bill, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th COLING*, COLING '04. ACL, Stroudsburg, PA, USA.

Durme, Benjamin Van and Lenhart Schubert. 2008. Open knowledge extraction through compositional language processing. In *In Proceedings of Semantics in Text Processing*.

Erekhinskaya, Tatiana N., Meghana Satpute, and Dan I. Moldovan. 2014. Multilingual extended wordnet knowledge base: Semantic parsing and translation of glosses. In *LREC*, pages 2990–2994.

Ferraro, Francis, Nasrin Mostafazadeh, Ting-Hao Huang, Lucy Vanderwende, Jacob Devlin, Michel Galley, and Margaret Mitchell. 2015. A survey of current datasets for vision and language research. In *Proceedings of the 2015 Conference on Em-*

pirical Methods in Natural Language Processing, pages 207–213. Association for Computational Linguistics, Lisbon, Portugal.

Fillmore, Charles J. 1968. The case for case. In Emmon Bach and Robert T. Harms, editors, *Universals in Linguistic Theory*, pages 0–88. Holt, Rinehart and Winston, New York.

Flor, Michael and Swapna Somasundaran. 2017. Sentiment analysis and lexical cohesion for the story cloze task. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics (LSDSem)*. Association for Computational Linguistics, Valencia, Spain.

Forster, E.M. 1927. *Aspects of the Novel*. Edward Arnold, London.

Ganitkevitch, Juri, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of NAACL-HLT*, pages 758–764. ACL, Atlanta, Georgia.

Gerevini, Alfonso, Lenhart K. Schubert, and Stephanie Schaeffer. 1995. The temporal reasoning tools timegraph i-ii. *International Journal on Artificial Intelligence Tools*, 4(1-2):281–300.

Giampiccolo, Danilo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, RTE '07, pages 1–9. ACL, Stroudsburg, PA, USA.

Girju, Roxana, Preslav Nakov, Vivi Nastase, Stan Szpakowicz, Peter Turney, and Deniz Yuret. 2007. Semeval- 2007 task 04: Classification of semantic relations between nominals. abs/1206.5333.

Goel, Pranav and Anil Kumar Singh. 2017. IIT (BHU): System description for LSDSem'17. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics (LSDSem)*. Association for Computational Linguistics, Valencia, Spain.

Goodwin, Travis, Bryan Rink, Kirk Roberts, and Sanda M. Harabagiu. 2012. Utdhlt: Copacetic system for choosing plausible alternatives. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 461–466. Association for Computational Linguistics, Stroudsburg, PA, USA.

Gordon, Andrew S, Cosmin Adrian Bejan, and Kenji Sagae. 2011. Commonsense causal reasoning using millions of personal stories. In *AAAI*.

Gordon, Andrew S., Zornitsa Kozareva, and Melissa Roemmele. 2012. SemEval-2012 Task 7: Choice of Plausible Alternatives: An Evaluation of Commonsense Causal Reasoning. In *Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012)*. Montreal, Canada.

Gordon, Andrew S. and Reid Swanson. 2009. Identifying Personal Stories in Millions of Weblog Entries. In *Third International Conference on Weblogs and Social Media, Data Challenge Workshop*. San Jose, CA.

Gordon, Jonathan. 2014. *Inferential Commonsense Knowledge from Text*. Ph.D. thesis.

Harris, Z. 1985. Distributional structure. In *Katz, J. J. (ed.), The Philosophy of Linguistics. New York: Oxford University Press.*, pages 26–47.

Hendrickx, Iris, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó. Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, pages 33–38. Association for Computational Linguistics, Stroudsburg, PA, USA.

Hermann, Karl Moritz, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1693–1701. Curran Associates, Inc.

Hierchia, G. and S. McConnell-Ginet. 2001. *Meaning and Grammar: An Introduction to Semantics*. MIT Press.

Hobbs, Jerry R. 2005. Toward a useful concept of causality for lexical semantics. *Journal of Semantics*, 22(2):181–209.

Huang, Po-Sen, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, CIKM '13, pages 2333–2338. ACM, New York, NY, USA.

Huang, Ting-Hao (Kenneth), Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, C. Lawrence Zitnick, Devi Parikh, Lucy Vanderwende, Michel Galley, and Margaret Mitchell. 2016. Visual storytelling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies, pages 1233–1239. Association for Computational Linguistics, San Diego, California.

Ide, N. and J. Vronis. 1994. Knowledge extraction from machine-readable dictionaries: An evaluation. In *In P. Steffens, editor, Machine Translation and the Lexicon. Springer-Verlag, Germany.*

Ikuta, Rei, Will Styler, Mariah Hamang, Tim O'Gorman, and Martha Palmer. 2014. Challenges of adding causation to richer event descriptions. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 12–20. Association for Computational Linguistics, Baltimore, Maryland, USA.

Jans, Bram, Steven Bethard, Ivan Vulić, and Marie Francine Moens. 2012. Skip n-grams and ranking functions for predicting script events. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 336–344. Association for Computational Linguistics.

Khemlani, Sangeet, Aron K Barbey, and Philip Nicholas Johnson-Laird. 2014. Causal reasoning with mental models. *Frontiers in Human Neuroscience*, 8(849).

Kiros, Ryan, Yukun Zhu, Ruslan Salakhutdinov, Richard S Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. *NIPS*.

Le, Quoc V and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*. Beijing, China.

Levary, David, Jean-Pierre Eckmann, Elisha Moses, and Tsvi Tlusty. 2012. Loops and self-reference in the construction of dictionaries. *Phys. Rev. X*.

Levesque, Hector J. 2011. The winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*. AAAI.

Levin, Beth. 2000. Aspect, lexical semantic representation, and argument expression. pages 413–429. Cambridge University Press, Cambridge.

Lewis, David. 1973. *Counterfactuals*. Blackwell Publishers, Oxford.

Lin, Chin-Yew and Franz Josef Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL ’04. Association for Computational Linguistics, Stroudsburg, PA, USA.

Lin, Dekang and Patrick Pantel. 2001. Dirt: Discovery of inference rules from text. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’01, pages 323–328. ACM, New York, NY, USA.

Liu, H. and P. Singh. 2004. Conceptnet - a practical commonsense reasoning tool-kit. *BT Technology Journal*, 22(4):211–226.

Llorens, Hector, Nathanael Chambers, Naushad UzZaman, Nasrin Mostafazadeh, James Allen, and James Pustejovsky. 2015. Semeval-2015 task 5: Qa tempeval - evaluating temporal information understanding with question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 792–800. Association for Computational Linguistics, Denver, Colorado.

Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Manshadi, Mehdi, Reid Swanson, and Andrew S. Gordon. 2008. Learning a Probabilistic Model of Event Sequences From Internet Weblog Stories. In *21st Conference of the Florida AI Society, Applied Natural Language Processing Track*. Coconut Grove, FL.

McIntyre, Neil and Mirella Lapata. 2009. Learning to tell tales: A data-driven approach to story generation. 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*, pages 217–225. Singapore.

Mehdi H. Manshadi, Mary Swift., James Allen. 2008. Towards a universal underspecified semantic representation. In *In 13th Conference on Formal Grammar.*

Melamud, Oren, Ido Dagan, Jacob Goldberger, and Idan Szpektor. 2013. Using lexical expansion to learn inference rules from sparse data. In *In Proceedings of ACL 2013.*

Méndez, Gonzalo, Pablo Gervás, and Carlos León. 2014. A model of character affinity for agent-based story generation. In *9th International Conference on Knowledge, Information and Creativity Support Systems*. Springer-Verlag, Springer-Verlag, Limassol, Cyprus.

Mihaylov, Todor and Anette Frank. 2017. Simple story ending selection baselines. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics (LSDSem)*. Association for Computational Linguistics, Valencia, Spain.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference* *on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119.

Miller., G. 1995. Wordnet: A lexical database for english. In *In Communications of the ACM.*

Minsky, Marvin. 1975. Minsky's frame system theory. In *Proceedings of the 1975 Workshop on Theoretical Issues in Natural Language Processing*, TINLAP '75, pages 104–116. Association for Computational Linguistics, Stroudsburg, PA, USA.

Mirza, Paramita and Sara Tonelli. 2014. An analysis of causality between events and its relation to temporal information. In *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, August 23-29, 2014, Dublin, Ireland*, pages 2097–2106.

Moldovan, Dan I., Christine Clark, Sanda M. Harabagiu, and Daniel Hodges. 2007. Cogex: A semantically and contextually enriched logic prover for question answering. *J. Applied Logic*, 5(1):49–69.

Moldovan, Dan I. and Vasile Rus. 2001. Explaining answers with extended wordnet. In *ACL.*

Mostafazadeh, Nasrin and James F. Allen. 2015. Learning semantically rich event inference rules using definition of verbs. In *Computational Linguistics and Intelligent Text Processing - 16th International Conference, CICLing 2015, Cairo, Egypt, April 14-20, 2015, Proceedings, Part I*, pages 402–416.

Mostafazadeh, Nasrin, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016a. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of NAACL HLT*. Association for Computational Linguistics, San Diego, California.

Mostafazadeh, Nasrin, Alyson Grealish, Nathanael Chambers, James Allen, and Lucy Vanderwende. 2016b. Caters: Causal and temporal relation scheme for semantic annotation of event structures. In *Proceedings of the Fourth Workshop on Events*, pages 51–61. Association for Computational Linguistics, San Diego, California.

Mostafazadeh, Nasrin, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. 2016c. Generating natural questions about an image. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1802–1813. Association for Computational Linguistics, Berlin, Germany.

Mostafazadeh, Nasrin, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. 2017. Lsdsem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51. Association for Computational Linguistics, Valencia, Spain.

Mostafazadeh, Nasrin, Lucy Vanderwende, Wen-tau Yih, Pushmeet Kohli, and James Allen. 2016d. Story cloze evaluator: Vector space representation evaluation by predicting what happens next. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 24–29. Association for Computational Linguistics, Berlin, Germany.

Mueller, Erik T. 2002. Understanding script-based stories using commonsense reasoning. *Cognitive Systems Research*, 5:2004.

Mueller, Erik T. 2007. Modeling space and time in narratives about restaurants. *LLC*, 22(1):67–84.

Neeleman, Ad and Hans Van De Koot. 2012. The theta system: Argument structure at the interface. *The Linguistic Expression of Causation*, pages 20–51.

Nguyen, Kiem-Hieu, Xavier Tannier, Olivier Ferret, and Romaric Besançon. 2015. Generative event schema induction with entity disambiguation. In *Proceedings of the 53rd annual meeting of the Association for Computational Linguistics (ACL-15)*.

Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *CL*, 29(1):19–51.

Peng, Haoruo, Daniel Khashabi, and Dan Roth. 2015. Solving hard coreference problems. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Picard, O., A. Blondin-Masse, S. Harnad, O. Marcotte, G. Chicoisne, and Y. Gargouri. 2009. Hierarchies in dictionary denition space. In NIPS Workshop on Analyzing Networks and Learning With Graphs.

Pichotta, Karl and Raymond J Mooney. 2014a. Statistical script learning with multi-argument events. *EACL 2014*, page 220.

Pichotta, Karl and Raymond J. Mooney. 2014b. Statistical script learning with multi-argument events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*. Gothenburg, Sweden.

Prasad, Rashmi, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K. Joshi, and Bonnie L. Webber. 2008. The penn discourse treebank 2.0. In *LREC*. European Language Resources Association.

Prasad, Rashmi, Aravind Joshi, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, and Bonnie Webber. 2005. The penn discourse treebank as a resource for natural language generation. In *In Proc. of the Corpus Linguistics Workshop on Using Corpora for Natural Language Generation*, pages 25–32.

Pustejovsky, James, Jos Castao, Robert Ingria, Roser Saur, Robert Gaizauskas, Andrea Setzer, and Graham Katz. 2003. Timeml: Robust specification of event and temporal expressions in text. In *Fifth International Workshop on Computational Semantics (IWCS-5*.

Pustejovsky, James and Amber Stubbs. 2011. Increasing informativeness in temporal annotation. In *Proceedings of the 5th Linguistic Annotation Workshop*, LAW V '11, pages 152–160. Association for Computational Linguistics, Stroudsburg, PA, USA.

Quirk, Chris, Chris Brockett, and William Dolan. 2004. Monolingual machine translation for paraphrase generation.

Radinsky, Kira, Sagie Davidovich, and Shaul Markovitch. 2012. Learning to predict from textual data. *J. Artif. Intell. Res. (JAIR)*, 45:641–684.

Rahman, Altaf and Vincent Ng. 2012. Resolving complex cases of definite pronouns: The winograd schema challenge. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 777–789.

Ramesh, Balaji Polepalli, Rashmi Prasad, Tim Miller, Brian Harrington, and Hong Yu. 2012. Automatic discourse connective detection in biomedical text. *JAMIA*, 19(5):800–808.

Regneri, Michaela, Alexander Koller, and Manfred Pinkal. 2010. Learning script knowledge with web experiments. In *Proceedings of the 48th Annual Meeting of the*

Association for Computational Linguistics, pages 979–988. Association for Computational Linguistics.

Riaz, Mehwish and Roxana Girju. 2013. *Proceedings of the SIGDIAL 2013 Conference*, chapter Toward a Better Understanding of Causality between Verbal Events: Extraction and Analysis of the Causal Power of Verb-Verb Associations, pages 21–30. Association for Computational Linguistics.

Richardson, Matthew, Christopher J. C. Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *EMNLP*, pages 193–203. ACL.

Riedl, M. and Carlos León. 2008. Toward vignette-based story generation for drama management systems. In *Workshop on Integrating Technologies for Interactive Stories - 2nd International Conference on INtelligent TEchnologies for interactive enterTAINment*.

Rink, Bryan, Cosmin Bejan, and Sanda Harabagiu. 2010. Learning textual graph patterns to detect causal event relations.

Roemmele, Melissa, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of Plausible Alternatives: An Evaluation of Commonsense Causal Reasoning. In *AAAI Spring Symposium on Logical Formalizations of Commonsense Reasoning*. Stanford University.

Roemmele, Melissa, Sosuke Kobayashi, Naoya Inoue, and Andrew Gordon. 2017. An RNN-based binary classifier for the story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics (LSDSem)*. Association for Computational Linguistics, Valencia, Spain.

Rudinger, Rachel, Pushpendre Rastogi, Francis Ferraro, and Benjamin Van Durme. 2015. Script induction as language modeling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP-15)*.

Sammons, Mark, V.G.Vinod Vydiswaran, and Dan Roth. 2012. Recognizing Textual Entailment. In Daniel M. Bikel and Imed Zitouni, editors, *Multilingual Natural Language Applications: From Theory to Practice*, chapter 6, pages 209–258. IBM Press, Pearson.

Schank, Roger C. and Robert P. Abelson. 1977. *Scripts, Plans, Goals and Understanding: an Inquiry into Human Knowledge Structures*. L. Erlbaum, Hillsdale, NJ.

Schenk, Niko and Christian Chiarcos. 2017. Resource-lean modeling of coherence in commonsense stories. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics (LSDSem)*. Association for Computational Linguistics, Valencia, Spain.

Schubert, Lenhart. 2002. Can we derive general world knowledge from texts? In *Proceedings of the Second HLT*, HLT ’02, pages 94–97. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Schubert, Lenhart K. and Chung Hee Hwang. 2000. Episodic logic meets little red riding hood: A comprehensive, natural representation for language understanding. In *Natural Language Processing and Knowledge Representation: Language for Knowledge and Knowledge for Language*. MIT/AAAI Press.

Schwartz, Roy, Maarten Sap, Ioannis Konstas, Leila Zilles, Yejin Choi, and Noah A. Smith. 2017. Story cloze task: UW NLP system. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics (LSDSem)*. Association for Computational Linguistics, Valencia, Spain.

Singh, P. 2001. The public acquisition of commonsense knowledge.

Singh, Push, Thomas Lin, Erik T. Mueller, Grace Lim, Travell Perkins, and Wan Li Zhu. 2002. Open mind common sense: Knowledge acquisition from the general public. In *On the Move to Meaningful Internet Systems, 2002 - DOA/CoopIS/ODBASE 2002 Confederated International Conferences DOA, CoopIS and ODBASE 2002*, pages 1223–1237. Springer-Verlag, London, UK, UK.

Speer, Robert and Joshua Chin. 2016. An ensemble method to produce high-quality word embeddings. *arXiv preprint arXiv:1604.01692*.

Steven Bethard, Sara Klingenstein, William Corvey and James H. Martin. 2008. Building a corpus of temporal-causal structure. In Bente Maegaard Joseph Mariani Jan Odijk Stelios Piperidis Daniel Tapias Nicoletta Calzolari (Conference Chair), Khalid Choukri, editor, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*. European Language Resources Association (ELRA), Marrakech, Morocco. Http://www.lrec-conf.org/proceedings/lrec2008/.

Styler, IV William F., Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C. de Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, and James Pustejovsky. 2014. Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143–154.

Swanson, Reid and Andrew S. Gordon. 2008. Say Anything: A Massively collaborative Open Domain Story Writing Companion. In *First International Conference on Interactive Digital Storytelling*. Erfurt, Germany.

Szpektor, Idan, Eyal Shnarch, and Ido Dagan. 2007. Instance-based evaluation of entailment rule acquisition. In John A. Carroll, Antal van den Bosch, and Annie Zaenen, editors, *In Proceeding of ACL Conference*. ACL.

Szpektor, Idan, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. 2004. Scaling web-based acquisition of entailment relations. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 41–48. ACL.

Talmy, Leonard. 1988. Force dynamics in language and cognition. *Cognitive Science*, 12(1):49–100.

Talmy, Leonard. 2003. *Toward a cognitive semantics*, volume 1. MIT press.

Tarjan, Robert. 1972. Depth first search and linear graph algorithms. *SIAM Journal on Computing*.

Taylor, Wilson L. 1953. ”Cloze procedure”: a new tool for measuring readability. *Journalism quarterly*.

Turner, Scott R. 1994. The creative process: A computer model of storytelling. *Hillsdale: Lawrence Erlbaum.*

UzZaman, Naushad, Hector Llorens, James F. Allen, Leon Derczynski, Marc Verhagen, and James Pustejovsky. 2012. Tempeval-3: Evaluating events, time expressions, and temporal relations. *CoRR*, abs/1206.5333.

Verberne, Suzan, Lou Boves, Nelleke Oostdijk, and Peter-Arno Coppen. 2007. Evaluating discourse-based answer extraction for why-question answering. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’07, pages 735–736. ACM, New York, NY, USA.

Weisman, Hila, Jonathan Berant, Idan Szpektor, and Ido Dagan. 2012. Learning verb inference rules from linguistically-motivated evidence. In *Proceedings of EMNLP-CoNLL*, pages 194–204. ACL, Jeju Island, Korea.

Weston, Jason, Antoine Bordes, Sumit Chopra, and Tomas Mikolov. 2015. Towards ai-complete question answering: A set of prerequisite toy tasks. *CoRR*, abs/1502.05698.

Wilson, Theresa, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 347–354. Vancouver, British Columbia, Canada.

Winograd, Terry. 1972. *Understanding Natural Language*. Academic Press, Inc., Orlando, FL, USA.

Wolfe, Travis, Benjamin Van Durme, Mark Dredze, Nicholas Andrews, Charley Beller, Chris Callison-Burch, Jay DeYoung, Justin Snyder, Jonathan Weese, Tan Xu, and Xuchen Yao. 2013. Parma: A predicate argument aligner. In *Proceedings of ACL short*.

Wolff, Phillip. 2007. Representing causation. *Journal of Experiment Psychology: General*, 136:82111.

Wolff, Phillip and Grace Song. 2003. Models of causation and the semantics of causal verbs. *Cognitive Psychology*, 47(3):276–332.

Zhu, Yukun, Ryan Kiros, Richard Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *arXiv preprint arXiv:1506.06724*.