Narrative Embedding: Re-Contextualization through Attention

Sean Wilner
Vody LLC
sean@vody.com

Daniel Woolridge
Vody LLC
dan@vody.com

Madeleine Glick
Vody LLC
madeleine@vody.com

Abstract

Narrative analysis is becoming increasingly important for a number of linguistic tasks including summarization, knowledge extraction, and question answering. We present a novel approach for narrative event representation using attention to re-contextualize events across the whole story. Comparing to previous analysis we find an unexpected attachment of event semantics to predicate tokens within a popular transformer model. We test the utility of our approach on narrative completion prediction, achieving state of the art performance on Multiple Choice Narrative Cloze and scoring competitively on the Story Cloze Task.

1 Introduction

Common sequences of events describing real-world interactions are a fount of useful information for how we interact with our world. For example, the prototypical “restaurant schema" of Schank and Abelson (2013) might contain the sequence of events in Story A of Figure 1. Knowledge of this plausible sequence of events allows us to both predict likely subsequent events in a novel ‘restaurant experience’ as well as infer unstated events when observing a recounting of the same. Since written stories often provide the minimum required for human comprehension, the stated narrative is liable to be rife with omission, leaving the reader to infer the missing events. For this reason, understanding these chains of events is central to understanding the narrative as a whole.

Each story is made up of constituent events, and by developing representations of events we can in turn produce a representation of the story at large. Event representation has been a topic of substantial interest to the field (Chambers and Jurafsky, 2008, 2009; Granroth-Wilding and Clark, 2016; Wang et al., 2017; Mostafazadeh et al., 2016b) and as the wealth of unstructured narrative data continues to grow, to industry as well. Indeed, industry has seen narrative representations applied both directly in the case of film and book analysis (Scriptbook, 2021) as well as indirectly as a means of communication in automated data explanation applications (Narrative Science, 2021).

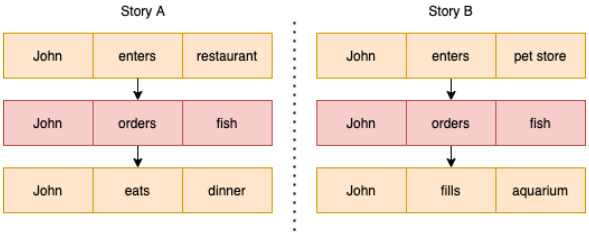

Figure 1: A selection of events from two story lines. Each narrative contains the event "John orders fish", however the meaning of the event within the context of each story is substantially different.

Our goal is to provide an unsupervised approach to narrative representation by utilizing and combining advances in representational techniques at both the semantic and structural level. Our approach is based on ‘re-contextualization’, a process of embedding representations through transformer blocks at two levels of granularity. First, similarly to standard transformer models, we contextualize events within a single sentence to provide local context. Second, we extract events from each sentence to contextualize them across an entire story or story arc. The motivation for this approach is to allow our representation to differentiate between events which, though potentially drawn from identical or nearly-identical sentences, represent substantially different narrative components (e.g. "John orders fish" from Story A versus Story B in Figure 1). In the process, we produce two models that demonstrate competitive performance on the Multiple Choice Narrative Cloze (MCNC) task and the Story Cloze task (SCT) respectively.

Specifically, our contributions herein can be summarized by the following:

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 1393–1405

- A novel approach to re-contextualization and analysis of the improvement it provides to narrative tasks
- A re-contextualization-based model for narrative completion that displays state of the art performance on MCNC and competitive performance on SCT
- An exploration of BERT transformer semantic attachment for event information

2 Related Work

2.1 Narrative Evaluation Frameworks

Our work is largely influenced by Chambers and Jurafsky (2008, 2009) and Granroth-Wilding and Clark (2016) who began and extended respectively a technique for automated unsupervised narrative extraction and decomposition. Their fundamental claim is that narratives can be decomposed into events which in turn are comprised of a predicate and its relevant roles (e.g. subject, direct object, etc.). Using chains of these events, models can be constructed to explicitly (as in the case of Chambers and Jurafsky (2009)) or implicitly (as in Granroth-Wilding and Clark (2016)) encode something analogous to a *script* (Schank and Abelson, 2013). Scripts can be thought of as collections of events which would be expected to occur in the same story.

Cloze or modified Cloze tasks are often used to analyze narrative representation. Originally, this niche was filled by the Narrative Cloze Task (NCT) (Chambers and Jurafsky, 2008, 2009) which left a missing event as a blank to be filled in from all possible events. However, NCT proved to be too under-constrained and frequency based language models routinely outperformed more complicated and knowledge-rich narrative techniques (Chambers, 2017). To address this, several other evaluations have been put forth such as Multiple Choice Narrative Cloze (MCNC) (Granroth-Wilding and Clark, 2016) where the objective is to select between a set of five candidate replacement events, one of which is the correct original event rather than the unbounded set of potential completions used before. Even more constrained is the Story Cloze Task (SCT) (Mostafazadeh et al., 2016b, 2017) which presents the first four sentences of a five sentence story and two candidate sentences for story completion. All together, these evaluation frameworks will allow us to train, tune, and evaluate the combined effectiveness of the techniques employed by our model.

The application of Cloze tasks to higher-level domains such as narrative have been criticized as being innately tied to language modeling and thus failing to accurately measure script knowledge (Rudinger et al., 2015). Nonetheless, keeping with common practice we use Cloze tasks here for their practicality and ease of measurement.

2.2 Computational Approaches

Recurrent models and Transformers (Vaswani et al., 2017) have shown state of the art performance for narrative tasks. Lv et al. (2019) use a Long Short-Term Memory (LSTM) model to encode contextually represented events where the breadth of that context is controlled through a self-attention mechanism. Additionally, Li et al. (2019) investigated several pre-trained transformers and pre-training regimes, achieving 90.3% accuracy on the newest SCT dataset, a considerable improvement over the previous state of the art at 64.4%. Similarly, attention has shown promise on MCNC as well (Wang et al., 2017; Lv et al., 2019). Wang et al. (2017) used a Deep Memory Network (Weston et al., 2015; Mikolov et al., 2014) consisting of an RNN with attention over event composition embeddings in the style of Granroth-Wilding and Clark (2016), where word embddings of event components consisting of the predicate, subject, direct object, and head noun of the prepositional phrase are passed through a feed-forward neural network. This Deep Memory Network encodes context-sensitive temporal information into events. The novel architecture we propose is similar in purpose though different in structure. We use transformers and transformer-attention-blocks instead of Recurrent Neural Networks (RNNs) to manage re-contextualization.

Our use of subsequent attentional mechanisms in an effort to “re-contextualize" events in the scope of their narratives as a whole is a continuation of ideas put forward by Wang et al. (2019) and Hu et al. (2017). Wang et al. (2019) propose a hierarchical encoder/decoder stack to generate story completions for SCT by treating it essentially as a translation task where the first four sentences are ‘translated’ into a fifth completion. The hierarchical encoder they present generates a contextualized representation at each of the sentences levels using an LSTM. Those representations are then re-represented with another layer of LSTM over the sentence level embeddings. Hu et al. (2017)

take a similar tack by treating larger-scope events (e.g. going to a restaurant as in Story A of Figure 1) as singular units comprised of collections of "subevents" (e.g. ordering food as in Figure 1). They use a series of 3-stacked LSTMs to embed each "subevent", and then again to determine the embedding of the larger-scope event, and lastly to predict the subsequent "subevent" to come next.

Outside of LSTM models, re-contextualization has been explored with adjacency graph updates where an event begins with an embedding produced as in Granroth-Wilding and Clark (2016) but is adjusted based on the other events in a story using the connected sub-graph of an event co-occurrence graph (Li et al., 2018).[1]

While transformers (Vaswani et al., 2017) have shown time and again how powerful they are at capturing many aspects of higher-order textual information (Devlin et al., 2019), to the best of these authors' knowledge there is no report on the use of transformers for explicit narrative event representation. Similarly, we are not aware of prior publications looking at the distribution of event semantics within transformer embeddings. In order to address these questions, we explore the interplay between transformers and narrative events across several models and tasks.

3 Datasets

For our experiments, we make use of two datasets: English Gigaword Corpus for Multiple Choice Narrative Cloze Task and the Story Cloze Task Corpus for the Story Cloze task (Mostafazadeh et al., 2016a; Sharma et al., 2018).

The English Gigaword Corpus consists of New York Times news articles containing a training set of 830,643 documents. This dataset was then parsed with spaCy (Honnibal and Montani, 2017) and embedded with the 'bert-base-uncased' pre-trained model (Devlin et al., 2019). Train/test splitting was done the same as in Granroth-Wilding and Clark (2016).

The Story Cloze Task (SCT), a premiere benchmark narrative task featured as the shared task of LSDSem 2017 (Mostafazadeh et al., 2017) consists of 49255 five sentence stories and 3744 SCT examples which have two candidate completions. SCT is an especially good dataset for evaluation of our approach since one of the goals in its construction was to make the story completions as close to statistically indistinguishable from one another as possible. This then requires a more complete narrative understanding to differentiate between alternative endings. As a result, SCT places a stronger emphasis on event disambiguation and cohesive interpretation than on joint probabilities of events.

More specifically, we use the SCT_1.5 dataset provided by Sharma et al. (2018) which addresses inherent story-completion biases found in the original SCT dataset. The improvement afforded by SCT_1.5 is especially important for transformer based models since it is impractical to know what elements of the original sentence bled into the semantics of any given token. For instance, the sentiment of the sentence as a whole may adjust the semantic representation of an otherwise semantically neutral verb in the sentence, allowing our classifier to pick up on sentiment shift through sentences (a feature for which SCT_1.0 was strongly biased).

The training and test sets of SCT_1.0 and SCT_1.5 have numerous instances of overlap with SCT_1.5 sharing 1556 of its 1571 stories with SCT_1.0. The test set overlap is less ubiquitous, with partial and complete overlaps including instances of word adjustment (e.g. "Forever 21" → "Forever Twenty-One") of 913 out of 1872 test stories. When removing all of the overlapping instances, we find an additional 959 instances from SCT_1.0 testing which we can add to SCT_1.5 training, alongiside the validation set of SCT_1.5. For the purposes of comparison, comparative results reported here will be on a withheld portion of the SCT_1.5 validation set since the accompanying test set is blind, and total submissions to determine accuracy are limited. Final best models' results are also submitted to the public leader-boards for a ground-truth blind assessment.

For both MCNC and Story Cloze, in keeping with common practice on the tasks, we report a raw accuracy metric.[2]

4 Methods

We explore narrative representation through its application to two tasks using three models: Two models for Multiple Choice Narrative Cloze (MCNC) – Transformer Event Composition (Event-

[1] N.B. the authors note that the resulting hidden states calculated during the adjacency graph updating phase reflect a similar computation to a Gated Recurrent Unit (GRU) and this distinction from LSTM models may be more academic than of practical import.

[2] See Table 4 of Mostafazadeh et al. 2017

Comp) & Transformer Event Sequence with Attention (Re-Context) – and one for the Story Cloze Task. These three models aim to answer the following 3 questions:

1. Transformer Event Composition: What impact does BERT have on standard event composition models?
2. Transformer Event Sequence with Attention: What impact does re-contextualization have on BERT embeddings for event prediction?
3. Story Cloze: Do our results generalize to more controlled and higher-level narrative completion tasks such as SCT?

We present and analyze two models on MCNC. Model 1 provides a comparison against the original event-comp model from Granroth-Wilding and Clark (2016) and allows determination of the direct impact of transformer (here BERT) embeddings on MCNC. The second model aims to capture narrative context via a novel architecture using an attention layer over a larger-scope window of narrative events. The idea behind this is to take locally (intrasentential) contextualized event embeddings and re-contextualize them within the scope of the greater narrative (intersentential). This mechanism is what allows us to tackle the type of event confusion displayed in Figure 1. The Story Cloze Task model, Model 3, follows the same representational architecture as our second MCNC model.

4.1 Transformer Event Composition Methods

The input to our approach is constructed by parsing stories and extracting syntactic event structure. This is accomplished with the aid of the spaCy dependency parser (Honnibal and Montani, 2017). Events are constructed from the dependency structure by capturing each predicate and its syntactic attachments (namely nominal subject, direct object, and prepositional relation). We ignore copular verbs, as in Granroth-Wilding and Clark (2016), since they are unlikely to represent an event. To create a fixed length input, the embedding vector for any syntactic relation not found for a given predicate is zeroed out. The event input is then a four-tuple of $(verb,\ nsubj,\ dobj,\ prep)$.

Our composition model uses the same architecture as described in Granroth-Wilding and Clark (2016) where events are given by the 4-tuple of verb, nominative subject, direct object, and preposition, but replaces the Word2Vec (Mikolov et al., 2013) embeddings they used with BERT embeddings.[3] For example, given the event $baked(John, cake)$, we would feed "John baked a cake." into BERT and capture the semantics of 'baked', 'cake', and 'John'. Our model works over pairs of events with a target label of whether they belong to a common narrative. This solves the MCNC task by selecting labels to maximize the average score given by the model over all pairs involving the multiple choice candidates.

	Full Event	Just Verbs
Random	20.0	20.0
P&M(2016)	43.17	—
C&J(2008)	30.92	—
G-W&C(2016)	49.57	24.57
W et al.(2017)	46.67	—
Y&H(2018)	48.84	—
EventComp(ours)	92.13	92.22
Re-Context(ours)	—	86.92

Table 1: Multiple Choice Narrative Cloze percent accuracy scores for our three models with baselines.

Since verbs have been shown to have a substantial signal in story detection, segmentation, and completion (Eisenberg and Finlayson, 2017; Granroth-Wilding and Clark, 2016), we also explore the relative performance of reducing our event representations to only the BERT output vector for the verbs.[4]

4.2 Transformer Event Composition Results

As we can see in Table 1, plain Word2Vec embeddings (G-W&C) outperform a random baseline, but are rather lackluster when compared to BERT. Worth noting is how much worse "Just Verbs" performed in Granroth-Wilding and Clark (2016). At face value, this is an unsurprising result. The actors involved (as communicated by the nominative subject and direct object) and the surrounding information such as location, time, or manner (as determined by the prepositional relations) are intrinsic parts of the event. On the other hand, the fact that the "Verbs Only" case for BERT embeddings

[3] Since BERT does not embed words but rather tokenized word-pieces (which are often but not always full words) embeddings are taken as the average over all tokens comprising the lexical entries for each event component.

[4] N.B. this does not mean only verbs play a part in the semantics since transformers contextualize their representations.

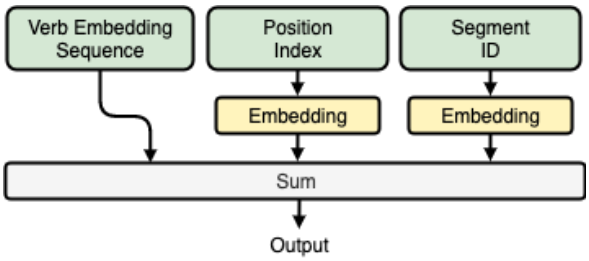

Figure 2: The input structure of Re-Context that combines a learned sequence and segment embeddings with the verb embedding additively.

performs as well as the full event composition[5] is noteworthy given the past history of comparatively poor performance of verb only event representations in the literature (Granroth-Wilding and Clark, 2016).

The relative improvement of "Verbs Only" suggests that predicate embeddings encode a surprisingly substantial amount of the relevant event information. More work is needed to more fully explore what other information, if any, transformers syntactically segregate. Given the compactness and comparable strength of predicate embeddings as event proxies, we use them in lieu of longer-form event representations in the following experiments.

4.3 Event Sequence with Attention Methods

To directly address the event-confusion problem presented in Figure 1, we propose a model, Re-Context, to develop event representations that are both locally and globally context-sensitive. By using BERT embeddings, each token is given a locally (within sentence) contextualized representation. To understand how each event interacts within the larger scope of the narrative we implement a multi-headed attention layer that reads in event embeddings and adds in positional information, ultimately re-contextualizing them with another transformer block. As mentioned above, events here are represented by the pre-trained BERT embeddings of their corresponding predicates. As a whole, the Re-Context model consists of four components: Input Embedding, Attention Blocks, Masking, and Binary Classification.

The Input Embedding follows the same structure as BERT's input embedding where the input is ultimately the sum of the event (verb) BERT embedding with a learned segment embedding and a learned positional embedding (see Figure 2). The positions here are the textual ordering of the events. We use the same set of training examples and so our input is a collection of 5 sequential events.

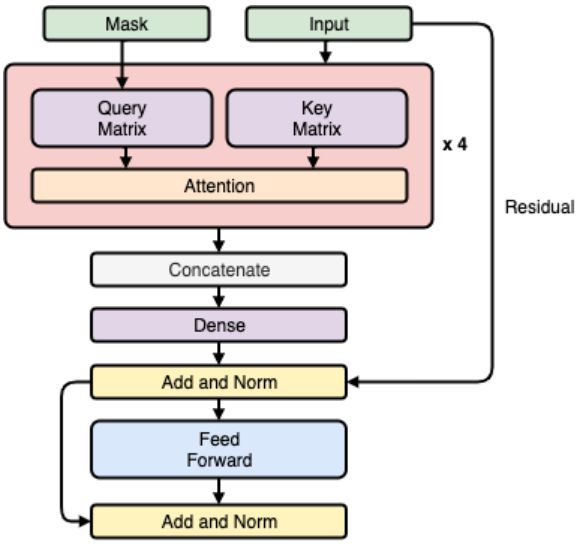

Figure 3: The attentional block component of Re-Context responsible for re-representing each verb based on the context verbs.

The Attention Block is the core of the Re-Context model (see Figure 3). It begins with a multi-headed attention layer with 4 heads to help the model contextualize each event with respect to the greater narrative. These contextualized event representations are then summed and normalized with the residual of the input layer. The normalized embeddings are then fed through a two layer feed forward network that up-scales them by a factor of four then down-scales them to their original dimension. Finally the embeddings are once more summed and normalized with the residual of the previous normalized layer.[6]

Masking here is a pre-training task to help the model discover more optimal weights for the input and attention block components. To do this we zero out the embedding of one of the events before passing it to the Input Embedding component. We simultaneously pass it a segment ID corresponding to '[MASK]'. The masked event is then value-masked from the multi-headed attention layer so that no other event can attend to it. The goal of the masking task is to attempt to reconstruct the original event embedding as the output embedding

[5]When breaking down the WSJ Corpus dataset into years, the Kolmogorov-Smirnov Test over the sets of scores from the two models yields a statistic of 0.21 with a p-value of 0.80, nowhere near strong enough to suggest differing distributions.

[6]For all network specifics including layer dimensionality, activation functions, and regularization hyperparameters, see Appendix A.

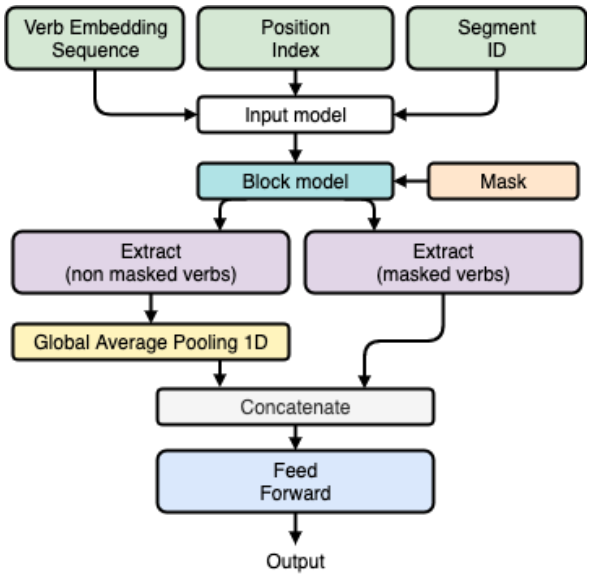

Figure 4: The complete binary classifier based on Re-Context used for solving Multiple Choice Narrative Cloze.

for the masked event.

Binary Classification (see Figure 4) is the comprehensive model that labels event pairs for MCNC. Ultimately, all the other components are made in service of this final classifier. The Binary Classifier takes in the Masking pre-trained Input and Attention Blocks, applying them to ordered event predicates sequentially. The classifier itself is built off of the Attention Block's output which it separates into two categories: context & candidate. The context is averaged and concatenated with the candidate before being fed into two feed-forward densely-connected layers. The output of the final layer is the binary classification of whether the candidate event belongs to the narrative.

Similar to previous models, Re-Context solves the MCNC task by identifying which candidate event has the highest positive label assigned by the Binary Classifier.

4.4 Event Sequence with Attention Results

Continuing the results analysis from above, in Table 1 we see that the attentional model, Re-Context, under-performs the simpler compositional model, EventComp. However, all the models we present represent a substantial improvement over previous state of the art.

Since BERT embeddings have access to the full scope of each sentence to which they belong, and some sentences have multiple events in them, we cannot be sure that there was no information bleed between the withheld events and those making up the feature set. That is, the event we want to replace in the Cloze task could be 'secretly' encoded into the semantics of the words making up at least some of the feature set used. If this were the cause for our high accuracy, we would expect to see a significantly higher score for the "Full Event" case as opposed to "Just Verb" since there would be more opportunities for signal transfer that would better imply the correct missing event. We see no such pattern. To control for this in future studies, masking could be employed during semantic embedding generation to avoid any possible pollution of the feature set, although this may yield significantly worse embeddings depending upon the scope of the masking. Fortunately, SCT has no such pollution concerns since target events are in withheld sentences, and thus we explore our model's performance upon it.

4.5 Story Cloze Model

To generate story completions for the Story Cloze Task, we use a similar architecture to the above Event Sequence with Attention model, Re-Context. The Binary Classification therein describes our task quite well in that we have two potential endings and must select the more optimal one.

One point of difference lies in the fact that we have a fixed sentence window rather than a fixed event window. That is, a single sentence may contain multiple events. To reduce this task to the same five-event learning style as above, we explored several approaches:

1. Average verb embedding
2. Aggregate selection
3. Sentence embedding

Approach 1. represents each sentence by the average of all its verbs. This has the advantage of being a consistent and comprehensive selection. The obvious downside is that many such verbs are often more structural than salient to the narrative (Swanson et al., 2014) as in the example "The truth is Bob ate the cake." where the root verb 'is' acts as a structural connector to the important event of $ate(Bob,\ cake)$. Consequently, more salient events can easily be watered down by less relevant ones. A less obvious downside to this approach is that it fails to directly represent events since the embeddings are now a collection of every event within a sentence.

Approach 2. employs event combinations. In

it, each story corresponds to a set of length five event (verb) representations made up of all possible combinations, one verb taken from each sentence. For example, given the four-sentence story: "Bob baked a cake. He thought the cake had cooled. He ate the cake. He burned his mouth." we would have two four-event groupings, namely 'baked' → 'thought' → 'ate' → 'burned' and 'baked' → 'cooled' → 'ate' → 'burned'. We then compute labels and report accuracy for several different scoring methods: as an average of the scores of all combinations, as the max of those scores, or as the average over the top $\frac{1}{4}$ scores. The only disadvantage to this technique is the commensurate increase in computational cost due to the combinatorial increase in feature set size.

In our third approach, we explore whole sentence embedding using SBERT (Reimers and Gurevych, 2019). This has the advantage of generating five embeddings directly, but shares the same disadvantage in representational precision as our averaging method since it also does not operate at an individual event level, and thus no explicit event representation is possible.

For all uses of BERT for Story Cloze, we report results on two pre-trained models: the base-uncased and large-cased models. For SBERT, we use the pre-trained 'roberta-large-nli-stsb-mean-tokens'[7] due to its high performance on Semantic Textual Similarity benchmarks.

Both in order to understand the relative contribution of the re-contextualization process our model employs and because of the favorable performance of the simpler compositional model on MCNC above, we also run all three approaches without the multi-headed attention network. The specific structure for these tests is a simple feed forward network taking in two events, sentence embeddings, or average of events and outputting a binary decision much in the same way as our EventComp approach to MCNC.

Training epochs, dropout, and batch-size were hand-tuned for the Multiple Choice Narrative Cloze, maximizing accuracy for a randomly generated train/test split. We maintain those same hyper-parameters throughout the study (with the exception of epochs which were reduced for the Aggregate selection method to a value of 3 for Masking and Binary pre-training due to the substantially increased size of dataset, and tuned for the Binary classifier on a random train/test split, but reported on for other train/test splits).[8] Results for SCT are reported as the median of 3 runs over randomly generated train/test splits. To ensure that any cross-contamination presented by the potentially overlapping nature with our tuning set with the testing set is not responsible for our relatively high performance, we test our highest scoring model against the withheld test set for Story Cloze 1.5.[9]

4.6 Story Cloze Results

	BASE-U	LRG-C	RoBERTa
Avg. Verb	81.4	77.2	—
Agg. Avg.	83.7	81.4	—
Agg. Max	83.0	80.1	—
Agg. 1/4	82.1	79.2	—
SBERT	—	—	79.3
Random	50.0	50.0	50.0

Table 2: Story Cloze Task percent accuracy scores for the three approaches' models (BERT-base-uncased, BERT-large-cased, and STS-pretrained RoBERTa) alongside a random baseline. Numbers reported are the median scores from 3 runs tested against a withheld 20% of the SCT_1.5 val dataset.

As we can see in Table 2, the results for the BERT-based models align quite well with our expectation in that the Average Verbs method is consistently below the Aggregate methods. This difference in score is more stark when compared with the Aggregate Average. This comparison is especially pertinent since the only difference between the two methods is if the inputs or outputs of the model are averaged. The relative input contribution of each verb in the Average method is consistent between the two cases, but in the Aggregate method each input with a structural verb quite likely co-occurs with multiple salient verbs. This discrepancy suggests that the wash-out effect of structurally important but semantically vacuous verbs can be overcome so long as there is a significant amount of narrative-salient verbs present in the example. In other words, by averaging over the scores instead of averaging over the verbs themselves, the more relevant verbs can play a larger role in the scoring of the model rather than being watered-down by

[7] https://github.com/UKPLab/sentence-transformers

[8] All hyperparameters are minimally adjusted and their specific values are reported in Appendix A

[9] The use of SCT 1.0 testing set as part of the training set for SCT 1.5 does not pose a problem as the task encourages the use of any (including outside) resources.(Mostafazadeh et al., 2017)

	BASE-U	LRG-C	RoBERTa
Avg Verb	77.9	77.9	—
Agg Avg	78.2	77.6	—
SBERT	—	—	79.9

Table 3: Story Cloze Task percent accuracy without re-contextualization.

other less relevant verbs within the same sentence. These results suggest that the Aggregate Average model is robust to a few non-salient verbs.

Our top score of 83.7% was achieved by the Aggregate Average method. The relative difference between the performance of the Aggregate metrics is slight but consistent between the BERT-base-uncased and BERT-large-cased models. This suggests that the salient events are not one-to-one with sentences and that each sentence may convey several meaningful events, otherwise we would expect to see Aggregate Max outperform the average, or at the very least Aggregate Top 1/4 which acts as a compromise between the two metrics.

Aggregate Average outperforming Aggregate 1/4 may be the result of several factors. In Aggregate 1/4, the top quarter is selected as the highest performing set of verbs. This may be largely driven by a single high-scoring verb should there be several verbs present in a single sentence, and so, many of the top $\frac{1}{4}$ could be all combinations that include that single verb. This would make the model susceptible to any contraindicative event in the narrative. Alternatively, it could be that there are more than $\frac{3}{4}$ 'relevant' verbs and so restricting the set, while removing more semantically vacuous verbs, also removed a significant signal that proved important to discernment.

The most striking comparison is between the verb-based methods and the SBERT method where the full sentence embeddings under-perform even the raw average of the verbs per sentence. This provides compelling evidence that BERT is directly attaching complete event semantics to predicates since SBERT generally outperforms BERT word-embedding averages on sentence-level Natural Language Understanding tasks (Reimers and Gurevych, 2019). It is worth noting that this relative performance difference is much weaker in the BERT-large-cased model suggesting that BERT-base-uncased stores more of the event information on the predicate than does BERT-large-cased.

Comparing Table 3 to Table 2 we can see, for both BERT models, re-contextualization improves the Aggregate Average method's accuracy. Interestingly, these improvements do not extend to the average verbs and sentence embeddings. This suggests that while event re-contextualization is useful for explicit event representation, that utility does not extend as well to our implicit event representations. This is doubly interesting given the poorer performance of re-contextualization on MCNC and suggests that re-contextualization is more relevant to higher order narrative tasks such as SCT. Moreover, we suspect that MCNC performance is more impacted by the ability to capture underlying joint probabilities of events, rather than any understanding of narratives as a whole. Thus the event ambiguity that re-contextualization aims to solve is less impactful than on SCT where word choice is controlled for.

Our approach for SCT performs favorably against the common baselines as shown in Table 4. We believe that Table 2 shows a higher accuracy for our model than Table 4 because the work put into normalizing and balancing STC_1.5 makes it a generally more difficult task. Table 4 reports results from running our method 14 times over the training data using different, randomized training order, ultimately choosing the completions by group consensus (where there was no consensus – less than 40% of the time – we did not report a label since the public leader-boards allows for scores on partial submissions).

In Table 4 the Sentiment, Word2Vec, and EndingReg scores are taken from Sharma et al. (2018) and cogcomp from Chaturvedi et al. (2017). We currently sit competitively on the competition leader-boards.[10] Li et al. (2019), which holds the first place on SCT_1.5, does not directly address event representation and instead reinterpret the task as a Next Sentence Prediction (NSP) task. NSP is one of the two pre-training tasks on which BERT is optimized, making this a natural approach to using BERT for SCT.

One of the driving thrusts of the Narrative Cloze, Multiple Choice Narrative Cloze, and Story Cloze tasks is to find methods to produce narrative scripts or schemas (Chambers and Jurafsky, 2009; Wang et al., 2017; Chambers, 2013; Pichotta and Mooney, 2014). Construction of these predicated abstractive models requires that individual events be distinguishable, and it is non-obvious how to use aggre-

[10] https://competitions.codalab.org/competitions/15333

	Accuracy
Random	50.0
Sentiment	52.5
Word2Vec	59.4
CogComp	60.8
EndingReg	64.4
AggBERT (ours)	79.0
Li et al.	90.3
Human	100

Table 4: Story Cloze Task accuracy benchmarks.

gate methods for such a task.

Finally, the high accuracy reported on SCT_1.5 suggests that our concerns about semantic bleed discussed in Subsection 4.4 are unlikely to be the cause of the high MCNC accuracy as such bleed cannot account for performance on SCT_1.5 given the separation of the target sentences from the input sequence representation.

5 Discussion & Conclusion

We find that BERT embeddings represent a substantial improvement to the MCNC state of the art. Our novel attentional re-contextualization of events provides a competitive Story Cloze Task performance yielding the highest score amongst models using explicit event representation. Of broader interest to the field are our findings that for Cloze tasks BERT attaches enough event semantics to predicates that they can function as full event proxies without loss of performance. This finding suggests possible avenues for further research to discover what other syntactic entities carry broader semantic aggregates in BERT.

Our overall findings and contributions can be summarized as follows:

- Verbs carry event semantics in BERT
- Re-contextualization using attention is helpful when using explicit events on harder tasks
- We achieve state of the art MCNC scores and place competitively on the Story Cloze Task

Acknowledgments

This research and document have been helped immeasurably by input from Ben Franco, Justin Houghton, Jeremy Houghton, Stephanie Horbaczewski, and the rest of the Vody Team. The authors would like to pay special thanks to Josh Houghton whose help in the ideation and production of this research and document was deeply appreciated.

We would also like to extend our deepest thanks to all the reviewers whose insightful comments helped to substantially improve this paper.

References

Nathanael Chambers. 2013. Event schema induction with a probabilistic entity-driven model. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Seattle, Washington, USA. Association for Computational Linguistics.

Nathanael Chambers. 2017. Behind the scenes of an evolving event cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 41–45, Valencia, Spain. Association for Computational Linguistics.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised learning of narrative event chains. In *Proceedings of ACL-08: HLT*, pages 789–797, Columbus, Ohio. Association for Computational Linguistics.

Nathanael Chambers 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*, pages 602–610, Suntec, Singapore. Association for Computational Linguistics.

Snigdha Chaturvedi, Haoruo Peng, and Dan Roth. 2017. Story comprehension for predicting what happens next. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1603–1614, Copenhagen, Denmark. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Joshua Eisenberg and Mark Finlayson. 2017. A simpler and more generalizable story detector using verb and character features. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2708–2715, Copenhagen, Denmark. Association for Computational Linguistics.

Mark Granroth-Wilding and Stephen Clark. 2016. What happens next? event prediction using a compositional neural network model. In *Proceedings*

of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA, pages 2727–2733. AAAI Press.

Matthew Honnibal and Ines Montani. 2017. Spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *Convolutional Neural Networks and Incremental Parsing*, 7(1).

Linmei Hu, Juanzi Li, Liqiang Nie, Xiaoli Li, and Chao Shao. 2017. What happens next? future subevent prediction using contextual hierarchical LSTM. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3450–3456. AAAI Press.

Zhongyang Li, Xiao Ding, and Ting Liu. 2018. Constructing narrative event evolutionary graph for script event prediction. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4201–4207. ijcai.org.

Zhongyang Li, Xiao Ding, and Ting Liu. 2019. Story ending prediction by transferable BERT. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 1800–1806. ijcai.org.

Shangwen Lv, Wanhui Qian, Longtao Huang, Jizhong Han, and Songlin Hu. 2019. Sam-net: Integrating event-level and chain-level attentions to predict what happens next. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6802–6809. AAAI Press.

Tomas Mikolov, Armand Joulin, Sumit Chopra, Michaël Mathieu, and Marc'Aurelio Ranzato. 2014. Learning longer memory in recurrent neural networks. *ArXiv preprint*, abs/1412.7753.

Tomás Mikolov, 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.

Nasrin Mostafazadeh, 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 the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. 2016b. 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, Berlin, Germany. Association for Computational Linguistics.

Nasrin Mostafazadeh, 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, Valencia, Spain. Association for Computational Linguistics.

Narrative Science. 2021. Narrative science.

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, Gothenburg, Sweden. Association for Computational Linguistics.

Karl Pichotta and Raymond J. Mooney. 2016. Learning statistical scripts with LSTM recurrent neural networks. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 2800–2806. AAAI Press.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Rachel Rudinger, 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*, pages 1681–1686, Lisbon, Portugal. Association for Computational Linguistics.

Roger C Schank and Robert P Abelson. 2013. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press.

Scriptbook. 2021. Scriptbook: Hard science, better content.

Rishi Sharma, James Allen, Omid Bakhshandeh, and Nasrin Mostafazadeh. 2018. Tackling the story ending biases in the story cloze test. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*,

pages 752–757, Melbourne, Australia. Association for Computational Linguistics.

Reid Swanson, Elahe Rahimtoroghi, Thomas Corcoran, and Marilyn Walker. 2014. Identifying narrative clause types in personal stories. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 171–180, Philadelphia, PA, U.S.A. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Bingning Wang, Ting Yao, Qi Zhang, Jingfang Xu, Kang Liu, Zhixing Tian, and Jun Zhao. 2019. Unsupervised story comprehension with hierarchical encoder-decoder. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 93–100.

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, Copenhagen, Denmark. Association for Computational Linguistics.

Jason Weston, Sumit Chopra, and Antoine Bordes. 2015. Memory networks. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Wenlin Yao and Ruihong Huang. 2018. Temporal event knowledge acquisition via identifying narratives. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 537–547, Melbourne, Australia. Association for Computational Linguistics.

A Appendix

A.1 MCNC Event Composition Models

Both the full event and verbs only models share similar architecture. For the full event model, it consists of two hidden layers, one for composing each event and a second for reducing the dimensions of the joined events.

Network Specifics:

1. Break input apart into two events using a Lambda layer (each with shape (4, 768) for full or (1, 768) for verbs only) then flatten the events
2. Pass each event into a different dense layer for composition with size 768 and a SELU activation (distinct layers since the event order is relevant as the first event is context and the second is the candidate completion)
3. Concatenate the two and pass them into another dense layer with size 72 and again a SELU activation
4. Finally, the output is a two unit layer with softmax activation

The models were compiled using Adam as the optimizer and with a categorical cross-entropy loss.

A.2 MCNC Attention Model

The attention models each have 2 steps to them, one for the mask pre-training and one for binary classification. The model is built upon 2 sub-models which the masking task aims to pre-train.

Input Model:

1. An input shape of [(5, 768), (5, 1), (5, 1)]
2. The (5, 768) is the BERT embeddings and don't require any further processing before combination
3. The two other inputs are position and sequence number, each of which are fed into an embedding layer of dimension 768
4. Finally, all three embedding vectors are summed for each of the 5 events.

Block Model:

1. Input is a vector for each event (shape of (5, 768)) and a boolean masking vector of shape (5,)
2. We construct multi-headed attention with 4 heads where the value matrix is the key matrix and the keys and queries are $\frac{2}{3}$ the embeddings size, namely 512
3. All the attention outputs are concatenated together and passed into a dense layer with SELU activation of size 768 to reduce their dimensions from 2048 back down to the block input size of 768
4. The new vectors are summed and normalized with a residual of the input
5. The model passes the vectors through two feed forward dense layers the first of size $4 \cdot 768 = 3072$ and the second brings the dimensionality back down to 768, both with SELU activation
6. Finally, the model once again goes through summation and normalization with a residual, this time using a residual of the previous normalization step.

Masking Task Model:

1. Input of event embeddings (verb embeddings) with one zeroed out and a boolean mask array set to the zeroed out event
2. Construct position and sequence vectors (sequence is 1 for context events and 2 for candidate completions events)
3. Run the Input Model
4. Run the Block Model
5. Select out the masked input event and pass it through a size 768 dense layer with SELU activation
6. Finally pass to an output Dense layer of 768 with tanh activation

Compiled with an Adam optimizer and mean squared error loss.

Binary Task Model:

1. Input of event embeddings (verb embeddings)
2. Construct an empty mask, position, and sequence vectors (sequence is 1 for context events and 2 for candidate completions events)
3. Run the Input Model

4. Run the Block Model

5. Flatten the output and compose it with a dense layer sized to be log-linear in size (61) with SELU activation and activity regularization using L2 with a 0.001 penalty

6. Add in Gaussian noise with a standard deviation of 0.01

7. Finally pass to the size 2 output layer with softmax activation

Compiled with an Adam optimizer and categorical cross-entropy loss.

A.3 SCT Attention Model

The SCT attention model functions the same way as the attention model from MCNC with one notable difference. Before the classification layer for the binary task, the SCT model uses an average pooling over all the context events to reduce dimensionality into the final classification layer, thereby reducing the layer size from $5 \cdot 768$ to $2 \cdot 768$. We found this helpful given the smaller training set for SCT as opposed to MCNC.

A.4 SCT_1.0 Test Set Cleaning

To use the SCT_1.0 test set as part of the training for SCT_1.5 we need to first remove all overlapping stories. Since there are several stories that have partial overlap or small changes, we checked for identical match on *any* sentence in the four context sentences. We thus dropped any stories from SCT_1.0 test set for which there was any sentence found in any stories in SCT_1.5 test set.