CHATTER: A Character Attribution Dataset for Narrative Understanding

Sabyasachee Baruah and Shrikanth Narayanan
Signal Analysis & Interpretation Laboratory
University of Southern California
sbaruah@usc.edu shri@usc.edu

Abstract

Computational narrative understanding studies the identification, description, and interaction of the elements of a narrative: characters, attributes, events, and relations. Narrative research has given considerable attention to defining and classifying character types. However, these character-type taxonomies do not generalize well because they are small, too simple, or specific to a domain. We require robust and reliable benchmarks to test whether narrative models truly understand the nuances of the character's development in the story. Our work addresses this by curating the CHATTER dataset that labels whether a character portrays some attribute for 88124 character-attribute pairs, encompassing 2998 characters, 12967 attributes and 660 movies. We validate a subset of CHATTER, called CHATTEREVAL, using human annotations to serve as a benchmark to evaluate the character attribution task in movie scripts. CHATTEREVAL also assesses narrative understanding and the long-context modeling capacity of language models.

1 Introduction

Narrativity occurs when characters interact with each other, triggering events that are temporally, spatially, and causally connected. This sequence of events forms the story. Piper et al. (2021) provided a symbolic definition of narrativity in which they asserted that narrativity occurs when the narrator $\mathcal{A}$ tells the perceiver $\mathcal{B}$ that some agent $\mathcal{C}$ performed the action $\mathcal{D}$ on another agent $\mathcal{E}$ at place $\mathcal{F}$ and time $\mathcal{G}$ for some reason $\mathcal{H}$. Baruah and Narayanan (2024) used this definition to identify four main elements of any narrative: characters, attributes, events, and relations. Labatut and Bost (2019) explored different types of character interactions and emphasized the central role characters play in narratives. Characters drive the plot forward through their actions, develop attributes, arouse tension and emotion in the story by creating conflicts or bonds with other characters, and embody tropes and stereotypes to relate to the audience. The vitality of characters in narratives makes character understanding an essential task in narrative research.

Dataset	Domain	Type	Size
Finlayson (2015)	Folklore	Archetype	282
Skowron et al. (2016)	Movies	Role	2010
Brahman et al. (2021)	Literature	Description	9499
Sang et al. (2022)	Movies	Personality	28653
Yu et al. (2023)	Literature	Traits	52002
CHATTER	Movies	Tropes	88124
CHATTEREVAL	Movies	Tropes	1061

Table 1: Comparison with other attribution datasets in terms of the attribute type and size. Size denotes the number of character-attribute pairs. CHATTER is the largest character attribution dataset collected so far.

Narratologists have explored various approaches to operationalize the character-understanding task. For example, Inoue et al. (2022) presented character understanding as a suite of document-level tasks that included gender and role identification of the character, *cloze* tasks, quote attribution, and question answering. Li et al. (2023) adopted coreference resolution, character linking, and speaker guessing tasks, and Azab et al. (2019) used character relationships and relatedness to evaluate character representations. We organized the character-understanding tasks into the following categories. **1) Identification** tasks find the unique set of characters and their mentions. It includes entity recognition, entity linking, and coreference resolution. The **2) Quotation** task maps utterances to characters. **3) Attribution** tasks, such as personality classification, persona modeling, and description generation, describe the character. The **4) *Cloze*** task asks the model to fill in the correct character name given an anonymized character description, story summary, or story excerpt. **5) Relation** tasks, such as relation classification, draw similarities between characters.

Among these tasks, character attribution is the

Proceedings of the The 7th Workshop on Narrative Understanding, pages 52–63
May 4, 2025 ©2025 Association for Computational Linguistics

most challenging because there exists a multitude of ways to qualify a character, such as personality (Sang et al., 2022), adjectives (Yu et al., 2023), persona (Bamman et al., 2013, 2014), archetypes (Finlayson, 2015), roles (Skowron et al., 2016), and descriptions (Brahman et al., 2021). Each of these approaches has drawbacks. For example, personality scales such as Big5 and MBTI cannot capture all the variation in characterization and can be difficult to interpret (Zillig et al., 2002). Adjective descriptors are too general and apply only to a limited context in the story. Persona roles are not explicitly defined. Archetypes (Propp, 1968; Jung, 2014) are abstract concepts specific to the domain of interest. Character descriptions are detailed, freeform, and scalable to any number of characters. However, we cannot factor them into simpler components or use them to compare between different characters.

We require a robust attribution taxonomy that can scale across different narratives, characters, and domains, is well-defined and discrete for effective character comparison, and necessitates document-level understanding to model them accurately. We developed the CHATTER dataset to fulfill this need. The CHATTER dataset uses tropes from the TVTropes website as the attribute type to describe characters. TVTropes editors and moderators comprehensively define every trope with illustrative examples from multiple media sources. Tropes cover a wider range of character descriptions than personality types and archetypes, and require longer context-understanding, compared to traits and adjectives, to accurately ascertain if a character portrays some trope. Unlike character descriptions, we can efficiently compare characters and their experiences using these well-defined tropes (Wang et al., 2021).

The CHATTER dataset contains labels indicating whether the character portrayed the trope in the movies they appeared in. It contains 88124 character-trope pairs. We drew our characters primarily from Hollywood movies across various genres. It also provides the full-length screenplays of the movies, averaging about 25K words per screenplay. We define the character attribution task as a **binary classification task** where, given a character-trope pair and the screenplays of the movies where the character appears, the model should predict whether the character portrayed the trope. We validate a subset of the CHATTER data, referred to as CHATTEREVAL, using human annotations to establish an evaluation benchmark for the character attribution task. We compared the zero-shot and few-shot performance of LLMs and CHATTER's labels to assess the suitability of using the CHATTER dataset as a training set for the attribution task. The dataset is available at https://drive.google.com/drive/folders/11egMhs-zkWSASe7zJENwHg17-6VOeXDU?usp=sharing.

2 Data

2.1 Tropes

Tropes are storytelling devices or conventions used by the writer to easily convey some story notion to the reader. They act like narrative motifs that readers can easily recognize, saving time and effort for the writer as they can omit details the readers can infer from the trope. For example, the *AntiHero* is a very popular trope that describes a protagonist who lacks traditional heroic qualities, often cynical and flawed, yet ultimately performs heroic actions. The character Severus Snape in the Harry Potter stories portrays the *AntiHero* trope as he gives Harry a hard time but stays loyal to Dumbledore (mentor) and works secretly to defeat Voldemort (antagonist).

Tropes can also relate to inanimate objects, events, locations or the environment. We focus only on character tropes in our work. The scope of a trope can vary greatly. Some tropes like *PetTheDog* (villain performing an act of kindness) are portrayed over short contexts, typically a single scene, whereas others like *HiddenDepths* (revealing unexpected talents as the story progresses) are portrayed over a longer context. The attribution model should be able to extract information from different points in the narrative and reason over it to identify the portrayed tropes.

We use the character trope labels of TVTropes[1]. TVTropes is a community-driven website, similar to Wikipedia, that catalogs tropes with definitions and examples. Fans of a creative work discuss and post tropes they identify in the narrative. Each trope has a dedicated page which contains its definition, illustrations, and portrayal examples from TV shows, movies, literature, animation, video games, and print media. TVTropes moderators ensure that the fan-edited content is correct. We collect the trope annotations from TVTropes to build the CHATTER dataset.

It is important to note that the character trope annotations we collect from TVTropes are those

[1] https://tvtropes.org

perceived by the reader. These might not align with the *actual* tropes intended by the creator. Since there is no quantifiable agreement on the published content, we treat the TVTropes data as a noisy source of character attribution.

2.2 Screenplays

We used movies as the source of our narratives. We chose the cinematic domain over the literary domain because it had more TVTropes labels, and we supposed it would be easier to find raters more knowledgeable about movies than books. Additionally, movies allow us to extend the attribution task to the multimodal domain, offering more opportunities for future research directions.

We used publicly available movie screenplays from the ScriptsonScreen[2] website. Each script is mapped to an IMDB[3] identifier so we can uniquely identify the movie. Most movies in our dataset are produced in the US or the UK after 1980. The average script size is about 25K words. We apply a named entity classifier and name alias generator to map the character names in the script to a unique character in the IMDB cast list. We preprocess the screenplays to find scenes, dialogues and descriptions using Baruah and Narayanan's (2023) screenplay parser. In total, our dataset contains screenplays of 660 movies.

2.3 CHATTER

We build the CHATTER dataset of character-trope pairs using the tropes of TVTropes and the screenplays of ScriptsonScreen. First, we download the movie screenplays from ScriptsonScreen, parse and map them to an IMDB page, and match the characters occurring in the document to a character in the IMDB cast list. Second, we search for the character in TVTropes and retrieve their character page. Third, we collect the tropes portrayed by the character from the character page, as labeled by the TVTropes community. Fourth, we search for the trope page and fetch its definition. We also summarize the definition using GPT-4 for the annotation task. The average size of the trope definition and its summary is 344 and 44 words, respectively.

We need good negative samples to evaluate the specificity of the attribution model. However, TVTropes does not provide this information because it does not label tropes *not* portrayed by the character. Instead, we analyze the trope definition to find antonym tropes and create the negative character-trope pairs. For example, the definition of the *AntiHero* trope contains the sentence –

```
...Compare and contrast this trope with
its antithesis, the AntiVillain...
```

– which indicates that the *AntiVillain* trope is contrary to *AntiHero*. We search the trope definitions to retrieve opposing tropes by checking if negation words such as "contrast", "opposite" and "counterpart" exist within a five-word context window of the antonym trope mention[4]. For each positive character-trope pair, we create a negative pair by either – 1) selecting an antonym trope from the trope definition or 2) choosing a trope that is absent in any of the positive character-trope pairs for that character. It should be harder for the attribution model to distinguish antonym tropes from tropes portrayed by the character because they are much more closely related to the portrayed tropes than a trope randomly sampled with the second method. Therefore, the former method creates hard negatives and the latter method creates soft negatives. We randomly choose between the two methods with 50% probability to get a good mix of hard and soft negatives. We add the negative samples to complete the construction of the CHATTER dataset.

3 Evaluation Data

We can use CHATTER to train attribution models, but we require a more reliable dataset for evaluation. We annotate a subset of CHATTER using human raters and create the CHATTEREVAL dataset.

3.1 Annotation

We sampled movies from our dataset released after 2010 and collected the corresponding character-trope pairs for annotation. We employed workers on the Amazon MTurk[5] crowdsource annotation platform. We selected workers with high reputation and experience[6] and tested them on two separate qualification tasks, each containing five questions that asked them to decide whether the character portrayed the given trope. The task showed them the picture of the character, the movies that starred the character with Wikipedia links, a link to the fandom page[7] where the worker can read more about the

[2] https://scripts-onscreen.com

[3] https://imdb.com

[4] The full list of negation words is included in the dataset

[5] https://www.mturk.com

[6] >98% approval rate and worked on >5000 HITs

[7] https://www.fandom.com

character, and the summarized trope definition with a link to the trope's TVTropes page. Appendix A shows the annotation interface. We also surveyed the workers about the movies they had watched. We qualified 69 workers who correctly answered at least 80% of the questions and had watched at least ten movies out of the ones sampled for evaluation.

The qualified workers labeled whether the character portrayed the trope on a Likert scale: *no*, *maybe no*, *not sure*, *maybe yes* and *yes*. To reduce the cognitive load on the workers, they annotated the character-trope pairs in batches, each batch containing characters from at most ten movies. Before every batch, we inform the workers about the movies they will encounter to prepare them for the annotation task. After each batch, we estimate the worker's performance by calculating the accuracy of their ratings against the CHATTER labels and record how much time they spent on the task. We dropped workers whose accuracy dropped below 50% or those speeding through the samples, disqualifying them from working on future batches. Of the 69 workers, we disqualified 10 raters and were left with 59 reliable workers.

The annotation task ran for one month between August 11 to September 19 2024. Throughout the task, we maintained a communication channel for the workers in the TurkerNation Slack Workspace, where we responded to any questions the workers had about the task. We pay $1.5 for each question which turns out to be $18/hour because the workers spent an average of 5 minutes per question. The entire task cost us about $10K. We collected three annotations per character-trope pair, totaling 5683 annotations. The Krippendorff inter-rater reliability score is 0.448, indicating moderate agreement.

3.2 CHATTEREVAL

We aggregate the annotations to build the CHATTEREVAL dataset. The character-trope pairs do not all show clear agreement. Character attribution, like sentiment analysis, is a subjective task, and whether a character portrays some trope is perceived differently by people. We need to assign a definite binary label to each annotated sample for the character attribution task. We also need to drop the very ambiguous samples and those with insufficient reliable ratings to ensure high quality.

The MTurk workers answer *yes*, *maybe yes*, *not sure*, *maybe no* or *no* on each question. We map this ordinal scale to a numeric range by mapping the labels to 2, 1, 0, -1 and -2, respectively. Each character-trope pair is annotated by at most three reliable workers. We sum the label values for each sample to get an integer score s between -6 and 6. The higher the absolute score, the greater is the agreement among the raters. We drop samples whose absolute integer score falls below 3. For the remaining samples, we obtain the annotation confidence w by normalizing the integer score s between 0 and 1: $w = (|s| - 2)/4$. The confidence score takes values 0.25, 0.5, 0.75 and 1. Attribution models can use these scores to weigh their evaluation metrics. The sample gets the label 1 (character portrays the trope) if $s > 0$, else 0 (character does not portray the trope). We include the annotations of the individual raters in CHATTEREVAL to encourage multiannotator modeling.

Dataset	Characters	Tropes	Movies	Samples
CHATTER	2998	12967	660	88124
CHATTEREVAL	271	896	78	1061

Table 2: CHATTER and CHATTEREVAL's sizes

	Min	Max	Avg	95%tile
Movies/Character	1	5	1.04	1.0
Tropes/Character	1	1107	29.40	91.0
Words/Script	3051	87738	24614.98	34011.35
Tokens/Script	5149	158038	41952.05	58933.09
Words/Segment	3	28423	2981.28	9983.8
Tokens/Segment	4	43742	4350.89	14317.15

Table 3: CHATTER and CHATTEREVAL's statistics

4 Data Statistics

CHATTER contains 88124 character-trope pairs with an almost 50-50 split between positive and negative pairs. It covers 2998 unique characters from 660 movies and 12967 tropes. CHATTEREVAL contains 1061 human-annotated character-trope pairs, covering 271 characters, 896 tropes, and 78 movies. It contains 555 positive and 506 negative pairs. Tables 2 and 3 show some data statistics of CHATTER and CHATTEREVAL.

Most characters appear in a single movie script in our dataset. CHATTER contains attribution labels for about 30 tropes for each character. Segments indicate the portions of a movie script where a character speaks or is mentioned. As shown in Table 3, the maximum size of a movie script or a character segment can exceed 87K and 28K words, respectively. This corresponds to 158K and 43K tokens, respectively (We used the Llama3 (Dubey et al., 2024) tiktoken-based BPE tokenizer). There-

Character	Movies	Trope	Label
Benoit Blanc	*Knives Out (2019)*, *Glass Onion (2022)*	The **NiceGuy** trope describes a character who is kind, friendly, morally good, and socially pleasant.	1
Mark Watney	*The Martian (2015)*	The **EarnYourHappyEnding** trope involves characters enduring significant hardship, anguish, and grief, but ultimately achieving a happy ending through hard work or love	1
Arthur Fleck	*Joker (2019)*	The **TheDogBitesBack** trope occurs when a villain is attacked or betrayed by an abused subordinate or victim who seizes the chance for revenge	0
Dr. King Schultz	*Django Unchained (2012)*	The **SoreLoser** trope describes a character who reacts to defeat with anger, accusations, and bad behavior	0

Table 4: Character-Trope examples from the CHATTEREVAL dataset

fore, we must use a model capable of handling long contexts for the attribution task.

Most movies in our dataset have been produced in the UK or the US after 1980. They cover a wide range of genres. The top five genres – *Action*, *Drama*, *Thriller*, *Adventure*, and *Comedy* – cover more than 50% of the movies. Table 4 shows qualitative examples from the CHATTEREVAL dataset. The tropes can relate to the character's attitude and personality (*NiceGuy*), some experience or incident (*SoreLoser*), or their overall story (*EarnYourHappyEnding*). The character attribution task entails that the model understands the trope definition, reads the scripts of the movies where the character appears, and, based on that, decides whether the character portrays the trope. Appendix D lists the most commonly occurring tropes and their definitions to visualize the trope space.

Our dataset does not contain scripts for all the movies where the character appears. For example, the character Arthur Fleck "Joker" has appeared in multiple movies, comics, and TV shows, but CHATTER only contains the movie script of the *Joker (2019)* movie. The TVTropes contributors and our raters draw their knowledge of the character from all sources. However, the attribution model predicts the attribute labels based only on the screenplay and its pretraining data. Therefore, intractable character-trope pairs could exist for which the model has insufficient information. In future work, we will investigate ways to find such intractable samples in our dataset.

5 Experiments

5.1 Models

We establish baselines on CHATTEREVAL using zero-shot and few-shot prompting. We used two closed-source models, Gemini-1.5-Flash (Reid et al., 2024) and GPT-4o-mini (Hurst et al., 2024), and three open-source models, Phi-3-small-7B-128k-Instruct (Abdin et al., 2024), Llama-3.1-8B-Instruct (Dubey et al., 2024) and Mistral-Nemo-Instruct-12B-2407 (Jiang et al., 2023), in our experiments. We selected these models because they can handle long contexts (128K tokens).

We experiment with four prompting strategies. **1) Priors** - We prompt the model with the character name, the list of movies where the character has appeared, and the trope definition. We do not include any screenplay content and ask the model to find the attribution label based solely on its prior knowledge about the character. **2) Script** - We include the full movie script in the prompt. **3) Segment (Zero-shot)** - We include the segments of the movie script where the character speaks or is mentioned. **4) Segment (Few-shot)** - We include the character segments and two examples from the CHATTEREVAL dataset. We select examples of the same trope. We also experimented with selecting random examples or examples of the same character, but they both performed worse than selecting same-trope examples. We do not apply few-shot prompting with movie scripts because the prompt size becomes too large. Appendix E describes the prompt template we used.

For the last three settings – **Script**, **Segment (Zero-shot)**, and **Segment (Few-shot)** – we replaced character names in the movie scripts or the character segments with random alphanumeric identifiers. We hypothesize that the LLM could be generating its response by recognizing the character name and using the knowledge it had accrued about the character from its pretraining datasets. The character-anonymization step possibly prevents the LLM from relying on its prior knowledge about the character and forces it to decide the attribution label based solely on the given movie script

Prompt	Model	Acc	P	R	F1
	Random	50.0	52.3	50.0	51.1
	CHATTER	*80.6*	*81.0*	*82.2*	*81.6*
Priors	Gemini-1.5	**81.9**	**91.4**	**72.2**	**80.7**
	GPT-4o	76.2	98.1	55.6	71.0
	Phi-7B	56.6	89.1	19.5	32.0
	Llama-8B	71.0	68.9	81.4	74.6
	Mistral-12B	73.1	83.0	61.3	70.5
Script	Gemini-1.5	72.4	83.1	59.4	69.3
	GPT-4o	**73.9**	**71.2**	**84.4**	**77.2**
	Phi-7B	65.5	77.4	48.1	59.3
	Llama-8B	61.4	61.4	70.7	65.7
	Mistral-12B	56.1	60.5	45.7	52.1
Segment (Zero-Shot)	Gemini-1.5	73.5	86.6	58.4	69.8
	GPT-4o	**78.5**	**77.4**	**83.3**	**80.3**
	Phi-7B	73.3	77.1	68.8	72.7
	Llama-8B	69.8	65.6	87.7	75.1
	Mistral-12B	69.3	81.4	53.2	64.3
Segment (Few-Shot)	Gemini-1.5	65.5	93.4	36.7	52.7
	GPT-4o	**74.5**	**79.5**	**69.2**	**74.0**
	Phi-7B	62.3	65.5	68.1	66.8
	Llama-8B	60.8	61.0	75.3	67.4
	Mistral-12B	62.0	62.0	73.6	67.3

Table 5: Prior, Zero-shot, and Few-shot performance on CHATTEREVAL for the character attribution binary classification task. The **CHATTER** row uses the CHATTER's labels as predictions. We bolden the model row with the best accuracy (or F1) for each prompting strategy.

or character segments. We observed that prompting with the non-anonymized documents produced better performance, validating our hypothesis. Future work should explore more effective prompting strategies to nullify the effect of the model's priors on the generated response.

We perform greedy decoding. We also evaluate CHATTER's labels against the annotations of CHATTEREVAL to assess its suitability as a training set for character attribution. We use permutation tests at $\alpha = 0.05$ to compare the performance of different prompting strategies and models. We apply Bonferroni correction to correct for multiple comparisons.

5.2 Results

Table 5 shows the performance of the different prompting strategies. The closed-sourced models were significantly more accurate than the open-sourced models for all strategies. This is expected because the closed-sourced models supposedly contain more parameters and have been trained on more extensive data. Gemini-1.5 has the strongest prior knowledge, followed by GPT-4o. They performed equally well when we prompted them on the full movie script. However, GPT-4o's zero-shot and few-shot accuracy on character segments was significantly better than Gemini's. There was no significant difference between the accuracy scores of Phi-7B, Llama-8B, and Mistral-12B for any of the prompting strategies, except for priors where Phi-7B performed worse than random.

Comparing the different prompting strategies, we observed that the zero-shot accuracy on character segments was significantly better than the few-shot accuracy across all the models. This discrepancy in performance is surprising because prompting models usually provide better results with exemplars in the prompt. A possible reason could be that the two examples in the prompt are insufficient to represent the task adequately. Increasing the number of prompts is not viable because of the context limit. A possible solution could be summarizing the character segments to fit more examples in the prompt.

Prompting the character segments usually performed as well or better than prompting the full movie scripts. However, in most cases, the model's prior performance already showed strong results. This confirms that these models were probably pretrained on the tropes data from TVTropes, making it harder to evaluate the true character attribution capability of the model. The huge gap in performance between the priors of the closed-source models and the zero-shot or few-shot prompting results of the open-sourced models shows that there is much scope for improvement for the open-sourced models. This is important because, given a movie script (or any other narrative document), we do not always want to prompt it using commercial APIs because they might contain private or unpublished information. Training local attribution models on the character-trope pairs of CHATTER should narrow this performance gap.

There is no significant difference between the results of the strongest-performing model and using the labels of the CHATTER dataset as predictions. Therefore, CHATTER can serve as a good training source for the character attribution task. The strong zero-shot performance of the closed-sourced models suggests that we can use them to create good-quality synthetic training data.

We also observe that the recall scores of the models are usually lower than their precision. A possible reason could be the sizeable prompt size, which makes it difficult for the model to pinpoint the relevant sections from the script or the character segments. While training character attribution models, we should further preprocess the data for more effective learning.

6 Related Work

Several past studies have curated character attribution datasets for narrative understanding. Finlayson (2015) annotated seven character archetypes in Russian folktales to create the ProppLearner corpus. Skowron et al. (2016) labeled hero, antagonist, sidekick, spouse and supporting roles in action-genre movies. Brahman et al. (2021) collected character descriptions from online study guides such as LitChart and Sparknotes, and created the LiSCU dataset for the character identification and description generation task. Sang et al. (2022) curated MBTI personality types from the personality database for movie characters. Yu et al. (2023) annotated reader's notes for character traits in Chinese-translated Gutenberg books. Baruah and Narayanan (2024) annotated screenplay excerpts for character attributes and evaluated in-context and chain-of-thought learning methods on the attribution task. Table 1 compares these datasets against CHATTER and CHATTEREVAL.

7 Conclusion

We proposed the CHATTER and CHATTEREVAL datasets for the narrative character attribution task. We addressed the limitations of previous datasets by curating a resource that is scalable, generalizable, well-defined and discrete. Experiments showed that CHATTER can serve as a reliable training set and CHATTEREVAL can be used as the evaluation benchmark for character attribution modeling. Future work includes developing character attribution models on our datasets to aid creators and writers in analyzing their narratives.

8 Limitations

We define the character attribution task as a binary classification task, where, given the character-trope pair and the screenplays of the movies in which the character appeared, the model should predict whether or not the character portrayed the trope. This formulation has some limitations. First, the screenplay's narrative may not exactly resemble the story told by the movie because of tweaks made during the filming process. Publicly available screenplays are rarely the final script but drafts from earlier in the production stage. Second, movies could have visual cues like the nonverbal behavior of the character that are missed by the screenplay text. Lastly, a character can appear in multiple movies, and our dataset does not contain scripts for all of them. These limitations have important implications because the TVTropes contributors and our raters draw their knowledge of the character from all sources. In contrast, the attribution model predicts the attribute labels based only on the screenplay. There could be character-attribute pairs for which the model has insufficient information and could lead to poor recall, as shown by the performance of the models in Table 5.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Mahmoud Azab, Noriyuki Kojima, Jia Deng, and Rada Mihalcea. 2019. Representing movie characters in dialogues. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 99–109, Hong Kong, China. Association for Computational Linguistics.

David Bamman, Brendan O'Connor, and Noah A. Smith. 2013. Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361, Sofia, Bulgaria. Association for Computational Linguistics.

David Bamman, Ted Underwood, and Noah A. Smith. 2014. A Bayesian mixed effects model of literary character. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 370–379, Baltimore, Maryland. Association for Computational Linguistics.

Sabyasachee Baruah and Shrikanth Narayanan. 2023. Character coreference resolution in movie screenplays. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10300–10313, Toronto, Canada. Association for Computational Linguistics.

Sabyasachee Baruah and Shrikanth Narayanan. 2024. Character attribute extraction from movie scripts using llms. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8270–8275.

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.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Mark A. Finlayson. 2015. ProppLearner: Deeply annotating a corpus of Russian folktales to enable the machine learning of a Russian formalist theory. *Digital Scholarship in the Humanities*, 32(2):284–300.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Naoya Inoue, Charuta Pethe, Allen Kim, and Steven Skiena. 2022. Learning and evaluating character representations in novels. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1008–1019, Dublin, Ireland. Association for Computational Linguistics.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Carl Gustav Jung. 2014. *The archetypes and the collective unconscious*. Routledge.

Vincent Labatut and Xavier Bost. 2019. Extraction and analysis of fictional character networks: A survey. *ACM Computing Surveys (CSUR)*, 52(5):1–40.

Dawei Li, Hengyuan Zhang, Yanran Li, and Shiping Yang. 2023. Multi-level contrastive learning for script-based character understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5995–6013, Singapore. Association for Computational Linguistics.

Andrew Piper, Richard Jean So, and David Bamman. 2021. Narrative theory for computational narrative understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 298–311, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Vladimir Propp. 1968. Morphology of the folktale. *U of Texas P*.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Yisi Sang, Xiangyang Mou, Mo Yu, Dakuo Wang, Jing Li, and Jeffrey Stanton. 2022. MBTI personality prediction for fictional characters using movie scripts. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6715–6724, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Marcin Skowron, Martin Trapp, Sabine Payr, and Robert Trappl. 2016. Automatic identification of character types from film dialogs. *Applied Artificial Intelligence*, 30(10):942–973.

Zhilin Wang, Weizhe Lin, and Xiaodong Wu. 2021. Learning similarity between movie characters and its potential implications on understanding human experiences. In *Proceedings of the Third Workshop on Narrative Understanding*, pages 24–35, Virtual. Association for Computational Linguistics.

Mo Yu, Jiangnan Li, Shunyu Yao, Wenjie Pang, Xiaochen Zhou, Zhou Xiao, Fandong Meng, and Jie Zhou. 2023. Personality understanding of fictional characters during book reading. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14784–14802, Toronto, Canada. Association for Computational Linguistics.

Lisa M Pytlik Zillig, Scott H Hemenover, and Richard A Dienstbier. 2002. What do we assess when we assess a big 5 trait? a content analysis of the affective, behavioral, and cognitive processes represented in big 5 personality inventories. *Personality and Social Psychology Bulletin*, 28(6):847–858.

A Annotation

We give the following annotation instructions to the Amazon MTurk workers.

```
A trope is a storytelling device or
convention the storyteller uses to
describe situations the audience can
easily recognize, often used to
stereotype characters. Think of them
as character attributes.

Your task is as follows:

1. Identify the movie character from the
   pictures and the given movie(s). If
   you are having trouble recognizing
   the character, click on the Character
   Page link(s).

2. Read the definition of the character
   trope. You can open the link to know
   more.

3. Choose yes, maybe yes, not sure,
   maybe no, or no to answer whether the
  character portrays or is associated with
   the trope.

4. (Optional) Explain your choice in the
   provided text area or leave any other
   comment behind.
```

Figure 1 shows the annotation interface that the workers use to label character attribution.

B TVTropes

TVTropes is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License. This license allows for the copy and redistribution of the material in any medium or format for non-commercial purposes. CHATTER is published for academic research purposes and does not infringe the TVTropes license.

C ScriptsonScreen

ScriptsonScreen aggregates movie scripts from different websites such as IMSDB[8], Dailyscripts[9] and Script Slug[10]. All the movie scripts we use are licensed for fair use and available for public download.

D TROPES

There are 445 tropes in total that have at least one character portraying them in our dataset. We could not figure out a systematic way to cluster them because the tropes are very different from each other. Therefore, we simply list the most common tropes and their abbreviated definition. Tables 6 and 7 list the tropes that are portrayed by at least two characters in our dataset.

E Prompt

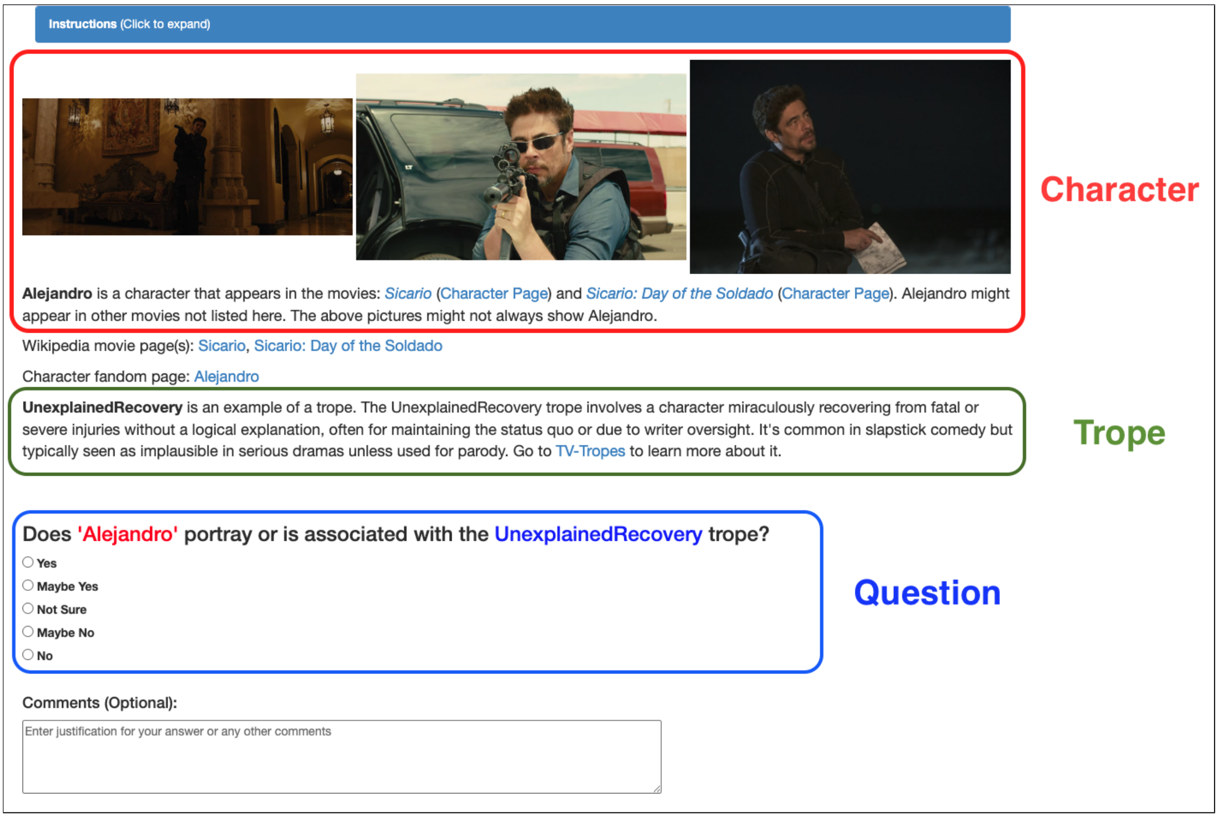
We used the following prompt template in our experiments.

```
A character trope is a story-telling
device used by the writer to describe
characters.

Given below is the definition of the
$TROPE$ trope enclosed between the
tags <TROPE> and </TROPE>.
Following that is a movie script
enclosed between the
tags <SCRIPT> and </SCRIPT>.
The character "$CHARACTER$" appears
in the movie script.

Read the movie script carefully
and based on that answer yes or no
if the character "$CHARACTER$"
portrays or is associated with the
$TROPE$ trope.
If yes, give a brief explanation.
Answer based only on the movie script.
Do not rely on your prior knowledge.

<TROPE>
$DEFINITION$
</TROPE>

<SCRIPT>
$SCRIPT$
</SCRIPT>

Does the character "$CHARACTER$"
portray or is associated with
the $TROPE$ trope in the above
movie script?
Answer yes or no.
```

[8] https:https://imsdb.com/

[9] https://www.dailyscript.com/

[10] https://www.scriptslug.com/



Instructions (Click to expand)

Alejandro is a character that appears in the movies: *Sicario* (Character Page) and *Sicario: Day of the Soldado* (Character Page). Alejandro might appear in other movies not listed here. The above pictures might not always show Alejandro.

Wikipedia movie page(s): Sicario, Sicario: Day of the Soldado

Character fandom page: Alejandro

UnexplainedRecovery is an example of a trope. The UnexplainedRecovery trope involves a character miraculously recovering from fatal or severe injuries without a logical explanation, often for maintaining the status quo or due to writer oversight. It's common in slapstick comedy but typically seen as implausible in serious dramas unless used for parody. Go to TV-Tropes to learn more about it.

Does 'Alejandro' portray or is associated with the UnexplainedRecovery trope?

○ Yes
○ Maybe Yes
○ Not Sure
○ Maybe No
○ No

Comments (Optional):

Enter justification for your answer or any other comments

Figure 1: Interface of the annotation task.

```
If yes, give a brief explanation.
Do not use MarkDown.
```

We replace $CHARACTER$, $TROPE$, $DEFINITION$ and $SCRIPT$ with the character name, trope name, definition of the trope and the movie script during prompting.

	Trope	Definition
1.	AdaptationalAttractiveness	Plain characters are portrayed as attractive in adaptations
2.	AdaptationPersonalityChange	Character personality changes during medium adaptations
3.	AffablyEvil	Charming villain with kindness despite malevolent intentions
4.	AntiVillain	Heroic goals achieved through questionable or evil means
5.	AudienceSurrogate	Character audience identifies with for sympathy and relatability
6.	BadLiar	Character fails to lie convincingly; creates transparent falsehoods
7.	BerserkButton	Character's minor trigger causes explosive anger reaction
8.	BitchInSheepsClothing	Deceptive character appears kind but is secretly villainous
9.	BlueIsHeroic	Blue signifies heroism through calm, disciplined characters
10.	CatchPhrase	Repetitive distinctive phrase by a character or category
11.	ChildrenAreInnocent	Children embody innocence and purity, contrasting adult corruption
12.	Cloudcuckoolander	Cheerfully eccentric character, detached from reality, unexpectedly wise
13.	ControlFreak	Obsessively enforces rules, hindering effectiveness and dissent
14.	CynicismCatalyst	Trauma shifts idealistic character to cynicism and moral ambiguity
15.	DarkAndTroubledPast	Character's tragic past shapes their personality and behavior
16.	DeadpanSnarker	Sarcastic character who critiques and deflates others' egos
17.	Determinator	Unyielding persistence towards goals, regardless of challenges
18.	DramaQueen	Excessively dramatic characters who overreact and seek attention
19.	EvenEvilHasStandards	Villain shows moral boundaries despite overall remorselessness
20.	EveryoneHasStandards	Characters uphold personal standards despite their own flaws
21.	EvilWearsBlack	Villains wear black, symbolizing darkness and aggression
22.	FauxAffablyEvil	Polite villains hiding true cruelty for manipulation and enjoyment
23.	Gaslighting	Manipulating perception to induce doubt and confusion
24.	GentleGiant	Big, kind character defying intimidating appearance; gentle and reliable
25.	GoingNative	Character embraces new culture, rejects original society
26.	GrinOfRage	Smiling while angry, used for intimidation or provocation
27.	GuileHero	Cunning hero uses wit and charm for noble goals
28.	HairTriggerTemper	Explosive anger at minor irritations; unpredictable and dangerous
29.	HeroAntagonist	Good antagonist opposing protagonist for noble reasons
30.	HiddenDepths	Characters unveil unexpected talents, deepening their complexity
31.	Hypocrite	Authority figures failing to uphold their own ideals
32.	ItsAllAboutMe	Self-centered character believes world revolves around them
33.	Jerkass	Self-centered character creating conflict for comedic/dramatic effect
34.	JerkassHasAPoint	Morally flawed character speaks uncomfortable but true points
35.	KarmaHoudiniWarranty	Villain faces late justice, satisfying audience's desire for retribution
36.	KickTheDog	Character's cruel act establishes evil, shifts audience sympathy
37.	LackOfEmpathy	Characters recognize emotions but lack emotional connection
38.	LargeHam	Flamboyant, over-the-top character adding drama and charisma
39.	LaserGuidedKarma	Immediate consequences for characters' actions reinforce moral lessons
40.	LivingMacGuffin	Person drives quests due to intrinsic value or attributes
41.	LoveAtFirstSight	Instant deep love between characters upon first meeting
42.	LoveInterest	Romantic character involved with another, often archetypal roles
43.	ManlyTears	Stoic male character cries from strong, dignified emotions
44.	MeaningfulName	Character names reflect traits or roles meaningfully
45.	MoralityPet	Villain's bond with innocent character prompts redemption
46.	MorphicResonance	Characters retain recognizable traits across different forms
47.	MyGodWhatHaveIDone	Character regrets harmful actions, prompting remorse and conflict
48.	Narcissist	Character obsessed with self-admiration and validation, often hostile
49.	NeverBareheaded	Character always wears headgear, never seen bare-headed
50.	NiceGirl	Kind, friendly character contrasting cynical figures; endearing presence

Table 6: Tropes and their definitions (first part)

	Trope	Definition
51.	NiceGuy	Kind, morally good character contrasting with cynicism
52.	NiceJobBreakingItHero	Hero's victory causes unintended negative consequences
53.	NoCelebritiesWereHarmed	Parody characters resembling real celebrities, often with altered names
54.	NoSocialSkills	Characters lacking social awareness, often blunt but intelligent
55.	OhCrap	Character realizes impending disaster, leading to panic or horror
56.	OnlyOneName	Characters known by only one name, no identifiers
57.	OnlySaneWoman	Rational character amidst chaotic, irrational peers; often frustrated
58.	PayEvilUntoEvil	Revenge-driven morality blurs hero-villain boundaries
59.	PosthumousCharacter	Dead character influences plot through memories or narratives
60.	ReasonableAuthorityFigure	Open-minded leader who evaluates heroes' claims rationally
61.	RebelliousSpirit	Character defies norms, follows personal rules, often anti-heroic
62.	ShadowArchetype	Character reflecting protagonist's denied flaws, causing conflict and growth
63.	SirSwearsALot	Character known for excessive swearing, revealing deeper traits
64.	StepfordSmiler	Cheerful facade hides inner turmoil and psychological issues
65.	TheChessmaster	Cunning strategist who manipulates events for personal gain
66.	TheDon	Ruthless crime patriarch with moral codes, protective yet shrewd
67.	TheHeart	Caretaker and moral compass of the team
68.	TheImmune	Character immune to disease, pivotal for finding cure
69.	TheJerkIndex	Characters exhibiting rudeness contrasting with polite behavior
70.	TheSociopath	Character lacking empathy, manipulative, and morally ambiguous
71.	TopHeavyGuy	Exaggerated large upper body, skinny legs
72.	TragicVillain	Sympathetic villain regrets their actions, seeks redemption
73.	UncertainDoom	Ambiguous fate of a character, survival uncertain
74.	UndignifiedDeath	Ridiculous, embarrassing deaths blending humor and tragedy
75.	UnwittingInstigatorOfDoom	Unintentional catalyst for disaster, unaware of their role

Table 7: Tropes and their definitions (second part)