Event Causality Is Key to Computational Story Understanding

Anonymous ACL submission

Abstract

Cognitive science research and symbolic AI techniques suggest that event causality provides vital information for story understanding. However, machine learning systems for story understanding rarely employ event causality, partially due to the lack of methods that reliably identify open-world causal event relations. Leveraging recent progress in large language models, we present the first method for event causality identification that leads to material improvements in computational story understanding. Our technique sets a new state of the art on the COPES dataset (Wang et al., 2023c) for causal event relation identification. Further, in the downstream story quality evaluation task, the identified causal relations lead to 3.6-16.6% relative improvement on correlation with human ratings. In the multimodal story video-text alignment task, we attain 4.1-10.9% increase on Clip Accuracy and 4.2-13.5% increase on Sentence IoU. The findings indicate substantial untapped potential for event causality in computational story understanding.

1 Introduction

Stories manifest in various forms in modern society, such as myths, fables, gossip, comic books, bedtime rituals for children, and million-dollar theatrical productions. Stories are theorized to play crucial roles in civilization, from building collective identities (Lincoln, 1999) to familiarizing readers with social skills (Oatley, 2008). Research on computational story generation (Guan et al., 2020; Ammanabrolu et al., 2021; Xie et al., 2022; Yang et al., 2022; Hong et al., 2023; Yang et al., 2023) and understanding (Du et al., 2021; Xu et al., 2022; Andrus et al., 2022; Sang et al., 2022; Dong et al., 2023) has gained traction in recent years.

Converging evidence points to the information value of event causality in story understanding. Cognitive science indicates that humans heavily rely on event causality in story comprehension (Fletcher and Bloom, 1988; Graesser et al., 2003), as reflected by experiments on event recall and prediction (Trabasso and Van Den Broek, 1985; Keefe and McDaniel, 1993). The symbolic AI approach to computational story generation also makes extensive use of human-crafted event causal relations (Meehan, 1976; Young et al., 1994; Li and Riedl, 2010; Porteous et al., 2011; Soo et al., 2016). Anecdotally, merely adding the word "causal" to the ChatGPT prompt of Wang et al. (2023b) leads to a 3% relative boost in story evaluation (§5). However, event causal relations are rarely utilized by deep learning-based methods for story understanding, possibly due to the difficulty in identifying event causal relations in an open-world setting.

In this paper, we argue that causal structures — the story events and the causal relations among them — offer crucial and operationalizable information for computational story understanding; we further propose an easy-to-use technique for extracting such relations by prompting large language models (LLMs). With few-shot in-context learning, we enable LLMs to reconstruct causal structures from open-domain, free-form story text.

To verify the validity of the extracted causal structures, we first compare them against human-annotated causal relations of Mostafazadeh et al. (2020) and Wang et al. (2023c), leveraging a diverse range of LLMs. Empirically, the proposed method performs comparably with and sometimes surpasses supervised state-of-the-art baselines.

However, even if the causal structures are correct, they may not be of value to story understanding. To examine the value of the causal structures, we conduct further tests on two downstream tasks: story quality evaluation (Guan et al., 2021) and story video-text alignment (Dogan et al., 2018; Sun et al., 2022). Incorporating the extracted event causal structures into the models improves Kendall's tau coefficient by 6.4%-15.6% relatively in story evaluation and improves Clip Accuracy by 4.1-10.9%

and Sentence IoU by 4.2-13.5% in story video-text alignment.

In summary, the experimental results indicate that (1) the simple prompting technique we propose can identify story causal relations with high accuracy, and (2) the identified story structures indeed benefit story understanding tasks. Since the identified structures coincide substantially with human-annotated causal relations, we argue the empirical evidence supports the thesis that automatically extracted event causality facilitates computational story understanding. Our contributions can be summarized as:

- We propose a simple prompt-based technique for identifying event causal structures from free-form stories in diverse domains.
- With the proposed technique, we set a new state of the art on the COPES event causality benchmark (Wang et al., 2023c).
- To our best knowledge, this is the first work to demonstrate the practical benefits of causal story structures in automated story understanding, leading to substantial improvements on two distinct tasks, the text-only story quality evaluation and the multimodal story video-text alignment.

2 Related Work

2.1 Cognitive Science on Event Causality in Story Understanding

Cognitive science research indicates event causality offers crucial information in human narrative comprehension. Gernsbacher (1997) discover that when two events in a sentence are joined by a causal connective, the second event is better memorized than if the two events are joined by a non-causal connective. Trabasso and Van Den Broek (1985) and Van den Broek et al. (1996) find that story events with more causal connections are better recalled than those with fewer causal connections. Furthermore, event causality also influences the prediction of future events in a narrative. Keefe and McDaniel (1993) discover that, immediately after reading an event, words related to the possible effects of the event are recognized fast than unrelated words. Therefore, we postulate that event causality offers important information for story understanding.

2.2 Event Causality in Story Generation

Event causality has been widely used in computational story generation. Early works in symbolic story generation constructed story plans from human-written action templates that stipulate the preconditions and effects of actions (Lebowitz, 1985; Young et al., 1994; Bae and Young, 2008; Riedl and Young, 2010; Li and Riedl, 2010; Brenner, 2010; Swartjes, 2010). A story plans orders the story events so that the preconditions of later events are fulfilled by the effects of prior events.

The reliance on human-crafted knowledge limits story planners to narrow domains. Recent works attempt to utilize large language models commonsense to acquire action templates (Ye et al., 2022; Simon and Muise, 2022; Spiliopoulou et al., 2022; Kelly et al., 2023). Ammanabrolu et al. (2021) attempt to build story graphs with neural networks trained to perform causal relation completion. However, we are not aware of the utility of the extracted templates or story graphs in story understanding tasks. Compared to story planning, which sometimes can operate with a known list of people, objects, and actions, story understanding needs to deal with a vast assortment of narratives in open-world settings. As a result, the ability to identify causal structures in arbitrary stories becomes crucial (Caselli et al., 2021).

2.3 Commonsense Causal Reasoning

The objective of Commonsense Causal Reasoning (CCR) is to identify commonsense causal relations between events from text (Kuipers, 1984; Roemmele et al., 2011; Zhang et al., 2022), which is distinct from causal relation identification that require domain expertise, such as medical knowledge (Gurulingappa et al., 2012). Such causal relations are often heavily dependent on context, such as participants, time, and locations of events. For example, the two events "cooking at home" and "cooking at a restaurant" likely have different causes. The former is likely caused by hunger for food, but the latter is likely caused by the job requirement. COPA (Roemmele et al., 2011), GLUCOSE (Mostafazadeh et al., 2020), and COPES (Wang et al., 2023c) are prototypical datasets for CCR. In this paper, we focus on GLUCOSE and COPES, as their problem formulations contain more story context than COPA.

Prior works test LLMs on COPA (Wei et al., 2021; Anil et al., 2023; Gao et al., 2023), and LMs

on GLUCOSE and COPES (Li et al., 2022; Colon-Hernandez et al., 2023; Wang et al., 2023a,c). However, to the best of our knowledge, this work is the first to quantitatively explore the ability of ChatGPT-3.5 to understand Contextualized CCR (*i.e.,* GLUCOSE and COPES) and its impact on downstream tasks.

2.4 Open-ended Generated Story Evaluation

A critical ingredient in story generation research is the automatic evaluation of story quality, as human comparisons can be expensive and difficult to replicate. Metrics that involve direct comparisons against gold references, such as BLEU (Papineni et al., 2002), have limited applicability as there is no single correct story for each writing prompt. Researchers have proposed supervisedly trained techniques (Ghazarian et al., 2019; Sellam et al., 2020; Guan and Huang, 2020) and in-context learning methods based on LLMs (Wang et al., 2023b; Chiang and Lee, 2023; Shen et al., 2023). In this work, we demonstrate that providing event causality information in the LLM context can further enhance the correlation between LLM ratings and human ratings.

3 Methodology

Our objective is to acquire a *causal graph*, a directed graph that contains events as nodes and causal relations as directed edges. At the core of our approach is an LLM prompt that includes an instruction and a list of story events, as shown in Figure 1. The prompt can contain a number of examples, though we show only one due to length considerations. The prompt requests the LLM to detect and output causal relations among the events. For simplicity, we consider each sentence in the story as an event.

The output format for the causal relations is `Edge: (Node A -> Node B)`. In preliminary experiments, we find that the arrow "->" notation, similar to the influential DOT language (Gansner et al., 2015) of graph representation, tends to yield better results than other notations we tried.

4 Event Causality Extraction

To assess the quality of LLM-extracted event causal relations, we compare against two human-annotated benchmarks: COPES (Wang et al., 2023c) and GLUCOSE (Mostafazadeh et al., 2020).

```
Here is a list of nodes (events) from a
story event graph.  We want you to fill
in the edges of the event graph with
causal connections between nodes. An event
graph contains nodes and edges.  Each
node represents an event, and each edge
represents the causal connection between
two events.

Example Input:
Node 0:  When Dan goes to school in the
morning, he has to take the bus.
Node 1: One day Dan was running late, and
missed the bus to school.
Node 2:  Dan called his friend Pete, and
asked for a ride to school.
Node 3:  Pete gave Dan a ride to school,
but Dan was late for his first class.
Node 4: Luckily Dan wasn't late for any of
his other classes that day.
Example Output:
Edge 0: (Node 0 -> Node 1)
Edge 1: (Node 1 -> Node 2)
Edge 2: (Node 2 -> Node 3)
Edge 3: (Node 1 -> Node 3)
Edge 4: (Node 3 -> Node 4)
(continue with another five demonstrations)

Now, it is your turn to construct the event
graph for the following event list.
Event List:
Node 0: <S1>
Node 1: <S2>
Node 2: <S3>
Node 3: <S4>
Node 4: <S5>
Output:
```

Figure 1: The LLM prompt for event causal relationship extraction.

It is worth noting that the purpose of these experiments is not to seek state-of-the-art performance, but to verify the identified causal relations are of decent quality. However, we still manage to beat state-of-the-art baselines on COPES.

Task Definition and Datasets For the COPES task, the input is a pair of events and the output is whether or not a causal relationship exists between the events. COPES contains 340 stories and 1360 event pairs from ROCStories, split 50/50 into the validation set and the test set.

The GLUCOSE dataset contains a number of causal dimensions. We select only Dimensions 1 and 6, which concern causality between events. Given a story and one of its events, the task is to identify all of the direct causes or effects of the event from the story. GLUCOSE paraphrases the identified causes and effects as well as the current

event in a subject-verb-object format and applies reference-based evaluation such as BLEU. As our technique only outputs causal relations and does not perform paraphrasing, we directly use the original sentences from the story.

Model For event causality extraction, we experiment with four advanced LLMs: Llama2-13B-chat (Touvron et al., 2023), Falcon-instruction-40B[1], Yi-34B-chat[2] and ChatGPT (3.5-turbo)[3] (Ouyang et al., 2022).

Evaluation For COPES, we follow Wang et al. (2023c) and report accuracy, Micro F1, and Marco F1. We use 6 randomly selected stories from the validation set as in-prompt examples and report performance on the COPES test set.

For GLUCOSE, we choose in-prompt examples randomly from the training set and conduct evaluation on the GLUCOSE test set of 293 stories. As of evaluation metrics, our main objective is to evaluate if the model can accurately distinguish between causally related, positive event pairs and unrelated, negative event pairs. Hence, we compute the precision and recall of the predicted positive class, and combine them as the F1 score. For completeness, we also adopt BLEU[4] following Mostafazadeh et al. (2020) as well as the BERTscore (Zhang et al., 2019) and Sentence BERT similarity (Reimers and Gurevych, 2019a), but these metrics mostly evaluate the surface form and are not as important as F1.

Baselines For COPES, we directly compare our model with COLA (Wang et al., 2023c). Also, we compare against ROCK (Zhang et al., 2022) and ClozePromptScore (Tamborrino et al., 2020) as replicated and reported by Wang et al. (2023c).

For GLUCOSE, we replicated the two models used by Mostafazadeh et al. (2020) as baselines. We train the same networks (T5 and GPT2-large) on the training split. However, the training set of Mostafazadeh et al. (2020) contains only positive examples, or pairs of causally related events. In order to handle causually unrelated, negative event pairs, which are abundant in open-world stories, we exhaustively add all negative event pairs to the training set, yielding 590K training samples. After that, from pretrained LLM weights, we train baseline network to output a description of the cause or effect for positive cases or "Nil" for negative cases. See Appendix A.2 for more details.

[1] https://falconllm.tii.ae/falcon.html

[2] https://01.ai/

[3] https://platform.openai.com/docs/models/gpt-3-5

[4] Implementation from Post (2018)

	Acc.	Micro F1	Marco F1
Supervised			
ClozePromptScore	62.06	45.57	58.22
ROCK	66.47	51.90	63.08
COLA	70.29	57.38	67.29
Few-shot (Ours)			
Falcon-40B-instuct	65.74	41.60	58.68
Llama-2-13B-chat	71.47	47.58	63.99
Yi-34B-chat	72.94	55.98	68.22
ChatGPT-3.5	**74.41**	**57.56**	**69.62**

Table 1: Performance on COPES.

	F1	BLEU	BERTScore	BERT Similarity.
Supervised				
GPT-2_{large}	59.54	28.92	79.86	84.64
$T5_{large}$	**61.50**	**31.75**	**84.34**	**88.77**
Few-Shot (Ours)				
Falcon	28.57	13.43	38.65	25.68
Llama-2	51.70	19.77	58.22	54.82
Yi	57.95	18.95	77.42	84.32
ChatGPT	60.75	21.20	75.33	80.89

Table 2: The average BLEU, BERTScore, BERT Similarity, and F1 score averaged over dimensions 1 & 6.

Results In Table 1, we show performance on COPES. With only 6 example stories, our technique with ChatGPT outperforms the state-of-the-art (SOTA) supervised model, COLA, by 4.2% in accuracy and 2.3% in Marco F1. Furthermore, our method is robust and can generalize to different LLMs. We observe our technique with Yi-34B-chat also outperforms the SOTA performance and Llama-2-13B outperforms ClozePromptScore.

Table 2 shows performance on GLUCOSE. In terms of F1 score, our technique with ChatGPT-3.5 outperforms supervised GPT2-large model trained on 590K causal statements and performs comparably to the supervised T5 model. Among the open-source models, Yi-34B-chat performs best and can match the GPT2-large baseline on most metrics. On reference-based metrics, LLMs score lower than supervised models mainly because supervised models are trained to imitate paraphrased annotations of GLUCOSE whereas while the few-shot LLMs are not trained or prompted to paraphrase. Hence, the reference-based metrics underestimate the actual performance of our technique. However,

on BERT similarity, our few-shot technique with Yi-34B-chat closes the gap with GPT-2.

5 Open-domain Story Evaluation

Having established that the event causality that we identified are quite accurate, we still need to verify if the event causality provide valuable information to actual story understanding tasks. To this end, we conduct two tests. This section describes the first test, automatic story quality evaluation. We generate the quality ratings from ChatGPT with and without the automatically extracted event causal relations, and compare how they correlate with human ratings.

Approach: Quality Ratings Conditioned on Causal Graphs We propose a two-stage prompting method that scores the quality of a story conditioned on its causal graph. First, we prompt ChatGPT to generate the causal graph of the story, using the same prompt in Figure 1. Then, we include the causal graph, which contains a list of causal relations between event descriptions, in a scoring prompt that asks ChatGPT to generate an overall score for story quality. The scoring prompt is derived from Wang et al. (2023b) (see Appendix B.1). We test both the zero-shot and the few-shot settings.

Dataset The dataset we use is OpenMEVA (Guan et al., 2021) which trains five different story generation models on ROCStories (Mostafazadeh et al., 2016) and another five on WritingPrompt (WP) (Fan et al., 2018). After that, each model generates stories from 200 writing prompts. As a result, OpenMEVA consists of two parts: OpenMEVA-ROC and OpenMEVA-WP, each containing 1,000 generated stories. Each story is evaluated by five human annotators, each assigning a score between 1 and 5. The final score for the story is the average of the five.

OpenMEVA-ROC and OpenMEVA-WP have different characteristics. Stories in OpenMEVA-ROC always contain 5 sentences and, on average, 34 words. The lengths of stories in OpenMEVA-WP are more varied, with an average of 20 sentences and 194 words. Qualitatively, we observe that OpenMEVA-ROC mostly retain the style of ROCStories, where the sentences are simple and describe clear-cut events. In comparison, OpenMEVA-WP is much more diverse, containing much non-event content such as conversations and monologues, and much more vague event boundaries. We show two examples in Figure 10 of the Appendix.

Evaluation We employ correlation to assess the similarity between ChatGPT scores and human ratings. We report three well-established correlations: Pearson correlation (Benesty et al., 2009), Spearman rank correlation (Zar, 2005), and Kendall's tau coefficient (Kendall, 1938). All of these measures have values ranging from -1 to 1, with values closer to 1 indicating a stronger positive correlation.

We calculate the correlation between human ratings and ChatGPT ratings at two aggregation levels: (1) dataset level, where we measure the correlation between two scoring systems across the entire dataset, and (2) writing prompt level, where we compute the correlation between the two scoring systems for the five stories generated for each writing prompt and then average the results. The formula can be found in Appendix B.1.

Baselines We compare our method with seven baselines, including two reference-based methods, one hybrid method, two reference-free methods, and two LLM methods. First, the reference-based baselines rate the computer-generated stories by matching them against a human-written story for the same writing prompt. The two baselines are (1) BERTScore (Zhang et al., 2019) and (2) BARTScore+CNN+Para (Yuan et al., 2021), which computes the perplexity of the text conditioned on the reference text. The hybrid method is (3) BLEURT (Sellam et al., 2020), a neural network that Guan and Huang (2020) adapt to evaluate machine stories against a reference.

The two reference-free baselines are (4) perplexity on GPT-2 (Radford et al., 2019), which gives higher rankings to stories with lower perplexity and (5) UNION (Guan and Huang, 2020), a neural network trained to discriminate machine stories from human stories; machine stories that are more similar to human stories are considered better.

In addition, we compare against the ChatGPT prompt of Wang et al. (2023b), which rates the OpenMEVA-ROC dataset on a scale of 1-5 stars. For fair comparisons, we also replicate this baseline using the same few-shot settings and the same ChatGPT-3.5 model. Finally, we create another variation (ChatGPT-"causal") by adding the word "causal" to the prompt of Wang et al. (2023b). Details can be found in Appendix B.1.

Metrics	OpenMEVA-ROC (n=1000)						OpenMEVA-WP (n=1000)					
	Writing Prompt Level			Dataset Level			Writing Prompt Level			Dataset Level		
	Pear.	Spear.	Kend.	Pear.	Spear.	Kend.	Pear.	Spear.	Kend.	Pear.	Spear.	Kend.
BART+CNN+Para	0.050	0.064	0.062	0.062	0.074	0.043	0.014	0.046	0.045	0.083	0.077	0.053
BERTScore-F1	0.144	0.131	0.103	0.127	0.113	0.079	0.089	0.085	0.077	0.033	0.031	0.022
BLEURT in-domain*	-	-	-	0.316	-	-	-	-	-	0.212	-	-
Perplexity	0.330	0.324	0.265	0.255	0.306	0.213	0.373	**0.381**	0.318	0.303	0.324	0.225
UNION in-domain*	-	-	-	0.412	-	-	-	-	-	0.326	-	-
UNION cross-domain*	-	-	-	0.213	-	-	-	-	-	0.229	-	-
Original Wang et al. (2023b)	0.490	0.472	0.427	0.439	0.415	0.342	-	-	-	-	-	-
ChatGPT zero-shot												
Repl. Wang et al. (2023b)♠	0.526	0.520	0.472	0.446	0.436	0.366	0.281	0.257	0.236	0.203	0.199	0.165
ChatGPT-"causal"	0.531	0.522	0.474	0.460	0.451	0.379	0.301	0.275	0.246	0.215	0.215	0.183
ChatGPT-causal-graph	0.576	0.562	0.510	0.520	0.505	0.423	0.331	0.299	0.273	0.277	0.277	0.230
ChatGPT in-domain few-shot												
Repl. Wang et al. (2023b)♠	0.553	0.526	0.466	0.498	0.496	0.398	0.313	0.291	0.257	0.269	0.262	0.208
ChatGPT-"causal"	0.560	0.537	0.480	0.501	0.503	0.402	0.327	0.305	0.270	0.276	0.276	0.218
ChatGPT-causal-graph	**0.592**	**0.575**	**0.520**	**0.526**	**0.514**	**0.425**	0.339	0.313	0.285	0.284	0.294	0.246
ChatGPT cross-domain few-shot												
Repl. Wang et al. (2023b)♠	0.519	0.500	0.433	0.462	0.452	0.348	0.373	0.345	0.301	0.293	0.297	0.228
ChatGPT-"causal"	0.506	0.513	0.449	0.461	0.463	0.357	**0.404**	0.353	0.303	0.314	0.316	0.243
ChatGPT-causal-graph	0.547	0.530	0.459	0.498	0.482	0.370	0.387	0.367	**0.323**	**0.328**	**0.328**	**0.258**

∗: Results are taken from OpenMEVA benchmark Guan et al. (2021)

♠: For fair comparison, We replicate Wang et al. (2023b) using the same few-shot settings and ChatGPT model (`gpt3.5-turbo-0613, temp=0`) as in other experiments.

Table 3: Writing prompt-level and dataset-level correlations on OpenMEVA. (Spear.: Spearman correlation; Pear.: Pearson correlation; Kend.: Kendall's Tau).

Results and Discussion We show results in Table 3, where our approach is denoted as ChatGPT-causal-graph. We report three settings, zero-shot, in-domain few-shot, and cross-domain few-shot. In zero-shot, we do not include any examples in the LLM prompt. In in-domain few-shot, we include in the prompt two example stories we wrote manually in the style of OpenMEVA-ROC and OpenMEVA-WP respectively. In cross-domain few-shot, we include two example stories from OpenMEVA-ROC when testing on OpenMEVA-WP and vice versa.

We observe that event causality provides significant improvements, especially in zero-shot settings. On zero-shot ROC, our method achieves relative improvements of 8.05% to 16.59% over the best baseline. On few-shot ROC, our method achieves relative improvements of 3.65% to 11.59% over the best baseline. On zero-shot WP and few-shot WP, causality graph brings even greater gains over the (Wang et al., 2023b) baseline, due to the low baseline performance. However, on WP stories, we surpass all baselines only in the cross-domain few-shot condition.

The performance on WP warrants further analysis. Our technique performs worse with in-domain examples than cross-domain examples, which seems to contradict machine learning commonsense. We attribute this to two reasons. First, the WP stories are longer and hence more difficult to understand as in-context examples. Second, WP stories are less event-centric and have more vague event boundaries than ROC. This may cause errors in ChatGPT-extracted causal graphs, which hurt in-context learning. Instead, when we use the more correct causal graphs from ROC as examples in the prompt, performance improves.

Though the WP results suggest that our technique for causal graph extraction may not work equally well in all story domains, we emphasize that this is the first work that has ever shown causal graphs provide benefits for *any* computational story understanding task.

6 Story Video-text Alignment

The second test task for the automatically extracted event causal relations is story video-text alignment. Due to the wide use of storytelling techniques, aligning the video and the text modalities requires significant story understanding (Sun et al., 2022). We show that preceding events on the causal graph provide crucial context for this task. Considering the domain gaps between this multimodal task and previous pure text tasks, this experiment supports

the argument that our proposed technique can cope with a broad range of real-world stories.

Task Definition and Datasets The task starts with a movie summary video from YouTube, which contains shots selected from a movie and human narration of the main plotline. The video has been segmented into a number of clips, and the narration has been transcribed into text and segmented into chunks. However, due to modality differences, the text chunks and video clips at the same time may not match each other semantically. The task is weakly supervised; we need to find the correct semantic alignment without training on gold alignment labels.

We adopt the Synopsis of Movie Narratives (SyMoN) dataset (Sun et al., 2022) for training, which contains 5,193 movie summary videos. The test set comes from the YouTube Movie Summary (YMS) dataset (Dogan et al., 2018), which has gold alignment labels. We report results on two different splits and textual chunking levels.

Approach: Context-aware video-text alignment To align a video sequence and a text sequence, we follow a three-step procedure. First, we encode each video clip and each text chunk together with their temporal and causal contexts. Then, we calculate the cosine similarity between each video-text pair. Finally, we calculate the overall sequence alignment from the individual similarity scores using Dynamic Time Warping (DTW), a sequence alignment algorithm detailed in Appendix B.2.

We finetune pretrained visual and textual encoders from UniVL (Luo et al., 2020). The visual encoder contains an S3D network that encodes 1-second clips into tokens, followed by a Transformer. The text encoder is a Transformer. We denote the encoded features for the ith text chunk as $\boldsymbol{t}_i$ and the encoded features of the ith video clip as $\boldsymbol{v}_i$. The feature vectors are normalized to unit length, so that cosine similarity is simply dot product. With randomly sampled negative text features $\boldsymbol{t}_k$ and video features $\boldsymbol{v}_k$, we finetune the encoders by minimizing the contrastive NCELoss (Gutmann and Hyvarinen, 2010),

$$L_{\text{NCE}} = \frac{1}{N}\sum_{i=1}^{N} -\boldsymbol{v}_i^\top \boldsymbol{t}_i + \log\Big(\exp \boldsymbol{v}_i^\top \boldsymbol{t}_i + \sum_{k\neq i}^{K} \exp \boldsymbol{v}_i^\top \boldsymbol{t}_k + \sum_{k\neq i}^{K} \exp \boldsymbol{v}_k^\top \boldsymbol{t}_i\Big), \tag{1}$$

where N is the total number of training samples and K is the number of negative samples.

Note that the training set of Sun et al. (2022) does not contain human-annotated alignment. Therefore, we adopt a weakly supervised approach that considers video clips and text chucks with similar temporal positions as positive pairs and randomly sample negative pairs during training.

When encoding the current video and text, the visual and textual Transformer networks also take in a number of contextual items (video clips or text chunks). These contextual item values are retrieved from a memory bank, which stores the item values from all layers of the Transformer network. After retrieval, the item values are fed to the corresponding Transformer encoder layers. We will discuss how the context is constructed momentarily.

Context Identification Here we distinguish between two types of contexts, temporal context and causal context. The temporal context contains m number of items (video clips or textual chunks) that immediately precede the current item being encoded. The causal context contains c items preceding the current item on the causal graph extracted from the entire story text by ChatGPT.

The causal context for a textual chuck can be directly identified from the causal graph, but the causal context for a video clip requires some extra processing. During training, we first find the textual causal predecessors from the causal graph. Next, the video clips temporally closest to the text predecessor chunks are deemed as the causal context for the current video clip. During inference, we align the two sequences incrementally from the beginning. Given a text predecessor, we use the aligned portion of the two sequences to locate the corresponding video clips as the causal context. The process is illustrated in Figure 2.

Evaluation Metrics Following Dogan et al. (2018), we use two evaluation metrics: Clip Accuracy, defined as the temporal proportion of correctly aligned video segments, and Sentence IoU, defined as the intersection-over-union between the aligned video durations and the ground-truth durations.

Baselines Our main baseline is the temporal context only, ablated version of our technique, which uses $c + m$ temporal context items instead of c causal context items and m temporal context items. Additionally, we compare against (1) the Minimal Distance and Dynamic Time Warping baseline in

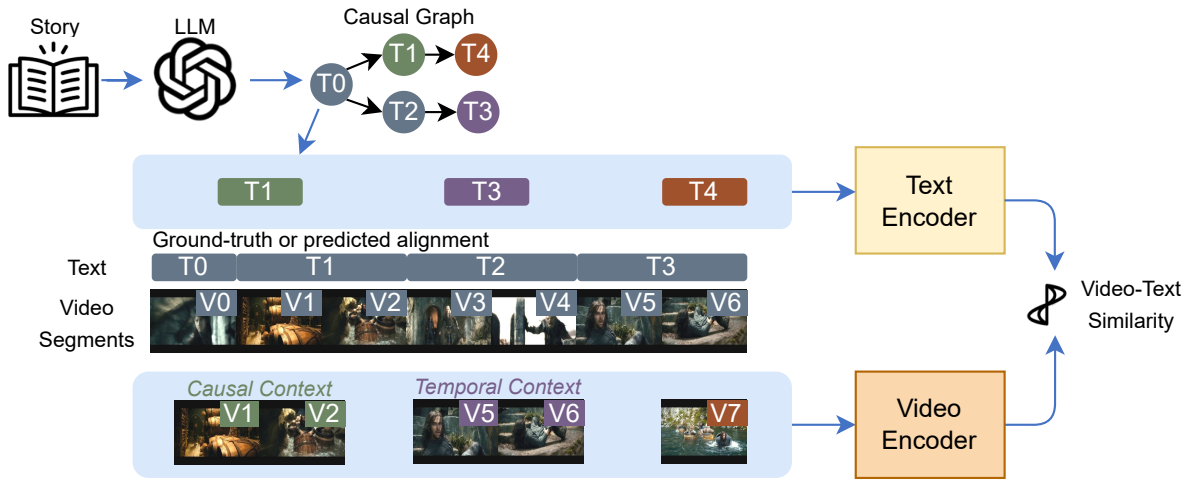

Figure 2: The process of video context identification. The causal context is marked in green, the temporal context in purple, and the current item in red. For illustration purposes, the number of causal and temporal context items are both set to 2.

NeuMATCH (Dogan et al., 2018), and (2) the Minimal Distance baseline in SyMoN (Sun et al., 2022). Both baselines did not utilize context.

Results and Discussion As shown in Table 4, incorporating causal context from the identified causal graph yields improvements across the board. The highest improvement for Clip Accuracy is 10.9% and the highest improvement for Sentence IoU is 13.5%. Note the SyMoN test split contains 65 videos whereas the NeuMATCH test split contains only 15 videos. Hence, the SyMoN split numbers may be more trustworthy. In the sentence-level SyMoN split, which is arguably more natural than the sub-sentence level, adding event causality improves Clip Accuracy by 7.7% and Sentence IoU by 8.0%. These results convincingly demonstrate that the automatically extracted causal graphs provide real benefits in the story video- text alignment task, even though the multimodal task clearly differs from the text-only tasks considered earlier.

	Clip Acc.	Sent. IoU
NeuMATCH Split (sub-sentence level)		
NeuMATCH-MD (Supervised)	4.0	2.4
NeuMATCH-DTW (Supervised)	10.3	7.5
SyMoN-MD	5.9	2.7
Temporal Context-DTW	12.3	7.1
Causal+Temporal Context-DTW	**23.2** (↑**10.9**)	**18.4** (↑ 10.9)
SyMoN Split (sub-sentence level)		
SyMoN-MD	10.1	1.9
Temporal Context-DTW	10.2	8.0
Causal+Temporal Context-DTW	**24.2** (↑ 8.2)	**21.5** (↑**13.5**)
NeuMATCH Split (sentence level)		
SyMoN-MD	7.4	3.4
Temporal Context-DTW	29.2	18.3
Causal+Temporal Context-DTW	**33.3** (↑ 4.1)	**22.5** (↑ 4.2)
SyMoN Split (sentence level)		
SyMoN-MD	7.7	3.3
Temporal Context-DTW	32.5	19.6
Causal+Temporal Context-DTW	**40.2** (↑ 7.7)	**27.6** (↑ 8.0)

Table 4: Alignment performance on YMS. The improvement over baseline is shown in the parenthesis.

7 Conclusion

In this paper, we propose a simple and effective in-context-learning method for extracting event causality from stories with LLMs. We match and outperform supervised baselines in event causality extraction. Furthermore, We validate the quality of the extracted event causality by applying them in downstream story understanding tasks. Experiments show that event causality assists story evaluation and video-text alignment, indicating the critical role of event causality in story understanding.

Limitations

The scope of our research is limited to causality between events. As such, the results may not extend to other types of causality (e.g. causality between events and emotion, location or possession states). Additionally, our technique works best on stories with clear event boundaries. When the stories contain dialogues or when the events are unclear, the improvements achieved by the causal graphs are limited.

Furthermore, our exploration of event causality is confined to fiction stories and does not involve

other domains such as news and tweets. While stories are a reflection of real life, their distribution emphasizes drama over realism. Therefore, it is not immediately clear if the event causality could play similar roles in other domains.

Our research on event causality for automatic story evaluation is primarily focused on the overall score, while some other studies delve into scoring different dimensions of quality, such as coherence, logicality, and relevance (Chhun et al., 2022; Ke et al., 2022; Xie et al., 2023). We argue that event causality may contribute to more than one dimension, since a clear and accurate causal graph implies relevance among events, logical causality, and overall coherence. Exploring how event causality contributes to the quality dimensions could be an interesting line of future research.

Broader Impact

In this paper, we explore the use of LLM-extracted event causality in story understanding. We recognize LLMs may inadvertently contain bias derived from training data. Furthermore, the story content we use may contain the biases of their creators, as well as social biases from the time periods of production.

Consequently, the causal relationships generated in our study are not intended as unbiased presentations of social norms. For this reason, we urge researchers to take caution when relying on LLMs or stories as a source for learning cultural and social conventions.

References

Prithviraj Ammanabrolu, Wesley Cheung, William Broniec, and Mark O Riedl. 2021. Automated storytelling via causal, commonsense plot ordering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5859–5867.

Berkeley R Andrus, Yeganeh Nasiri, Shilong Cui, Benjamin Cullen, and Nancy Fulda. 2022. Enhanced story comprehension for large language models through dynamic document-based knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10436–10444.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Byung-Chull Bae and R Michael Young. 2008. A use of flashback and foreshadowing for surprise arousal in narrative using a plan-based approach. In *Interactive Storytelling: First Joint International Conference on Interactive Digital Storytelling, ICIDS 2008 Erfurt, Germany, November 26-29, 2008 Proceedings 1*, pages 156–167. Springer.

Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. 2009. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 37–40. Springer.

Michael Brenner. 2010. Creating dynamic story plots with continual multiagent planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, pages 1517–1522.

Tommaso Caselli, Eduard Hovy, Martha Palmer, and Piek Vossen. 2021. *Computational Analysis of Storylines: Making Sense of Events*. Cambridge University Press.

Cyril Chhun, Pierre Colombo, Chloé Clavel, and Fabian M Suchanek. 2022. Of human criteria and automatic metrics: A benchmark of the evaluation of story generation. *arXiv preprint arXiv:2208.11646*.

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 2023 Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Pedro Colon-Hernandez, Henry Lieberman, Yida Xin, Claire Yin, Cynthia Breazeal, and Peter Chin. 2023. Adversarial transformer language models for contextual commonsense inference. *arXiv preprint arXiv:2302.05406*.

Pelin Dogan, Boyang Li, Leonid Sigal, and Markus Gross. 2018. A neural multi-sequence alignment technique (neumatch). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8749–8758.

Yijiang Dong, Lara Martin, and Chris Callison-Burch. 2023. Corrpus: Code-based structured prompting for neurosymbolic story understanding. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13152–13168.

Li Du, Xiao Ding, Ting Liu, and Bing Qin. 2021. Learning event graph knowledge for abductive reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5181–5190.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia. Association for Computational Linguistics.

Charles R Fletcher and Charles P Bloom. 1988. Causal reasoning in the comprehension of simple narrative texts. *Journal of Memory and language*, 27(3):235–244.

Emden R. Gansner, Eleftherios Koutsofios, and Stephen North. 2015. Drawing graphs with dot. Technical report.

Jinglong Gao, Xiao Ding, Bing Qin, and Ting Liu. 2023. Is chatgpt a good causal reasoner? a comprehensive evaluation. *arXiv preprint arXiv:2305.07375*.

Morton Ann Gernsbacher. 1997. Coherence cues mapping during comprehension. In *Processing inter-clausal relationships*, pages 3–21. Psychology Press.

Sarik Ghazarian, Johnny Tian-Zheng Wei, Aram Galstyan, and Nanyun Peng. 2019. Better automatic evaluation of open-domain dialogue systems with contextualized embeddings. *arXiv preprint arXiv:1904.10635*.

Arthur C Graesser, Brent Olde, and Bianca Klettke. 2003. How does the mind construct and represent stories? *Narrative impact: Social and cognitive foundations*, page 121.

Jian Guan, Fei Huang, Zhihao Zhao, Xiaoyan Zhu, and Minlie Huang. 2020. A knowledge-enhanced pretraining model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, 8:93–108.

Jian Guan and Minlie Huang. 2020. Union: An unreferenced metric for evaluating open-ended story generation. *arXiv preprint arXiv:2009.07602*.

Jian Guan, Zhexin Zhang, Zhuoer Feng, Zitao Liu, Wenbiao Ding, Xiaoxi Mao, Changjie Fan, and Minlie Huang. 2021. Openmeva: A benchmark for evaluating open-ended story generation metrics. In *Proceedings of the 2021 Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of biomedical informatics*, 45(5):885–892.

Michael U Gutmann and Aapo Hyvarinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. pages 297–304.

Xudong Hong, Asad Sayeed, Khushboo Mehra, Vera Demberg, and Bernt Schiele. 2023. Visual writing prompts: Character-grounded story generation with curated image sequences. *Transactions of the Association for Computational Linguistics*, 11:565–581.

Pei Ke, Hao Zhou, Yankai Lin, Peng Li, Jie Zhou, Xiaoyan Zhu, and Minlie Huang. 2022. Ctrleval: An unsupervised reference-free metric for evaluating controlled text generation. *arXiv preprint arXiv:2204.00862*.

Dennis E Keefe and Mark A McDaniel. 1993. The time course and durability of predictive inferences. *Journal of memory and language*, 32(4):446–463.

Jack Kelly, Alex Calderwood, Noah Wardrip-Fruin, and Michael Mateas. 2023. There and back again: Extracting formal domains for controllable neurosymbolic story authoring. In *AIIDE*.

Maurice G Kendall. 1938. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93.

Benjamin Kuipers. 1984. Commonsense reasoning about causality: deriving behavior from structure. *Artificial intelligence*, 24(1-3):169–203.

Michael Lebowitz. 1985. Story-telling as planning and learning. *Poetics*, 14(6):483–502.

Boyang Li and Mark Riedl. 2010. An offline planning approach to game plotline adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 6, pages 45–50.

Bryan Li, Lara J Martin, and Chris Callison-Burch. 2022. Cis2: A simplified commonsense inference evaluation for story prose. In *ACL Workshop*.

Bruce Lincoln. 1999. *Theorizing myth: Narrative, ideology, and scholarship*. University of Chicago Press.

Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. 2020. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*.

J. Meehan. 1976. *The Metanovel: Writing Stories by Computers*. Ph.D. thesis, Yale University.

N. Mostafazadeh, Aditya Kalyanpur, Lori Moon, David W. Buchanan, Lauren Berkowitz, Or Biran, and Jennifer Chu-Carroll. 2020. Glucose: Generalized and contextualized story explanations. In *Conference on Empirical Methods in Natural Language Processing*.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and evaluation framework for deeper understanding of commonsense stories. *arXiv preprinto arXiv:1604.01696*.

Keith Oatley. 2008. The mind's flight simulator. *The Psychologist*, 21:1030–1032.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Julie Porteous, Jonathan Teutenberg, Fred Charles, and Marc Cavazza. 2011. Controlling narrative time in interactive storytelling. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 449–456.

Matt Post. 2018. A call for clarity in reporting bleu scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Nils Reimers and Iryna Gurevych. 2019a. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019b. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Mark O Riedl and Robert Michael Young. 2010. Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research*, 39:217–268.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*.

Yisi Sang, Xiangyang Mou, Mo Yu, Shunyu Yao, Jing Li, and Jeffrey Stanton. 2022. Tvshowguess: Character comprehension in stories as speaker guessing. In *NAACL*.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696*.

Chenhui Shen, Liying Cheng, Yang You, and Lidong Bing. 2023. Are large language models good evaluators for abstractive summarization? *arXiv preprint arXiv:2305.13091*.

Nisha Simon and Christian Muise. 2022. Tattletale: Storytelling with planning and large language models. In *ICAPS Workshop on Scheduling and Planning Applications workshop*.

Von-Wun Soo, Tai-Hsun Chen, and Chi-Mou Lee. 2016. Generate causal story plots by monte carlo tree search based on common sense ontology. In *2016 Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS)*, pages 610–615. IEEE.

Evangelia Spiliopoulou, Artidoro Pagnoni, Yonatan Bisk, and Eduard Hovy. 2022. Events realm: Event reasoning of entity states via language models. *arXiv preprint arXiv:2211.05392*.

Yidan Sun, Qin Chao, Yangfeng Ji, and Boyang Li. 2022. Synopses of movie narratives: a video-language dataset for story understanding. *arXiv preprint arXiv:2203.05711*.

Ivo Martinus Theodorus Swartjes. 2010. *Whose story is it anyway? How improv informs agency and authorship of emergent narrative*. Ph.D. thesis, University of Twente.

Alexandre Tamborrino, Nicola Pellicano, Baptiste Pannier, Pascal Voitot, and Louise Naudin. 2020. Pre-training is (almost) all you need: An application to commonsense reasoning. *arXiv preprint arXiv:2004.14074*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Tom Trabasso and Paul Van Den Broek. 1985. Causal thinking and the representation of narrative events. *Journal of memory and language*, 24(5):612–630.

Paul Van den Broek, Elizabeth Pugzles Lorch, and Richard Thurlow. 1996. Children's and adults' memory for television stories: The role of causal factors, story-grammar categories, and hierarchical level. *Child development*, 67(6):3010–3028.

Hecong Wang, Erqian Xu, Pinxin Liu, Zijian Meng, and Zhen Bai. 2023a. Contextualized multi-step commonsense reasoning through context extension.

Jiaan Wang, Yunlong Liang, Fandong Meng, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023b. Is chatgpt a good nlg evaluator? a preliminary study. *arXiv preprint arXiv:2303.04048*.

Zhaowei Wang, Quyet V. Do, Hongming Zhang, Jiayao Zhang, Weiqi Wang, Tianqing Fang, Yangqiu Song, Ginny Wong, and Simon See. 2023c. COLA: Contextualized commonsense causal reasoning from the causal inference perspective. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5253–5271, Toronto, Canada. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Yuqiang Xie, Yue Hu, Yunpeng Li, Guanqun Bi, Luxi Xing, and Wei Peng. 2022. Psychology-guided controllable story generation. *COLING*.

Zhuohan Xie, Trevor Cohn, and Jey Han Lau. 2023. The next chapter: A study of large language models in storytelling. In *Proceedings of the 16th International Natural Language Generation Conference*, pages 323–351.

Ying Xu, Dakuo Wang, Mo Yu, Daniel Ritchie, Bingsheng Yao, Tongshuang Wu, Zheng Zhang, Toby Jia-Jun Li, Nora Bradford, Branda Sun, Tran Bao Hoang, Yisi Sang, Yufang Hou, Xiaojuan Ma, Diyi Yang, Nanyun Peng, Zhou Yu, and Mark Warschauer. 2022. Fantastic questions and where to find them: FairytaleQA – an authentic dataset for narrative comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Kevin Yang, Dan Klein, Nanyun Peng, and Yuandong Tian. 2023. DOC: Improving long story coherence with detailed outline control. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3378–3465, Toronto, Canada. Association for Computational Linguistics.

Kevin Yang, Yuandong Tian, Nanyun Peng, and Dan Klein. 2022. Re3: Generating longer stories with recursive reprompting and revision. In *Conference on Empirical Methods in Natural Language Processing*.

Anbang Ye, Christopher Cui, Taiwei Shi, and Mark O. Riedl. 2022. Neural story planning. *arXiv Preprint 2212.08718*.

R Michael Young, Martha E Pollack, and Johanna D Moore. 1994. Decomposition and causality in partial-order planning. In *AIPS*, pages 188–194.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34:27263–27277.

Jerrold H Zar. 2005. Spearman rank correlation. *Encyclopedia of Biostatistics*, 7.

Jiayao Zhang, Hongming Zhang, Weijie Su, and Dan Roth. 2022. Rock: Causal inference principles for reasoning about commonsense causality. In *International Conference on Machine Learning*, pages 26750–26771. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

A Causality Extraction on Glucose

A.1 GLUCOSE Causality Dimensions

GLUCOSE (Mostafazadeh et al., 2020) divides causality within a story into ten dimensions. Here X represents the current event:

1. Event that directly causes or enables X
2. Emotion or basic human drive that motivates X
3. Location state that enables X
4. Possession state that enables X
5. Other attributes enabling X
6. Event that X directly causes or enables
7. An emotion that is caused by X
8. A change in location that X results in
9. A change of possession that X results in
10. Other changes in property that X results in

A.2 GLUCOSE Finetuning Setup

We finetuned two types of LMs, Decoder-only (GPT2) and Encoder-Decoder (T5), on 590K causal statements from GLUCOSE. These statements comprise 290K positive samples and 300K negative samples. The statements that explain the causal relationship between events, states or emotions are positive samples generated by AMT workers from the original GLUCOSE dataset. When there was no causal relationship between two events/states/emotions, there was no statement generated by AMT works. In this case, we generate a simple negative statement: "No, escaped." for such events/states/emotions.

GPT-2 We finetuned gpt2-large on 4 NVIDIA A6000 GPUs with a learning rate of $3e-5$ for 10 epochs. The batchsize is set to 64. The weight decay factor is $5e-4$. 15% of the input tokens are masked at random.

T5 We finetuned T5-large on 4 NVIDIA A6000 GPUs for 10 epochs, using a batch size of 32 on each GPU. The learning rate was $5e-4$ under the cosine schedule with a warmup for the first 500 steps, and we adjusted the weight decay factor to $1e-2$. No masked tokens were applied.

A.3 Prompts

Aside from the prompt shown in Figure 1, we design 11 additional prompts for event causality extraction. These prompts are designed to handle all ten GLUCOSE causality dimensions, as well as paraphrase the event descriptions in the GLUCOSE format.

Basic Prompt We show the basic prompt we use for causality extraction in Figure 3, all of the other prompts in this section are variations of this prompt.

```
Your job is to find all the causalities in
a story.
You will be given a list of events in the
story. An event can be caused by another
event, an emotion, a possession state,
a location state or some other property.
Similarly, the effect of an event can be
another event, an emotion, a possession
state, a location state or some other
property. For every event in the story,
find all its causes and effects. For the
description of events, you should write
the event id in the parentheses after the
description. For descrption of emotions,
possession states, location states or
other properties, write the type of the
description in the parantheses after the
description.
Example Input:
Event 0: When Dan goes to school in the
morning, he has to take the bus.
Event 1: One day Dan was running late, and
missed the bus to school.
Event 2: Dan called his friend Pete, and
asked for a ride to school.
Example Output:
Dan's routine of taking the bus to
school(Event 0) >Results in> Dan taking
the bus to school(other property)
Dan takes the bus to school(other property)
>Causes/Enables> Dan to miss the bus (Event
1).
Dan missing the bus (Event 1)
>Causes/Enables> Dan to call his friend
Pete for a ride (Event 2)
Input:
Event 0: <S1>
Event 1: <S2>
Event 2: <S3>

Output:
[Output from ChatGPT]
```

Figure 3: The basic prompt for event causality

Separate Cause and Effect Dimensions 1 to 5 represent the causes of the event and Dimensions 6 to 10 represent the effects of the event, it seems natural to generate the causes and effects separately.

```
Identify and describe the causal
relationships among the events in
the narrative, highlighting how one event
leads to or influences another.
```

Figure 4: The prompt instruction generated by ChatGPT

The prompt instructions remain unchanged, but when generating dimensions 1 to 5, the causal statements of dimensions 6 to 10 are removed from the example outputs.

Different Definitions of Causality Note that in the above prompt, we do not formally define causality. Next, we experiment with 4 different definitions of causality by changing the first line of the prompt to the following:

- Multifactorial: Your job is to find all the causalities in a story using the multifactorial definition of causality: A causes B when, in combination with other factors, it is a necessary or sufficient condition for the occurrence of event B.

- Interventionist: Your job is to find all the causalities in a story using the interventionist definition of causality: A causes B when changing or intervening in the occurrence of A results in a corresponding change in the occurrence of B.

- Probabilistic: Your job is to find all the causalities in a story using the probabilistic definition of causality: A causes B when the likelihood or probability of B happening is significantly higher when A occurs compared to when A does not.

- Counterfactual: Your job is to find all the causalities in a story using the counterfactual definition of causality: A causes B if and only if when A does not happen, B will not happen.

ChatGPT Generated Instruction We input the examples into ChatGPT and asked ChatGPT to generate an instruction. To maximize reproducibility, we set the temperature to 0 when using the OpenAI API. We replace the instruction part of the prompt with instruction shown in Figure 4

Natural Language Form Output Keeping the instructions and example input of the prompt unchanged, change the format of the example output so that it is a grammatically correct sentence, see Figure 5.

```
Example Output:
Dan's routine of taking the bus to
school(Event 0) results in Dan taking the
bus to school(other property)
Dan takes the bus to school(other property)
enables Dan to miss the bus (Event 1).
Dan missing the bus (Event 1) causes Dan to
call his friend Pete for a ride (Event 2)
```

Figure 5: Constrain the output format to a natural language.

ChatGPT Generated Format Ask ChatGPT to generate free-form causality, and replace the output format in the prompt with the ChatGPT-generated format shown in Figure 6:

```
Example Output:
Original Event ID: 0
Event: Dan goes to school in the morning,
Dan takes the bus
Effect: Dan missed the bus to school
Original Event ID: 1
Event: Dan was running late, and missed the
bus to school
Cause: Dan takes the bus to school
Effect: Dan asks for a ride to school
Emotional Effect: Dan feels worried
Original Event ID: 2
Event: Dan calls his friend Pete and asks
for a ride to school
Motivation: Dan feels worried
Cause: Dan missed the bus to school
Effect: Pete gave Dan a ride to school
Emotional Effect: Dan feels relieved
```

Figure 6: The output format generated by ChatGPT

Curated Examples For the above prompts, we use 6 examples randomly chosen from GLUCOSE training set. Since GLUCOSE gives a quality score for each annotator, we pick examples annotated by high-quality annotators to create this prompt.

Causal Graph The prompt for generating causal graph is shown in Figure 1.

Event Chain Figure 7 shows the prompt for generating event causality in the form of event chains.

```
Here is a list of events from a story.
Trace the domino effect of events in the
story and explain how one event led to the
next.
Example Input:
Event 0: When Dan goes to school in the
morning, he has to take the bus.
Event 1: One day Dan was running late, and
missed the bus to school.
Event 2: Dan called his friend Pete, and
asked for a ride to school.
Example Output:
Chain 0: Event 0 -> Dan takes the bus to
school(other property) -> Event 1 -> Event
2
Input:
Event 0: <S1>
Event 1: <S2>
Event 2: <S3>

Output:
[Output from ChatGPT]
```

Figure 7: The prompt for event causality extraction in the form of event chains.

A.4 Ensemble of Prompts for GLUCOSE

While suitable for extracting event causality, the prompt shown in Figure 1 does not consider GLUCOSE dimensions and cannot extract causality between events and states. To show that ChatGPT can extract the causality of all ten GLUCOSE dimensions, we engineer 11 other prompts that generate dimensions in the causal statements. However, we note that different prompts perform well on different dimensions.

Therefore, we assemble the results from different prompts to achieve better results. Here, a simple voting scheme is adopted, a sentence is marked as having a causality of dimension d only when the causality is detected with more than n prompts, n is a predefined threshold set to 4. See Table 5 and Table 6 for results.

A.5 Evaluation

Before event causality extraction, we first divide the story text into a sequence of sentences with the NLTK sent_tokenizer: https://www.nltk.org/api/nltk.tokenize.html

For BLEU score, we calculate with the SacreBLEU implementation (Post, 2018), with equal weights up to 4-grams at corpus level on the three-reference test set.

We use the sentence-transformer implementation (Reimers and Gurevych, 2019b) to calculate

the BERTscore, using the average of token embeddings from the `bert-nli-mean-tokens` model without considering the "idf" weight of each token. The average of token embeddings is also used in calculating the BERT Similarity.

The results reported in Table 2 and 1 of the main paper and the results presented in this section are from a single run.

A.6 Experiment with different prompts

In Table 5, we show the results on ChatGPT using different prompts, we show the results average over 10 dims. In Table 6 shows the results averaged over dimensions 1 and 6.

	BLEU	BERTScore	BERT Similar.	F1
Multifactorial	31.78	57.74	52.75	44.33
Interventionist	31.70	57.27	52.10	43.94
Probabilistic	29.83	56.97	51.70	43.37
Counterfactual	31.51	49.92	42.51	39.32
No Explicit Definition	28.04	57.12	51.98	**46.60**
Causal Chain	18.28	46.10	38.23	36.28
Causal Graph	8.10	46.96	40.39	39.75
Natural Language Output	24.23	52.67	46.12	40.38
ChatGPT instruction	19.52	44.42	35.63	35.43
ChatGPT format	15.98	51.85	45.86	41.7
Curated Examples	24.68	57.22	52.50	40.14
Basic Prompt	20.51	50.32	43.74	38.42
Ensemble	**33.06**	**63.15**	**60.46**	**50.70**

Table 5: Results of different prompts on GPT 3.5, averaged over 10 dimensions.

	BLEU	BERTScore	BERT Similar.	F1
Multifactorial	35.95	60.45	55.82	52.85
Interventionist	36.11	59.50	54.24	52.28
Probabilistic	32.13	59.18	53.79	50.90
Counterfactual	**37.54**	54.98	48.89	49.12
No Explicit Definition	30.37	59.90	54.71	59.82
Causal Chain	23.29	49.63	42.01	43.71
Causal Graph	21.2	**75.33**	**80.89**	**60.75**
Natural Language Output	31.66	68.51	66.57	54.44
ChatGPT instruction	30.12	55.31	50.73	50.96
ChatGPT format	24.23	68.44	68.93	54.75
Curated Examples	33.97	69.74	69.40	55.34
Basic Prompt	30.20	65.96	65.61	54.11
Ensemble	35.29	72.71	73.32	60.58

Table 6: Results of different prompts on GPT 3.5, averaged over dimensions 1 and 6.

A.7 Separately Report Results of 10 Dimensions

Tables 7, 8, 9, 10 show the BLEU score, BERTScore, BERT Similarity and F1 score of each of the GLUCOSE dimensions.

```
Score the following storyline given the
beginning of the story with one to five
stars.
Where one star means "Nonsense",
two stars mean "The storyline has some
connections with the beginning, but is not
understandable",
three stars mean "The storyline has some
causal connections with the beginning and
is understandable",
four stars mean "The storyline is causally
consistent with the beginning and possibly
involves a few grammar mistakes",
and five stars mean "Perfect storyline with
causal connections and perfect grammar".

The beginning of the story: <prompt>
Storyline: <generated story>
Stars:
```

Figure 8: The story scoring prompt from Wang et al. (2023b) with the word "causal" inserted.

B Event Causality in Story Understanding

B.1 Open-ended Generated Story Evaluation

Story Evaluation The prompt for scoring the story is first introduced by Wang et al. (2023b), and we insert a single word 'causal' into it for exploratory experiments, shown in Figure 8.

Figure 9 demonstrates the prompt for the method "ChatGPT-causal-graph" on WP under the few-shot cross-domain setting, as introduced in §5

Correlation Computation We calculate average correlations at two aggregation levels: dataset level and writing prompt level.

Given a set of N writing prompt sentences and M generative language models. The story generated by the mth model using the nth writing prompt is denoted as $T_{n,m}$. The scoring for $T_{n,m}$ from ChatGPT or human workers are denoted as $S_L(T_{n,m})$ or $S_{human}(T_{n,m})$.

Dataset Level

$$\begin{aligned}\text{Corr}_\text{d} =& \rho([S_L(T_{1,1}), \ldots, S_L(T_{M,N})], \\ & [S_{human}(T_{1,1}), \ldots, S_{human}(T_{M,N})])\end{aligned} \tag{2}$$

Writing Prompt Level

$$\begin{aligned}\text{Corr}_\text{p} =& \frac{1}{N}\sum_{n=1}^{N}(\rho([S_L(T_{1,n}), \ldots, S_L(T_{M,n})], \\ & [S_{human}(T_{1,n}), \ldots, S_{human}(T_{M,n})]))\end{aligned} \tag{3}$$

```
Score the following storyline given the
beginning of the story with one to five
stars.
Where one star means "Nonsense",
two stars mean "The storyline has some
connections with the beginning, but is not
understandable",
three stars mean "The storyline has some
causal connections with the beginning and
is understandable",
four stars mean "The storyline is causally
consistent with the beginning and possibly
involves a few grammar mistakes",
and five stars mean "Perfect storyline with
causal connections and perfect grammar".

We also provide causal connections
analyzed by experts, where each event is
represented as a node, and the causal
connections between these nodes are listed.

Here are two examples:
Example1:
The beginning of the story: i was sitting
on the bench today.
Storyline: i heard my neighbor 's dogs
barking. i looked at his dog and realized
it was a Monday. the monday still runs
through monday. i called my neighbor to
let her know how i felt.
Event graph:
Edge 0: (Node 0 -> Node 1)
Edge 1: (Node 1 -> Node 2)
Edge 2: (Node 2 -> Node 3)
Edge 3: (Node 2 -> Node 4)
Stars: 1.4

Example2:
The beginning of the story: [FEMALE] is an
actress who just turned 19.
Storyline: she had a bad acting performance
in an upcoming movie. she was disappointed.
then, she decided to try something
different. she filmed herself playing in
her very own version.
Event graph:
Edge 0: (Node 0 -> Node 1)
Edge 1: (Node 1 -> Node 2)
Edge 2: (Node 2 -> Node 3)
Edge 3: (Node 3 -> Node 4)
Stars: 5
(End of examples)

The beginning of the story: <prompt>
Storyline: <generated story>
Event graph:
[list the event graph generated with
ChatGPT, here is just an example:]
Edge 0: (Node 0 -> Node 1)
Edge 1: (Node 1 -> Node 2)
Edge 2: (Node 2 -> Node 3)
Edge 3: (Node 3 -> Node 4)
Your score should reward stories with rich
causal chains and penalize those that lack
or have confusing causal chains.
```

Figure 9: The story prompt we propose.

OpenMEVA-WP sample
Prompt: "All children up to 14 years of age have vanished, and no more kids can be born. How does the last generation of mankind spends their remaining years until the last human dies?"
Generated: '"My name is Sarah and I'm 14 years old. I wish I could tell you what happened to my parents. I don't know how I'm going to tell you this, but it's only 9 years after I left my dad's house, and I'm having a lot of nightmares about this. They said that there weren't enough kids anymore, and even their children weren't enough. I feel as though I'm just sitting here, with a voice like all those other moms and dads that can't look at me. But I feel like I know how to feel, what else do they want from me. I can see how it was with our parents, their relationship seemed to be drifting apart, but they would always have this dialogue that had almost always turned on them. "Why aren't we moving in? We're broke." "We won't be moving, we're just leaving." "We won't be back there, we won't get back." "We won't. No one will love us." "And I will stay here, so will my Mom and Dad." But then there was one giant quiet thing to stand there. That silence was deafening."'
Human Evaluation: [4,4,4,4,5]

OpenMEVA-ROC sample
Prompt: "[FEMALE] is an actress who just turned 19."
Generated: "she had a bad acting performance in an upcoming movie. she was disappointed. then, she decided to try something different. she filmed herself playing in her very own version."
Human Evaluation: [5,5,5,5,5]

Figure 10: Example of OpenMEVA stories.

B.2 Video-Text Alignment

Training Following UniVL, we use S3D to extract 1 video feature per second. The video clips are trimmed or appended to 4 seconds. A video clip A is represented as 4 video feature vectors $\{v_1^A, \ldots, v_4^A\}$. The text chunks are trimmed or appended to 64 tokens. The video features and text tokens are then passed into the video and text encoders. Both encoders are 12-layer Transformers.

The model is trained on SyMoN with an initial learning rate of 5×10^{-5} and cosine learning rate decay. We use a batch size of 256 and train for 40 epochs. The first epoch applies linear warm-up of learning rates. SGD with momentum of 0.9 is used for optimization and a weight decay term of 0.5 is added for regularization. 60% of the text are masked at random.

The number of parameters within the models is 153,784,064. The model is trained for 2.8 days on 1 NVIDIA A6000 1015 GPU.

Evaluation In YMS, a text chunk may correspond to multiple video clips, whereas a video clip

may correspond to one or zero text chunks. During inference, we first calculated the pair-wise similarity between every video clip and text chunk. Then we calculate the global alignment with DTW (introduced momentarily). If the similarity between an aligned video clip and text chunk falls below a threshold, tuned on the validation set, the video clip is consideredto not match anything.

The YMS dataset consists of 94 movie summary videos in total. In the NeuMatch split, the test set consists of 15 videos and the validation set consists of 12 videos. In the SyMoN split, the test set and validation set contain 65 and 29 videos respectively.

The results in Table. 4 are from a single run.

Dynamic Time Wrapping DTW uses dynamic programming to find the best correspondence between two sequences based on distance (or similarity), the final alignment corresponds to the shortest distance or highest similarity.

We use DTW to align a sequence of video clips $V = (v_1, \ldots, v_N)$ and a sequence of text chunks $T = (t_1, \ldots, t_M)$. We first assume that v_1 is aligned to t_1, thus the cost of matching v_1 and t_1 is $c(1,1) = 0$, and the cost of match v_1 with $t_j (j \neq 1)$ is $c(1,j) = \infty$, and vice versa. Then, the minimal cost of aligning $(v_1, \ldots, v_i)$ with $(t_1, \ldots, t_j)$, can be calculated as:

$$\begin{aligned} c(i,j) = \min(&c(i-1,j) + d(i,j), \\ &c(i,j-1) + d(i,j), \\ &c(i-1,j-1) + d(i,j)) \end{aligned} \tag{4}$$

where $d(i,j)$ denotes the distance between v_i and t_j. Since we have the cosine similarity between each video-text pairs, the distance can be calculated as $d(i,j) = 1 - s(i,j)$, where $s(i,j)$ is the cosine similarity between v_i and t_j.

C Licensing Information

The Glucose dataset is licensed under the Creative Commons Attribution-NonCommercial 4.0 International Public License. The COPES dataset is licensed under the MIT License. The OpenMEVA dataset is from https://github.com/thu-coai/UNION. The SyMoN dataset is from https://github.com/insundaycathy/SYMON. The YMS dataset is from https://github.com/pelindogan/NeuMATCH/tree/master.

ChatGPT is under the GNU AFFERO GENERAL PUBLIC LICENSE Version 3. Llama-2 is licensed under LLAMA 2 COMMUNITY LICENSE AGREEMENT. Yi-34b-chat is licensed under Yi Series Models Community License Agreement Version: 2.1. Falcon is licensed under Apache License Version 2.0. The Union model is from https://github.com/thu-coai/UNION. The UniVL model is licensed under the MIT License.

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5	Dim 6	Dim 7	Dim 8	Dim 9	Dim 10
$GPT2_{large}$	28.34	18.42	22.30	40.72	1.74	29.49	31.58	32.26	**43.83**	**56.36**
$T5_{large}$	30.70	**23.20**	**32.10**	**45.50**	**9.30**	32.80	**39.60**	**44.10**	29.30	0.60
$ChatGPT_{causal\ graph}$	**35.29**	19.95	27.56	40.61	6.19	**35.30**	33.84	33.01	24.14	5.47

Table 7: BLEU Score on Glucose

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5	Dim 6	Dim 7	Dim 8	Dim 9	Dim 10
$GPT2_{large}$	50.63	54.79	62.74	75.63	**98.01**	53.58	61.94	85.06	**93.84**	**99.34**
$T5_{large}$	56.92	68.40	**76.85**	**78.77**	96.26	59.48	**73.10**	**89.94**	91.48	97.18
$ChatGPT_{causal\ graph}$	**61.82**	**77.83**	75.26	74.56	91.68	**62.78**	68.94	86.92	88.61	96.15

Table 8: BERTScore on Glucose

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5	Dim 6	Dim 7	Dim 8	Dim 9	Dim 10
$GPT2_{large}$	48.91	53.91	61.36	75.06	**97.97**	52.02	61.40	84.76	**93.63**	**99.36**
$T5_{large}$	55.38	67.81	**76.08**	**78.33**	96.19	58.02	**72.72**	**89.73**	91.26	97.13
$ChatGPT_{causal\ graph}$	**60.30**	**76.85**	74.27	73.87	91.53	**61.39**	68.48	86.63	88.26	96.06

Table 9: BERT Similarity on Glucose

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5	Dim 6	Dim 7	Dim 8	Dim 9	Dim 10
$GPT2_{large}$	58.28	44.35	46.63	62.78	0	60.68	**55.16**	47.27	**46.15**	**33.33**
$T5_{large}$	60.03	**45.59**	**52.40**	**64.05**	**14.29**	**63.25**	52.60	**54.26**	35.14	0
$ChatGPT_{causal\ graph}$	**60.36**	29.19	36.20	55.45	7.55	60.80	54.87	42.11	36.00	8.33

Table 10: F1 on Glucose