

# Llamas Don't Understand Fiction: Application and Evaluation of Large Language Models for Knowledge Extraction from Short Stories in English

Arianna Graciotti<sup>1,2</sup> , Franziska Pannach<sup>1</sup> , Valentina Presutti<sup>1</sup> , and Federico Pianzola<sup>1</sup> 

<sup>1</sup> Centre for Language and Cognition, University of Groningen, Groningen, Netherlands

<sup>2</sup> Department of Languages, Literatures and Modern Cultures, University of Bologna, Bologna, Italy

## Abstract

Extracting event knowledge from unstructured text is a well-known challenge in Natural Language Processing (NLP) and is particularly difficult when dealing with fiction. Subtext, rather than explicit information, and figurative style in fictional narratives, complicate event extraction. Recent advances in Large Language Models (LLMs) have improved performance across various NLP tasks. However, their effectiveness in extracting events from fiction remains underexplored. In this article, we evaluate the performance of open-weights LLMs to extract character death events from fictional narratives in English. These events are defined as triples consisting of *Victim*, *Perpetrator*, and *Mode of Demise*. We cast Knowledge Extraction (KE) as a zero-shot task and evaluate our approach on a manually annotated benchmark of fanfiction stories. Our results show that LLMs struggle with KE from fiction, with a maximum F1-score of 0.45 across the elements constituting the triples and, at most, 25% of death events correctly extracted. A detailed error analysis reveals that most errors stem from missed death events and from direct presentation modes, such as direct speech, which significantly impair extraction performance. Moreover, KE accuracy declines as the story length increases, while LLMs' background knowledge leakage contributes to false positives. These findings provide domain-specific insights into the challenges of KE in fiction.

**Keywords:** Event Extraction, Fiction, Human-Centered Evaluation, LLMs, Zero/Few-Shot Extraction

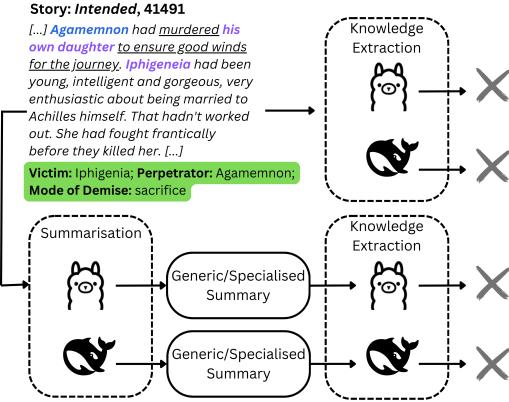
## 1 Introduction

This work presents an effort to evaluate Large Language Models (LLMs) for automatic knowledge extraction (KE) from fiction, and to understand the impact of stories' features on the models' performance, such as the presentation mode, the point of view from which the event is presented, and the stories' word-length and word-entropy. Entertainment companies increasingly use NLP techniques to improve the services they offer to their audience [5], and fiction is a domain well known to language models, since it has been included in the pre-training of most of them [6; 8; 12]. However, fictional texts pose unique challenges for some common NLP tasks, such as entity extraction, co-reference resolution, and event extraction [3; 33]. Fiction notably uses information gaps to generate effects such as suspense, curiosity, and surprise [34], so information about events and characters is not always presented in a straightforward way. In addition, novelists pay attention to the aesthetic quality of their writing, often using figurative language. This linguistic and narrative complexity makes tasks like event detection in fiction challenging.

---

Arianna Graciotti, Franziska Pannach, Valentina Presutti, and Federico Pianzola. "Llamas Don't Understand Fiction: Application and Evaluation of Large Language Models for Knowledge Extraction from Short Stories in English." In: *Computational Humanities Research 2025*, ed. by Taylor Arnold, Margherita Fantoli, and Ruben Ros. Vol. 3. Anthology of Computers and the Humanities. 2025, 4–31. <https://doi.org/10.63744/iCGYNUN0uUAe>.

© 2025 by the authors. Licensed under Creative Commons Attribution 4.0 International (CC BY 4.0).



**Figure 1:** An excerpt from a short story in our dataset, with the human-annotated character death event triple highlighted in green. References to the victim appear in purple, to the perpetrator in blue, and the mode of demise is underlined.

To test the KE capabilities of LLMs, we selected one type of event, character deaths, and one specific subdomain, fanfiction on ancient Greek mythology. The mention of death events varies across the data, from explicit and detailed descriptions to hints and allusions. Some stories in the corpus should be well known to LLMs, e.g. *Agamemnon* sacrificing his daughter *Iphigenia* (see Figure 1). However, fanfiction adaptations often change relevant plot points. We can thus analyse how the background knowledge of a model in the training data influences the performance of knowledge extraction when the original and the adaptations differ.

Death events serve as an exemplary case study due to their narrative prominence in plot development, making them ideal for refining automated event extraction methodologies. Technical advancements in KE can enable comparative literary studies at scale: by systematically identifying and analyzing story traits (e.g., death patterns, causal relationships, or thematic motifs), researchers can trace how narrative conventions evolve across cultural traditions, genres, or historical periods.

Our main contributions can be summarised as follows:

- An analysis of the KE capabilities of open-weights LLMs (llama3.1:70b and deepseek-r1:70b) within the domain of fiction;
- A comparative evaluation of KE performance from two types of input: full story texts and LLM-generated summaries;
- A detailed error analysis that highlights specific challenges faced by LLMs on KE in the domain of fiction, including handling events presented in different modes, such as direct and indirect speech or thought, and other domain-specific complexities.

All code, data, and evaluation outputs are publicly available on GitHub<sup>1</sup>, including the chain-of-thought output of deepseek-r1:70b.

## 2 Related Work

KE of events in fiction has been mainly addressed as a combined dependency-parsing and supersense-tagging task [2; 27], as a semantic roles labelling task – e.g., using PropBank [19; 28] or FrameNet [1; 25] – and as a QA task [16; 37]. A first challenge for fiction is that it requires readers and language models to connect information across large spans of text to identify what happens in a story. However, most of the documents used to train LLMs for downstream tasks

<sup>1</sup> <https://github.com/GOLEM-lab/llms-ke-fiction>

are quite short. Therefore, it is not beneficial to use models that excel in general extractive QA tasks, because their good performance in short passage datasets, such as the Stanford Question Answering Dataset [29, SQuAD 1.1], where most texts do not reach 350 words, does not scale to longer texts. Even fiction-specific datasets rarely include long texts, e.g., FairytaleQA [36] has an average of 370 words per story and MCTest [30] has 200 words per story. NarrativeQA [18] and LFED [37] are two exceptions, including full-length books and movie scripts. Indeed, analyses of these two datasets show that language models struggle with QA in fiction. Better insights are needed to understand which factors impact the performance of KE from fiction.

A second challenge is that QA datasets primarily address generic reading comprehension and are not suitable for systematic KE tasks, such as causality linking between events [23], character profiling [38], and relationship identification [39]. For an in-depth understanding of the story, more fine-grained information on events and their participants is needed. But in this case, annotation is labour-intensive and datasets of annotated events in fiction are somewhat limited: LitBank [33] uses samples of 2,000 words from 100 novels and EvENT [35] has six full texts for a total of ca. 291,000 tokens.

Given the scarcity of specialised resources for fiction, leveraging LLMs’ zero-shot capabilities for KE might be an efficient solution. To evaluate this opportunity, we focus on extracting character death events, a type of event that is overrepresented in fiction [24] due to its potential for narrative engagement. Indeed, murders, natural disasters, and other tragedies are often crucial events in the plot development of novels. Reliably extracting this information can open up new possibilities for the large-scale comparative analysis of literature.

### 3 Methodology

In this section, we outline the methodology used to assess the ability of LLMs to perform KE from fiction. We begin by defining the task in Section 3.1 and detailing the framework for evaluating LLMs’ KE capabilities. Our approach consists of two main steps: knowledge extraction and summarisation. Next, in Section 3.2, we introduce the dataset, a corpus of short stories with manually annotated death events. We then describe the models evaluated in our study (Section 3.3) before presenting the evaluation criteria used for assessment (Section 4).

#### 3.1 Task

**Knowledge extraction** We approach the recognition of death events in fictional narratives by framing it as a KE task. Specifically, we prompt an LLM to identify death events within a given input and return a triple comprising the three constituent elements of each event: (1) the victim, (2) the perpetrator, (3) the mode of demise (e.g. *Medusa*, *Perseus*, *beheading*). The prompt used for this task is provided in Appendix B.1. The extracted triples are compared against the manually annotated gold standard benchmark.

**Summarisation** The input to the LLM can be either the full story or an automatically generated summary of the story. The motivation for introducing a preliminary summarisation step is that, in fiction, events are often implicit or presented using figurative language. Summarising stories may help in making events more explicit in the summaries and, therefore, easier to extract [22]. For the summarisation step, we explore two types of summaries: (1) a generic summary, generated by prompting the LLM to provide a summary of the story capturing all the events; (2) a task-specific specialised summary, generated by prompting the LLM to explicitly focus on identifying and reporting character deaths. The specific prompts for generating these summaries are detailed in Appendix B.2.

### 3.2 Dataset

#Stories	#Death Events	#Victims	#Perpetrators	Modes of Demise (All)	Modes of Demise (Unique)
59	246	234	169	177	63

**Table 1:** Summary statistics of stories, death events, and modes of demise.

Metric	Mean	Median	Std Dev	Min	Max
Words per story	2,560	1,686	2,083.5	400	7,367
Word entropy	8.15	8.13	0.49	7.05	9.25

**Table 2:** Dataset words per story and word entropy statistics.

#### 3.2.1 Statistics

Our dataset contains 59 stories (see Table 1) collected from the fan fiction platform Archive of Our Own [13]. As reported in Table 2, the stories in our dataset feature a median of 1,681 words per story and 8.13 word entropy. The latter value is used in the error analysis presented in Section ??.

Following [15], we compute textual features and readers’ reception characteristics of the stories<sup>2</sup> and report them in Table 7 in Appendix A. As textual features, we report metrics such as the Dale-Chall New Readability score, which measures the stylistic simplicity of a given text (the higher the score, the more complex the story), and the Hurst exponent ( $H$ ), which is used as a proxy to measure the narrative complexity of the story’s sentiment arc ( $H > 0.5$  indicates an higher predictability of the narrative and is often associated to bestsellers, while  $H < 0.5$  indicates more complex narratives and is frequently related to highbrow fiction [7]). Readers’ reception features include kudos and hits, metrics harvested from Archive of Our Own that indicate the number of likes a story received and the number of times it was visited. The ratio of these metrics indicates appreciation [13; 26]. Interestingly, our dataset has a higher Dale-Chall New mean score than the fanfiction corpus reported in [15] (6.72 vs 5.73), and a slightly higher  $H$  (0.57 vs 0.59). We calculated Spearman  $\rho$  between hits and appreciation and each of the textual features reported, but found no correlation.

#### 3.2.2 Annotation

Three annotators worked on the corpus in different phases: one annotator (an expert on Greek myths) selected the short stories. Each of the stories was published under the category (i.e. *fandom*) *Ancient Greek Religion and Lore*. They were filtered by length (ca. 400-10.000 words), as shorter publications are often poems or very short stories without many events, and by the author-supplied label *Major Character Death*. This ensured that at least one character death was present in the story, but the annotation included minor character deaths as well, where they occurred.

The domain expert then annotated the death events in the texts as triples of *Victim*, *Perpetrator* and *Mode of Demise* (e.g., *Medusa*, *Perseus*, *beheading*), and extracted textual evidence supporting each triple. The textual evidence might be an explicit statement or the semantically closest statement indicating the event (e.g. *I saw her floating body*). Co-references were always resolved, and the proper name of the character was used in the triple. Information on the three constituents might be presented in different parts of the text, e.g., the perpetrator appears only later in the text, while the victim is mentioned early. In these cases, multiple statements of support were extracted and separated by “[...].” Not all cases require the three constituents (Victim, Perpetrator, and Mode of Demise) to be present. For instance, not every character death occurs at the hand of a second

<sup>2</sup> Calculated with [https://github.com/centre-for-humanities-computing/fabula\\_pipeline](https://github.com/centre-for-humanities-computing/fabula_pipeline).

character. One or more of the triple constituents might be unspecified without even implied mention (e.g. *Artemis killed her children*). In these cases, the Mode of Demise is *unspecified*. Modes of Demise were aligned, e.g. *turned to stone*, *transformed into a rock*, and *petrification* were subsumed under *petrification*.

In a subsequent quality control step, a second annotator checked all the annotations and verified that the selected textual evidence was complete and correct. A third annotator (an expert narratologist) annotated the presentation modes and the perspective of the narrator (first-, second-, third-person). We categorize presentation modes into five types: (i) *direct speech*, (ii) *direct thought*, (iii) *indirect speech*, (iv) *indirect thought*, and (v) *narrator*.<sup>3</sup> This distinction is relevant because the presentation mode influences the epistemological status of events. For example, speech, thoughts, and dreams decrease the certainty about the actual occurrence of the reported event. This can pose challenges for LLMs, as it may lower their confidence during the KE task. We report in Appendix D an overview of these modes, including a representative example sentence from the dataset for each.

### 3.3 Models

We employed 11lama3.1:70b [21] and deepseek-R1-Distill-11lama-70B, a reasoning model based on 11lama-3.3-70B-Instruct [10]. The choice of open-weight LLMs was necessary to locally process full-text stories without breaching copyright or data privacy associated with sending content to proprietary, remotely hosted models. The selected models were utilised to both generate summaries of the input stories and extract death events from the tested inputs (full-text stories, generic summaries, and specialised summaries).

### 3.4 Evaluation

The evaluation compares the extracted KE outputs to the gold annotations provided in the dataset. We conduct two types of evaluations: (1) a *fine-grained* evaluation, which assesses each component of a death event (victim, perpetrator, and mode of demise) and (2) a *coarse-grained* evaluation, which evaluates the death event triple as a whole.

**\*Fine-grained evaluation** The fine-grained evaluation assesses the correctness of individual elements within each death event triple. Specifically, we compare the extracted Victim, Perpetrator, and Mode of Demise against the manually annotated references. This enables the identification of true positives (TP), false positives (FP), and false negatives (FN), which are then used to compute standard precision ( $P$ ), recall ( $R$ ), and F1 score ( $F1$ ) for each constituent element. We adopt a soft approach: for *unspecified* constituents, we consider *null* predictions by the LLM as true negatives (TN), even when the model did not detect any other constituent.

Input	11lama3.1:70b									deepseek-r1:70b								
	Victim			Perpetrator			Mode of Demise			Victim			Perpetrator			Mode of Demise		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Generic Summary	0.85	0.27	0.41	0.74	0.21	0.32	0.55	0.17	0.26	0.84	0.24	0.37	0.59	0.17	0.26	0.54	0.17	0.26
Specialised Summary	<b>0.86</b>	<u>0.44</u>	<u>0.58</u>	<u>0.76</u>	<u>0.33</u>	<u>0.46</u>	<u>0.67</u>	<u>0.22</u>	<u>0.33</u>	<u>0.89</u>	<u>0.42</u>	<u>0.57</u>	<u>0.64</u>	<u>0.30</u>	<u>0.41</u>	<u>0.49</u>	<u>0.26</u>	<u>0.34</u>
Story	0.77	0.35	0.48	0.53	0.19	0.28	0.56	<u>0.26</u>	<u>0.35</u>	0.81	<u>0.37</u>	<u>0.51</u>	<u>0.54</u>	<u>0.22</u>	<u>0.31</u>	<u>0.55</u>	<u>0.25</u>	<u>0.34</u>

**Table 3:** Fine-grained evaluation metrics (Precision, Recall, F1) for Victim, Perpetrator, and Mode of Demise across the three input types for the two models tested. The best score is in bold, and the second-best is underlined.

**\*Coarse-grained evaluation** Building upon the fine-grained evaluation, the coarse-grained evaluation assesses the correctness of each death event triple as a whole. A triple is classified as:

<sup>3</sup> The categories *indirect speech* and *indirect thought* also include modes sometimes called *free indirect speech*, *represented thought*, etc. [32]

(a) *Correct*, if all its constituent elements (victim, perpetrator, and mode of demise) are TPs or TNs; (b) *Partially Correct*, if at least one of its constituent elements is a FP or FN; or (c) *Incorrect*, if none of its constituent elements is a TP or a TN. For each input type (full story, generic summary, and specialised summary), we compute the proportion of correct, partially correct, and incorrect death events relative to the total number of extracted death events.

Input	11ama3.1:70b			deepseek-r1:70b		
	Corr. (n, %)	Part. (n, %)	Inc. (n, %)	Corr. (n, %)	Part. (n, %)	Inc. (n, %)
Generic Summary	(31) 12%	(30) <u>12%</u>	(188) 76%	(26) 10%	(31) <u>12%</u>	(198) 78%
Specialised Summary	(63) <b>25%</b>	(36) <b>14%</b>	(155) <b>61%</b>	(49) <u>19%</u>	(49) <b>19%</b>	(159) <b>62%</b>
Story	(51) <u>20%</u>	(31) <u>12%</u>	(178) <u>68%</u>	(57) <b>22%</b>	(31) <u>12%</u>	(177) <u>67%</u>

**Table 4:** Coarse-grained evaluation results of extracted death event triples across three input types for the two models tested. Each triple is classified as *Correct* (Corr.), *Partially Correct* (Part.), or *Incorrect* (Inc.). The total number of extracted death event triples varies across model runs and input types, as models may extract different numbers of incorrect triples per run.

\*Manual review for evaluation validity Initially, this comparison is performed automatically using string-matching. However, since automatic metrics often struggle to fully capture the nuances of KE outputs, as with free-form QA outputs [9], especially with generative outputs [17], all comparison pairs are subsequently reviewed manually. This manual revision also accounts for cases where the victim or perpetrator is referred to using nominal or pronominal references, ensuring that models are not penalised for unresolved coreference (e.g. *his own daughter* for *Iphigenia*, in Figure 1). Examples illustrating how manual evaluation improves the fairness of model assessment are provided in Appendix 10, specifically in Tables 11, 12. The manual revision process was performed by the same experts who annotated the dataset (cf. Section 3.2).

## 4 Discussion

This section presents the highlights of our experimental results. We then analyse errors, first examining the types of errors in KE, then assessing whether certain stylistic features of the text, like the death event presentation mode or the narrator’s point of view (POV), are likely to elicit more errors.

### 4.1 Results highlights

\*Specialised summaries ensure the best KE performance By explicitly focusing on death events, the specialised summary prompt encourages the LLM to generate more exhaustive descriptions of these events, making them easier to extract through the KE prompt. The results of the fine-grained evaluation, presented in Table 3, confirm that for both tested models, the most advantageous input for the KE step is the specialised summary, ensuring the highest precision, recall, and *F1* score for both the Victim and Perpetrator elements. It is worth highlighting that the second-best precision values for the Victim and the Perpetrator are obtained with the generic summary in the input. A different trend is observed for the Mode of Demise. With 11ama3.1:70b, the highest *F1* score is obtained when the full story is used as input. Conversely, for deepseek-r1:70b, the *F1* score for Mode of Demise remains the same whether the input is the specialised summary or the full story. The results of the coarse-grained evaluation, reported in Table 4, further support this trend. For 11ama3.1:70b, the specialised summary yields the highest number of correctly and partially correctly extracted death events, along with the lowest number of incorrectly extracted ones. For deepseek-r1:70b, the highest number of correctly extracted death events is observed when using the full story as input. However, the specialised summary remains the input that ultimately results

in the lowest number of incorrect extractions. We report detailed examples of cases of successful extraction of death events from the specialised summary as compared to the other types of input in Appendix F.1.

\*Victim identification rarely causes partial errors A deeper analysis into the most recurring element responsible for a partially correct assessment reveals that the Victim is almost never the mistaken element, except a small proportion (3%) of cases in which 11ama3.1:70b extracts from the full story and deepseek-r1:70b from the generic summary. The identification of the Victim consistently achieves the highest  $F_1$  across all inputs and models tested. The most recurring error in partially correct death events when the input is a summary, whether generic or specialised, is the Mode of Demise. This element accounts for 74% of partially correct assessment in 11ama3.1:70b and 67% in deepseek-r1:70b. However, when the input is the story, the errors are more evenly distributed between the Perpetrator and the Mode of Demise: 42% and 55% for 11ama3.1:70b, and 52% and 48% for deepseek-r1:70b, respectively. These findings suggest that when events are recounted in summaries, key elements such as the Victim and Perpetrator are more consistently retained. In contrast, the Mode of Demise is seen as a lower-priority detail and may be omitted or described less explicitly. This supports the hypothesis that summarisation tends to preserve the most salient aspects of an event while filtering out less central details.

**LLMs struggle with knowledge extraction from fiction** The overall performance on the KE task is limited. The highest average  $F_1$  score across the three death event elements with the best-performing input is 0.45 for 11ama3.1:70b, while for deepseek-r1:70b is 0.44. Additionally, the percentage of correctly extracted death events is notably low, with a maximum of only 25%. These findings show the challenges of KE from fiction.

## 4.2 Error Analysis

### Missed death events dominate error types

In Table 8 and Table 9 (see Appendix C), we categorise errors into different types to analyse the prevalence of specific issues in extracting death events from different inputs.

First, we differentiate errors according to their scope: (a) errors at the summarisation level and (b) errors at the KE level. The former occurs when a death event is either missing from the summary or inaccurately represented in it.<sup>4</sup> The latter arises when a death event is correctly reported in the summary but is either not extracted or extracted incorrectly during the KE process. Errors at the KE level are applicable across all settings, regardless of whether the input is the generic summary, the specialised summary, or the full story.

Second, we classify errors belonging to each scope based on their type. Errors at the summarisation level can occur in two main forms: (i) a wrong death event reported, such as when a death event is reported by the LLM in the summary but not present in the full story, or when there is a mismatch between the Victim, the Perpetrator, or the Mode of Demise reported in the summary and those manually annotated, and (ii) a missing death event from the summary. We further divide the latter error type into two subcategories: (i) a missing death event of a major character, when the omitted death event pertains to a character so relevant to the story that it is explicitly named with a proper noun, and (ii) a missing death event of a minor character, when the omitted death event pertains to a character identified through a genitive construct (e.g., *Philomela's father*) or a generic descriptor (e.g., *villagers*).<sup>5</sup> Errors at the KE level can manifest in two main forms: (a) a wrong death event extracted, where the death event triple extracted from a given input contains an incorrect value for one or more of its constituent elements, and (b) a missed death event, where the

<sup>4</sup> These errors are specific to KE settings where the inputs are the generic or specialised summaries and do not apply (*N/A*) to the KE setting in which the input is the full story.

<sup>5</sup> The distinction between missed minor and major death events is particularly relevant for evaluating summary quality, as, in certain use cases, missing the deaths of minor characters may not significantly impact the usability of the summary.

death event triple is erroneously not extracted from the input. Similar to errors at the summarisation level, we further divide the missed death event scenario into two subcategories: (i) a missed death event of a major character, and (ii) a missed death event of a minor character.

Our analysis reveals that the vast majority of errors at the summarisation level across both models tested involve death events present in the story but omitted from the summary. For `llama3.1:70b`, missing death events of major characters constitute the highest proportion of summarisation errors (48% for generic and 44% for specialised summaries), followed by missing death events of minor characters (32% and 31%, respectively). A similar pattern emerges for `deepseek-r1:70b`, with missing death events of major characters accounting for 41% (generic) and 36% (specialised) of errors, and minor character deaths representing 36% and 32%, respectively. Wrong death events reported remain relatively low across both models (7% for generic and 13% for specialised summaries for `llama3.1:70b`, 8% and 22% for `deepseek-r1:70b`). Combined, missing death events account for approximately 75–80% of all summarisation errors, highlighting the models’ tendency to omit rather than fabricate or misrepresent death events in the summaries.

At the KE level, the distribution of error types varies considerably depending on the input. When extracting from summaries (generic or specialised), errors are relatively low and distributed across categories, with wrong extractions of 6% and 7% for `llama3.1:70b`, and 3% and 1.5% for `deepseek-r1:70b`, and missed death events of 8% and 5% for `llama3.1:70b` and 7% and 4.5% for `deepseek-r1:70b`. Considering the full story as an input, for `llama3.1:70b` processing full stories, missed death events account for 79% of all KE errors (46% major characters, 33% minor characters), while for `deepseek-r1:70b`, this figure reaches 73% (34% major, 39% minor). Wrong extractions are also notably higher when processing full stories (20% and 27%) compared to summary inputs, suggesting that the increased complexity and length of full narratives pose substantial challenges for accurate KE.

In general, missed death events dominate extraction errors, confirming a trend already observed in Table 3: for each constituent element of the death event triples, *Recall* in KE extraction is very low. This trend underscores that most errors can be attributed either to the LLM failing to identify and report a death event in the summary or to the LLM failing to recognise the death event directly from the full story.

\*Indirect presentation modes and third-person narration facilitate KE

Table 5 highlights the influence of presentation modes on the correctness of death event extraction across the input types tested. For all input types and both LLMs, death events presented in *direct speech* are the most challenging to extract, exhibiting the lowest rates of correct extractions. However, prior summarisation appears to mitigate this issue by making such events more explicit, ultimately facilitating their extraction in the KE step. An example of this phenomenon is provided in Table 13 reported in Appendix F.1. In this case, although death events are clearly stated in direct speech within the excerpt of the original story, both LLMs fail to extract them when the input is the full story. The same applies when the input is the generic summary. However, when the specialised summary is used as input, the death events are reformulated more explicitly, enabling successful extraction by the KE step. We report other similar examples in Appendix F.2.

Conversely, death events presented in indirect modes, either through the narrator or indirect thought, are consistently easier to extract across all input types for both LLMs. This suggests that the indirect presentation of a death event inherently facilitates extraction, reinforcing the benefit of incorporating a summarisation step. Since summarisation tends to transform direct speech into indirect reporting, it helps structure the information in a way that better aligns with LLMs’ KE capabilities. Table 6 corroborates these findings, showing that KE is more accurate when stories are narrated in the third person. Across all models and input types tested, third-person narratives yield the highest proportion of correct extractions while also exhibiting the lowest number

of missed death events. This trend suggests that third-person narration provides a clearer, more explicit presentation of events, reducing ambiguity and improving LLMs’ accuracy in extracting key information.

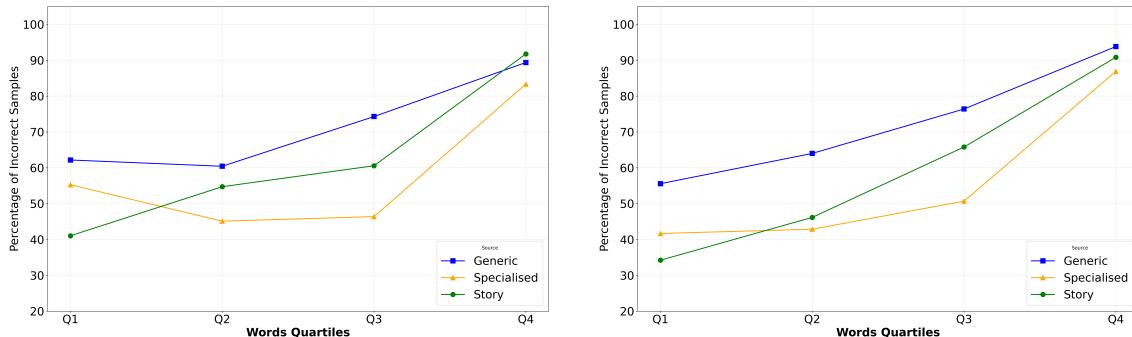
11ama3.1:70b											
Presentation mode	Total	Generic Summary			Specialised Summary			Story			
		Corr.	Part.	Miss.	Corr.	Part.	Miss.	Corr.	Part.	Miss.	
narrator	165	15%	<b>11%</b>	74%	<b>34%</b>	15%	52%	<b>27%</b>	11%	<b>62%</b>	
direct speech	38	8%	<b>11%</b>	82%	3%	<b>21%</b>	76%	0%	13%	87%	
indirect thought	37	<b>19%</b>	8%	<b>73%</b>	14%	11%	76%	16%	<b>22%</b>	<b>62%</b>	
indirect speech	5	0%	20%	80%	40%	0%	60%	20%	20%	60%	
direct thought	3	0%	100%	0%	100%	0%	0%	100%	0%	0%	

deepseek-r1:70b											
Presentation mode	Total	Generic Summary			Specialised Summary			Story			
		Corr.	Part.	Miss.	Corr.	Part.	Miss.	Corr.	Part.	Miss.	
narrator	165	12%	<b>13%</b>	75%	<b>23%</b>	22%	55%	25%	<b>13%</b>	62%	
direct speech	38	5%	8%	87%	8%	8%	84%	5%	8%	87%	
indirect thought	37	<b>14%</b>	11%	76%	19%	19%	62%	<b>35%</b>	8%	<b>57%</b>	
indirect speech	5	0%	20%	80%	20%	40%	40%	20%	20%	60%	
direct thought	3	0%	67%	33%	33%	33%	33%	0%	67%	33%	

**Table 5:** Error analysis by presentation mode for 11ama3.1:70b and deepseek-r1:70b. Results for presentation modes with a low number of samples are reported but excluded from the comparison.

\*KE worsens as story length and word entropy increase



**Figure 2:** Comparison of incorrectly extracted death events by word count quartiles. Left: 11ama3.1:70b. Right: deepseek-r1:70b. Q1-Q4 word count ranges: [400-1000], [1085-1677], [1686-3488], [4072-7367]. Each quartile contains 14-15 works. Each plot includes one line for each input type (generic, specialised, story).

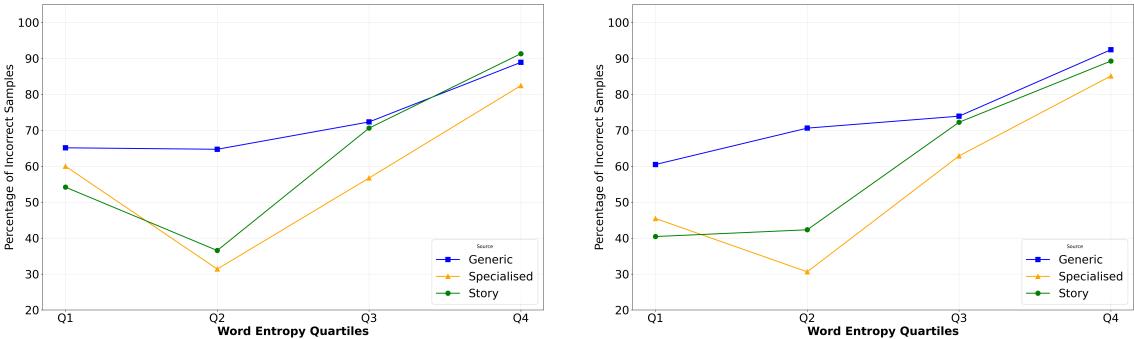
Long contexts pose challenges for LLMs [20]. Table 2 shows substantial variation in story length, ranging from short (400 words) to long (7367 words) narratives. Word entropy, a measure that indicates the predictability of words’ co-occurrence in a text [4; 31], with higher levels signifying higher unpredictability, varies in a range from 7.05 to 9.25 in our corpus. Figures 2 and 3 demonstrate that incorrect death event extractions increase with both story length and word entropy. Specialised summaries produce the fewest incorrect extractions in most conditions. However, for

11lama3.1:70b											
Narrator POV	Total	Generic Summary			Specialised Summary			Story			
		Corr.	Part.	Miss.	Corr.	Part.	Miss.	Corr.	Part.	Miss.	
third	126	18%	15%	67%	27%	15%	58%	23%	12%	65%	
first	112	11%	7%	83%	25%	12%	62%	20%	<b>14%</b>	66%	
second	10	0%	22%	78%	20%	40%	40%	20%	10%	70%	

deepseek-r1:70b											
Narrator POV	Total	Generic Summary			Specialised Summary			Story			
		Corr.	Part.	Miss.	Corr.	Part.	Miss.	Corr.	Part.	Miss.	
third	126	15%	16%	69%	26%	16%	58%	30%	10%	60%	
first	112	5%	10%	85%	13%	<b>25%</b>	62%	16%	<b>14%</b>	70%	
second	10	30%	0%	70%	30%	10%	60%	20%	20%	60%	

**Table 6:** Error analysis by narrator’s point of view (POV) for 11lama3.1:70b and deepseek-r1:70b. Results for narrator’s POVs with a low number of samples are reported but excluded from the comparison.



**Figure 3:** Comparison of incorrectly extracted death events by word entropy quartiles. Left: 11lama3.1:70b. Right: deepseek-r1:70b. Q1-Q4 entropy ranges: [7.1-7.7], [7.8-8.1], [8.1-8.5], [8.5-9.2]. Each quartile contains 14-15 works. Each plot includes one line for each input type (generic, specialised, story).

the shortest and most predictable stories (first quartiles of word count and entropy), full stories yield more accurate extractions than specialised summaries. The advantages of specialised summaries peak at intermediate complexity levels (second and third quartiles), where the gap between specialised summaries and other input types is largest. At the extreme quartile (longest stories and highest entropy), all input types show similar performance levels, with specialised summaries maintaining slight advantages. This pattern suggests that summarisation becomes increasingly beneficial as text length and predictability increase, but offers diminishing returns for the most challenging texts.

We calculate the Spearman correlation to examine the relationship between incorrectly extracted death event triples and the word length and entropy of the stories to which they belong. Our analysis reveals that summarisation consistently reduces the correlation between KE errors and story length and entropy metrics across both evaluated LLMs. For full stories, we observed the strongest correlations (all statistically significant at  $p < 0.001$ ), with `deepseek-r1:70b` showing correlations of 0.49 for word count and 0.46 for word entropy, while `11ama3.1:70b` exhibited correlations of 0.44 and 0.42, respectively. When using specialised summaries, these correlations decreased moderately to 0.41 for `deepseek-r1:70b`, and more substantially to 0.31 and 0.30 for `11ama3.1:70b`. Generic summaries showed the weakest correlations, with values dropping to 0.32 and 0.30 for `deepseek-r1:70b`, and 0.27 and 0.24 for `11ama3.1:70b`. This consistent pattern across both LLMs suggests that summarisation can effectively reduce the dependency between extraction accuracy and challenging stories, which are characterised by higher word length and entropy.

We also conducted quartile analysis and Spearman’s  $\rho$  computations for textual features and reader reception metrics, but found no clear trends or correlations with extraction errors. These results are reported in the Appendix G, Table 7.

#### LLMs’ background knowledge may hinder KE quality

As seen in Tables 3 and 4 death event extraction from fiction is characterised by low Recall, while Precision tends to achieve higher scores. However, false positives can still occur. LLMs may generate hallucinations [14], incorporating their background knowledge of the story’s characters and settings into the summary or in the KE step. This phenomenon is especially significant in the context of fanfiction, where characters and narrative universes are often drawn from pre-existing works. Consequently, the LLM may inject information into the summary based on its prior training rather than the input text. This issue is especially relevant to our case study, which uses narratives inspired by Greek mythology, a domain well-known and extensively documented online.

We observed an illustrative example of this phenomenon with `11ama3.1:70b`. When generating a generic summary, the LLM produced the sentence: “However, Clytaemnestra appears behind him, seeking revenge for their daughter Iphigenia’s death, and strikes Agamemnon down with her sword.” From this summary sentence, our KE step incorrectly identified a death event with Iphigenia as the Victim, Agamemnon as the Perpetrator, and stabbing as the Mode of Demise; all three elements of this extracted triple were false positives. Crucially, this event does not appear anywhere in the corresponding full story.<sup>6</sup> The presence of the characters *Agamemnon*, *Cassandra*, and *Clytemnestra* in the actual narrative appears to have prompted the LLM to incorporate background knowledge about Iphigenia’s death—a well-documented mythological antecedent that serves as motivation for Clytemnestra’s actions in the Greek tragedy—even though this backstory was not present in the input text. This demonstrates how models can inadvertently inject external knowledge into summaries, leading to extracted information that, while mythologically accurate, constitutes a false positive for our task. We report further examples of such false positives in Appendix F.3.

<sup>6</sup> The full story has title *Cassandra’s Song* and workid 734141

## 5 Conclusion

This research investigated the feasibility of using LLMs for automated knowledge extraction of death events from fictional narratives, specifically Greek mythology fanfiction. We explored the impact of different input presentations: full text, generic summaries, and specialised summaries tailored for death event extraction. Our findings indicate that specialised summaries can offer some performance improvements, especially with shorter narratives, but gains are modest (+0.09 F1), leaving the task challenging. However, this advantage diminishes with increasing story length and word entropy, where all input types lead to low performance. Correlation analysis reveals stronger associations between extraction errors and story length and word entropy when processing full stories. These correlations weaken with summary inputs, suggesting that summarisation may be a beneficial preprocessing strategy for event extraction from fiction. We also showed examples of the detrimental effects of knowledge leakage from pre-training, leading to false-positive extractions. This issue is particularly prominent in domains like Greek mythology, where widely available background information can contaminate the LLM’s output. Moreover, the way information is presented significantly impacts KE performance. First-person narratives and direct speech consistently yielded worse extractions compared to other narrative perspectives, likely due to their less explicit presentation of events.

Overall, the performance of unsupervised LLMs in this KE task was poor, with most problems due to the low Recall. In other words, LLMs may perform sufficiently well to extract information from passages that are relatively simple and more explicit, but still perform extremely bad with passages that are stylistically more unpredictable. Given that indirect communication and artistic uses of language are characteristics of many narratives and fictional texts, the observed limitations suggest that relying solely on unsupervised LLMs for accurate and reliable information extraction from such texts is currently not feasible. Future research should explore alternative KE methods, domain-specific fine-tuning, and dataset diversification to address these challenges.

The specific case study addressed by this research is illustrative of the potential of KE for comparative analysis. Mythology is a rich domain with a long history of adaptation and can serve as an exemplary context for studying how fictional characters meet their demise. By matching the mode of demise in modern adaptations (fanfiction) to their original counterparts, we can investigate innovation and the reframing of plots in the reception of the Ancient Greek mythological tradition. Additionally, fanfiction comes in various forms—from poetic retellings to dialogues between ancient characters, to streams-of-consciousness narratives. This study, therefore, provides a valuable starting point for event extraction across different domains, such as the analysis of screenplays.

## 6 Limitations

This work has several limitations that we plan to address in subsequent studies. We performed automatic alignment of Modes of Demise manually, as LLMs output cannot always be matched directly (e.g. *turning to stone* and *petrification*). A valuable contribution to the study of events in fiction would be an automatic or semi-automatic method to align these categories, where applicable.

We plan to extend the study of event extraction to more domains. While fanfiction stories on Ancient Greek mythology serve as an interesting case study, we aim to include other types of fiction in subsequent studies.

We also plan to extend the dataset to more than one event category. As demonstrated in this paper, death events are expressed in many forms, which allows us to test the capabilities of KE models in various scenarios and across presentation modes. However, other categories, such as the development of relationships between characters, are equally interesting. Further studies in this direction will allow us to come to more generalizable conclusions about automatic KE from

fictional stories.

## 7 Ethical considerations

We followed best practices for using online fandom data: obtaining permission, attribution, giving back, and learning community norms [11]. People within fan communities often have protective views regarding their data and its use by researchers. One of the risks is that of amplifying fan content to an audience it was never intended for, which could compromise the privacy and context of that data. Fan-generated data are highly contextual to the owner and their specific privacy needs. To address these concerns, we gained explicit permission from fans whose stories are being quoted and analysed in this research.

## 8 Acknowledgments

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

## References

- [1] Aguilar, Jacqueline, Beller, Charley, McNamee, Paul, Van Durme, Benjamin, Strassel, Stephanie, Song, Zhiyi, and Ellis, Joe. “A Comparison of the Events and Relations Across ACE, ERE, TAC-KBP, and FrameNet Annotation Standards”. In: ed. by Teruko Mitamura, Eduard Hovy, and Martha Palmer. Baltimore, Maryland, USA, June 2014, pp. 45–53. DOI: 10.3115/v1/W14-2907. URL: W14-2907/.
- [2] Bamman, David. “BookNLP”. 2021. URL: <https://github.com/booknlp/booknlp>.
- [3] Bamman, David, Lewke, Olivia, and Mansoor, Anya. “An Annotated Dataset of Coreference in English Literature”. eng. In: ed. by Nicoletta Calzolari et al. Marseille, France: European Language Resources Association, May 2020, pp. 44–54. ISBN: 979-10-95546-34-4. URL: 2020.lrec-1.6/.
- [4] Bentz, Christian and Alikaniotis, Dimitrios. “The word entropy of natural languages”. 2016. arXiv: 1606.06996 [cs.CL]. URL: <https://arxiv.org/abs/1606.06996>.
- [5] Bhat, Gayatri, Saluja, Avneesh, Dye, Melody, and Florjanczyk, Jan. “Hierarchical Encoders for Modeling and Interpreting Screenplays”. In: ed. by Nader Akoury, Faeze Brahman, Snigdha Chaturvedi, Elizabeth Clark, Mohit Iyyer, and Lara J. Martin. Virtual, June 2021, pp. 1–12. DOI: 10.18653/v1/2021.nuse-1.1. URL: 2021.nuse-1.1/.
- [6] Biderman, Stella, Bicheno, Kieran, and Gao, Leo. “Datasheet for the Pile”. In: *CoRR abs/2201.07311* (2022). arXiv: 2201.07311. URL: <https://arxiv.org/abs/2201.07311>.
- [7] Bizzoni, Yuri, Feldkamp, Pascale, Lassen, Ida Marie, Jacobsen, Mia, Thomsen, Mads Rosendahl, and Nielbo, Kristoffer. “Good Books are Complex Matters: Gauging Complexity Profiles Across Diverse Categories of Perceived Literary Quality”. 2024. arXiv: 2404.04022 [cs.CL]. URL: <https://arxiv.org/abs/2404.04022>.
- [8] Chang, Kent, Cramer, Mackenzie, Soni, Sandeep, and Bamman, David. “Speak, Memory: An Archaeology of Books Known to ChatGPT/GPT-4”. In: ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore, Dec. 2023, pp. 7312–7327. DOI: 10.18653/v1/2023.emnlp-main.453. URL: 2023.emnlp-main.453/.
- [9] Chen, Anthony, Stanovsky, Gabriel, Singh, Sameer, and Gardner, Matt. “Evaluating Question Answering Evaluation”. In: ed. by Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. Hong Kong, China, Nov. 2019, pp. 119–124. DOI: 10.18653/v1/D19-5817. URL: D19-5817/.
- [10] DeepSeek-AI. “DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning”. 2025. arXiv: 2501.12948 [cs.CL]. URL: <https://arxiv.org/abs/2501.12948>.
- [11] Dym, Brianna and Fiesler, Casey. “Ethical and privacy considerations for research using online fandom data”. In: *Transformative Works and Cultures* 33 (June 2020). ISSN: 1941-2258. DOI: 10.3983/twc.2020.1733.
- [12] Elazar, Yanai et al. “What’s In My Big Data?” In: *The Twelfth International Conference on Learning Representations*. 2024. URL: <https://openreview.net/forum?id=RvfPn0kPV4>.
- [13] Fiesler, Casey, Morrison, Shannon, and Bruckman, Amy S. “An archive of their own: A case study of feminist HCI and values in design”. In: *Proceedings of the 2016 CHI conference on human factors in computing systems*. 2016, pp. 2574–2585.

- [14] Huang, Lei et al. “A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions”. In: *ACM Trans. Inf. Syst.* 43, no. 2 (Jan. 2025). ISSN: 1046-8188. DOI: 10.1145/3703155.
- [15] Jacobsen, Mia, Bizzoni, Yuri, Moreira, Pascale Feldkamp, and Nielbo, Kristoffer L. “Patterns of Quality: Comparing Reader Reception Across Fanfiction and Commercially Published Literature”. English. In: *2024 Computational Humanities Research Conference, CHR 2024*. Vol. 3834. CEUR Workshop Proceedings. Publisher Copyright: © 2024 Copyright for this paper by its authors.; 2024 Computational Humanities Research Conference, CHR 2024 ; Conference date: 04-12-2024 Through 06-12-2024. CEUR-WS, 2024, pp. 718–739.
- [16] Jing, Yimin, Xiong, Deyi, and Yan, Zhen. “BiPaR: A Bilingual Parallel Dataset for Multi-lingual and Cross-lingual Reading Comprehension on Novels”. In: ed. by Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan. Hong Kong, China, Nov. 2019, pp. 2452–2462. DOI: 10.18653/v1/D19-1249. URL: D19-1249/.
- [17] Kamalloo, Ehsan, Dziri, Nouha, Clarke, Charles, and Rafiei, Davood. “Evaluating Open-Domain Question Answering in the Era of Large Language Models”. In: ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada, July 2023, pp. 5591–5606. DOI: 10.18653/v1/2023.acl-long.307. URL: 2023.acl-long.307/.
- [18] Kočiský, Tomáš, Schwarz, Jonathan, Blunsom, Phil, Dyer, Chris, Hermann, Karl Moritz, Melis, Gábor, and Grefenstette, Edward. “The NarrativeQA Reading Comprehension Challenge”. In: *Transactions of the Association for Computational Linguistics* 6 (2018), ed. by Lillian Lee, Mark Johnson, Kristina Toutanova, and Brian Roark, pp. 317–328. DOI: 10.1162/tacl\_a\_00023. URL: Q18-1023/.
- [19] Li, Sha, Zhan, Qiusi, Conger, Kathryn, Palmer, Martha, Ji, Heng, and Han, Jiawei. “GLEN: General-Purpose Event Detection for Thousands of Types”. In: ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore, Dec. 2023, pp. 2823–2838. DOI: 10.18653/v1/2023.emnlp-main.170. URL: 2023.emnlp-main.170/.
- [20] Liu, Nelson F., Lin, Kevin, Hewitt, John, Paranjape, Ashwin, Bevilacqua, Michele, Petroni, Fabio, and Liang, Percy. “Lost in the Middle: How Language Models Use Long Contexts”. In: *Transactions of the Association for Computational Linguistics* 12 (2024), pp. 157–173. DOI: 10.1162/tacl\_a\_00638. URL: 2024.tacl-1.9/.
- [21] Llama Team, AI @ Meta. “The Llama 3 Herd of Models”. 2024. arXiv: 2407.21783 [cs.AI]. URL: <https://arxiv.org/abs/2407.21783>.
- [22] Lucy, Li, Griffiths, Camilla, Levine, Sarah, Eberhardt, Jennifer L., Demszky, Dorottya, and Bamman, David. “Tell, Don’t Show: Leveraging Language Models’ Abstractive Retellings to Model Literary Themes”. 2025. arXiv: 2505.23166 [cs.CL]. URL: <https://arxiv.org/abs/2505.23166>.
- [23] Meehan, Margaret, Malenfant, Dane, and Piper, Andrew. “Causality Mining in Fiction”. In: *Text2Story@ECIR*. 2022. URL: <https://api.semanticscholar.org/CorpusID:248089390>.
- [24] Morin, Olivier, Acerbi, Alberto, and Sobchuk, Oleg. “Why people die in novels: testing the ordeal simulation hypothesis”. In: *Palgrave Communications* 5, no. 62 (June 2019). DOI: <https://doi.org/10.1057/s41599-019-0267-0>.
- [25] Palmer, Martha, Bonial, Claire, and McCarthy, Diana. “SemLink+: FrameNet, VerbNet and Event Ontologies”. In: ed. by Miriam R. L. Petrucc and Gerard de Melo. Baltimore, MD, USA, June 2014, pp. 13–17. DOI: 10.3115/v1/W14-3004. URL: W14-3004/.

- [26] Pianzola, Federico, Acerbi, Alberto, and Rebora, Simone. “Cultural Accumulation and Improvement in Online Fan Fiction”. In: *CHR 2020: Workshop on Computational Humanities Research*. Vol. 2723. CEUR Workshop Proceedings. Amsterdam, The Netherlands, Nov. 2020, pp. 2–11. URL: <http://ceur-ws.org/Vol-2723/>.
- [27] Piper, Andrew. “What Do Characters Do? The Embodied Agency of Fictional Characters”. In: *Journal of Computational Literary Studies* 2 (1 Mar. 2024), pp. 1–12, ISSN: 2940-1348. DOI: 10.48694/jcls.3589.
- [28] Pradhan, Sameer, Bonn, Julia, Myers, Skatje, Conger, Kathryn, O’gorman, Tim, Gung, James, Wright-bettner, Kristin, and Palmer, Martha. “PropBank Comes of Age—Larger, Smarter, and more Diverse”. In: ed. by Vivi Nastase, Ellie Pavlick, Mohammad Taher Pilehvar, Jose Camacho-Collados, and Alessandro Raganato. Seattle, Washington, July 2022, pp. 278–288. DOI: 10.18653/v1/2022.starsem-1.24. URL: 2022.starsem-1.24/.
- [29] Rajpurkar, Pranav, Zhang, Jian, Lopyrev, Konstantin, and Liang, Percy. “SQuAD: 100,000+ Questions for Machine Comprehension of Text”. In: ed. by Jian Su, Kevin Duh, and Xavier Carreras. Austin, Texas, Nov. 2016, pp. 2383–2392. DOI: 10.18653/v1/D16-1264. URL: D16-1264/.
- [30] Richardson, Matthew, Burges, Christopher J.C., and Renshaw, Erin. “MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text”. In: ed. by David Yarowsky, Timothy Baldwin, Anna Korhonen, Karen Livescu, and Steven Bethard. Seattle, Washington, USA, Oct. 2013, pp. 193–203. URL: D13-1020/.
- [31] Schlechtweg, Dominik, Eckmann, Stefanie, Santus, Enrico, Schulte im Walde, Sabine, and Hole, Daniel. “German in Flux: Detecting Metaphoric Change via Word Entropy”. In: ed. by Roger Levy and Lucia Specia. Vancouver, Canada, Aug. 2017, pp. 354–367. DOI: 10.18653/v1/K17-1036. URL: K17-1036/.
- [32] Semino, Elena and Short, Mick. *Corpus Stylistics: Speech, Writing and Thought Presentation in a Corpus of English Writing*. London: Routledge, 2004.
- [33] Sims, Matthew, Park, Jong Ho, and Bamman, David. “Literary Event Detection”. In: ed. by Anna Korhonen, David Traum, and Lluís Márquez. Florence, Italy, July 2019, pp. 3623–3634. DOI: 10.18653/v1/P19-1353. URL: P19-1353/.
- [34] Steg, Max, Slot, Karlo, and Pianzola, Federico. “Computational Detection of Narrativity: A Comparison Using Textual Features and Reader Response”. In: ed. by Stefania Degaetano, Anna Kazantseva, Nils Reiter, and Stan Szpakowicz. Gyeongju, Republic of Korea: International Conference on Computational Linguistics, Oct. 2022, pp. 105–114. URL: 2022.latechclf1-1.13/.
- [35] Vauth, Michael and Gius, Evelyn. “Event Annotations of Prose”. In: *Journal of Open Humanities Data* (Aug. 2022). DOI: 10.5334/johd.83.
- [36] Xu, Ying et al. “Fantastic Questions and Where to Find Them: FairytaleQA – An Authentic Dataset for Narrative Comprehension”. In: ed. by Smaranda Muresan, Preslav Nakov, and Aline Villavicencio. Dublin, Ireland, May 2022, pp. 447–460. DOI: 10.18653/v1/2022.acl-long.34. URL: 2022.acl-long.34/.
- [37] Yu, Linhao, Liu, Qun, and Xiong, Deyi. “LFED: A Literary Fiction Evaluation Dataset for Large Language Models”. In: ed. by Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue. Torino, Italia: ELRA and ICCL, May 2024, pp. 10466–10475. URL: 2024.lrec-main.915/.

- [38] Yuan, Xinfeng, Yuan, Siyu, Cui, Yuhang, Lin, Tianhe, Wang, Xintao, Xu, Rui, Chen, Jiangjie, and Yang, Deqing. “Evaluating Character Understanding of Large Language Models via Character Profiling from Fictional Works”. In: ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA, Nov. 2024, pp. 8015–8036. DOI: 10.18653/v1/2024.emnlp-main.456. URL: 2024.emnlp-main.456/.
- [39] Zhao, Runcong, Zhu, Qinglin, Xu, Hainiu, Li, Jiazheng, Zhou, Yuxiang, He, Yulan, and Gui, Lin. “Large Language Models Fall Short: Understanding Complex Relationships in Detective Narratives”. In: ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand, Aug. 2024, pp. 7618–7638. DOI: 10.18653/v1/2024.findings-acl.454. URL: 2024.findings-acl.454/.

## A Dataset textual features and reader reception statistics

Metric	Mean	Median	Std Dev	Min	Max
<i>Textual Features</i>					
Dale-Chall readability	6.72	6.65	0.61	5.91	8.89
Hurst exponent	0.59	0.57	0.11	0.28	0.91
<i>Reader Reception Metrics</i>					
Kudos	42.3	21	47	1	190
Hits	1,135.1	488	2,268.5	9	16,809
Appreciation (kudos/hits)	0.061	0.049	0.061	0.006	0.446

**Table 7:** Dataset textual features and reader reception statistics. Appreciation is calculated as kudos/hits ratio.

In this section, we report Table 7, in which we compute textual features and reader reception characteristics of the stories.

## B Prompts

This appendix contains all the prompts used in our experiments.

### B.1 Knowledge Extraction prompts

#### B.1.1 Knowledge Extraction from Summaries

Below is the prompt for extracting character death events from summaries:

```
Answer the following questions based on the provided story summary:  
<STORY SUMMARY>
```

For each death event reported in the summary, answer the following:

1. Character Death (Yes/No)
2. Mode of Demise
3. Victim
4. Perpetrator

Your answers must be made using a single word or as few words as possible. For example:

Character Death: yes; Mode of Demise: tearing to pieces;  
Victim: Orpheus; Perpetrator: Maenads.

There may be multiple death events in the summary. Please provide separate answers for each event.

You MUST STRICTLY RELY on the PROVIDED SUMMARY ONLY. You MUST NOT provide answers based on any information outside the text.

Each group of answers for each death event should be output in JSONL format, as in the following example:

```
{"id": "<id>", "death_event_1": {"Character Death": "", "Mode of Demise": "", "Victim": "", "Perpetrator": ""}, "death_event_2": {"Character Death": "", "Mode of Demise": "", "Victim": "", "Perpetrator": ""}}
```

For this summary, use <ID> as the work identifier. You MUST return the generated JSONL only. Do not write anything else.

### B.1.2 Knowledge Extraction from Stories

Below is the prompt for extracting character death events from full stories:

Answer the following questions based on the provided short story:  
<STORY>

For each death event reported in the story, identify the following:

1. Character Death (Yes/No)
2. Mode of Demise
3. Victim
4. Perpetrator

Your answers must be made using a single word or as few words as possible. For example:

Character Death: yes; Mode of Demise: tearing to pieces;  
Victim: Orpheus; Perpetrator: Maenads.

There may be multiple death events in the story. Please provide separate answers for each event.

You MUST STRICTLY RELY on the PROVIDED STORY ONLY. You MUST NOT provide answers based on any information outside the text.

Each group of answers for each death event should be output in JSONL format, as in the following example:

```
{"id": "<id>", "death_event_1": {"Character Death": "", "Mode of Demise": "", "Victim": "", "Perpetrator": ""}, "death_event_2": {"Character Death": "", "Mode of Demise": "", "Victim": "", "Perpetrator": ""}}
```

For this story, use <ID> as the work identifier. You MUST return the generated JSONL only. Do not write anything else.

## B.2 Summarisation Prompts

### B.2.1 Generic Summary

Below is the prompt used for generic summary generation:

You are an expert in short story summarisation. Create a summary of the provided short story:

<STORY>

Rely STRICTLY on the provided text. It is FORBIDDEN to include any information that is not present in the text.

Your output is a coherent and cohesive summary that encapsulates the essence of the given short story in a few sentences.

Make sure to capture all the events of the story. Return the generated summary only. Do not write anything else.

### B.2.2 Specialised Summary

We report here the prompt used for specialised summary generation:

You are an expert in short story summarisation. Create a summary of the provided short story:

<STORY>

Make sure to include information about all the character deaths mentioned, specifying who is/are the murderer(s), what is/are the mode(s) of demise, who is/are the victim(s), who is/are the perpetrator(s). Rely STRICTLY on the provided text. It is FORBIDDEN to include any information that is not present in the text.

Your output is a coherent and cohesive summary that encapsulates the essence of the given short story in a few sentences.

Make sure to capture all the events of the story. Return the generated summary only. Do not write anything else.

## C Error Type Distribution Across Input Types

This section provides a detailed breakdown of error types occurring in the character death event extraction task across both models tested (llama3.1:70b and deepseek-r1:70b). The errors are categorised according to their scope (summarisation vs. knowledge extraction) and type (wrong events vs. missed events, further divided by character importance).

Table 8 presents the error distribution when using generic and specialised summaries as inputs, encompassing both summarisation-level errors (where death events are incorrectly represented or omitted) and knowledge extraction errors (where death events are not correctly extracted). Table 9 focuses on knowledge extraction errors when the full story serves as the input, where summarisation-level errors are not applicable. The percentages represent the proportion of each error type relative to the total number of errors observed in each experimental condition.

Error types	1lrama3.1:70b		deepseek-r1:70b	
	Generic	Specialised	Generic	Specialised
<b>Summarisation</b>				
Wrong death event reported	7%	13%	8%	22%
Missing death event (Major character)	<b>48%</b>	<b>44%</b>	<b>41%</b>	<b>36%</b>
Missing death event (Minor character)	<u>32%</u>	<u>31%</u>	<u>36%</u>	<u>32%</u>
<b>Knowledge Extraction</b>				
Wrong death event extracted	6%	7%	3%	1.5%
Missed death event (Major character)	8%	5%	7%	4.5%
Missed death event (Minor character)	0%	0%	5%	4%

**Table 8:** Categories of error types when extracting from summaries (generic and specialised) across two models. Highest in bold, second highest underlined.

Error types	1lrama3.1:70b	deepseek-r1:70b
	Story	Story
<b>Knowledge Extraction</b>		
Wrong death event extracted	20%	27%
Missed death event (Major character)	<b>46%</b>	<u>34%</u>
Missed death event (Minor character)	<u>33%</u>	<b>39%</b>

**Table 9:** Categories of error types when extracting from full stories across two models. Highest in bold, second highest underlined.

## D Presentation modes and narrator POV examples

We report in Table 10 examples of death events labelled by presentation mode and narrator POV.

## E Manual review for evaluation examples

To better understand the limitations of automatic evaluation and the necessity of manual review, we analyze specific cases where string-matching fail to capture the correct KE annotations. Tables 11, 12 presents illustrative examples of such cases, focusing on instances where the LLM’s outputs differ from gold annotations due to unresolved coreference or lexical variations.

One common issue arises when the Victim or Perpetrator is referred to using pronouns or indirect nominal references rather than explicit mentions. In the first example, the manually annotated Victim is *Theofylaktos*, while the Victim automatically extracted from the different input types are, respectively, *Medusa's human partner*, *old man*, *the male protagonist*. Considering the story, all of those three extractions can be considered correct. However, in the story, the name *Theofylaktos* only occurs once, making it difficult for the model to extract it as the Victim in a zero-shot setting. Furthermore, the story from which this death event is extracted is 5,714 words long, which is almost double the average word length in our dataset and almost five times the median. While an automatic string-matching approach may classify such cases as incorrect, manual evaluation allows for resolving these references, preventing penalisation. Similarly, variations in lexical choices—such as synonyms for the Mode of Demise—can lead to false negatives under strict string comparison. In the second example, the manually annotated Mode of Demise is *slaying*, while the one automatically extracted from the generic summary is *pickaxe*. From the sentence reported in the table, it is straightforward to see that *pickaxe* is the instrument used for the murder. We decided to consider this case correct and to consider extractions such as *murder* or *beating* wrong instead as too generic.

<b>Presentation Mode</b>	<b>POV</b>	<b>Example</b>	<b>ID</b>	<b>Title</b>
Narrator	Third person	The room was still locked, the servant hanged, suitors dead, Penelope and Telemachus safe.	57811963	<i>Witch, Goddes, and home again</i>
Direct Speech	Second person	and you arch up into him as he says, "Because I killed Patroclus, Achilles won't let me live."	17464	<i>Requiem</i>
Indirect Thought	Third person	Her own nightmares about miners involved hanged men, ropes still around their necks, accusing her.	30281253	<i>Cast Down but Not Destroyed</i>
Indirect Speech	Third person	From the soldier she learned that Menoiteus had exiled him to Pthita for a supposed murder of another boy.	24350869	<i>Mela</i>
Direct Thought	First person	He will bring me to his wife, who schemes with his cousin to kill him in revenge for the murder of her Iphigenia. She will ease him up with flattery before trapping him in a net in a bathtub, like a large mackerel caught by a fisherman, and will slay him with a sword.	47136421	<i>Cassandra on the beach</i>

**Table 10:** Examples of presentation modes for death events, including corresponding IDs and titles from the dataset.

---

*In the Eye of the Beholder*, 51106009

*Direct speech, first: The change is gradual, like a slow-tide, but unstoppable. His flesh turns to gray stone, and it moves up his body. His hands in mine become cold, and he moves away. He keeps his eyes locked onto mine.*  
*(Theofylaktos, Medusa, petrification)*

---

**Model:** llama3.1:70b

---

Input	Victim	Perpetrator	Mode of Demise	Summary Sentence
Generic Summary	<b>Medusa's human partner</b>	-	-	The narrator, Medusa, lives with a human man for many years, hiding her true reptilian form from him. As he ages and his health declines, he eventually discovers her secret but is not repelled by it. In fact, he finds her snakes beautiful and asks to see her eyes, which shine like mother of pearl. With his dying breath, he thanks Medusa for a life well-lived and a painless death, and turns into stone as he passes away. Despite her grief, Medusa feels her heart is light, knowing she will see him again in the afterlife and that their love has given her the strength to carry on.
Specialised Summary	<b>old man</b>	<b>Medusa</b>	<b>petrification</b>	Medusa's lover, an old man with weak eyesight, dies after being petrified by Medusa's gaze at his own request, allowing him to die painlessly and with dignity. He is the victim, and Medusa is the unwitting perpetrator of his demise. Athena is indirectly responsible for his death, as she had previously laughed in Medusa's head, foreshadowing the event.
Story	<b>the male protagonist</b>	<b>Athena</b>	<b>petrification</b>	N/A

**Table 11:** Example of manual evaluation handling alignment for Victim and Perpetrator.

---

*Cast Down but Not Destroyed*, 30281253

*Indirect thought, third: Talking about Dixon’s murder wasn’t proper, not in front of his widow.” [...] ”Raising the pick over her head had taken all Corrie’s strength, but the wet, crushing noise when it entered Dixon’s head had made the effort worth it. (Dixon, Corrie, slaying)*

---

**Model:** llama3.1:70b

Input	Victim	Perpetrator	Mode of Demise	Summary Sentence
Generic Summary	<b>Dixon</b>	<b>Corrie</b>	<b>pickaxe</b>	The story unfolds as a tale of secrets, lies, and revenge, ultimately revealing that Corrie killed her husband with a pickaxe, seeking justice for the suffering he caused.
Specialised Summary	<b>Dixon</b>	<b>Corrie</b>	<b>murder</b>	It is implied that Corrie herself was the one who killed Dixon with a pickaxe, as she recalls the effort and noise of the act.
Story	<b>Dixon Worth</b>	<b>Corrie Worth</b>	<b>beating</b>	N/A

**Table 12:** Example of manual evaluation handling alignment for Mode of Demise.

## F Further examples of errors

### F.1 Successful extraction from summaries examples

We report in Table 13 examples of cases in which the summarisation mitigates the difficulty of extracting death events conveyed in direct speech, making them more explicit.

In Table 15, we present three further examples of death events extracted across the three types of input tested in this study. In the first example, the death event is presented by the narrator. Extraction is successful in both summary-based settings but fails when the input is the full story. We hypothesise that the figurative and implicit nature of the death event expression in the full story likely misled the model. Conversely, the summaries render the death event more explicitly, even though the mode of demise—men being killed through petrification—is fantastical and concerns a minor character (generic “men”). This explicitness in the summaries likely facilitates correct extraction. The second example illustrates a death event presented as an indirect thought. Specifically, it is a prophecy by Cassandra, forecasting her own demise at the hands of Clytemnestra. This presentation mode introduces additional complexity for extraction. Consequently, extraction fails when the input consists of either the generic summary or the full story. However, in the specialised summary, the death event is articulated more explicitly, enabling successful extraction during the KE step of our method. The third example involves Hector’s death at the hands of Achilles, presented in indirect speech. As seen in Section 4, death events conveyed through indirect speech are generally more likely to be extracted successfully. Consistent with this observation, the extraction process succeeds across all input types for this example. However, when the full story is used as input, a false positive occurs in the mode of demise.

### F.2 Failed KE from direct speech additional examples

As discussed in Section 4, death events presented in direct speech are difficult for the models to extract. We report in Table 14 an additional example of a death event occurring in a direct speech that cannot be extracted from the story and the specialised summary, but correctly extracted from the generic summary.

---

*I have died everyday waiting for you, 38886291*

Direct speech, first: “Look at how he will be remembered now. Killing Hector, killing Troilus.”.  
(Hector/Troilus, Achilles, unspecified)

---

**Model:** llama3.1:70b

Input	Victim	Perpetrator	Mode of Demise	Summary Sentence
Generic Summary	-	-	-	-
Specialised Summary	<b>Hector/Troilus</b>	<b>Achilles</b>	-	Achilles’ lover, Patroclus, confronts Achilles’ mother, Thetis, about how she has ruined him and how he will be remembered for his cruel killings in war, specifically the murders of Hector and Troilus.
Story	-	-	-	N/A

**Model:** deepseek-r1:70b

Generic Summary	-	-	-	-
Specialised Summary	<b>Hector/Troilus</b>	<b>Achilles</b>	-	Achilles is portrayed as a complex figure, remembered both for his heroic deeds in battle and his intimate, human moments with Patroclus. His killings of Hector and Troilus are highlighted as acts of cruelty driven by grief, leaving a legacy tied to war and bloodshed.
Story	-	-	-	N/A

**Table 13:** For each example, we include the *story title*, ID, and the quote from the story reporting the death event. Each quote is introduced with the corresponding presentation and narrator POV and followed, in round brackets, by the gold-standard victim, perpetrator, mode of demise. Correct predicted elements are shown in **bold**, with incorrect predictions highlighted in red.

---

*Medusa, 1087311*

**Narrator, Third person:** Medusa had killed hundreds of men with a single glance; this newcomer was nothing but another nameless body to be added to the list of deceased. No one who met her gaze ever breathed again. (men, Medusa, petrification)

---

**Model:** llama3.1:70b

Input	Victim	Perpetrator	Mode of Demise	Summary Sentence
Generic Summary	<b>men</b>	<b>Medusa</b>	<b>turning to stone</b>	Medusa, cursed by Poseidon for her beauty, has been exiled to an island where she can turn men to stone with a single gaze, and has amassed hundreds of statues over time.
Specialised Summary	<b>men</b>	<b>Medusa</b>	<b>turned to stone</b>	She has been living on an island, turning hundreds of men who dared to approach her into statues.
Story	-	-	-	N/A

*Cassandra on the Beach, 47136421*

**Direct thought, first person:** She will then turn to me, bloodlust in her eyes, swinging the same sword towards me. I will sit there calmly as she kills me, because I knew this was coming, and I cannot wish for death to come soon enough. (Cassandra, Agamemnon's wife, slaying)

Input	Victim	Perpetrator	Mode of Demise	Summary Sentence
Generic Summary	-	-	-	-
Specialised Summary	<b>Cassandra</b>	<b>Clytemnestra</b>	<b>killed with sword</b>	Cassandra herself will be given to Agamemnon, who will rape her, and she will bear him twins before he is murdered by his wife Clytemnestra in revenge for the murder of her daughter Iphigenia; Clytemnestra will then kill Cassandra with the same sword.
Story	-	-	-	N/A

*Requiem, 17464*

**Narrator, second person:** You watch when Achilles strikes your brother down, watch your brother's glorious form fall under the wrath of a goddess's son, and you can almost hear your brother telling you again that he hates you, that you are the reason he won't live to see the sun set this day. (Hector, Achilles, slaying)

Input	Victim	Perpetrator	Mode of Demise	Summary Sentence
Generic Summary	<b>Hector</b>	<b>Achilles</b>	-	The story then cuts back to the present, where Paris puts on his armor, preparing for battle after Hector's death at the hands of Achilles.
Specialised Summary	<b>Hector</b>	<b>Achilles</b>	-	He reflects on his unrequited love for his brother Hector, who is now dead, murdered by Achilles as revenge for killing Patroclus.
Story	<b>Hector</b>	<b>Achilles</b>	<b>struck down</b>	N/A

**Table 14:** Examples of death events expressed in direct speech whose extraction is missed from the story and the specialised summary, but correct from the generic summary.

*Aulis Through Her Eyes*, 56575177

Direct speech, third: "...Because he plans to kill your daughter and sacrifice her to Artemis." [...] "How can you not? This is your daughter and now you are about to kill your own daughter with the knife in your hands! What kind of man would have a heart like this?" [...] "Mother, I cannot bear to leave you! I cannot bear to leave this world and enter the dark realm of Hades.". (*Iphigenia, Agamemnon, sacrifice*)

**Model:** 11ama3.1:70b

Input	Victim	Perpetrator	Mode of Demise	Summary Sentence
Generic Summary	<b>Iphigenia</b>	<b>Agamemnon</b>	<b>sacrifice</b>	Klytemnestra, queen of Mycenae, devises a plan for revenge against her husband King Agamemnon, who had sacrificed their daughter Iphigenia before leaving for Troy.
Specialised Summary	-	-	-	-
Story	-	-	-	-

**Table 15:** Examples of death events extracted from the different inputs tested in this study. For each example, we include the *story title*, ID, and the quote from the story reporting the death event. Each quote is introduced with the corresponding presentation and narrator POV and followed, in round brackets, by the gold-standard victim, perpetrator, mode of demise. Correct predicted elements are shown in **bold**, with incorrect predictions highlighted in red.

*In The Minotaur's Maze*, 50126794

*You had no idea how Asterion could have survived all this time. He had been killed! But apparently, he hadn't gotten the memo.*

**Model:** 11ama3.1:70b

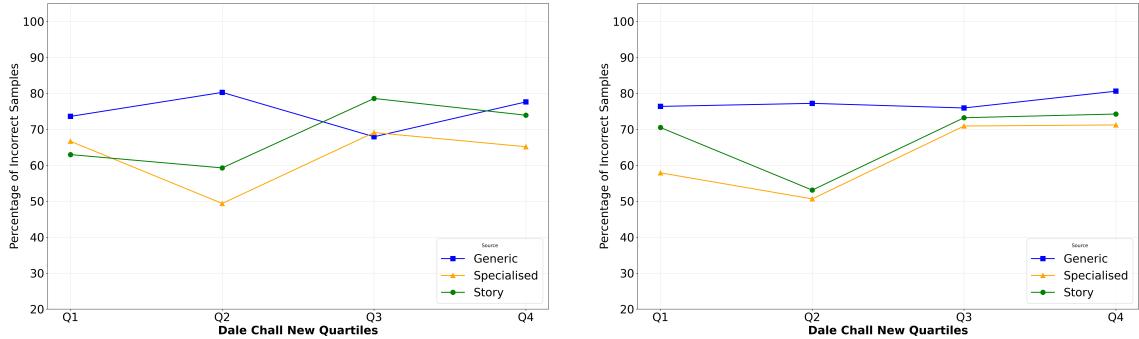
Input	Victim	Perpetrator	Mode of Demise	Summary Sentence
Generic Summary	-	-	-	N/A
Specialised Summary	<b>Asterion</b>	-	-	In this dark fantasy short story, an explorer seeking ancient relics discovers the entrance to the fabled Minotaur labyrinth and navigates its treacherous paths. However, they soon find themselves face-to-face with Asterion, the supposedly slain Minotaur, who has somehow survived for thousands of years.
Story	<b>Asterion the Minotaur</b>	-	-	N/A

**Table 16:** Examples of FP death event extracted from the specialised summaries and the story. For each example, the predicted elements of the triple (victim, perpetrator, and mode of demise) are reported, with incorrect predictions highlighted in red.

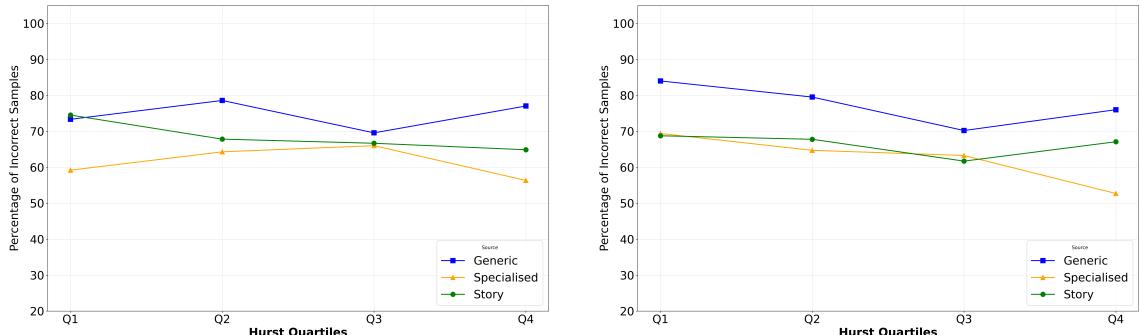
### F.3 False positives examples

As discussed in Section 4, the major problems LLMs face when performing KE from fiction are Recall issues and missed death events. Nevertheless, false positive extractions still occur, particularly when dealing with characters’ thoughts and dreams. Table 16 presents an example of a false positive extraction of a death event: the protagonist speculates about Asterion’s death, but Asterion has not died and is still alive. However, the model incorrectly extracts Asterion’s death as an actual event, leading to a false positive.

## G KE performances trend and Textual Features and Reader Reception metrics

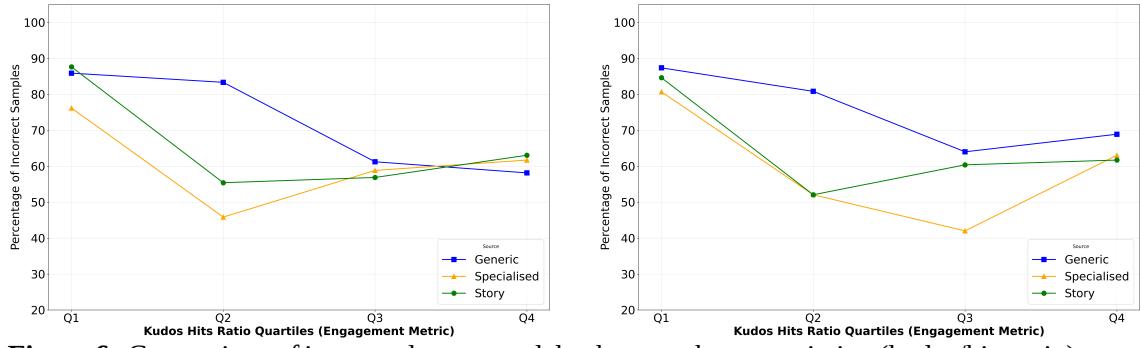


**Figure 4:** Comparison of incorrectly extracted death events by Dale-Chall readability quartiles. Left: 11lama3.1:70b. Right: deepseek-r1:70b. Q1-Q4 Dale-Chall ranges: [5.910, 6.249], [6.266, 6.637], [6.647, 6.979], [6.993, 8.886]. Each quartile contains 14-15 works. Each plot includes one line for each input type (generic, specialised, story).



**Figure 5:** Comparison of incorrectly extracted death events by Hurst exponent quartiles. Left: 11lama3.1:70b. Right: deepseek-r1:70b. Q1-Q4 Hurst ranges: [0.280, 0.510], [0.520, 0.570], [0.580, 0.640], [0.670, 0.910]. Each quartile contains between 12 and 17 works. Each plot includes one line for each input type (generic, specialised, story).

We conducted quartile analysis and Spearman  $\rho$  computation for textual features and reader reception metrics reported in Table 7 to investigate potential correlations with extraction errors. The analysis examined Dale-Chall readability (Figure 4), Hurst exponent (Figure 5), and appreciation metrics (Figure 6) across both models and all input types. Spearman  $\rho$  computation revealed no significant correlations for readability and Hurst exponent metrics. However, quartile analysis showed a trend of decreasing incorrect extractions as story appreciation increases, suggesting that stories with higher reader engagement may be easier for LLMs to process for death event extraction. This indicates that while text features do not substantially influence extraction accuracy, reader reception metrics may provide some predictive value for model performance.



**Figure 6:** Comparison of incorrectly extracted death events by appreciation (kudos/hits ratio) quartiles. Left: llama3.1:70b. Right: deepseek-r1:70b. Q1-Q4 appreciation ranges: [0.006, 0.029], [0.031, 0.048], [0.049, 0.071], [0.075, 0.446]. Each quartile contains 14-15 works. Each plot includes one line for each input type (generic, specialised, story).