Weakly-Supervised Modeling of Contextualized Event Embedding for Discourse Relations

I-Ta Lee, Maria Leonor Pacheco, Dan Goldwasser
Department of Computer Science
Purdue University
West Lafayette, IN, USA
{lee2226, pachecog, dgoldwas}@purdue.edu

Abstract

Representing, and reasoning over, long narratives requires models that can deal with complex event structures connected through multiple relationship types. This paper suggests to represent this type of information as a narrative graph and learn contextualized event representations over it using a relational graph neural network model. We train our model to capture event relations, derived from the Penn Discourse Tree Bank, on a huge corpus, and show that our multi-relational contextualized event representation can improve performance when learning script knowledge without direct supervision and provide a better representation for the implicit discourse sense classification task.

1 Introduction

Representing world knowledge, and reasoning over it, to help improve language understanding is one of the longest standing AI goals. Structured knowledge representations such as *scripts* (Schank and Abelson, 1977) capture temporal relations between events to describe human-level representations of common scenarios. For example, the *Restaurant Script* captures the fact that food is first ordered and only then paid for. Initial works relied on manual script construction, a labor-intensive task that is hard to scale to the number of possible scenarios. More recent works focus on extracting this knowledge directly from text, using symbolic event representations (Chambers and Jurafsky, 2008) or more recently, exploiting representation learning advances and representing events using dense vectors, learned from data (Pichotta and Mooney, 2016a; Granroth-Wilding and Clark, 2016; Wang et al., 2017; Lee and Goldwasser, 2018; Li et al., 2018). While these works differ in the way the internal structure of the event is represented, broadly speaking, the resulting models resemble word-embedding approaches (Mikolov et al., 2013), representing event co-occurrence in a low-dimensional vector space, and as a result use vector similarity over their embedding to measure their relationship.

In this paper, we follow the observation that many natural language understanding tasks require a more expressive representation that can capture the context in which events appear (Goldwasser and Zhang, 2016) and consider multiple relations between events (Lee and Goldwasser, 2019), and going beyond simple event similarity to represent relations. To help explain the intuition behind it, consider the following example, consisting of a short story and a Multiple-Choice Narrative Cloze (MCNC) question (Granroth-Wilding and Clark, 2016), the standard evaluation for such models.

Example 1: ***Jenny gets coffee***

Jenny woke up very early and had some time to kill. She went outside and noticed that it was raining, so she went inside her favorite coffee-shop. She greeted the waiter ...

What happened next?

(a) she bought a new car.
(b) she ordered a steamy latte.
(c) she ordered a large breakfast
(d) she asked about open positions.

Events typically correspond to predicate-argument structures, and the narrative cloze task is modeled as ranking event pairs based on their similarity, using consecutive events as positive examples. Based on this approach, identifying that (a) is not a reasonable option is straight-forward, however, the task of separating between (b), (c) and (d) is much harder, and requires models that can reason about the broader context in which an event occurs, capturing the *cause* of entering the coffee-shop (i.e., killing time) and the activity most associated with it (i.e., ordering coffee).

To meet this challenge we suggest a multi-relational contextualized representation of events,

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 4962–4972
November 16 - 20, 2020. ©2020 Association for Computational Linguistics

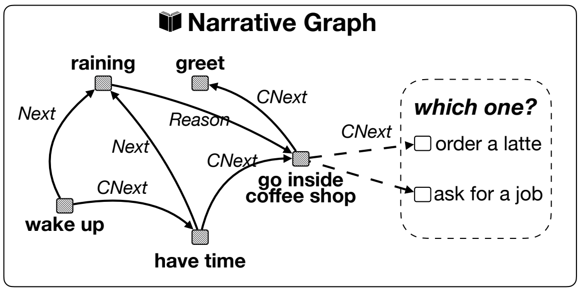

Figure 1: Narrative Graph extracted for Example 1. Some edges are omitted for clarity.

generalizing ideas from contextualized word representations (Peters et al., 2018; Devlin et al., 2018) to multi-relational narrative representation. Similar to contextualized word representations, we suggest learning an event representations which captures the narrative it is a part of. For example, the event *"she went inside the coffee-shop"* would be represented differently given different context, such as different weather conditions (*"it was sunny and warm"*), different time of the day (*"it was almost noon when she woke up"*) or if the protagonist needed employment. In each one of these cases, the relationship between the contextualized event representation and the answer candidates would be different. **Unlike** contextualized word embedding models, our challenging settings require dealing with complex internal event structure (associations between the predicate and the entities, and their semantic roles), long narrative text, often beyond the length that can be effectively represented using these models, as well as representing complex relationships between events, beyond co-occurrence. To identify the association between the question and the correct answer, (b), the contextualized event representation should capture the reason for entering the coffee-shop, in this case indicated by the discourse connective *"so"*.

We propose using Narrative Graph (NG) to represent the text, consisting of nodes, corresponding to events, and edges representing observed relations between events. These relations capture the sequential order of event occurrence, represented using the **Next** relationship, events sharing a coreferenced entity are connected via the **CNext** relationship. In addition, we represent discourse relations corresponding to six relations defined in the Penn Discourse Tree Bank (PDTB) (Prasad et al., 2007), which include **Before, After, Sync., Contrast, Reason** and **Result**. We rely on the discourse connectives associated with each relations to add these relations to the NG. Figure 1 provides an example of a partial narrative graph corresponding to the example above. We define the contextualized event embedding over this graph, by using a Relational Graph Convolution Network (R-GCN) (Schlichtkrull et al., 2018), a relational variant of the Graph Convolution Network architecture (GCN) (Kipf and Welling, 2016), which creates contextualized node representations by unfolding the graph structure recursively into a tree structure and learns a composition function, similar to a tree-based Recursive NN. This architecture allows us to take into account the narrative structure and the different discourse relations connecting events when embedding the event node.

We associate the event text, along with its local context, with each node, and use BERT (Devlin et al., 2018) to encode its initial representation, contextualized locally. During training, the error is back-propagated over the graph to train the narrative relationships' composition parameters, and then to BERT, to train the NG-contextualized representation of the individual event.

We define an unsupervised learning process, learning to recover removed edges from a given narrative graph and capture incorrect associations between event nodes and edges. This process allows the model to learn the association between the missing information and the observed context in the narrative graph. We use the New York Times section of English Gigaword (Parker et al., 2011) for training the model. We evaluate the model on MCNC and its relational variants, as well as the popular, and challenging, implicit discourse classification task (Xue et al., 2016).

2 Related Work

Statistical script learning is an unsupervised learning problem addressing the probabilities of event co-occurrence. Chambers and Jurafsky (2008) started the early work, using Pairwise Mutual Information (PMI) -based models to calculate the conditional probability distribution. In recent years, neural-based learning frameworks emerged, leading to a wave of model evolution. Granroth-Wilding and Clark (2016) combine Skip-Gram (Mikolov et al., 2013), a word embedding model, with neural networks for learning event representations. Pichotta and Mooney (2016b) built a Long Short Term Memory (LSTM) network to learn the next relation along coreferent event chains, modeling relationships over event sequences. Weber et al. (2017) constructed a high-dimensional tensor-based neural network, inspired by Computer Vision models, to learn event representations. Lee and Goldwasser (2018) and Wang et al. (2017) showed that adding event features, such as entity animacy, sentiments, or event order information, help commonsense inference. Li et al. (2018) started using graph structures, beyond pairwise or sequential models, to capture event context. Lee and Goldwasser (2019) made a multi-relational model capturing different relation types between events with translating-based objective functions, which is the closest work to this paper. In this paper, the NG model uses two-level (word and event) contextualizations, built on top of a pre-trained language model–BERT, coupled with multi-relational graph structures, to learn event representations.

In the literature, the definitions of events can be categorized in two ways: entity-centric or predicate-centric. Early works (Chambers and Jurafsky, 2008) operated on the entity-centric events, following coreference chains of a specific entity to model sequences of events. Predicate-GR (Granroth-Wilding and Clark, 2016) was a widely adopted event definition here, consisting of a pair of dependency type, such as subject or object, and predicate token, such as verbs. Recent works (Pichotta and Mooney, 2014; Lee and Goldwasser, 2018, 2019) moved to the predicate-centric events (also called multi-argument events). Each event was anchored at a predicate and considered related entity mentions and modifiers as context to the event. Other works focusing on event extraction (Walker et al., 2006) or relation extraction (Han et al., 2019) also adopted this definition, as it tends to capture more comprehensive view of events' semantics, aggregating information from multiple entities. In this paper, we also choose this definition, since our goal is to utilize events' context to model event relationships.

Graph neural models are often applied in Knowledge Base Completion. Early works used random-walk-based methods, aiming to build scalable neural models, such as DeepWalk (Perozzi et al., 2014), node2vec (Grover and Leskovec, 2016), LINE (Tang et al., 2015) and GraphSAGE (Hamilton et al., 2017). Graph Convolution Networks (GCN) introduced by Kipf and Welling (2016) provide an efficient way of aggregating features from neighboring nodes. Several GCN variants followed, Relational Graph Convolution Networks (R-GCN) (Schlichtkrull et al., 2018) added relation type information to address the multi-relational knowledge bases. Graph Attention Networks (GAT) (Veličković et al., 2017) manipulated attention layers for aggregating neighboring messages. Gated mechanisms (Marcheggiani and Titov, 2017; Dauphin et al., 2017) were proposed for mitigating the impact of data noise. GCN were also used for NLP applications, to represent structure (Marcheggiani and Titov, 2017) and social information (Li and Goldwasser, 2019). In this paper, we adopt R-GCN for NG, as modeling different types of relationships is crucial for event commonsense inference, as attested by Lee and Goldwasser (2019).

Discourse relations are crucial aspects for completing language understanding. Early works focused on identifying explicit and implicit discourse relations under supervised settings (Zhou et al., 2010; Park and Cardie, 2012; Biran and McKeown, 2013; Xue et al., 2016), while recent works mined discourse connectives to refine sentence representations unsupervisedly (Malmi et al., 2017; Nie et al., 2019; Sileo et al., 2019). Our work learns discourse relations between events by leveraging the fact that some explicit connectives and their categories are relatively easy to identify. We build a simplified discourse annotator that can be used to extract discourse relations between events without suffering from high noise.

3 Model

3.1 Overview

We propose a learning framework for constructing event embeddings, contextualized by a relational event graph. The proposed approach can be used for many discourse and narrative analysis tasks, that go beyond the sentence level.

The framework consists of two levels of hierarchical contextualizations. The first, defined at the word level, uses contextualized word embeddings, such as BERT (Devlin et al., 2018), which was applied successfully to various Natural Language Understanding (NLU) tasks. The second level, which is the main novelty of this paper, contextualizes *event*. Similar to words, events in different scenarios can have different meanings, e.g., a smile can mean positive or negative signs. As contextualized word embeddings tend to focus on local information, failing to capture high-level conceptual transitions, such as discourse relations, we

Relation Types between Events		
Complete Name	Abbrev.	#relations.
Next	**Next**	274M
Coreferent Next	**CNext**	66M
Temporal.Async.Precedence	**Before**	1.63M
Temporal.Async.Succession	**After**	1.52M
Temporal.Synchrony	**Sync.**	0.55M
Comparison.Contrast	**Contrast**	0.91M
Contingency.Cause.Reason	**Reason**	0.22M
Contingency.Cause.Result	**Result**	2.41M

Table 1: Statistics of event relations extracted from *New Youk Times* section of *English Gigawords* (Parker et al., 2011). 1.42M documents are used after excluding documents that are too long or too short.

suggest a new data structure to represent the input, called Narrative Graph (NG), which represents a document using its events and their relationships.

3.2 Preprocessing

Event Extraction We define events as verb predicates that have at least one dependency link to entity mentions. The dependency links include subject (nsubj), direct object (dobj), indirect object (iobj), prepositional words or noun modifiers (nmod)[1]. Along with the verb predicates, we take the sentence they appear in as their local (word-level) context. To further differentiate the representations of the events appearing in the same sentence, we take into account their predicate position as inputs. Each event appears in a NG as a node, and edges between nodes represent event relationships.

Relation Extraction The relation is defined as a triplet (e_h, r, e_t), where e_h and e_t are head and tail events, and r is the relation type. We extract eight types of relations, including two narrative relations, **CNext** and **Next**, and six discourse relations. All relations are directional. Table 1 summarizes statistics of the relations we extracted from the corpus English Gigaword (Parker et al., 2011). We explain each relation type as follows:

(1) The **CNext** relation stands for Coreferent Next relation, inspired by (Chambers and Jurafsky, 2008), capturing narrative relationships between events with shared entities on coreference chains[1]. Based on the procedure proposed by (Lee and Goldwasser, 2019), we first identify all possible events and connect pairs of the events with a **CNext** relation if they have entity mentions appearing in the same coreference chain. For example, "Jim *shot* John. John *died*." *shot* and *died* have the **CNext** relation *(shot, **CNext**, died)* because the entity John is the participant to both events in a sequential order.

(2) The **Next** relation is defined between events appearing in the neighboring sentences. It aims to capture the event relationship where two events are relevant but do not have shared participants. For example, "The weather turned bad. The rain started falling." has the relation *(turned, **Next**, falling)*. These two events have no shared participant but are clearly related.

(3) The six discourse relations (the last six rows in Table 1) are selected from PDTB for capturing transitions between events. For example, "Jenny fell asleep, because she was tired." has a relation **Reason** and the argument spans (*ARG1* and *ARG2*) are the two clauses. Instead of having relations over arguments spans, we adapt the relation definition to the event level, where e_h comes from *ARG1* and e_t comes from *ARG2*. Note that when getting sentence context for event predicates, we mask the discourse connective, such as "because", from the model, because we want the model to learn relationships between events, rather than a simple decision function of key words. More detailed relation definitions can be found in the PDTB annotation manual (Prasad et al., 2007).

Since the relations annotated in PDTB are not enough for generalizing event embeddings, we construct a rule-based discourse annotator. We first compile a list of discourse connectives by looking at the annotated relations in PDTB. To reduce the noise, only highly indicative connectives are considered. For example, "however" indicates **Contrast** relation and "in the meanwhile" denotes **Sync.** relation. We then search for the discourse connectives (CONN) in documents, and use three patterns to locate the argument spans:

1. {ARG1}. {CONN} {ARG2}.
2. {ARG1}, {CONN} {ARG2}.
3. {CONN} {ARG2}, {ARG1}.

where the first pattern has a discourse relation across two sentences while the other two have it in one sentence with multiple clauses. Since each argument span could have multiple events, we use all possible pairs. While the extracted relations are noisy, we demonstrate that they help in learning event representations in experiments.

Narrative Graph The extracted events and relations from a document form a NG. The NG is an

[1] Stanford CoreNLP (Manning et al., 2014) pipeline is used for extracting dependency trees and coreference resolutions.

event-level abstraction of the document, as depicted in Figure 1, describing typed relational transitions between events. In this paper, the NG is modeled with a graph neural network. We have to limit the graph size, as there are physical memory limitation when training the network. The size is controlled by two hyperparameters: s_{min} and s_{max}, standing for the minimum and maximum numbers of nodes.

3.3 Neural Architecture

We define two contextualized embedding functions:

$$\begin{aligned} e &= f_{word}(p, loc(p), ctx(p)), \\ e' &= f_{event}(e, g(e)), \end{aligned} \tag{1}$$

where p is the target event predicate; $loc(p)$ is the token offset of the predicate in the sentence; $ctx(.)$ is the local context function; $f_{word}(.)$ encodes p and get its contextualized word embedding e, representing the event with the local context; $g(.)$ is the event context function, retrieving all events and relations in the document, i.e., $g(e) = \{e^*, r | e^* \in doc(e), r \in doc(e)\}$; lastly, $f_{event}(.)$ encodes the event, along with its NG, and outputs the contextualized event embedding e'.

In this paper, we use BERT (Devlin et al., 2018) for f_{word}, Relational Graph Convolution Network (R-GCN) (Schlichtkrull et al., 2018) for f_{event}, and NG for the event context function $g(.)$. The following subsections will explain more in details. Note that this architecture setting is for demonstrating purposes. Our framework retains the flexibility of adopting other embedding and context functions.

Word-Level Contextualization Figure 2 visualizes the NG model. The input tokens are the event predicate along with its sentence context. We use BERT as the local (word-level) encoder. It has three embedding tables to represent the input, which are token embedddings, position embeddings, and token type embeddings. The token type embeddings were originally used for distinguishing input sentences for BERT's next-sentence pre-training task. Recent work (Han et al., 2019) has shown that an effective way to fine-tuned BERT for events is to encode special tokens, such as event predicates, with the token type (token_type_id). We adopt this idea and use the token type inputs to mark event predicates, i.e., $token\_type\_id = 1$ for predicate tokens and $token\_type\_id = 0$ otherwise. This method emphasizes predicates when encoding events and generates slightly different contextualized representations for different emphases, even in the same sentence. For the rest of this paper, unless mention explicitly, we encode events with BERT in this way. In our training procedure, we initialize our model with pre-trained BERT and fine-tune it, and represent each event with its predicate word embeddings output from BERT.

Event-Level Contextualization Graph Convolution Networks (GCN) (Kipf and Welling, 2016) were designed to process graph structures by propagating messages between local neighboring nodes through graph convolution. R-GCN (Schlichtkrull et al., 2018) adds relational considerations so that it can operate on multi-relational graphs[2]. The network is defined as follows:

$$h_i^{l+1} = ReLU\left(\sum_{r \in R} \sum_{u \in N_r(v_i)} \frac{1}{c_{i,r}} W_r^l h_u^l\right), \tag{2}$$

where h_i^{l+1} is the hidden representation for the node v_i at layer $l+1$; $N_r(v_i)$ is the set of neighboring nodes under the r relation; $c_{i,r}$ is the normalizatoin factor; W_r^l is the relation-specific parameters for layer l; and R is the set of relation types (in our case, the eight types denoted in Table 1).

The R-GCN is connected to BERT on top, taking only predicate word embedding to represent each event node. The node representations are contextualized by local neighbors according to NG. The number of R-GCN layers l_{rgcn} is a hyperparameter to control the order of neighbors to be considered.

Negative Sampling As the NG model is contextualized over NG, we have to create negative graphs by removing some edges and predict them. To do so, we first determine a set of hyperparameters: the number of truncated graphs n_{neg_g} created for each NG, the proportion of edges to be removed r_{neg_e} for each truncated graph, and the number of negative edges n_{neg_e} to be sampled for each removed edge. Once they are determined, we sample the edges to be removed by their relation type, based on a smoothed distribution, where we sample **Next**, **CNext**, and each discourse relation with probabilities 0.5, 0.2 and 0.05 respectively. The reason why we smooth the distribution is to avoid undersampling the rare relation types. For each sampled edge, we truncate its e_h, r, and e_t uniformly.

[2] We also have experimented with gated mechanism (Marcheggiani and Titov, 2017) for R-GCN to mitigate the noise from parsing errors. However, the performance is slightly worse.

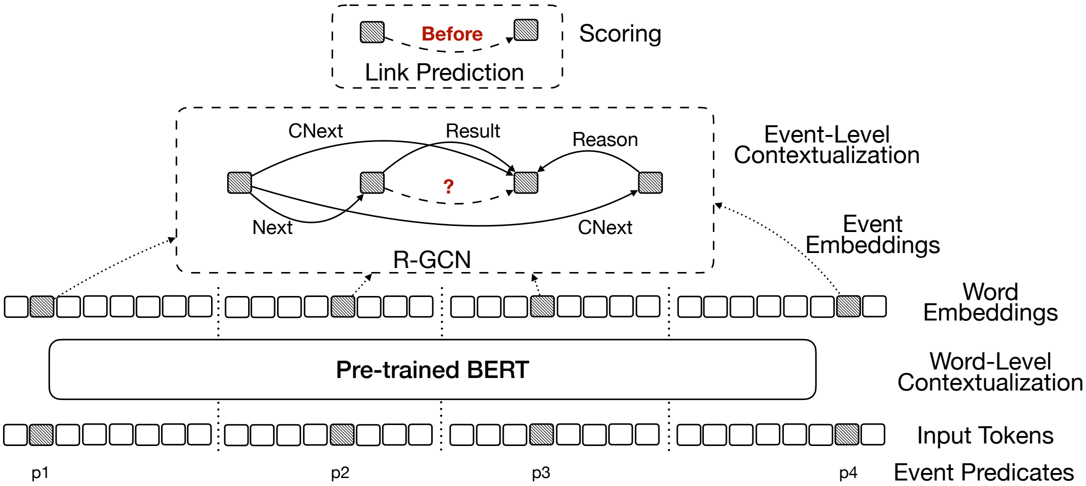

Figure 2: Neural architecture for the Narrative Graph model.

Objective There are two common objectives researchers have been using for optimizing graphical networks: node classifications and link predictions (Schlichtkrull et al., 2018). We select the latter one, as our goal is to capture structural transitions between events. However, it is possible to train for both objectives jointly within our framework, and we leave it for future work.

We score a target link (triplet) with a modified version of DistMult (Chang et al., 2014), an effective scoring function designed for knowledge base completion. The function is defined as follows:

$$f(h, r, t) = e_h^T W_r e_t, \tag{3}$$

where e_h and e_t are the representations for head and tail events of the triplet, and $W_r \in R^{d \times d}$ are relation-specific parameters. The original DistMult restricts W_r to a diagonal matrix to account for the huge amount of relation types existing in knowledge bases. We relax this as we need to address more fine-grained differences between relations, such as directionality[3].

The final loss function is the Cross-Entropy Loss with weighted classes:

$$\begin{aligned}\mathscr{L} = -\frac{1}{|\mathscr{T}|} \sum_{(h,r,t,y) \in \mathscr{T}} & y \log(\sigma(w_r f(h, r, t))) \\ & + (1 - y) \log(1 - \sigma(w_r f(h, r, t))),\end{aligned} \tag{4}$$

where $\mathscr{T}$ is the set of sampled triplets with labels; $\sigma(.)$ is the logistic sigmoid function; w_r is the class weight depending on relation type distributions; and y is the binary label.

4 Evaluations

Our evaluation consists of two parts. The first part conducts intrinsic evaluation, evaluating the basic characteristics of the NG model. In the second part extrinsic evaluation is performed, by using the NG event embedding for a downstream task–Implicit Discourse Relation Sense Classification (Xue et al., 2016), from CoNLL 2016. The source code and models used in this paper are publicly available[4].

4.1 Data and Experiment Settings

For pretraining and intrinsic evaluations, we use the NYT section of English Gigaword (Parker et al., 2011), which contains about 2M newswire documents. We filter out extremely short and long documents by limiting the number of graph nodes between 20 and 350 ($s_{min} = 20$ and $s_{max} = 350$). This leaves us 1.42M documents, and about 345M relations are extracted (see Table 1). The data splits follow (Granroth-Wilding and Clark, 2016)'s setting, dividing the documents into train/validation/test sets. Other hyperparameters are listed as follows: the number of R-GCN layers $l_{rgcn} = 2$, the number of truncated graphs $n_{neg\_g} = 4$, the ratio of edges to be removed $r_{neg\_e} = 0.05$, the number of negative edges per removed edge $n_{neg\_e} = 20$, the hidden layer size $d = 128$, the class weights in the loss function are inversely proportional to the class distribution given in Table 1.

For training the model, we use AdamW optimizer (Loshchilov and Hutter, 2017) with initial learning rate 0.0002. No warm-up steps are

[3] We have also tried other scoring functions, such as TransE families (Bordes et al., 2013), but DistMult outperforms them.

[4] https://github.com/doug919/narrative_graph_emnlp2020

used. The BERT encoder is initialized with BERT-Tiny (Turc et al., 2019), a distilled compact version of BERT to accommodate the large graph structure, and fine-tuned during training. We experiment with dropout rates $\{0, 0.1, 0.2, 0.4\}$ and use the model that achieves the best result in the validation set. The number of model parameters is 4812168. We search the hyperparameter for about 30 trials using a month, and use F1-macro score over Triplet Classification task (Table 5) for selecting the model. The expected validation performance is 58.89% F1-macro score. The final model is trained on four NVIDIA 1080Ti GPUs for 5 days.

For extrinsic evaluation, the data is from the CoNLL 2016 shared task, using their data splits (Xue et al., 2016).

4.2 Intrinsic Evaluation

The intrinsic evaluation consists of four tasks. The first task is Multiple-Choice Narrative Cloze (MCNC), proposed by Granroth-Wilding and Clark (2016), which measures the models' ability to recover a missing event given its coreferent event chain. The second evaluates the models' ability to identify the tail event, given the head event and relation, i.e., $(e_h, r, ?)$. The third evaluates the models' ability to detect the correct relations between two given events, i.e., $(e_h, ?, e_t)$. The fourth evaluation is a binary triplet classication, inspired by knowledge base completion, where a test triplet is given and the binary classifier identifies it is true or false.

Baselines Six baseline models are considered.

1. **Random**: makes random predictions.
2. **EventComp-BERT**: is an implementation of EventComp (Granroth-Wilding and Clark, 2016) but replace the event encoder with BERT. It uses a feed-forward neural network to compose a coherence score for event pairs based on coreference chains. It is a single-relational model that only considers **CNext**.
3. **EventLSTM-BERT**: is an attention-based LSTM model that captures event coreference chains. It is also a single-relational model (**CNext**). We follow (Wang et al., 2017)'s architecture and settings but use BERT for encoding events and remove the dynamic memory component.
4. **EventTransE-BERT**: is an implementation of EventTransE (Lee and Goldwasser, 2019), but replace the event encoder with BERT. It is a strong uncontextualized event embedding model, outperforming various models on the MCNC task. It trains on multi-relational data and a translating-based loss (TransE) is used for scoring event triplets.
5. **Event-BERT-sim**: uses the pre-trained BERT model without fine-tuning and scores event pairs with cosine similarity, which simply measures the embedding similarity between events. The relation type is not taken into account. This baseline gives the idea about how much performance gain can be acquired from word-level contextualization.
6. **Event-BERT-ft**: is fine-tuned (ft) using the same objective and data as the NG. However, the event-level contextualization, i.e., R-GCN layer, is skipped, so it is a pairwise event models powered by BERT. It is a multi-relational model and the loss function is identical to NG.

Multiple-Choice Narrative Cloze We begin with the popular benchmark–MCNC, which predicts the next event, given its preceding events. It was originally proposed by Chambers and Jurafsky (2008) as a ranking problem, which ranks all possible events given an event chain. However, the ranking metric over a huge set of event vocabularies is not easy to interpret for model comparisons. Granroth-Wilding and Clark (2016) thus adapted it to a multiple-choice setup, rendering a clear performance metric. (Lee and Goldwasser, 2019) further generalized it to the multi-relational setting. In this task, we follow (Granroth-Wilding and Clark, 2016)'s set-up. Each question has an input sequence of 8 events that are connected with **CNext**, and the target event has 4 negative and 1 positive choices. Since the question set released by previous works does not contain document information required by our NG model, we re-sample the question set with document information. 10000 test instances are sampled from the test split.

Table 2 lists the result. The first row shows the random baseline for 5 choices. The following three rows are single-relational models that only consider event co-occurrence with **CNext** relation. Event-BERT-sim uses event similarity without fine-tuning, which gives the basic performance. EventComp-BERT fits to the event pairs with **CNext** relation and perform better. The sequential model EventLSTM-BERT preforms very well, since this task set-up is perfect for sequential

Methods	Type	Validation	Test
Random	-	20.00	20.00
Event-BERT-sim	*S*	40.18	41.24
EventComp-BERT	*S*	54.12	53.86
EventLSTM-BERT	*S*	62.78	62.62
Event-BERT-ft	*M*	47.22	47.20
EventTransE-BERT	*M*	57.92	58.35
NG	*M*	**65.86**	**63.59**

Table 2: Accuracy scores (%) for MCNC. The task asks models to predict the target event, out of 5 choices, given a sequence of events with **CNext** relation. The model type *S* means single-relational models and *M* means multi-relational models.

Methods	Accuracy
Random	10.00
Event-BERT-sim	32.43
EventComp-BERT	55.16
Event-BERT-ft	50.10
EventTransE-BERT	58.16
NG	**60.94**

Table 3: Accuracy scores (%) over 10-choice MC-questions for CNext relation. Each question has the form $(e_h, r, ?)$, where the head event e_h and relation r are given and the model predicts the correct tail event.

models like LSTM. However, EventLSTM-BERT does not have the ability to digest multi-relational data. The rest three models are multi-relational models. NG outperforms EventLSTM-BERT significantly, since it encodes the narrative graph structure and other relation types. If we compare NG with Event-BERT-ft (NG without R-GCN), we can see that the graph structure improves the result with a large margin (18.64% absolute accuracy improvement in the test set), making NG the best performer over all the single- and multi-relational models.

Predict Coreferent Next Event In this task, we predict the tail event of a **CNext** relation. Unlike MCNC, where a sequence of coreferent events are given, we only take one event as the input and predict the other. We also adopt the multiple-choice setting, and to strengthens the evaluation, we increase the number of candidates to 10 to make the task more challenging. 5000 test instances are randomly sampled from the test split.

Table 3 shows the task result. We can see that even under this more challenging setting, NG can still outperforms all the models. Event-BERT-ft can be interpreted as "NG without R-GCN". We can see that without the event-level contextualization, the performance drops significantly (-10.84% absolute accuracy). The result denotes that as the high-level structure over events are contextualized in the embeddings, the NG model can make better predictions for events in various scenarios. The EventTransE-BERT is a strong competitor here, as it also benefits from multi-relational modeling, but, again, without the event-level contextualization, it performs worse than NG. This again attests the importance of encoding the narrative graph structure. Note that the EventLSTM-BERT cannot be applied here, as it requires a fixed length input.

Methods	Acc.	F1	MRR	Recall@3
Random	16.67	-	-	-
EventTransE-B	44.65	29.33	64.59	81.05
Event-BERT-ft	59.24	55.42	75.27	91.26
NG	**80.27**	**79.68**	**88.05**	**95.74**

Table 4: Predicting the discourse sense, out of 6 candidates, between two given events, i.e., $(e_h, ?, e_t)$.

Methods	Precision	Recall	F1
Event-BERT-sim	3.45	74.04	6.59
EventTransE-BERT	30.17	53.61	38.62
Event-BERT-ft	49.19	36.79	42.09
NG	**68.20**	**66.21**	**67.19**

Table 5: Binary classification for a given triplet (e_h, r, e_t). The scores are macro-averaged over the **minority** class. The validation performance is 56.91%.

Predict Discourse Sense In this task, the models predict the discourse sense for a given pair of events. It is a multi-class classification problem over 6 discourse senses used in this paper.

Table 4 shows the result with four different metrics, including accuracy, F1-macro score, Mean Reciprocal Rank (MRR), and Recall@3. The later two metrics evaluate models' ranking ability. We compare three multi-relational models in this task. The NG outperforms EventTransE-BERT, which means that the DistMulti objective for the other two models is more sensitive to relation types than TransE. The NG also outperforms Event-BERT-ft. Both models achieve high Recall@3, which means that over 90% of correct relations are ranked on the top half. The main difference between the two models is the event-level contextualization (R-GCN component), which brings in 21.03% absolute accuracy improvement. The NG model can both rank and select the answer with the most confidence.

Triplet Classification Table 5 shows the result for triplet classifications, where a triplet is given

	NG			
	Precision	**Recall**	**F1**	**#pos. / #neg.**
Next	59.53	89.86	71.62	149k / 2343k
CNext	46.80	40.38	43.36	59k / 1027k
Before	62.45	69.34	65.72	4.9k / 223k
After	82.18	58.31	68.22	4.5k / 217k
Sync.	73.35	59.08	65.45	1.9k / 179k
Contrast	67.97	76.15	71.83	6.2k / 244k
Reason	73.24	76.87	75.01	2.9k / 192k
Result	80.08	59.72	68.42	0.6k / 159k
macro-avg	68.20	66.21	67.19	-

Table 6: Triplet classification breakdown for *NG*. The scores are macro-averaged over the **minority** class.

and the task is to predict it is true or false. There are 229k positive and 4584k negative triplets, sampled with the smoothed class distribution described in the Negative Sampling section. Event-BERT-sim does not consider relation types, so it only measures event similarity based on BERT embeddings. The low precision and high recall shows that most events are similar if we do not take the relation type into account. Again, the big performance gain of NG over Event-BERT-ft is due to the NG-contextualized event embeddings, which offers high-level summary of documents.

Table 6 shows triplet classification results by type. Given the low positive-to-negative examples ratio, we report the F1 score over the minority class, and the macro-average over all these scores. We note the difficulty of predicting **CNext**, although it has the second highest number of examples. We attribute this to the noise generated by the coref resolution solver, as other relations have clearer signals for learning, and the fact that **CNext** is the only relation that connects two events that could be far away from each other in text. We leave this issue for future work.

4.3 Extrinsic Evaluation

The last evaluation is over a downstream task, Implicit Discourse Sense Classification, a subtask from CoNLL 2016 shared task (Xue et al., 2016). The task is a multi-class classification task with 15 discourse classes, including explicit and implicit relations. The explicit relations mean that the discourse connective, such as "because", exists in the text, providing clues for the sense prediction, while the implicit one does not have it. We only evaluate on the subtask for implicit relations as it is both challenging and useful for language understanding.

Methods	**Test**	**Blind**
PurdueNLP (Pacheco et al., 2016)	34.45	29.10
ecnucs (Wang and Lan, 2016)	40.91	34.18
ttr (Rutherford and Xue, 2016)	36.13	37.67
ELMo (Peters et al., 2018)	37.65	36.72
EvTransE* (Lee and Goldwasser, 2019)	39.05	38.35
NG*	**42.84**	**43.91**

Table 7: F1-micro scores for Implicit Discourse Sense Classifications. EvTransE is abbreviated for EventTransE. The start signs mean that its event representation is concatenated with *ELMo* word embeddings. The validation performance for NG is 46% F1 score.

Several baseline models are chosen from the leader board of the shared task, including the best and median systems (the first three rows). Following (Lee and Goldwasser, 2019)'s experiment setting, ELMo (Peters et al., 2018) is used as the basic features for the supervised classification. The input features feed to a self-attention layer and then two fully-connected hidden layers, with dimensions 256 and 128, are added on top for classifications. The EventTransE baseline concatenates its event representation with ELMo and feeds to the same network architecture. The NG model applies its event embeddings in the same way as EventTransE, and achieves the best performance.

5 Conclusions

We propose the Narrative Graph embedding model to learn contextualized event representations for disambiguating discourse relations. We use weak supervision, provided by the predictions of off-the-shelf NLP tools and a rule-based discourse annotator, to learn event representations capturing world knowledge useful for downstream tasks. Our model considers multiple discourse relations types, such as "contrast" or "cause". We evaluate our model on three intrinsic tasks, including triplet classification and event/relation predictions, as well as an extrinsic task–discourse relation classification. Our results show that the model can outperform competitive systems. In the future we intend to apply our model to discourse analysis tasks which require modeling long-range dependencies.

Acknowledgments

We thank the reviewers for their efforts and insights. This work was partially funded by the NSF and DARPA ASED program under contracts CNS-1814105 and 13000686.

References

Or Biran and Kathleen McKeown. 2013. Aggregated word pair features for implicit discourse relation disambiguation.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

Nathanael Chambers and Daniel Jurafsky. 2008. Unsupervised learning of narrative event chains. In *ACL*, volume 94305, pages 789–797.

Kai-Wei Chang, Wen-tau Yih, Bishan Yang, and Christopher Meek. 2014. Typed tensor decomposition of knowledge bases for relation extraction. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1568–1579.

Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. 2017. Language modeling with gated convolutional networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 933–941. JMLR. org.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dan Goldwasser and Xiao Zhang. 2016. Understanding satirical articles using common-sense. *Transactions of the Association for Computational Linguistics*, 4:537–549.

Mark Granroth-Wilding and Stephen Clark. 2016. What happens next? event prediction using a compositional neural network model. In *AAAI*, pages 2727–2733.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in neural information processing systems*, pages 1024–1034.

Rujun Han, Qiang Ning, and Nanyun Peng. 2019. Joint event and temporal relation extraction with shared representations and structured prediction. *arXiv preprint arXiv:1909.05360*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

I-Ta Lee and Dan Goldwasser. 2018. Feel: Featured event embedding learning. *AAAI*, pages 4840–4847.

I-Ta Lee and Dan Goldwasser. 2019. Multi-relational script learning for discourse relations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4214–4226.

Chang Li and Dan Goldwasser. 2019. Encoding social information with graph convolutional networks forpolitical perspective detection in news media. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2594–2604.

Zhongyang Li, Xiao Ding, and Ting Liu. 2018. Constructing narrative event evolutionary graph for script event prediction. *arXiv preprint arXiv:1805.05081*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Eric Malmi, Daniele Pighin, Sebastian Krause, and Mikhail Kozhevnikov. 2017. Automatic prediction of discourse connectives. *CoRR*, abs/1702.00992.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. *arXiv preprint arXiv:1703.04826*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Allen Nie, Erin Bennett, and Noah Goodman. 2019. Dissent: Learning sentence representations from explicit discourse relations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4497–4510.

Maria Leonor Pacheco, I-Ta Lee, Xiao Zhang, Abdullah Khan Zehady, Pranjal Daga, Di Jin, Ayush Parolia, and Dan Goldwasser. 2016. Adapting event embedding for implicit discourse relation recognition. *Proceedings of the CoNLL-16 shared task*, pages 136–142.

Joonsuk Park and Claire Cardie. 2012. Improving implicit discourse relation recognition through feature set optimization. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 108–112. Association for Computational Linguistics.

Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. English gigaword fifth edition ldc2011t07. dvd. *Philadelphia: Linguistic Data Consortium*.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Karl Pichotta and Raymond Mooney. 2014. Statistical script learning with multi-argument events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 220–229.

Karl Pichotta and Raymond J Mooney. 2016a. Learning statistical scripts with lstm recurrent neural networks. In *AAAI*.

Karl Pichotta and Raymond J Mooney. 2016b. Using sentence-level lstm language models for script inference. *arXiv preprint arXiv:1604.02993*.

Rashmi Prasad, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, Aravind Joshi, Livio Robaldo, and Bonnie L Webber. 2007. The penn discourse treebank 2.0 annotation manual.

Attapol Rutherford and Nianwen Xue. 2016. Robust non-explicit neural discourse parser in english and chinese. *Proceedings of the CoNLL-16 shared task*, pages 55–59.

Roger C Schank and Robert P Abelson. 1977. Scripts, plans, goals and understanding: An inquiry into human knowledge structures.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607. Springer.

Damien Sileo, Tim Van-De-Cruys, Camille Pradel, and Philippe Muller. 2019. Mining discourse markers for unsupervised sentence representation learning. *arXiv preprint arXiv:1903.11850*.

Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: On the importance of pre-training compact models.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 57:45.

Jianxiang Wang and Man Lan. 2016. Two end-to-end shallow discourse parsers for english and chinese in conll-2016 shared task. *Proceedings of the CoNLL-16 shared task*, pages 33–40.

Zhongqing Wang, Yue Zhang, and Ching-Yun Chang. 2017. Integrating order information and event relation for script event prediction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 57–67.

Noah Weber, Niranjan Balasubramanian, and Nathanael Chambers. 2017. Event representations with tensor-based compositions. *arXiv preprint arXiv:1711.07611*.

Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Attapol Rutherford, Bonnie Webber, Chuan Wang, and Hongmin Wang. 2016. Conll 2016 shared task on multilingual shallow discourse parsing. *CoNLL-16 shared task*.

Zhi-Min Zhou, Yu Xu, Zheng-Yu Niu, Man Lan, Jian Su, and Chew Lim Tan. 2010. Predicting discourse connectives for implicit discourse relation recognition. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1507–1514. Association for Computational Linguistics.