

# Large Language Models Fall Short: Understanding Complex Relationships in Detective Narratives

Runcong Zhao<sup>1\*</sup>, Qinglin Zhu<sup>1\*</sup>, Hainiu Xu<sup>1</sup>, Jiazheng Li<sup>1</sup>, Yuxiang Zhou<sup>1</sup>  
Yulan He<sup>1,2,3</sup>, Lin Gui<sup>1</sup>

<sup>1</sup>King’s College London, <sup>2</sup>University of Warwick, <sup>3</sup>The Alan Turing Institute  
{runcong.zhao, qinglin.1.zhu, hainiu.xu, jiazheng.li, yuxiang.zhou}@kcl.ac.uk  
{yulan.he, lin.1.gui}@kcl.ac.uk

## Abstract

Existing datasets for narrative understanding often fail to represent the complexity and uncertainty of relationships in real-life social scenarios. To address this gap, we introduce a new benchmark, *Conan*, designed for extracting and analysing intricate character relation graphs from detective narratives. Specifically, we designed hierarchical relationship categories and manually extracted and annotated role-oriented relationships from the perspectives of various characters, incorporating both public relationships known to most characters and secret ones known to only a few. Our experiments with advanced Large Language Models (LLMs) like GPT-3.5, GPT-4, and Llama2 reveal their limitations in inferencing complex relationships and handling longer narratives. The combination of the *Conan* dataset and our pipeline strategy is geared towards understanding the ability of LLMs to comprehend nuanced relational dynamics in narrative contexts.

## 1 Introduction

Tasks like multi-agent interaction (Park et al., 2023; Xu et al., 2023; Wang et al., 2023) and character-centric narrative understanding (Zhu et al., 2023) have recently gained significant attention. These tasks require a deeper understanding of complex relationships among multiple entities (Worth, 2004), thereby serving as a critical benchmark for assessing the reasoning capabilities of LLMs (Bubeck et al., 2023). Detective stories, where characters often adopt multiple identities or aliases that are revealed at various points, are the most appropriate testbed for assessing LLMs’ capability of deducing complex relationships. However, existing datasets designed for character-centric narrative understanding are either built on well-known stories that LLMs have been trained on (Brahman et al., 2021; Sang et al., 2022; Chen et al., 2023; Iyyer et al., 2016), or consist of simpler texts (Bamman et al., 2020; Stambach et al., 2022; Xu et al.,

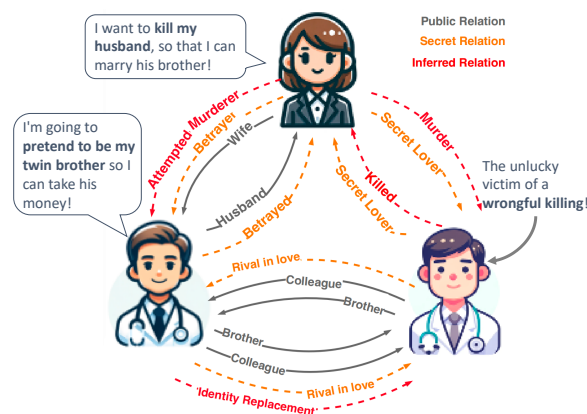


Figure 1: The example illustrates complex relationships of characters in narratives. Gray-colored relationships represent surface-level information, widely known to most characters. Orange-colored relationships, on the other hand, are secrets known to only one or very few individuals, often conflicting with the commonly known relationships; these are referred to as secret relationships. Red-colored relationships represent inferred information, meaning they are not explicitly stated in any character’s story but can be deduced by synthesising information from all characters collectively. LLMs struggle with such complex relationships in long narratives.

2022), such as children’s stories, where characters and their relationships are typically introduced when the characters first appear in the narrative (Zhao et al., 2023).

However, relationships between characters are often characterised by **incomplete and uncertain** information in reality. These references may involve descriptions from another character’s perspective, as in “A middle-aged man walked out of the main gate, with grey and white hair, and a slender build.”. Also, in real social scenarios, this is often not the case. For instance, as depicted in Figure 1, husband A may remain unaware that his brother B is the secret lover of his wife C and the biological father of his son D. Each individual may have differing interpretations of their relationships, with these perceptions potentially **conflicting** based on their own perspectives.

\*Equal contribution.

Misidentified relationships can greatly impact the core conflicts and plot of a story, as these complex relationships are often central to the narrative. Therefore, we have developed the benchmark, COntextual Narrative ANalysis (*Conan*) to understand complex relationships in detective narratives. We have also outlined the desired input-output format, identified three sub-tasks within this framework, and developed hierarchical relationship categories for evaluation, drawing insights from the field of social science. *Conan* is constructed to extract role-oriented relational graphs from detective narratives. It comprises detective narratives from various characters’ perspectives, along with annotated relationships that can be deduced from the narrative. As shown in Figure 1, it includes three types of relationships: (1) *Public Relations* that are known to most people; (2) *Secret Relations* that are known only to a few or even just one person and often conflicting with the widely known relationships; (3) *Inferred Relations*, which are not explicitly articulated in any single character’s story and must be deduced by combining information from multiple characters. This dataset can act as a benchmark to test the cognitive and inferential abilities of LLMs. Additionally, it holds the potential to improve LLM capabilities in areas such as narrative comprehension (Zhao et al., 2023), simulation agents (Park et al., 2023), and game agents (Xu et al., 2023; Wang et al., 2023).

Our quantitative experiments and qualitative analysis show our benchmark is challenging for cutting-edge LLMs (GPT-3.5, GPT-4, and Llama2). Our findings reveal that these models struggle primarily due to two reasons: (1) the complexity of information that necessitates inferential reasoning; (2) the length of narratives which demand efficient key information extraction. Additionally, our evaluation across three distinct strategies has pinpointed the most effective strategy for various scenarios, while also highlighting those that are inefficient and underperforming. Given the high operational costs of LLMs, these insights are valuable for saving time and resources in future research<sup>1</sup>.

In summary, we have made the following contributions:

- We have constructed and annotated the *Conan* dataset for the task, designed to evaluate and understand the inference capacity of LLMs.
- We have designed hierarchical relationship categories for evaluation, built on insights

<sup>1</sup>Our code and dataset are available at <https://github.com/BLPXSPG/Conan>

from social science and necessary empirical observations during the process of manual annotation and LLMs’ evaluation.

- To assess the performance of advanced LLMs, namely, GPT-3.5, GPT-4, and Llama2, we have conducted evaluations with three different strategies using our benchmark and have identified the most effective strategy for various scenarios.
- Our findings reveal that LLMs significantly underperform humans on *Conan*. We have carried out a series of experiments to validate our hypotheses regarding the causes of LLMs’ failure.

## 2 Task Definition

As shown in Figure 2, the input narrative  $N$  consists of background stories for  $k$  characters, represented as  $c_i \in C_k = \{c_1, c_2, \dots, c_k\}$ . The background story of each character  $N_{c_i}$  is crafted solely from the perspective of that particular character  $c_i$ , with all relationships and events in the story framed based on  $c_i$ ’s perception. Unlike the conventional narrative structure that revolves around a single protagonist, in our setup, the complete narrative  $N$  emerges as a collaborative novel, composed by intertwining the perspectives of these  $k$  characters, i.e.,  $N = \cup_{c_i \in C_k} N_{c_i}$ .

As depicted in Figure 1, for the same set of character relationships, there can be instances where some characters remain completely unaware of these connections, or where the perceptions of different characters are in direct contradiction. Moreover, the narrative may introduce additional characters beyond these  $k$  individuals. Some of these extra characters might play peripheral roles, while others could hold pivotal significance, such as the victim. Therefore, we define all characters that appeared in the story as  $C = \{c_1, c_2, \dots, c_K\}$ , where  $C_k \subseteq C$ . Hence, by defining  $R_C$  as the relationships among this group of characters  $C$ , we aim to perform the following three sub-tasks:

1. **Character Extraction.** Identify all characters in the given story, which can be  $N_{c_i}$  or  $N$ .
2. **Entity Linking.** Recognise character relationships from the perspective of a specific character  $c_i$ . This equals  $R_C|N_{c_i}$  formally.
3. **Relation Deduction.** Infer the actual relationships between characters by considering the collection of all character-centric narratives. Formally, this corresponds to  $R_C|N$ .

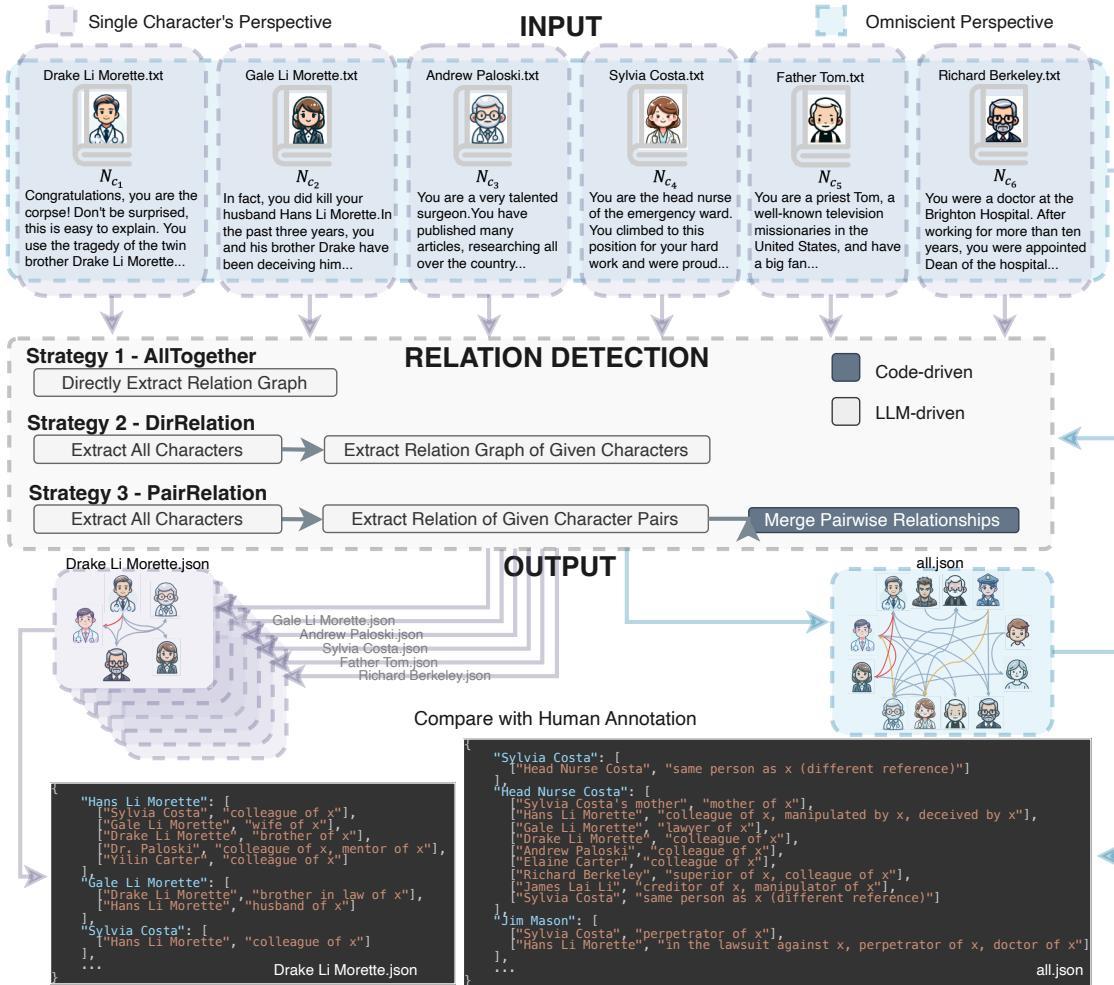


Figure 2: Input-Output Format and Benchmark Relation Detection Strategies. The input narrative consists of  $k$  background stories  $N_{c_i}$  that are uniquely created from the perspective of the character  $c_i$ . For each narrative, we have labelled it  $k + 1$  times, including  $k$  relationship graphs from **each character’s perspective** and one relationship graph from **omniscient perspective**. Our objective is to extract all characters from the given story, including those beyond the initial  $k$  characters, subsequently detect the relationships among all the extracted characters, even when they involve false or multiple identities, and finally uncover conflicting relationships to deduce the genuine nature of these relationships.

To understand the relationships of characters within detective narratives, the challenges mainly manifested in three types, corresponding to the aforementioned three sub-tasks: First, in detective narratives, a character might have different identities or aliases, and these identities may surface at different points in the story. Secondly, LLMs struggle with basic inference based on existing information due to a phenomenon known as reversal curse (Berglund et al., 2023). For example, when we know that  $A$  is  $B$ ’s father, it does not necessarily imply that  $B$  is  $A$ ’s child according to these models. Finally, the third challenge involves drawing inferences from the perspectives of multiple characters,

which sometimes yield conflicting or inconsistent information. For instance, from  $A$ ’s perspective,  $B$  might be regarded as his mother’s friend. However, from  $B$ ’s perspective, he has been the lover of  $A$ ’s mother and is actually  $A$ ’s biological father.

### 3 Dataset Construction

To investigate the capabilities of LLMs in comprehending detective narratives, we developed the first open benchmark dataset for this task. The construction process includes data collection and processing, and annotation of characters and their relationships.

### 3.1 Data Collection and Processing

**Collection** We gathered the original narratives from the popular Chinese murder mystery games, where each player has a predefined role description. We utilised the background stories of these playable characters to construct our dataset for the following reasons. Firstly, they feature complex character relationships that often pose challenges for LLMs. Secondly, these narratives offer more realistic scenarios compared to existing synthetic benchmarks. Lastly, while these stories are considerably longer than current benchmarks, they still maintain a level of conciseness suitable for human reading. These equilibria ensure that the workload for human annotators, in terms of reading and comprehending these narratives, remains manageable.

**Filtering** We selected 100 high quality narratives from a pool of 2,135, following the filter process described in Appendix A. Then we employed Adobe to extract text from scanned narratives and leveraged the capabilities of the ChatGPT model GPT-3.5 to rewrite and refine the text.

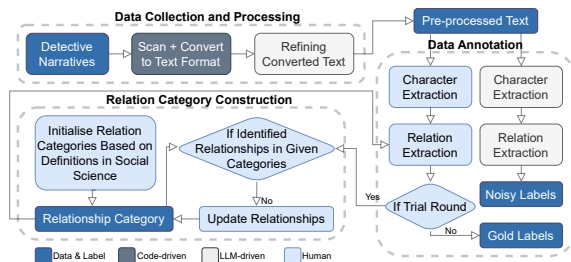


Figure 3: Dataset Construction.

### 3.2 Data Annotation and Evaluation

**Relation Category Construction** We started with relationship scheme extraction. Based on a relationship scheme of 53 casual relationships (Berscheid, 1994; DeVito, 2015), we carried out a trial run using 10 detective novels to expand upon these relationships. The resulting scheme contains 5 primary relationship categories, namely *Romantic*, *Family*, *Allies*, *Adversarial*, and *Business*, which expand to 163 fine-grained relationships denoted as  $R$ .

During this process, we found that the boundaries between some categories were not clearly defined, such as “mother”, “mother-in-law”, and “adoptive mother”. In contrast, mistaking “mother” for “stranger” is a more significant error than confusing these three specific relationships. To mitigate such ambiguities, we consolidated similar

categories into broader ones until no further merging was feasible. For instance, we merged “lover”, “boyfriend”, and “girlfriend” into a single category: “romantic relationships”.

Our goal for constructing this hierarchical category was to better evaluate LLM capabilities. For instance, recognising “biological daughter” as “child” at a higher level would not be considered incorrect. However, a singular category could not achieve this balance. We reported F1 scores for each level. Consequently, we established a hierarchical structure of relationship categories, comprising 5 top-level categories, 54 intermediate categories, and 163 detailed categories. Such a well-defined, detective-oriented relationship scheme helps to reduce the potential subjectivity during the annotation process.

Evaluating free-form relationships generated by LLMs can be extremely challenging, requiring either human evaluation or using LLMs for auto-evaluation. However, human evaluation is costly and subjective, while evaluation by LLMs has been shown to have notable disparities compared to human judgement. Therefore, it is better to annotate the relationships between characters, providing a consistent basis for evaluating this dataset.

**Labelling** We recruited four annotators, all of whom were fans of detective narratives, and conducted training sessions for them. Our complete annotation pipeline involves three tasks: (1). *Character Extraction*: Annotators read given detective novels to identify all characters appearing in the narrative. (2). *Entity Linking*: annotators closely examine the story from the perspective of a single character, and extract both explicit and implicit relationships. These relationships are structured as triplets in the format of  $(c_i, c_j, r_{i,j})$ , where  $r_{i,j}$  signifies the relationship between characters  $c_i$  and  $c_j$ . We specifically account for scenarios where  $i = j$  to handle the common detective novel plot where a single character has multiple identities (refer to Appendix C for details). (3). *Conflict Detection and Relationship Refinement*: As our dataset comprises narratives from multiple character perspectives, annotators are tasked with considering all available information to form a unified relationship graph from the  $k$  individual detective narratives. This step often involves resolving conflicting information that arises from the imperfect knowledge shared among characters and refining the final relationships through inference.

Consequently, for each story, our team of experts

would generate  $k$  distinct relationship graphs at the individual character level, and one consolidated graph that merges these individual perspectives.

**Inter-annotator Agreement** Following the agreement measure for triplets in previous works (Girju et al., 2007; Gurulingappa et al., 2012), we used F1-score as a criterion to measure inter-annotator agreement (IAA). We selected one challenging narrative and asked three annotators to annotate it following the annotation guidelines. This step served the purpose of evaluating the inter-annotator agreement and ensuring the quality of the annotations. The detailed annotation guideline and the calculation of the inter-annotator agreement score are in Appendix B. We calculate IAA for three steps in our task: 1). identifying characters,  $IAA(c_i)$ ; 2). determining if there are relationships between a given character pair,  $IAA(c_i, c_j)$ ; and 3). classifying the relationships between two characters when there are relationships between them,  $IAA(c_i, c_j, r_k)$ .  $IAA(c_i, c_j, r_k)_1$  assesses whether annotators agree on at least one relationship for each character pair.  $IAA(c_i, c_j, r_k)_{all}$  measures agreement on all relationship triplets. As shown in Table 1, annotators demonstrate high agreement in both character and relationship extraction. However, agreement decreases as the task becomes more subjective. For instance, identifying characters present in the narrative is more straightforward, but “*father’s friend of x*” might be labelled as “*acquaintance of x*” by one annotator and as unrelated by another.

Annotator	Character	Relation		
	$c_i$	$(c_i, c_j)$	$(c_i, c_j, r_k)_1$	$(c_i, c_j, r_k)_{all}$
1 & 2	0.978	0.894	0.962	0.873
1 & 3	0.978	0.800	0.916	0.738
2 & 3	0.966	0.756	0.907	0.736
Average	0.974	0.817	0.928	0.782

Table 1: Inter-annotator Agreement Score.

### 3.3 Data Statistics

The detailed statistics are shown in Table 2. We collected a total of 2,135 narratives, but the majority of them exhibited low quality. After manual selection, we identified 100 high-quality narratives. We first generated annotations automatically using GPT-4, categorised under the column *All* in Table 2. Our dataset has an average of 27,695 tokens per narrative, making it significantly longer and more complex than earlier datasets, particularly consid-

ering its detective-themed content.

We also recruited human experts to annotate 25 narratives, resulting in a total of 8,254 annotations. They were compensated at an hourly rate of \$31.92, with each narrative estimated to take about 10 hours to complete. However, due to quality concerns, we removed one annotated narrative from our final dataset as it ambiguously described the relationships between characters, making the annotation highly subjective. Consequently, our final dataset consists of 24 annotated narratives encompassing 7,951 relationships.

Given the quality concerns, we did not use GPT-4’s annotations for evaluation purposes. All experimental results reported in this paper were assessed using data annotated by human experts. However, the GPT-4 annotations could serve as an initial foundation for further annotation of this dataset in future research. The original narrative is in Chinese, but we also provide an English-translated version and conduct experiments on it, detailed in Appendix E.3.

## 4 Experiments

### 4.1 Baselines

We conducted our experiments using GPT-3.5, GPT-4 and Llama2-chat. We accessed GPT-3.5 and GPT-4 through the Azure API <sup>2</sup> with model version gpt-35-turbo-16k 0613(Default) and gpt-4-turbo 1106-Preview. The experiments were carried out with the default parameters of the interface, between October and December 2023. For Llama2, we used the HuggingFace model Llama-2-70b-chat-hf <sup>3</sup>. Inferences on this model were run directly using greedy decoding at a temperature setting of 0.

Dataset	All	Human
#Narratives	100	24
#Background stories	640	149
Avg. #Character per narrative	18.72	18.84
w/ narrative	7.40	7.21
w/o narrative	11.32	11.63
#Relationships	27,444	7,951
Avg. #Token per character story	4,327	4,539
Avg. #Token per narrative	27,695	28,182

Table 2: Dataset Statistics.

Given that the length of some narratives exceeds the input limit of LLMs, we segment the original

<sup>2</sup><https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/models>

<sup>3</sup><https://huggingface.co/meta-llama/Llama-2-70b-chat-hf>

narrative input into multiple parts. The relation graph is initially extracted from the first segment. Subsequently, this pre-established relation graph, coupled with the ensuing narrative segment, is used as input to instruct the model to update and refine the relationship graph. This iterative process allows for comprehensive relationship mapping despite the constraints of LLM input limitations.

We evaluate relation detection using three strategies: First, *AllTogether*, where we ask the model to directly output a relationship graph, including characters and their relationships. Second, *DirRelation*, to distinguish and minimise the influence of errors occurring in two separate stages, character extraction, and relationship extraction, we extract the characters from the narrative first, then utilise it alongside the narrative script to generate the relationship network. Third, *PairRelation*, where we initially extract characters, inquire about the relationship of each character pair, and finally, aggregate the results, merging them into a comprehensive relationship map. This strategy targets LLM’s problem of ignoring relationships in narratives, especially with long narratives.

The total running costs for using GPT-3.5 and GPT-4 in our experiments are approximately \$400 and \$2000, respectively. In addition, the running time for Llama2 in our experiments totalled 490 hours, utilising two 80G A100 graphics cards.

## 4.2 Corruption Rate

For generative language models, there is always a possibility that the output format may not follow the given instructions. Even though these models, are trained to follow specific formats like JSON, complex tasks such as *Conan* still demonstrate instances where models fail to comply with the provided format guidelines. Therefore, we classify cases failing to produce the desired format as corrupted cases. We calculate the corruption rate for various models and strategies as  $\frac{n_c}{n_{all}}$ , where  $n_c$  is the number of corrupted relationships and  $n_{all}$  is the number of total generated relationships. For each model, we add a post-processing step to remap the relationships not in the specified categories into desired categories. However, some outputs remain corrupted even after recategorising.

As indicated in Table 3, there was a significant reduction in the corruption rate following post-processing. However, our primary focus is the F1-score after self-correction. This step proved beneficial for Llama2 and GPT-3.5, but counterproductive for the best-performing model, GPT-

Strategy	Llama2		GPT-3.5		GPT-4	
	before	after	before	after	before	after
<i>Corruption Rate</i>						
AllTogether	0.445	0.316	0.310	0.032	0.143	<b>0.006</b>
DirRelation	0.448	0.286	0.266	0.022	0.164	<b>0.009</b>
PairRelation	0.563	0.336	0.160	<b>0.021</b>	0.180	0.032
<i>F1-score</i>						
AllTogether	0.030	<b>0.035</b>	0.028	<b>0.031</b>	<b>0.125</b>	0.119
DirRelation	0.050	<b>0.057</b>	0.052	<b>0.053</b>	<b>0.110</b>	0.103
PairRelation	0.020	<b>0.021</b>	0.025	<b>0.025</b>	<b>0.027</b>	0.026

Table 3: Comparison of corruption rate and F1-score before and after self-correction. The corruption rate is the percentage of output relationships that fail to comply with the provided format guidelines before and after self-correction.

4. Consequently, we chose to implement the self-correction step only for Llama2 and GPT-3.5 in our subsequent experiments.

## 4.3 Character Extraction

We hypothesised that character extraction would be a simple task for LLMs, and planned to use it primarily for relationship extraction. Surprisingly, LLMs struggled with this initial step. These findings contribute to the research community by highlighting unanticipated challenges in character extraction for LLMs that may not be immediately apparent. Besides the F1-score of character extraction for three baseline relation detection strategies, we also report the results that we ask LLMs to directly extract characters from the given narrative, which were used in *DirRelation* and *PairRelation*. We denote it as *DirCharacter* in Table 4.

Strategy	Llama2			GPT-3.5			GPT-4		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
<i>Information from Single Character’s Perspective</i>									
DirCharacter	0.636	0.605	0.620	0.664	0.697	0.680	0.613	0.782	<b>0.687</b>
AllTogether	0.665	0.489	0.564	0.229	0.373	0.283	0.755	0.789	<b>0.772</b>
DirRelation	0.725	0.540	0.619	0.644	0.678	0.660	0.658	0.768	0.709
PairRelation	0.575	0.615	0.594	0.643	0.714	0.677	0.612	0.780	0.686
<i>Information from All Characters’ Perspectives</i>									
DirCharacter	0.449	0.703	0.548	0.522	0.836	<b>0.643</b>	0.444	0.870	0.588
AllTogether	0.671	0.261	0.376	0.240	0.238	0.239	0.736	0.455	0.562
DirRelation	0.730	0.332	0.457	0.502	0.517	0.509	0.681	0.442	0.536
PairRelation	0.371	0.598	0.458	0.478	0.844	<b>0.610</b>	0.430	0.869	0.575

Table 4: Character extraction results. Note that simply instructing LLMs to extract relationships of given characters, as in our “*DirRelation*” and “*PairRelation*” approaches, doesn’t ensure consistent compliance with the instruction (the characters in the output could be different from what was given). Therefore, in addition to the “*DirCharacter*” approach, which directly asks LLMs to extract characters, we also evaluate the characters in the final output of the three strategies.

We can see that GPT-4 performs the best when

```

{
  "Gale Li Morette": [
    ["Hans Li Morette", "murderer of X", "victim of X", "husband of X"], "betrayed by x",
    ["Drake Li Morette", "husband of X", "extramarital affair with X", "lover of X"], "brother-in-law of x",
    ["Sylvia Costa", "customer of X", "helped by X"],
    ["Elaine Carter", "acquaintance of X"],
    ["James Riley", "Helper of X, Customer of X"]
  ],
  "Hans Li Morette": [
    ["Gale Li Morette", "victim of X", "wife of X", "husband of X", "murderer of X"], "betrayed of x",
    ["Drake Li Morette", "same person as x (different identity)", "X", "sibling of X"],
    ["Jim Mason", "in the lawsuit against x, victim of x, patient of x"],
    ["Sylvia Costa", "colleague of X"]
  ],
  "Drake Li Morette": [
    ["Gale Li Morette", "husband of X", "extramarital affair with X", "lover of X"], "sister-in-law of x",
    ["Hans Li Morette", "same person as x (different identity)", "X", "sibling of X", "deceived by X"]
  ],
  "Sylvia Costa": [
    ["Gale Li Morette", "customer of X", "helped by X"],
    ["Hans Li Morette", "Colleague of X"],
    ["Jim Mason": ["Hans Li Morette", "in the lawsuit against x, perpetrator of x, doctor of x"],
    ["James Riley": ["Gale Li Morette", "Helped by x, lawyer of x"]]]
  ]
}

```

Figure 4: Error Analysis. An example output for “Gale Li Morett.txt” using GPT-4 and “AllTogether”, the best-performing combination. Output relationships are labeled as correct (✓) or incorrect (x). The three main error types are: relationships identified by LLM but not in the narrative, relationships in the narrative but not identified by LLM, and relationships with incorrect direction (e.g., LLM outputs Gale as Hans’ husband instead of wife).

extracting information from a single character’s perspective. However, its performance drops significantly when extracting information based on all characters’ perspectives, a more pronounced drop than observed with Llama2 and GPT-3.5. Upon comparing extracted characters based on all characters’ information using different LLMs, we noted that GPT-4 tends to extract more details, averaging 31.92 characters per narrative. In contrast, Llama2 extracted 25.54 characters, and GPT-3.5 extracted 26.08 characters, which are significantly less than the GPT-4’s extraction. Consequently, these results demonstrate higher recall and precision for GPT-4 with shorter inputs, in comparison to the aforementioned models. Yet, for longer narratives, GPT-4’s precision suffers due to issues like character duplication (e.g., “Costa”, “Head Nurse Costa”, and “Sylvia Costa” all referring to the same character) and misclassification of entities such as organisation names.

Errors can include personal pronouns such as “You”, locations such as “Pavilion”, organisations such as “Xiao”, and objects like “Casket”. Additionally, hallucinations and low Recall also harm the performance. Common error examples can be found in Appendix E.1.

#### 4.4 Relation Extraction

After removing the corrupted cases, we calculate the F1-score based on the derived relationship triples. These triples consist of character  $i$ , character  $j$ , and the relationships between them, represented as  $(c_i, c_j, r_{i,j})$ . Here we define precision as  $\frac{n_p}{n_g}$ , where  $n_g$  is the number of generated triples after removing corrupted ones,  $n_p$  is the number of correct relationship triples among generated ones;

and recall as  $\frac{n_r}{n_l}$ , where  $n_l$  is the number of labelled triples,  $n_t$  is the number of matched triples among labelled ones.

Strategy	Llama2			GPT-3.5			GPT-4		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
<i>Information from Single Character’s Perspective</i>									
AllTogether	0.129	0.054	0.076	0.056	0.041	0.047	0.283	0.269	<b>0.276</b>
DirRelation	0.160	0.085	0.111	0.119	0.076	0.093	0.219	0.267	0.240
PairRelation	0.025	0.089	0.039	0.029	0.099	0.045	0.047	0.258	0.080
<i>Information from All Characters’ Perspectives</i>									
AllTogether	0.121	0.020	0.035	0.064	0.020	0.031	0.267	0.082	<b>0.125</b>
DirRelation	0.203	0.033	0.057	0.092	0.037	0.053	0.202	0.075	0.110
PairRelation	0.012	0.092	0.021	0.014	0.132	0.025	0.014	0.238	0.027

Table 5: F1-score of Relation Extraction.

As illustrated in Table 5, it is evident that both Llama2 and GPT-3.5 struggle significantly to extract relationships from long narratives. In comparison, GPT-4 demonstrates considerably better performance, although there remains a substantial gap compared to human understanding. As shown in Figure 4, even extracting relationships from a single character’s perspective is challenging. As anticipated, extracting relationships based on information from all characters’ perspectives is a more challenging task compared to extracting relationships based on information from a single character’s perspective. This increased difficulty arises due to two factors:

**More Complicated Information** Narratives from a single character’s perspective typically don’t contain self-contradictory content. This is because the information is based on that character’s perception, which often tends to be consistent and self-explanatory. Therefore, LLMs can just extract what is stated in the text. In contrast, information from all characters includes secrets, misunderstandings, lies generated for self-interest or specific goals, and

even delusions caused by illnesses. Deriving accurate character relationships from these potentially repetitive or contradictory pieces of information is naturally more complex. LLMs must navigate through various narrative layers, distinguishing between truth, deception, and perception to accurately infer relationships.

To validate our assumption, we also calculate the accuracy of the relationships that are inconsistent across different characters’ perspectives, which we identify as secret or inferred relations, as illustrated in Figure 1. We can see that the accuracy of those complicated relationships is lower compared to all labelled relationships in Table 6.

Strategy	Llama2		GPT-3.5		GPT-4	
	all	complex	all	complex	all	complex
AllTogether	0.020	0.012	0.020	0.017	0.082	0.072
DirRelation	0.033	0.012	0.037	0.028	0.075	0.064
PairRelation	0.092	0.045	0.132	0.081	0.238	0.157

Table 6: Comparison with Complex Relationships. The accuracy of all relationships and only the complex relationships.

**Longer Narrative** One reason we suspect for the low F1 scores is the “lost in the middle” phenomenon observed in long narratives, which has also been noticed earlier (Liu et al., 2023), where LLMs tend to put more attention on the beginning and the end of long inputs, often ignoring information in the middle. Consequently, additional information can enhance judgement for humans, however, when all characters’ information is combined as input for LLMs, this extra data does not lead to improved results.

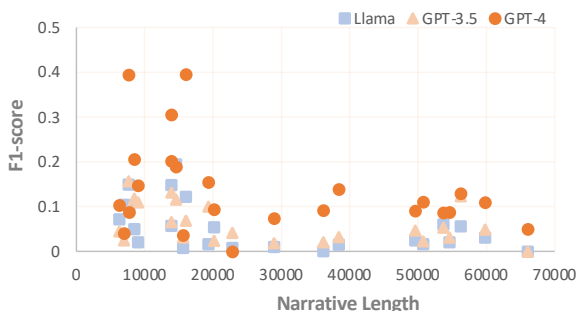


Figure 5: F1-score against the length of given narrative.

To validate our assumption, we plotted the F1-scores for each narrative against their respective lengths. The results suggest that longer narratives tend to yield lower F1-scores. However, shorter narratives do not guarantee higher F1-scores, as the results also heavily depend on the complexity

of each narrative. Therefore, narratives with short lengths but complicated relationships can exhibit lower F1-scores due to their inherent difficulty.

#### 4.5 Ablation Studies

**Impact of Character Extraction** To investigate the impact of character list quality on relationship extraction outcomes, we assessed the performance using both a gold standard and a model-generated noisy character list, as detailed in Table 7. Results show a significant performance increase with the provided characters. Table 5 reveals that while both Llama2 and GPT-3.5 showed improved results with DirRelation, GPT-4 performed better using AllTogether. Given GPT-4 did not do well in character extraction, its performance sets a ceiling for the efficacy of DirRelation and PairRelation, as their outcomes hinge on the quality of separately extracted characters.

Strategy	Llama2		GPT-3.5		GPT-4	
	gold	noisy	gold	noisy	gold	noisy
<i>Information from Single Character’s Perspective</i>						
DirRelation	0.124	0.111	0.123	0.093	<b>0.315</b>	0.240
PairRelation	0.071	0.039	0.083	0.045	0.167	0.080
<i>Information from All Characters’ Perspectives</i>						
DirRelation	0.053	0.057	0.064	0.053	<b>0.171</b>	0.110
PairRelation	0.046	0.021	0.055	0.025	0.113	0.027

Table 7: Impact Assessment of Character Extraction. We evaluated the impact on relationship extraction’s F1-score using both the gold standard character list and the model-generated noisy character list.

**Impact of Strategies** To investigate the impact of various relation detection strategies, we compared the approaches previously discussed. We noticed that directly asking models to extract all relationships resulted in low recall. We also found that when inquiring complex relationships between the given two characters, they perform better. Therefore, a straightforward solution was to inquire about the relationships between each pair of characters, which we termed as the PairRelation strategy. While this approach did increase recall, it also significantly amplified hallucinations. For instance, when the relationship between characters  $a$  and  $b$  wasn’t explicitly mentioned, the model often fabricated one. Additionally, this method is more costly and time-consuming: while originally we needed to calculate it at a complexity of  $O(k)$ , it now escalates to  $O(k^3)$ , where  $k$  is the number of characters. This translates to a hundredfold increase in cost for narratives with 10 characters.



Consequently, we infer that PairRelation is the least effective strategy. For GPT-4, when a reliable list of characters is available, DirRelation should be employed. In scenarios where a high-quality character list cannot be ensured, the AllTogether approach is preferable. For Llama2 and GPT-3.5, DirRelation is always the better strategy.

## 5 Related Works

**Character Extraction** Named Entity Recognition (NER) is a longstanding challenge in NLP (Wu et al., 2020; Aly et al., 2021). It involves locating and classifying named entities present in unstructured text. Character extraction is a specific area within NER, focusing on identifying characters involved in a given narrative (He et al., 2013; Bamman et al., 2020; Sang et al., 2022). In this process, models often confuse the targeted entities (characters) with other types of entities, such as organisations, items, locations, and so on. Another challenge is to merge all expressions in a given text that refer to the same entity, which is known as coreference resolution (Chen et al., 2017; Bohnet et al., 2022). This task becomes even more challenging when dealing with long-distance mentions (Massey et al., 2015; Yu et al., 2023a). In detective narratives, misidentified characters, entwined relationships, and secrets hidden by characters can further complicate this task (Zhao et al., 2023).

**Character-centric Information Extraction** Depending on various motivations, character-centric information extraction such as identifying motivations and emotional reactions (Rashkin et al., 2018; Kim and Klinger, 2019), roles (Stammach et al., 2022), appearance (Zhao et al., 2023), or personalities (Flekova and Gurevych, 2015; Yu et al., 2023b) can be conducted from the perspectives of these identified characters. Relation extraction plays a key role in character-centric information extraction (Chang et al., 2009; Labatut and Bost, 2019). Previous research exploring the narrative comprehension and inferential abilities of LLMs has typically relied on question answering (Yang and Choi, 2019; Xu et al., 2022; Gandhi et al., 2023), which may include questions about relationships but do not cover all aspects of character relationships. Alternatively, some research has been conducted on extracting relationships directly from sentences (You and Goldwasser, 2020; Mellace et al., 2020), typically focusing on verbs, such as *say*, *smile*, *look*. Relation extraction is frequently integrated with other character-centric tasks like dialogue genera-

tion (Chen et al., 2023), summarisation (Brahman et al., 2021), and tracking the evolution of relationships between characters over time (Iyyer et al., 2016).

## 6 Conclusion and Future Directions

This research introduces a task for LLMs to comprehend complex character relationships in detective narratives, utilising our newly created *Conan* dataset. This dataset highlights the current challenges for LLMs and aims to improve their ability in narrative contexts.

**Challenges** (1). *Enhancing Inference Capabilities of LLMs*. Our research reveals that LLMs face challenges in deciphering complex relationships, particularly when the input narrative contains conflicting information. To address this, recent advancements (Wei et al., 2022; Yao et al., 2023; Wang et al., 2024) can be leveraged to augment the inference capabilities of LLMs. (2). *Optimising Key Information Identification*. We observed that LLMs struggle to pinpoint key information in lengthy inputs. Therefore, employing methods like cosine similarity (Park et al., 2023) or Retrieval-Augmented Generation (RAG)<sup>4</sup> to help LLMs retrieve and focus on the most relevant or crucial information could be beneficial.

**Applications** (1). *Enhancing Narrative Understanding*. Our work can be used to analyse complex narratives in literature, films, and video games. It helps in understanding character dynamics and plot development. (2). *Interactive Agents*. AI-driven agents are widely used in many sectors, including chatbots that cater for both professional and emotional needs (Qian et al., 2023; Tu et al., 2023), interactive game development (Gao and Emami, 2023; Zhao et al., 2023), and digital life simulation (Cai et al., 2023). Understanding the relationships between characters in user inputs can enhance conversation quality, making these systems more empathetic and context-aware. (3). *Theory of Mind*. Our dataset, built on various characters' perspectives, naturally includes insights into how characters perceive relationships and how they think others perceive them (Premack and Woodruff, 1978). This makes it well-suitable for theory of mind tasks (Wimmer and Perner, 1983; Onishi and Baillargeon, 2005).

<sup>4</sup><https://github.com/weaviate/Verba>

## Limitations

We identify some of the limitations below:

**Annotator Influence** Despite our efforts to design clear annotation guidelines through trial rounds, individual interpretations of relationships in *Conan* may still differ. This is because annotators’ perceptions can influence how they see character dynamics, leading to some variations in how relationships are categorized.

**Limited Annotation Scope** Due to the cost of manual annotation, we could not annotate all 100 narratives with human labelers. To address this, we employed the best-performing model and strategy for annotation, but this may not fully capture all the relationships in the given narratives.

**Limited Evaluation of LLMs** Our experiments were limited to a single run of each LLM, GPT-3.5, GPT-4, and LLaMa. They represent the cutting-edge models at the time. But due to the high cost, we could not run more extensive evaluations across a broader range of popular LLMs, potentially missing insights from newer or less-known models.

## Ethics Statement

To respect copyright restrictions on the original Murder Mystery Game content, we follow established practices used in previous research (Fermann et al., 2018; Chen et al., 2023). Those interested in using the dataset will need to obtain the original game narratives themselves. But we are sharing the content for which we hold the copyright, including annotations we created for the game’s characters and their relationships; and content generated by LLMs based on those relationships. Additionally, we offer the code for preprocessing the original narratives to ensure others can replicate our experimental setup.

Please be aware that the detective narrative dataset may contain descriptions of violence, including violent events, actions, or characters. This content is provided solely for academic research and narrative analysis purposes. It is not intended to promote violence in any way. Users should be aware of potential exposure to violent content and use the dataset professionally and responsibly. This dataset is unsuitable for minors or those sensitive to violence.

## Acknowledgements

This work was supported in part by the UK Engineering and Physical Sciences Research Council

(EPSRC) through a New Horizons grant (grant no. EP/X019063/1) and a Turing AI Fellowship (grant no. EP/V020579/1, EP/V020579/2).

## References

- Rami Aly, Andreas Vlachos, and Ryan McDonald. 2021. [Leveraging type descriptions for zero-shot named entity recognition and classification](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, page 1516–1528, Online. Association for Computational Linguistics.
- David Bamman, Olivia Lewke, and Anya Mansoor. 2020. [An annotated dataset of coreference in English literature](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 44–54, Marseille, France. European Language Resources Association.
- Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. 2023. [The reversal curse: Lms trained on "a is b" fail to learn "b is a"](#). *CoRR*, abs/2309.12288.
- Ellen Berscheid. 1994. [Interpersonal relationships](#). *Annual Review of Psychology*, 45(1):79–129.
- Bernd Bohnet, Chris Alberti, and Michael Collins. 2022. [Coreference resolution through a seq2seq transition-based system](#). *CoRR*, abs/2211.12142.
- Faeze Brahman, Meng Huang, Oyvind Tafjord, Chao Zhao, Mrinmaya Sachan, and Snigdha Chaturvedi. 2021. [“let your characters tell their story”: A dataset for character-centric narrative understanding](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1734–1752, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott M. Lundberg, Harsha Nori, Hamid Palangi, Marco Túlio Ribeiro, and Yi Zhang. 2023. [Sparks of artificial general intelligence: Early experiments with GPT-4](#). *CoRR*, abs/2303.12712.
- Zhongang Cai, Jianping Jiang, Zhongfei Qing, Xinying Guo, Mingyuan Zhang, Zhengyu Lin, Haiyi Mei, Chen Wei, Ruisi Wang, Wanqi Yin, Xiangyu Fan, Han Du, Liang Pan, Peng Gao, Zhitao Yang, Yang Gao, Jiaqi Li, Tianxiang Ren, Yukun Wei, Xiaogang Wang, Chen Change Loy, Lei Yang, and Ziwei Liu. 2023. [Digital life project: Autonomous 3d characters with social intelligence](#). *CoRR*, abs/2312.04547.
- Jonathan Chang, Jordan Boyd-Graber, and David M Blei. 2009. [Connections between the lines: Augmenting social networks with text](#). In *Proceedings*

- of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, page 169–178, Paris, France. Association for Computing Machinery.
- Henry Y. Chen, Ethan Zhou, and Jinho D. Choi. 2017. [Robust coreference resolution and entity linking on dialogues: Character identification on tv show transcripts](#). In *Proceedings of the 21st Conference on Computational Natural Language Learning*, page 216–225, Vancouver, Canada. Association for Computational Linguistics.
- Nuo Chen, Yan Wang, Haiyun Jiang, Deng Cai, Yuhan Li, Ziyang Chen, Longyue Wang, and Jia Li. 2023. [Large language models meet harry potter: A dataset for aligning dialogue agents with characters](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 8506–8520. Association for Computational Linguistics.
- Joseph A DeVito. 2015. *The interpersonal communication book*, volume 14.
- Lucie Flekova and Iryna Gurevych. 2015. [Personality profiling of fictional characters using sense-level links between lexical resources](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, page 1805–1816, Lisbon, Portugal. Association for Computational Linguistics.
- Lea Frermann, Shay B. Cohen, and Mirella Lapata. 2018. [Whodunnit? crime drama as a case for natural language understanding](#). *Transactions of the Association for Computational Linguistics*, 6:1–15.
- Kanishk Gandhi, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah D. Goodman. 2023. [Understanding social reasoning in language models with language models](#). *CoRR*, abs/2306.15448.
- Qi Chen Gao and Ali Emami. 2023. [The turing quest: Can transformers make good npcs?](#) In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 93–103. Association for Computational Linguistics.
- Roxana Girju, Preslav Nakov, Vivi Nastase, Stan Szpakowicz, Peter Turney, and Deniz Yuret. 2007. [Semeval-2007 task 04: Classification of semantic relations between nominals](#). In *Proceedings of the Fourth International Workshop on Semantic Evaluations*, page 13–18, Prague, Czech Republic. 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.
- Hua He, Denilson Barbosa, and Grzegorz Kondrak. 2013. [Identification of speakers in novels](#). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, page 1312–1320, Sofia, Bulgaria. Association for Computational Linguistics.
- Mohit Iyyer, Anupam Guha, Snigdha Chaturvedi, Jordan Boyd-Graber, and Hal Daumé III. 2016. [Feuding families and former friends: Unsupervised learning for dynamic fictional relationships](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, page 1534–1544, San Diego, California. Association for Computational Linguistics.
- Evgeny Kim and Roman Klinger. 2019. [Frowning frodo, wincing leia, and a seriously great friendship: Learning to classify emotional relationships of fictional characters](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, page 647–653, Minneapolis, Minnesota. Association for Computational Linguistics.
- Vincent Labatut and Xavier Bost. 2019. [Extraction and analysis of fictional character networks: A survey](#). *ACM Comput. Surv.*, 52(5):89:1–89:40.
- Dongxu Li, Zhihao Yang, Yuquan Lan, Yunqi Zhang, Hui Zhao, and Gang Zhao. 2023. [Simple approach for aspect sentiment triplet extraction using span-based segment tagging and dual extractors](#). In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 2374–2378, Taipei, Taiwan. Association for Computing Machinery.
- Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. [Lost in the middle: How language models use long contexts](#). *CoRR*, abs/2307.03172.
- Philip Massey, Patrick Xia, David Bamman, and Noah A. Smith. 2015. [Annotating character relationships in literary texts](#). *CoRR*, abs/1512.00728.
- Simone Mellace, Vani Kanjirangat, and Alessandro Antonucci. 2020. [Relation clustering in narrative knowledge graphs](#). In *Proceedings of AI4Narratives - Workshop on Artificial Intelligence for Narratives in conjunction with the 29th International Joint Conference on Artificial Intelligence and the 17th Pacific Rim International Conference on Artificial Intelligence*, volume 2794 of *CEUR Workshop Proceedings*, pages 23–27, Yokohama, Japan. CEUR-WS.org.
- Kristine H Onishi and Renée Baillargeon. 2005. [Do 15-month-old infants understand false beliefs?](#) *science*, 308(5719):255–258.
- OpenAI. 2023. [Gpt-4 technical report](#). *CoRR*, abs/2303.08774.

- Joon Sung Park, Joseph C. O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. [Generative agents: Interactive simula-  
cra of human behavior](#). In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 2:1–2:22, San Francisco, CA, USA. Association for Computing Machinery.
- David Premack and Guy Woodruff. 1978. [Does the chimpanzee have a theory of mind?](#) *Behavioral and brain sciences*, 1(4):515–526.
- Chen Qian, Xin Cong, Wei Liu, Cheng Yang, Weize Chen, Yusheng Su, Yufan Dang, Jiahao Li, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2023. [Communicative agents for software development](#). *CoRR*, abs/2307.07924.
- Hannah Rashkin, Antoine Bosselut, Maarten Sap, Kevin Knight, and Yejin Choi. 2018. [Modeling naive psy-  
chology of characters in simple commonsense stories](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2289–2299, Melbourne, Australia. Association for Computational Linguistics.
- Yisi Sang, Xiangyang Mou, Mo Yu, Shunyu Yao, Jing Li, and Jeffrey Stanton. 2022. [TVShowGuess: Character comprehension in stories as speaker guessing](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4267–4287, Seattle, United States. Association for Computational Linguistics.
- Dominik Stambach, Maria Antoniak, and Elliott Ash. 2022. [Heroes, villains, and victims, and GPT-3: Automated extraction of character roles without training data](#). In *Proceedings of the 4th Workshop of Narrative Understanding*, pages 47–56, Seattle, United States. Association for Computational Linguistics.
- Quan Tu, Chuanqi Chen, Jinpeng Li, Yanran Li, Shuo Shang, Dongyan Zhao, Ran Wang, and Rui Yan. 2023. [Characterchat: Learning towards conversational ai with personalized social support](#). *CoRR*, abs/2308.10278.
- Shenzhi Wang, Chang Liu, Zilong Zheng, Siyuan Qi, Shuo Chen, Qisen Yang, Andrew Zhao, Chaofei Wang, Shiji Song, and Gao Huang. 2023. [Avalon's game of thoughts: Battle against deception through recursive contemplation](#). *CoRR*, abs/2310.01320.
- Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Hao-tian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P. Xing, and Zhiting Hu. 2024. [Promptagent: Strategic planning with language models enables expert-level prompt optimization](#). In *Proceedings of the 12th International Conference on Learning Representations*, Vienna, Austria. ICLR Press.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). In *Proceedings of the 36th Conference on Neural Information Processing Systems*, New Orleans, United States of America. Curran Associates, Inc.
- Heinz Wimmer and Josef Perner. 1983. [Beliefs about beliefs: Representation and constraining function of wrong beliefs in young children's understanding of deception](#). *Cognition*, 13(1):103–128.
- Sarah E Worth. 2004. [Narrative understanding and understanding narrative](#). *Contemporary Aesthetics (Journal Archive)*, 2(1):9.
- Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. [Scalable zero-shot entity linking with dense entity retrieval](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, page 6397–6407, Online. Association for Computational Linguistics.
- Ying Xu, Dakuo Wang, Mo Yu, Daniel Ritchie, Bing-sheng Yao, Tongshuang Wu, Zheng Zhang, Toby Li, Nora Bradford, Branda Sun, Tran 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*, pages 447–460, Dublin, Ireland. Association for Computational Linguistics.
- Yuzhuang Xu, Shuo Wang, Peng Li, Fuwen Luo, Xi-aolong Wang, Weidong Liu, and Yang Liu. 2023. [narrative playcation games: An empirical study on werewolf](#). *CoRR*, abs/2309.04658.
- Zhengzhe Yang and Jinho D. Choi. 2019. [Friendsqa: Open-domain question answering on tv show transcripts](#). In *Proceedings of the SIGdial 2019 Conference*, page 188–197, Stockholm, Sweden. Association for Computational Linguistics.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. 2023. [Tree of thoughts: Deliberate problem solving with large language models](#). In *Proceedings of the 37th Conference on Neural Information Processing Systems*, New Orleans, United States of America. Curran Associates, Inc.
- Keen You and Dan Goldwasser. 2020. [“where is this relationship going?”: Understanding relationship trajectories in narrative text](#). In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, page 168–178, Barcelona, Spain. Association for Computational Linguistics.
- Juntao Yu, Silviu Paun, Maris Camilleri, Paloma Garcia, Jon Chamberlain, Udo Kruschwitz, and Massimo Poesio. 2023a. [Aggregating crowdsourced and automatic judgments to scale up a corpus of anaphoric](#)

reference for fiction and Wikipedia texts. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 767–781, Dubrovnik, Croatia. Association for Computational Linguistics.

Mo Yu, Jiangnan Li, Shunyu Yao, Wenjie Pang, Xiaochen Zhou, Zhou Xiao, Fandong Meng, and Jie Zhou. 2023b. *Personality understanding of fictional characters during book reading*. In *Proceedings of 61st Annual Meeting of the Association for Computational Linguistics*, page 14784–14802, Toronto, Canada. Association for Computational Linguistics.

Runcong Zhao, Wenjia Zhang, Jiazheng Li, Lixing Zhu, Yanran Li, Yulan He, and Lin Gui. 2023. *Narrativeplay: Interactive narrative understanding*. *CoRR*, abs/2310.01459.

Lixing Zhu, Runcong Zhao, Lin Gui, and Yulan He. 2023. *Are NLP models good at tracing thoughts: An overview of narrative understanding*. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 10098–10121. Association for Computational Linguistics.

## A Data Selection

**Filtering** We excluded low quality narratives from our analysis due to various reasons: these included stories that were too simple with little content to analyse, those with entirely public background information offering limited merging opportunities, and narratives featuring supernatural elements. Additionally, we omitted stories with weak character relationships, such as groups of strangers in survival or mystery settings, and those with overly complex rules involving characters with multiple abilities. Poorly formatted narratives, those with an excessive number of false identities, and settings like psychiatric hospitals where characters have compromised memories were also filtered out.

**Processing** Extracting text from scanned narratives presents challenges with the incorrect recognition of text, a common issue with Chinese characters. This complexity is compounded by the presence of mixed or nested images within the content and the use of colored backgrounds. To address these challenges, we initially employed Adobe for the primary text extraction. Subsequently, we leveraged the capabilities of the ChatGPT model GPT-3.5, to rewrite and refine the text. This two-step process helps in correcting errors originating from the original scanned documents, ensuring a more accurate and reliable text output.

## B Annotation Guideline

In this section, we provide the annotation guidelines given to the annotators. For the sake of demonstration, we introduce 5 hypothetical characters: Alpha, Bravo, Charlie, Delta, and Echo.

### B.1 Disclaimers of Risks

Consider that the detective narrative dataset may contain descriptions of violent events, actions, or characters, we warn annotators of such content before the annotation process:

Please be advised that the detective narrative dataset may contain descriptions of violent events, actions, or characters. This content is included solely for academic, research, and narrative analysis purposes. It is not meant to glorify or trivialise violence in any form. Annotators should be aware of potential exposure to violent content and engage with the dataset professionally and responsibly.

### B.2 Definition of Terminologies

Before delving into the specific tasks of this project, it's crucial to define some key terms to ensure clarity and consistency:

**Character** This term denotes any intelligent entity within the narrative context. Examples of characters include humans, animals with demonstrated intelligence (like dogs), and artificial entities such as intelligent robots. However, it's important to note that not every entity qualifies as a *character*. For instance, collective groups (e.g., a gang), inanimate objects (e.g., an ordinary tree), or locations (e.g., a mansion) do not fall under the definition of a *character* for this project.

**Relationship** In the context of this project, a relationship refers to a social connection or interaction between two characters. A key aspect of relationships in this study is their directional nature, meaning that the perception of the relationship can vary depending on the character's viewpoint. For example, consider two characters, Alpha and Bravo. If they are friends, but Bravo is secretly aware of Alpha's betrayal, then Alpha might still perceive Bravo as a friend, while Bravo may no longer see Alpha in the same light.

**Conflicts** in the relationship graph can arise from two main sources: (1) Character Disagreement and (2) False Relationship, both resulting from partially observed or interpreted information. Below are explanations and examples for each type: *Character*

*Disagreement* happens when a character deliberately attempts to deceive others by assuming the identity of someone else. For example, if a male character Alpha commits a murder while disguised in female character Bravo’s clothes and wearing a wig, witnesses might mistakenly believe that Bravo is the murderer. *False Relationship* can arise under various scenarios, with extramarital affairs being a common cause. In such a scenario, Alpha may believe that Bravo is their father, while in reality, their biological father is Charlie.

### B.3 Annotation Tasks

In this project, you will work with multiple detective narratives, each offering a different character’s perspective on the same story. For instance, if Alpha, Bravo, Charlie, and Delta are all involved in a murder case and are suspects, you will receive four separate narratives. Each narrative will present the story from the viewpoint of one of these characters. Your task is to analyse these narratives and identify the underlying relationships between characters based on the varying perspectives provided. For each annotation task, we apply the following prompts for both human annotators and baselines.

**Character Extraction** We highlighted the variable(s) in blue in our designed prompt template below:

Identify and list all characters mentioned in the background story of {c}. Narrative Background: {Background} Please provide the response in JSON format for clarity and structure, listing each character distinctly. Example format: {"characters": ["character 1", "character 2", ...]}

c is the character name.

**Entity Linking** Extracting inter-character relationships and outputting the relationship graph, including characters and their relationships. This prompt is used for the AllTogether and human annotation.

Please classify the relationships between the characters using the given relationship categories {categories}. Note that multiple relationships categories can be applied to each character pair.

Character Background of {c}: {Background}

Provide the relationships in the following JSON format ONLY:

```
{
  "character name": [
    ["linked character", "relationship"],
```

```
    ["linked character", "relationship"]
  ],
  "character name": [
    ["linked character", "relationship"],
    ["linked character", "relationship"]
  ]
}
```

Given the character list, extracting inter-character relationships and outputting the relationship graph. This prompt is used for the DirRelation.

Using the given relationship categories {categories}, please classify the relationships between the characters listed below. Note that multiple relationships categories can be applied to each character pair.

List of Characters: {character list}

Character Background of {c}: {Background}

Provide the relationships in the following JSON format ONLY:

```
{
  "character name": [
    ["linked character", "relationship"],
    ["linked character", "relationship"]
  ],
  "character name": [
    ["linked character", "relationship"],
    ["linked character", "relationship"]
  ]
}
```

Given a pair of characters, extract their relationships from the provided narrative. This prompt is used for the PairRelation.

Using the given relationship categories {categories}, classify the relationship between characters {a} and {b} based on the character background of {c}. You may select multiple relationships.

Character Background: {Background}

Response in JSON format ONLY: {"a": ["{b}", "relationship 1, relationship 2"], "b": ["{a}", "relationship 1, relationship 2"]}

### B.4 Special Cases

During our trial run, we found there are some cases we need to clarify where annotators have different opinions on it:

**Perspective of Annotation** When reading through detective stories, comprehend the stories from an objective point of view. For instance, if Alpha helped Bravo but Bravo does not appreciate the help, you shall disregard Bravo’s emotion and still annotate that "Alpha is the helper of Bravo".

**Naming Convention** In our dataset, the same story will be conveyed from the perspective of numerous characters. Therefore, each character might use different names to refer to the same underlying character. For instance, if Charles is the son of Alpha, then the narrative from Alpha's perspective may refer to Charles exclusively using a nickname, Charlie. In this case, the relationship graph of Alpha shall *use the name that appeared in the narrative as the annotation*, which is Charlie in this case. However, when forming the overall relationship graph, *always use the formal name of the character as annotation*.

## C Relationship Schemes

In this section, we list our relationship categories.

- **Romantic Relationships**, which focus on the emotional, psychological, and sometimes physical intimacy between characters who are involved in a romantic or amorous setting. This can range from courtship to marriage and can include a variety of romantic scenarios, conflicts, or issues.
  - Wife of X
    - \* Concubine of X
  - Husband of X
  - Extramarital Affair with X
    - \* Secret Lover of X
  - Romantic relationships with X
    - \* Lover of X
    - \* Boyfriend of X
    - \* Girlfriend of X
  - Ex-romantic relationships with X
    - \* Ex-boyfriend of X
    - \* Ex-girlfriend of X
    - \* Ex-wife of X
    - \* Ex-husband of X
  - Admirer of X
    - \* Secret Admirer of X
    - \* Fondness of X
  - Admired by X
    - \* Liked by X
    - \* Secret Crush of X
  - Fiance of X
  - Fiancee of X
  - Co-wives of X
- **Family Relationships**, which are between members of a family unit, which are often critical to the protagonist's identity or quest. This can include parent-child dynamics, sibling bonds, and other familial relationships (Grandparent, Cousin, etc.). It often involves loyalty, tradition, unconditional love, dysfunction, generational conflict, and personal growth.
  - Father of X
    - \* Father-in-law of X
    - \* Future Father-in-law of X
    - \* Adoptive Father of X
    - \* Step-father of X
    - \* Biological Father of X
  - Mother of X
  - Mother-in-law of X
  - Future Mother-in-law of X
  - Adoptive Mother of X
  - Step-Mother of X
  - Biological Mother of X
  - Child of X
    - \* Son of X
    - \* Son-in-law of X
    - \* Future Son-in-law of X
    - \* Adoptive Son of X
    - \* Step-Son of X
    - \* Biological Son of X
    - \* Daughter of X
    - \* Daughter-in-law of X
    - \* Future Daughter-in-law of X
    - \* Adoptive Daughter of X
    - \* Step-Daughter of X
    - \* Biological Daughter of X
  - Sibling of X
    - \* Brother of X
    - \* Half Brother of X
    - \* Adoptive Brother of X
    - \* Step-brother of X
    - \* Elder Brother of X
    - \* Younger Brother of X
    - \* Twin Brother of X
    - \* Sister of X
    - \* Half Sister of X
    - \* Adoptive Sister of X
    - \* Step-sister of X
    - \* Elder Sister of X
    - \* Younger Sister of X
    - \* Twin Sister of X
  - Grandparent of X
    - \* Grandfather of X

- \* Grandmother of X
- Grandchild of X
  - \* Grandson of X
  - \* Granddaughter of X
- Relative of X
  - \* Sister-in-law of X
  - \* Brother-in-law of X
  - \* Nephew of X
  - \* Aunt of X
  - \* Uncle of X
  - \* Niece of X
  - \* Cousin of X
  - \* Future Relative of X
- Possibly Family of X
- **Ally Relationships**, which involve a cooperative bond formed between characters, often to achieve a common goal or to face a shared enemy. This relationship is usually strategic and can be either temporary or long-term. These relationships may not be rooted in emotional bonds but rather strategic interests.
  - Friend of X
    - \* Sworn Brother of X
  - Mentor of X
    - \* Teacher of X
  - Student of X
  - Informant of X
  - Information Receiver from X
  - Acquaintance of X
    - \* Classmate of X
    - \* Schoolmate of X
    - \* Neighbour of X
  - Helper of X
    - \* Saviour of X
  - Helped by X
    - \* Seeker of Help from X
    - \* Saved by X
  - Quest Companion of X
    - \* Crime Partner of X
  - Guest of X
  - Host of X
- **Antagonistic Relationships** refers to connections between individuals or groups at odds due to conflicting beliefs, loyalties, or goals.
  - Perpetrator of X
    - \* Bully of X
    - \* Murderer of X
  - Attempted Perpetrator of X
    - \* Attempted Murderer of X
  - Victim of X
    - \* Killed by X
  - Dislike of X
    - \* Jealous of X
    - \* Unsuccessful Helper of X
    - \* Suspect to X
  - Disliked by X
    - \* Jealous by X
  - Subject of investigation for X
  - Deceiver of X
    - \* Thief of X
    - \* Betrayer of X
    - \* Spy of X
  - Deceived by X
    - \* Betrayed by X
  - Suspected by X
    - \* Suspicious of X
    - \* Suspect of X
  - Adversary of X
    - \* Enemy of X
    - \* Hate of X
    - \* Hated by X
    - \* Rebellion against X
    - \* Rival of X
    - \* Rival in Love of X
    - \* X's Victim's Family
    - \* X's Enemy's Family
    - \* Perpetrator of X's Family
    - \* in the Lawsuit against X
  - Manipulator of X
  - Manipulated by X
- **Business Relationships**, refer to interactions and connections between characters that are primarily based on commercial, financial, or professional contexts. These relationships are often characterized by mutual interests, professional collaborations, or economic transactions.
  - Superior of X
    - \* Authority over X
    - \* Employer of X
    - \* Master of X
  - Subordinate of X
    - \* Employee of X
    - \* Servant of X
    - \* Guard of X



- \* Tour Guide of X
  - \* Minion of X
  - Debtor of X
  - Creditor of X
  - Colleague of X
  - Business Partner of X
  - Customer of X
    - \* Buyer of X
    - \* Tenant of X
    - \* Patient of X
  - Product Provider of X
    - \* Seller of X
    - \* Landlord of X
  - Service Provider of X
    - \* Lawyer of X
    - \* Messenger of X
    - \* Doctor of X
- **Special Relationships** In addition to the standard relationship labels, we also introduce a set of special relationship labels.
- Same Person as X (Different Reference)
  - Same Person as X (Different Identity)
  - Replaced X’s Identity
  - Stranger to X

“Same Person as X (Different Reference)” refers to different references for the same person, such as “Costa”, “Head Nurse Costa”, and “Sylvia Costa”. These can easily be identified by humans as referring to the same individual. “Same Person as X (Different Identity)” refers to a single individual having multiple identities. For example, “Elaine Carter” may be known as a doctor in a hospital, but in reality, she is a criminal detective named “Yilin Carter”. “Elaine Carter” is her false identity used to investigate an organ trafficking case in the hospital.

“Stranger” is primarily designed for PairRelation. In human annotation, we simply skip relationships not described in the narrative. However, with PairRelation, we might ask the model about the relationships between two characters who do not know each other. Therefore, we have designed this category for such situations and remove all relationship categories labelled as “Stranger to X” during the evaluation stage.

## C.1 Inter-annotator Agreement

Calculation of the F1-score for inter-annotator agreement follows the previous works (Girju et al., 2007; Gurulingappa et al., 2012; Li et al., 2023). For example, if the characters annotated by annotator 1 is  $C_1$ , the characters annotated by annotator 2 is  $C_2$ , then we have  $\text{precision} = \frac{|C_1 \cap C_2|}{|C_1|}$ ,  $\text{recall} = \frac{|C_1 \cap C_2|}{|C_2|}$ , and the F1-score =  $\frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$ .

## D Experimental Setup

### D.1 Zero-shot Setting

All the experiments reported were conducted in a zero-shot setting. Although few-shot learning has been proven to improve performance, we encountered challenges when attempting to apply it to long narratives in our dataset. Due to input token limitations, it is nearly impossible to implement a proper few-shot setting using the classic approach, where each example is typically just one sentence long. During our trial round, we discovered that using examples as part of a few-shot learning strategy can lead to hallucinations, where the model treats the characters and relationships from the examples as integral parts of the story, inadvertently incorporating them into the output.

As a result, we opted for a compromise solution, which gives relatively better results. We decided to provide desired output examples without including actual examples but rather an explanation, as the example given in B.3. This allows us to guide the model without the risk of incorporating example-specific content directly into the output.

## E Evaluation

### E.1 Character Extraction: Error Sample Analysis

We discussed the common issues we encountered in Section 4.3 Character Extraction. To be more specific, here is an extracted example:

```
"Tony Bijeli": [
  ["Frank Bijeli", "brother of x"],
  ["Frank Bijeli", "friend of x"],
  ["Father Tom", "admired by x"],
  ["Nurses", "possibly family of x"],
  ["Hospital staff", "stranger to x"]
]
```

"Nurses" and "Hospital staff" are clearly not the types of entities we aim to extract. Errors can also include personal pronouns such as "You" and "Them", locations such as "Hospital" and "Pavilion", organisations such as "Xiao", and inanimate objects like "Spirit Casket".

Strategy	Llama2			GPT-3.5			GPT-4		
	precision	recall	f1	precision	recall	f1	precision	recall	f1
<i>Information from Single Character's Perspective (detailed-level)</i>									
AllTogether	0.096	0.040	0.057	0.043	0.031	0.036	0.202	0.192	0.197
DirRelation	0.116	0.062	0.081	0.078	0.050	0.061	0.149	0.182	0.164
PairRelation	0.017	0.061	0.027	0.018	0.061	0.028	0.032	0.173	0.054
<i>Information from All Characters' Perspectives (detailed-level)</i>									
AllTogether	0.097	0.016	0.028	0.049	0.016	0.024	0.188	0.058	0.088
DirRelation	0.153	0.025	0.043	0.049	0.020	0.028	0.131	0.049	0.071
PairRelation	0.008	0.065	0.015	0.009	0.083	0.016	0.010	0.165	0.019
<i>Information from Single Character's Perspective (medium-level)</i>									
AllTogether	0.129	0.054	0.076	0.056	0.041	0.047	0.283	0.269	0.276
DirRelation	0.160	0.085	0.111	0.119	0.076	0.093	0.219	0.267	0.240
PairRelation	0.025	0.089	0.039	0.029	0.099	0.045	0.047	0.258	0.080
<i>Information from All Characters' Perspectives (medium-level)</i>									
AllTogether	0.121	0.02	0.035	0.064	0.020	0.031	0.267	0.082	0.125
DirRelation	0.203	0.033	0.057	0.092	0.037	0.053	0.202	0.075	0.110
PairRelation	0.012	0.092	0.021	0.014	0.132	0.025	0.014	0.238	0.027
<i>Information from Single Character's Perspective (high-level)</i>									
AllTogether	0.205	0.085	0.120	0.094	0.069	0.080	0.417	0.397	0.407
DirRelation	0.256	0.137	0.178	0.236	0.151	0.184	0.320	0.391	0.352
PairRelation	0.054	0.190	0.084	0.079	0.268	0.122	0.093	0.505	0.157
<i>Information from All Characters' Perspectives (high-level)</i>									
AllTogether	0.223	0.038	0.064	0.089	0.029	0.043	0.450	0.138	0.211
DirRelation	0.342	0.056	0.096	0.167	0.067	0.096	0.334	0.124	0.181
PairRelation	0.030	0.236	0.054	0.041	0.383	0.073	0.032	0.524	0.060

Table 8: Evaluation of three different relation detection strategies to extract relationships with Chinese narrative input.

Additionally, hallucinations sometimes involve characters that never manifested in the original narrative. Recall performance is also comparatively lower with lengthier and more intricate detective narrative inputs when compared to classic benchmark tasks.

## E.2 Performance Across Different Category Levels

Table 8 presents the performance of three LLMs – Llama2, GPT-3.5, and GPT-4 – across different relation detection strategies and levels of relationship categorisation. The performance is measured in terms of precision, recall, and F1-score. The relation detection strategies tested are AllTogether, DirRelation, and PairRelation, and the relationship categorisation levels are detailed, medium, and high. The analysis is divided based on whether the information was derived from a single character’s perspective or from all characters’ perspectives.

As previously mentioned, we developed a hierarchical structure for relationship categories, consisting of 5 top-level categories, 54 intermediate categories, and 163 detailed categories. In our main text, due to page limitations, we report only the re-

sults for the intermediate categories, which we consider the most representative. Detailed categories may have some degree of overlap, while top-level categories are too broad and may lose essential information. We present the full results across all category levels, affirming the observations consistent with those reported in the main text.

**Strategy Performance Variability** AllTogether: Generally, GPT-4 performs better in this strategy across all levels, indicating its superior ability to extract and understand relationships without additional character inputs. DirRelation: This strategy shows a consistent pattern of improved performance across all models, especially at the high-level categorization. This suggests that DirRelation, which involves direct relationship extraction, might be a more suitable approach for these models. PairRelation: Generally, this strategy has the lowest F1-scores among the three, which might be due to its nature of generating more hallucinations as it prompts for relationships between every pair of characters. We also realised that low precision, resulting from such fabrications, had a more detrimental effect than low recall, as it could lead to

complete misinformation for readers. For example, while failing to detect that “Drake” is the secret lover of “Gale” is a shortcoming, the model incorrectly predicting “Drake” as the father of “Gale” is a more severe error.

**Impact of Perspective** The models generally perform better when extracting information from a single character’s perspective compared to all characters’ perspectives. This could be due to the increased complexity and potential for contradictory information when considering multiple perspectives.

**Category Level Impact** Detailed-level: The models show relatively lower scores, likely due to the finer granularity of relationships, which increases complexity. Medium-level: This level appears to strike a balance between detail and breadth, with GPT-4 showing notably higher scores in the AllTogether strategy. High-level: Here, the models generally achieve higher F1-scores, particularly GPT-4, suggesting that broader categories are easier for the models to identify and classify.

**Model Comparison** GPT-4: This model consistently outperforms the others in almost all scenarios, especially in the AllTogether strategy, indicating its advanced capability in understanding complex relationships. Llama2 and GPT-3.5: These models show improvements in the DirRelation strategy, particularly at higher-level categories, but fall short compared to GPT-4.

### E.3 Translated Version

For our study, we primarily report results using the Chinese version for two main reasons. Firstly, the source material is originally in Chinese, so utilising the native language minimizes information loss during translation. Secondly, our experiments partly rely on the OpenAI API, which imposes content restrictions. Detective narratives often involve themes like murder, which can trigger OpenAI’s stringent content filters, especially in English. However, these restrictions are less frequent with Chinese inputs, likely due to biases in the training data of the LLMs.

We initially attempted to translate the Chinese version into English using GPT-3.5 and even GPT-4. However, we encountered challenges in controlling the translation process with these LLMs. Despite specifying in our prompts that we required direct translation, the LLMs sometimes performed summarisation instead. Additionally, the translations

were prone to hallucinations, introducing inaccuracies. Consequently, we decided to use Google Translator for the translation process to ensure more accurate and direct translations.

During the process of translating Chinese names into English, we observed that Google Translator inconsistently applied phonetic transliteration and semantic translation. This inconsistency resulted in the translated character names being quite confusing, as the approach to translation varied unpredictably between literal phonetic equivalents and meaning-based interpretations. Therefore, we initially used GPT-4 to translate all labelled character names from Chinese into English. Once the character names were translated, we replaced the original names in the corpus with these English versions. Following this step, we employed Google Translator to translate the entire corpus. This two-step process ensured a more consistent and clear translation of character names, while also effectively translating the broader narrative content.

In addition to our previous findings, we also report on experiments conducted using the English corpus. These experiments corroborate the results obtained from the Chinese corpus. However, it’s important to note that we limited these experiments to running on Llama2 and GPT-3.5 models. The decision to exclude GPT-4 from these tests was due to its higher costs.

The data indicates that self-correction significantly reduces the corruption rate across all strategies and models, with particularly notable improvements observed in GPT-3.5. GPT-3.5 benefits more from self-correction compared to Llama2, as evidenced by the greater percentage reductions in corruption rates. While all strategies show improvement with self-correction, the AllTogether strategy displays the most significant improvement for GPT-3.5, suggesting that this strategies might be more prone to initial corruption but also benefits greatly from correction.

The AllTogether and DirRelation show some improvement with self-correction. However, the improvements for Llama2 are generally minimal, and the PairRelation strategy does not benefit from self-correction. This suggests that while self-correction can lead to improvements in model performance, the extent of these improvements varies and can be limited for certain strategies or models.

In the context of a single character’s perspective, all strategies show relatively good performance. From all characters’ perspectives, the performance of both models decreases, indicating the complex-

Strategy	Llama2			GPT-3.5			GPT-4		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
<i>Information from Single Character's Perspective</i>									
DirRelation(Extracted)	0.160	0.085	0.111	0.119	0.076	0.093	0.219	0.267	0.240
DirRelation(Given)	0.170	0.097	0.124	0.155	0.102	0.123	0.293	0.342	0.315
PairRelation(Extracted)	0.025	0.089	0.039	0.029	0.099	0.045	0.047	0.258	0.080
PairRelation(Given)	0.045	0.181	0.071	0.056	0.158	0.083	0.105	0.402	0.167
<i>Information from All Characters' Perspectives</i>									
DirRelation(Extracted)	0.203	0.033	0.057	0.092	0.037	0.053	0.202	0.075	0.110
DirRelation(Given)	0.163	0.032	0.053	0.155	0.040	0.064	0.330	0.115	0.171
PairRelation(Extracted)	0.012	0.092	0.021	0.014	0.132	0.025	0.014	0.238	0.027
PairRelation(Given)	0.027	0.168	0.046	0.034	0.147	0.055	0.071	0.282	0.113

Table 9: Detailed Comparison of Character Extraction Impact.

Strategy	Llama2		GPT-3.5	
	before	after	before	after
AllTogether	0.312	<b>0.282</b>	0.426	<b>0.107</b>
DirRelation	0.245	<b>0.219</b>	0.296	<b>0.063</b>
PairRelation	0.226	<b>0.185</b>	0.303	<b>0.105</b>

Table 10: Corruption Rate for English Corpus. The percentage of output relationships that fail to comply with the provided format guidelines before and after self-correction.

Strategy	Llama2		GPT-3.5	
	before	after	before	after
AllTogether	0.069	<b>0.070</b>	0.044	<b>0.060</b>
DirRelation	0.088	<b>0.090</b>	0.038	<b>0.043</b>
PairRelation	0.025	0.025	0.014	0.014

Table 11: Performance Comparison for English Corpus. The F1-score before and after self-correction.

Strategy	Llama2			GPT-3.5		
	precision	recall	f1	precision	recall	f1
<i>Information from Single Character's Perspective</i>						
DirCharacter	0.628	0.634	0.631	0.633	0.631	0.632
AllTogether	0.808	0.528	0.639	0.522	0.572	0.546
DirRelation	0.718	0.619	0.665	0.663	0.620	0.641
PairRelation	0.692	0.645	0.668	0.575	0.637	0.605
<i>Information from All Characters' Perspectives</i>						
DirCharacter	0.450	0.744	0.561	0.485	0.776	0.597
AllTogether	0.840	0.310	0.453	0.776	0.305	0.438
DirRelation	0.789	0.346	0.481	0.575	0.618	0.595
PairRelation	0.502	0.557	0.528	0.370	0.765	0.499

Table 12: F1-score of Character Extraction for English corpus.

ity of handling multiple viewpoints. In both perspectives, DirRelation and PairRelation show more balanced precision and recall compared to AllTogether, especially for GPT-3.5. The PairRelation, while showing reasonable performance in a single character's perspective, drops in effectiveness when considering all characters' perspectives.

AllTogether: Generally, this strategy shows mod-

Strategy	Llama2			GPT-3.5		
	precision	recall	f1	precision	recall	f1
<i>Information from Single Character's Perspective (detailed-level)</i>						
AllTogether	0.136	0.081	0.102	0.088	0.076	0.082
DirRelation	0.147	0.113	0.128	0.095	0.081	0.088
PairRelation	0.022	0.11	0.037	0.016	0.072	0.026
<i>Information from All Characters' Perspectives (detailed-level)</i>						
AllTogether	0.144	0.029	0.049	0.112	0.022	0.037
DirRelation	0.170	0.041	0.066	0.046	0.024	0.031
PairRelation	0.011	0.09	0.019	0.005	0.068	0.009
<i>Information from Single Character's Perspective (medium-level)</i>						
AllTogether	0.178	0.107	0.134	0.121	0.104	0.112
DirRelation	0.191	0.147	0.166	0.132	0.113	0.122
PairRelation	0.032	0.157	0.053	0.026	0.116	0.042
<i>Information from All Characters' Perspectives (medium-level)</i>						
AllTogether	0.208	0.042	0.07	0.181	0.036	0.06
DirRelation	0.232	0.056	0.09	0.063	0.032	0.043
PairRelation	0.014	0.121	0.025	0.008	0.1	0.014
<i>Information from Single Character's Perspective (high-level)</i>						
AllTogether	0.280	0.167	0.209	0.194	0.167	0.179
DirRelation	0.270	0.208	0.235	0.222	0.190	0.205
PairRelation	0.062	0.308	0.103	0.059	0.268	0.097
<i>Information from All Characters' Perspectives (high-level)</i>						
AllTogether	0.345	0.07	0.117	0.307	0.061	0.102
DirRelation	0.397	0.095	0.154	0.151	0.078	0.103
PairRelation	0.029	0.246	0.052	0.023	0.301	0.042

Table 13: Evaluation of three different strategies to extract relationships with English narrative input.

erate to high precision but lower recall, leading to varied F1-scores. Its performance is more consistent in the single character's perspective. DirRelation: Exhibits balanced precision and recall, generally performing better than AllTogether in terms of F1-scores. It shows a notable improvement in F1 scores at the high-level perspective. PairRelation: Significantly lower precision across all categories, but higher recall in high-level information extraction. However, the low precision greatly affects its overall F1-scores.

After translation, all strategies exhibit improved performance with Llama2, which shows its better

understanding ability in English. GPT was also shown to have better performance in English corpus (OpenAI, 2023), however, in our experiment, GPT shows varying results with the English corpus compared to the Chinese one, likely due to content restrictions. Detective narratives often include themes like murder, which may activate OpenAI’s strict content filters. However, such restrictions are less common with Chinese inputs, probably because of biases in the training data of LLMs, resulting in decreased performance with the English corpus.

#### **E.4 Similarity Distance**

We attempted to incorporate cosine similarity measurement scores alongside F1-scores due to the unclear boundaries between some categories, and because LLMs sometimes produce correct outputs that don’t match the specified category. For instance, “*mother of x*” might be output as “*mother to x*”, which would be marked incorrect in metrics like accuracy and F1-score. The similarity scores aim to mitigate the impact of such formatting errors.

However, we discovered that cosine similarity scores are not a reliable measure, as they can yield unreasonable results. For instance, the similarity score between “*employer*” and “*stranger*” is 0.29, but it’s 0.42 between “*daughter*” and “*stranger*”, suggesting that a daughter is closer to being a stranger than an employer. This result is counterintuitive from a human perspective, leading us to exclude these similarity measurement results in this paper.