Adapting Large Language Models for Movie Domain with Narrative Understanding Tasks

Siqi Shen
University of Michigan
shensq@umich.edu

Amanmeet Garg
Amazon Inc.
amanmega@amazon.com

Abstract

Large language models (LLMs) have been deployed in a wide spectrum of domains and applications due to their strong language understanding capabilities obtained through pretraining. However, their performance on specific domain is usually suboptimal due to limited exposure to domain-specific tasks. Adapting LLMs to the movie domain poses unique challenges due to complex narratives that cannot be fully captured through subtitles or scripts alone. In this paper, we decompose movie understanding capability into a suite of narrative understanding tasks based on narrative theory. We construct a dataset for these tasks based on resources in the movie domain, and use it to examine the effect of different domain adaptation strategies. Our experiment results show the effectiveness of our approach in improving the narrative understanding of LLMs and highlight the trade-offs between domain-specific and general instruction capabilities.

1 Introduction

Large language models have revolutionized natural language processing with their ability to understand and generate text across diverse domains (Radford and Narasimhan, 2018). However, these models often struggle with specialized tasks in domains that are underrepresented in their training data, such as cinematic content. While domain adaptation has shown success in fields such as medicine (Yang et al., 2023), finance(Wu et al., 2023), and law(Cui et al., 2024), adapting LLMs to understand movie narratives remains an underexplored challenge despite cinema's widespread cultural impact.

Adapting an LLM to the movie domain, on the other hand, remains underexplored despite movies and TV shows having such a large audience. An adapted LLM could potentially enable and benefit many movie-related tasks, be it a better summarization of movie content or more accurate content moderation.

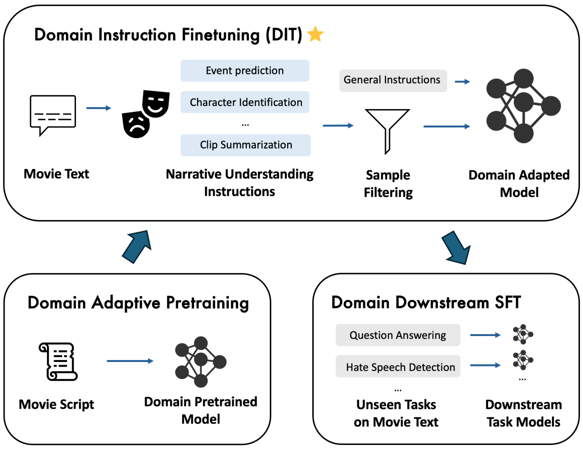

Figure 1: Different stages of adapting a generic LLM to the movie domain.

Understanding text in the movie domain, however, pose its unique set of challenges. Movie subtitles lack visual context and contain fragmented dialogues with interruptions, incomplete sentences, and repetitions. In addition, understanding a line sometimes requires knowledge of the story setups or the background of the characters. Existing work on domain adaptation of general-use LLMs generally falls into the following three categories. The first uses a domain-specific corpus $\mathcal{C}$ directly for pretraining without specifying domain tasks (Wu et al., 2023, 2024) referred to as Domain Adaptative Pretraining (DPT), which is only effective when substantial domain knowledge is unseen during the model's pretraining stage. The second category focuses on a single domain task t, for example, question answering, while generalization to unseen tasks $\mathcal{U}$ is not a main consideration(Wu et al., 2024; Singhal et al., 2025). The last category compiles a set of domain-related tasks $\mathcal{T}$ and uses all of them during training, aiming to broaden the coverage of domain use cases (Cui et al., 2024; Liu et al., 2023).

A narrative consists of two key components: the story elements (characters, events, and settings)

Proceedings of the 29th Conference on Computational Natural Language Learning, pages 187–200
July 31 - August 1, 2025 ©2025 Association for Computational Linguistics

and how these elements are presented (discourse). In movies, for example, a story includes both what happens (a detective solving a crime) and how it is told (revealing clues gradually to build suspense). This framework, established by Chatman (Chatman, 1980), provides a systematic way to analyze movie narratives by breaking them down into these fundamental components. Understanding these elements is crucial for teaching LLMs to comprehend movie content effectively.

We build the instruction dataset with movie text that focuses on narrative understanding capabilities. Our narrative understanding tasks $\mathcal{T}$ include predicting the main action or place of a movie clip, inferring the characters along with their interactions and relations, summarizing the subtitle, and segmenting the story according to plots. We control the quality of the dataset by sourcing from various datasets and databases in the domain, and curate instruction samples with both a suite of designed rules and LLM-based judges.

Using this dataset, we conduct domain adaptation with instruction finetuning on both narrative understanding and general instructions. We examine the effect of the training schema as well as data mixture, and illustrate that LLM can be effectively adapted to the movie domain with a trade-off between general instruction following. We also showcase the interplay of domain instruction finetuning with pretraining on movie text and in-domain downstream tasks.

Overall, our work offers insight on how to adapt an LLM to the movie domain and a better understanding of the effect of the adaptation procedure, with the following main contributions: (1) We propose a suite of tasks for adapting LLM to the movie domain based on narrative theory. (2) We collected an instruction dataset consisting of narrative understanding tasks with various quality control measures. (3) We conduct systematic experiments that analyze the effect of adapting LLMs on narrative understanding tasks, showing that adaptation also helps in the movie domain.

2 Related works

Instruction finetuning Instruction fine-tuning enables large language models (LLMs) to follow user instructions across various tasks by exposing them to diverse task instructions. Early works on instruction datasets, such as T0-SF (Sanh et al., 2021) and NaturalInstructions (Wang et al., 2022b) were mostly compiled by humans. It shows the potential of instruction finetuning in zero-shot and few-shot scenarios, particularly in generalization across unseen tasks (Wei et al., 2021; Chung et al., 2024). Methods like Self-Instruct (Wang et al., 2022a) that automatically generate new tasks have become popular alternatives to human annotation. It increases the diversity of tasks based on seed tasks in a bootstrapping way that drives the success of many open-source models (Taori et al., 2023; Xu et al., 2023; Peng et al., 2023).

Domain Adaptation Deploying an off-the-shelf LLM on domain tasks usually leads to suboptimal performances, thus there are attempts on adapting them to various domains. BloombergGPT (Wu et al., 2023) mixes the proprietorial finance text corpus with the general text corpus and is trained with the regular LLM pre-training task. Finetuning on domain-specific tasks is a more common and efficient solution for most application scenarios. FinGPT (Yang et al., 2023) adapts to the finance domain with a new task to predict the change in stock price, and sees improvement in in-domain tasks such as portfolio management. ChatLaw (Cui et al., 2024) construct a legal dataset comprising 10 major categories including case classification, statute prediction, and test its performance with Unified Qualification Exam for Legal Professionals. ChipNeMo (Liu et al., 2023) adopts LLM for chip design by training it in chip design documents and code. These existing works suggest that training general LLM on a selection set of domain-specific tasks can increase the model's general capability tasks in that domain.

3 Narrative Understandings in Movie Domain

3.1 Definition of Narrative

Narratives can be defined as stories in which a series of events or experiences unfold over time. Novel, fable, opera, and film are all good examples of narratives in different formats. Narratives are built upon different constituents, such as agent, scene, with the events organized in a particular order, and that structure is called the narrative structure. Despite the fact that narratology scholars have no consensus on what a narrative structure is, they offer different ways to comprehensively understand a narrative. In our work, we adopt the version of the narrative structures of Seymour Chatman (Chatman, 1980) as shown in Figure 5, with *Story* and

Task	Input	Output
subtitle_action	Look, see that? It's on. It worked. What do you mean it worked? Everything's off. Shutdown tripped the circuit breakers. Turn them back on, reboot a few systems	smoke, watch screen, press button
subtitle_place	The defendant, please.	court (inside)
synopsis_event	"Ron tries to sell medicine to the gay men from the support group he attended before. Nobody buys, and Ron leaves."	selling
synopsis_place	*same as above*	meeting room
subtitle_character	PersonA: Here is a support group that meets daily in Draddy Auditorium. I suggest you try it out and maybe go talk about your feelings, your concerns. PersonB: I'm dying. You telling me to go get a hug from a bunch of fag*ots?	PersonA: Eve; PersonB: Ron Woodroof
subtitle_interaction	*same as above*	suggests, rebukes
subtitle_relation	How you doing, Miss Clark? All right, Henry. Thank you. How you doing? If God is willing, Miss Clark. That's good.	acquaintance
short_subtitle_synopsis	*subtitle of a clip*	*synopsis of the clip*
long_subtitle_synopsis	*subtitle of a story part*	*synopsis of the story part*
synopsis_turning_point	(37) Later, Zira gets close to Taylor's cage and he grabs her note book and pencil.(38) Julius, (Buck Kartalian), the gorilla guard, enters the cage to beat up Taylor and retrieves the stolen items...	(41)

Table 1: Examples of narrative understanding tasks

Discourse as its two main constituents.

Story The *Story* of a narrative is the content occurring in the narrative, including events and existents. *Events* is also traditionally referred as plot, capturing key information about what is going on in a movie clip or a paragraph of text. That includes actions originating from characters such as "fell on to the ground" or things happening with other objects such as "It is raining here." *Existents*, on the other hand, instantiate events with concrete characters and settings. For example, the character who fell can be "Indiana Jones" and the setting can be "in a rainforest", and that immediately brings up the image of muddy ground and the thrill of treasure hunting to our mind.

Discourse Segments of content scattered around on their own do not give us anything interesting or thought-provoking, as they need to be delivered with some arrangements to make sense. That is done by the *Discourse* of a narrative, which is about the way to express the content. The discourse is in charge of both temporal and spatial arrangements. The temporal arrangement is more straightforward, for example, how does that events chronologically revolve and if there is non-linear storytelling etc. The spatial arrangement is more about the focus of spatial attention, maneuvering what the audience sees through the camera eye, and figuratively in verbal narratives.

3.2 Movie Narrative Understanding Tasks

Movie as a form of narrative consists of information in different modalities, including text, audio, and visual information. However, a lot of information can be inferred from the text alone. Taking a simple one-word line "Order!" One can guess that the setting of the story is probably in court or parliament. Also, if a host is announcing "Shari and Prakash are so happy that so many of you are here today joining in holy matrimony," then it is likely that Shari and Prakash are new spouses. It is expected that LLMs' capabilities on different narrative elements can help them adapt to the movie domain with better language understanding. We restrain our scope to the text modality, nonetheless, our general framework can be extended to a multimodal setting with Vision-Language models and visual narrative understanding tasks.

To better facilitate LLMs' narrative understanding on the movie domain, we propose a comprehensive array of tasks shown in Table 1, which cover each key element of a narrative at both the story level or discourse level.

For the *Story* element, we include several prediction tasks on the events including actions and happenings, as well as the existents of characters and settings. The tasks of event and place prediction from the synopsis take the summarized text description of the movie clip and answer the question based on that. It improves the model on extracting information about essential narrative components from the summarization. These prediction tasks are also conducted with subtitles as the input. Since subtitle is not as concise and well-formatted as the synopsis, this set of tasks is considered more difficult and often involves making inferences between the conversation lines. As character is another centerpiece making the content, we include a character disambiguation task that predicts the corresponding speaker given the subtitle and the story background. Story background is needed in this case to match the speaker to names, as there is usually no direct mention of it unless some speaker calls others' names. The character interaction prediction and character relation prediction go a step further and require the model to understand the dynamics between multiple characters.

At the *Discourse* level, we include a turning point prediction task as well as two summarization tasks. The turning points are crucial narrative moments that segment a movie into thematic story parts (Papalampidi et al., 2019). The story part is a larger unit than the scene, where a movie usually consists of several parts for setting up, complications of the plot, etc., and predicting the turning points requires an overall understanding of long and complex narratives. Summarization based on subtitles of a movie clip or a whole story part is also introduced, which requires correctly capturing the plot progression.

4 Movie Domain Adaptation

We discuss the sample collection procedures for each narrative understanding task in § 4.1 and the quality control measures in § 4.2. We discuss the method to perform domain adaptation in § 4.3.

4.1 Data Collection

We construct our samples around the subtitle and synopsis since they are the most available textual sources for movies. Among all our proposed narrative understanding tasks, most of them do not have a straightforward way to get labels directly from the subtitle only. Therefore, we look at existing human-annotated datasets on the movie domain for our need, as they may offer better fidelity than relying on synthesized data alone.

More specifically, we collect the place and action tags for movie scenes from MovieNet (Huang et al., 2020) for action prediction and place prediction based on subtitles. Each movie scene is further divided into movie shots in MovieNet, we aggregate the subtitles of movie shots that belong to the same scene, and match the place or action tag as its label. We collect the same information from (Vicol et al., 2018) with the difference that the event and place now match the synopsis of the clip instead of the subtitle, which is a short descriptive sentence in natural language. For tasks centered on characters, we collect characters occurring within a movie clip from MovieGraphs, and obtain their relations and interactions as well. We also keep a record of the corresponding subtitles and timestamps for all samples. The turning point prediction task is based on the segmentation of the story parts in MovieNet by tracking the sentence index where the synopsis turns into a new story part. The clip summarization and story summarization tasks are constructed by matching the subtitle with its synopsis at a clip and story part level correspondingly.

Instruct prompt construction We manually construct a prompt template for each narrative understanding task and use the collected labels to instantiate training samples. We specify the requirement for each individual task in the user prompt and use a format aligned with the Alpaca (Taori et al., 2023). The specific prompts that we used can be found in Table 8.

4.2 Sample Refinement

The multimodal nature of the MovieNet and MovieGraphs datasets presents challenges for text-only analysis. For instance, when aggregating subtitles from consecutive shots, redundant dialogues may appear, particularly in scenes with minimal conversation. For example, a subtitle can sometimes occur in multiple consecutive movie shots in MovieNet, which can introduce repeated utterances when aggregated into a movie clip. That usually happens when the main characters are not having a lot of conversation and the subtitle lingers for more than intended.

There are also many noises in the annotated samples. For example, a lot of subtitles contain very few verbal exchanges in MovieGraphs, since the annotation is at the clip level, which is usually less than one minute in length. Also, the subtitles are split by timestamps with no information of speakers available, which makes it more difficult to make sense of whether it is the same speaker talking.

Besides, a common issue with most tasks is that textual data alone does not contain sufficient information to infer the desired answer, especially for clips with multiple labels. For example, inferring multiple locations like *[desert, doorway, living room, yard (outside)]* for a single scene would be very hard, and it is a problem rooted in scene parsing of the datasets we use. It is the same case for character labels, there are labels such as *bridesmaid #4*, which sometimes just appear in the clip without saying anything. All these issues require more careful preprocessing and sample selection as follows to curate a useful dataset.

Rule-based filtering To avoid samples with too much ambiguity, we enforce some restrictions on the labels. For character-related tasks, we keep samples with exactly two named characters and remove samples with more than one interaction and relationship. The order of the character is tracked as there are relationships such as "parent", which are not mutual. We also keep only samples with one place and event label, as that corresponds to correctly segmented scenes. For actions, we keep the actions that are salient in the clip with a duration of more than 4 seconds.

Introducing script information Since subtitles are noisy and lack information of the speakers at an utterance level, we decide to use the dialogue in the script to replace the subtitle. Using a clip from the movie Indiana Jones and the Last Crusade as an example (Table 10), it is apparent that the script provides richer information, including the description of each scene and the speaker information for all the dialogues. We use all scripts on The Internet Movie Script Database (IMSDb)[1] , and use the TMDB API [2] to get the meta-information for each title. The labels from MovieNet and MovieGraphs are based on the IMDB ID, so we get the mapping from IMDB to TMDB ID, and then map the samples to the corresponding script files.

Note that subtitle does not strictly follow the dialogue provided in the script, and is more like an improvisation from the actors. Therefore, finding the corresponding segment of script for a subtitle itself is a long-text retrieval task. We use fuzzy string match for each utterance in the subtitle to an utterance in the script based on editing distance. An utterance in the script may be split into multiple utterances in the subtitle, so we use a partial ratio match, which matches the shorter utterance with substrings of the same length in the script utterance. We add dummy tokens to the script utterance to make sure it is the longer one, and cut off with a match score of 90. We then collect the matched utterance from the script to replace the original subtitles for our tasks. Our method expects high precision with lower recall compared with the embedding-based retrieval method for ignoring sentences with the same semantics, which meets our requirement for data filtering.

Modified character prediction task It usually does not make much sense to predict a named character from the subtitle alone unless there is a name called out in the clip, while providing a synopsis makes it a named entity recognition task without making an inference. Therefore, we modify the character prediction task and let the model match the character to the speakers. This task requires an understanding of the synopsis and dialogue, and can have samples constructed without any human labeling.

4.2.1 LLM-based sample selection

We take the common assumption that verifying an answer is easier than generating one, and we do another round of filtering using LLM for all tasks other than summarization tasks and turning point prediction. We provide the LLM with data samples and ask whether the expected output can be inferred from the subtitle as well as the synopsis for some tasks. We also ask the model to give an explanation for the decision. The prompts that we use for sample filtering can be found in Table 7. We only keep samples that are deemed to be inferable from the input and use them in both training and testing. We also provide the explanation at the training time so that the model generates the label corresponding to the task and gives an explanation.

4.3 Domain Adaptation with Instruction Finetuning

We adapt LLM to the movie domain following the instruction finetuning paradigm (Shi et al., 2024; Zhang et al., 2023a). Instruction finetuning is an approach that finetune pretrained LLMs on a va-

[1] https://imsdb.com/

[2] https://www.themoviedb.org/

riety of prompts and tasks in the form of natural language, such that the model learns the desired capability while being able to follow different instructions for practical use. To maintain models' instruction-following capability, we mix the movie domain-specific tasks with samples from general instruction dataset. We train the model in a regular supervised finetuning setup using causal language model loss, with the difference that we mask out the loss on the instruction. That encourages the model to learn the narrative understanding tasks itself given the input rather than completing the input, which sometimes consists of movie text much longer than the expected output.

5 Experiments & Results

5.1 Experimental Setup

Dataset We split the samples for each task by movie titles using the train-test split from the MovieNet and MovieGraphs datasets. This prevents any information leakage from the model seeing the same movie content from different samples during the training run. We train models with a mixture of narrative understanding tasks and general instructions from LIMA (Zhou et al., 2024) and GPT-4-LLM (Peng et al., 2023). General instructions are sampled from these two datasets to the specified amount, with the order of the samples shuffled randomly. The number of samples for each individual task group can be found in Table 6.

Models We conducted the experiments on the instruction-finetuned version of the LLama3 and LLama3.1 family (Dubey et al., 2024), as it generally leads to better performance for continued instruction fine-tuning (Zhang et al., 2023b). We use a standard setting for fine-tuning and train each model for 3 epochs with a learning rate of 1e-5 and a weight decay of 1e-2 following AlpaGasus (Chen et al., 2023). We used a total batch size of 64 on 8 A100 GPUs with DeepSpeed Stage3 [3]. We keep multiple checkpoints for each setting for evaluation, as loss in the evaluation set is a poor indicator of the quality of text generation (Zhou et al., 2024).

Evaluation Metrics Story elements tasks that predict events (interaction, action) and existent (character, place) expect output that is usually a word or a phrase for the corresponding element. We use exact match to check if the ground-truth label is correctly generated by the model while ignoring the generated explanation. For interaction prediction, we extract the lemma of the root verb in the output and ground truth before matching.

For discourse tasks, we report traditional generation metrics including BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) for clip and story summarization. We report the average distance from the predicted index to the ground truth for the turning point prediction.

5.2 Domain Instruction Finetuning (DIT)

Training Method for DIT We first investigate what LLMs' zero-shot capability is on different narrative understanding tasks $\mathcal{T}$. We also examine how different training methods affect the domain adaptation procedure, including the parameter-efficient training (PEFT) method LoRA (Hu et al., 2021) and instruction embedding noise NEFTune (Jain et al., 2023). We show the average metrics for the story tasks and the discourse tasks in Table 2. The metrics for the individual tasks are available in the Appendix.

Model	Story	Discourse	
	Acc.	BLEU-2	RougeL
LLAMA3.1-8B	0.151	0.092	0.169
LLAMA3.1-70B	0.214	0.124	0.194
Finetune	**0.321**	**0.150**	**0.194**
LoRA-8-16	0.273	0.146	0.193
w/o NEFTune	0.304	0.147	0.191
LLAMA3-8B-Finetune	0.310	0.147	0.191

Table 2: Narrative tasks metrics for different ablations

Our results show that fine-tuning on narrative understanding tasks greatly improves models' performance on all training tasks. For LoRA finetuning, we test different LoRA rank and alpha with a dropout equal to 0.01 and a learning rate of 3e-4 following Alpaca-LoRA[4]. The LoRA finetuned model shows lower average accuracy on story tasks and generation metrics for the summarization tasks. Increasing the rank and alpha does not increase the results correspondingly either. Removing the embedding noise NEFTune from the instruction hurts the performance, so we keep it in all our following experiments.

We also compare the Llama3.1-8B model with the 70B version and its earlier variant Llama3. Although the 70B variant has a stronger zero-shot performance than the 8B model, it has a much

[3] https://github.com/microsoft/DeepSpeed

[4] https://github.com/tloen/alpaca-lora

Model	Story	Discourse		
	Acc.	BLEU-2	Rouge-L	Distance
Zero-shot	0.151	0.092	0.169	3.08
+Discourse	0.036	0.148	0.193	3.02
+Story	0.312	0.053	0.128	4.68
+Domain	0.317	0.149	0.193	2.79
+Domain +General	**0.322**	**0.150**	**0.194**	**2.52**

Table 3: Effect of data composition for Domain Instruction Finetuning

lower performance than the finetuned model. The Llama3 model appears to be a little worse than the more recent Llama3.1 after the same finetuning.

Effect of Data Composition for DIT We then look into what is the best data composition for the instruction finetuning. We finetune the model with all narrative understanding tasks from our dataset combined with general instructions and examine whether a different mix ratio affects the outcomes. To examine whether learning some narrative understanding tasks helps the others, we also trained the model with either the tasks for the story element or discourse.

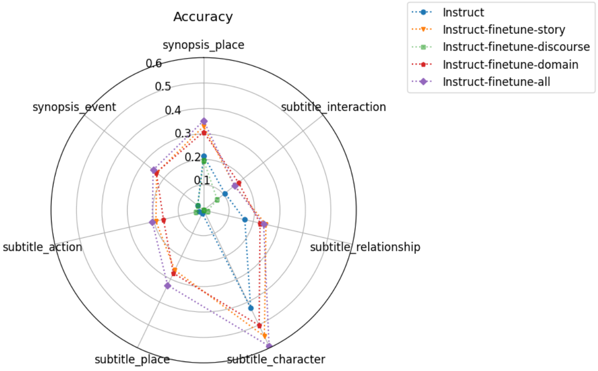

Figure 2: Story narrative understanding tasks performance trained with different data mixtures

Table 3 shows the model's average metrics on story element tasks and discourse tasks respectively, and the accuracies on individual prediction tasks are shown in Figure 2. While the general LLM fails to predict actions or places from the subtitle out-of-the-box, training only on discourse worsens the accuracy to close to zero. This suggests that training on story tasks or discourse tasks does not generalize to the other category of tasks since the model overfits to the specific input format and tasks. Finetuning on all movie domain tasks increases the performance across all tasks in both categories of tasks, while adding generation instructions benefits the model's performance as well. However, as shown in Figure 4, oversampling general instructions does not improve the model's training task performance any further.

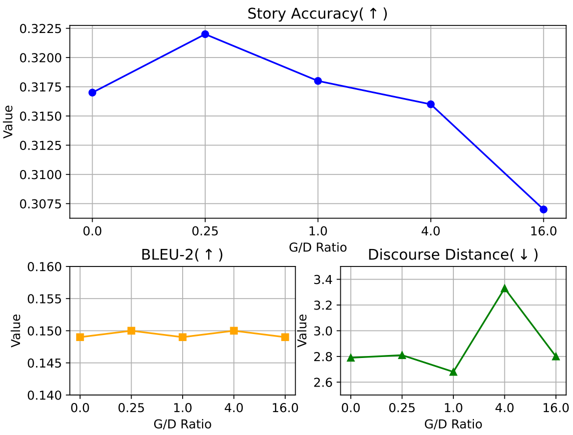

Figure 3: Effect of General/Domain instructions Ratio on narrative understanding tasks performance

Generalization on unseen tasks with DIT We check the performance on three out-of-distribution benchmarks to see how the finetuned model generalizes to unseen tasks $\mathcal{U}$ or maintains its original capability.

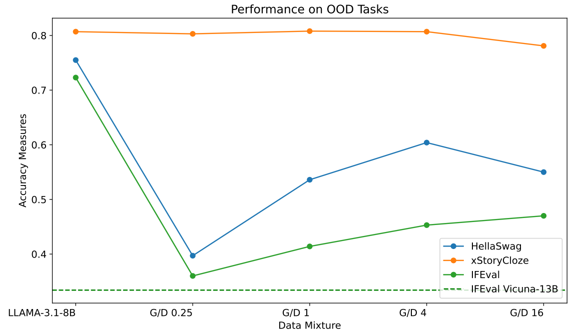

Figure 4: Models performance on out-of-domain tasks with different General/Domain instruction Ratio. The most left value are the base model without domain instruction finetuning.

Hellaswag (Zellers et al., 2019) and the English part of xStoryCloze (Lin et al., 2022) both test the model's commonsense natural language inference capability by letting it choose a sentence to finish a passage or story. Meanwhile, IFEval (Zhou et al., 2023) focuses mainly on the ability to follow various instructions. We obtain benchmarking results with the *lm-evaluation-harness* library from EleutherAI [5]. Figure 4 shows that the performance of xStoryCloze remains intact as a text completion task without complex formatting, while the performance on Hellaswag and IFEval drops dras-

[5] https://github.com/EleutherAI/lm-evaluation-harness

Model	Story Acc.	BLEU-2	Rouge-L	HellaSwag	xStoryCloze	IFEval
Baseline	0.151	0.092	0.169	**0.755**	**0.807**	**0.723**
DIT	0.322	**0.150**	**0.194**	0.397	0.803	0.360
DPT+DIT	**0.331**	0.144	0.187	0.277	0.772	0.349

Table 4: Effects of Domain Pretraining on Domain Instruction Finetuning. The metrics for narrative understanding tasks are the averaged results.

Model	NarrativeQA	Hate Speech	Political	LGBTQ	Religious
Zero-shot	0.190	0.771	0.450	0.692	0.839
SFT	0.373	0.931	0.922	**0.923**	0.934
DIT-SFT	**0.375**	0.928	0.915	0.922	**0.934**
DPT-DIT-SFT	0.369	**0.934**	**0.925**	0.920	0.933

Table 5: Results on downstream task supervised finetuning. BLEU-2 for narrativeQA. Accuracy for harmful content detection datasets.

tically after the instruction finetuning. The model regains some of the instruction-following capability with an increased amount of general instructions, but is not able to recover to the original level. This suggests the necessity of a high-quality and large enough general instruction dataset, for domain adaptation while maintaining the instruction-following capability. The reinforcement learning from human feedback stage could also potentially help.

5.3 Interplay of Domain Adaptation Stages

Impact of DPT on domain and unseen tasks We also examine whether DPT influences the effectiveness of subsequent domain instruction finetuning as in Wu et al. (2023). We train the LLM on all the movie scripts as a text completion task for one epoch, before conducting the same instruction finetuning as in previous setups.

The results in Table 4 show that although pretraining increases the accuracy of story element prediction, it decreases the performance on the discourse tasks, and it is detrimental to model's general instruction-following capability. This aligns with the results of studies in other domains (Wu et al., 2024).

Impact of domain adaptation on downstream tasks We further investigate whether the domain adaptation uniformly improves the model's performance on downstream tasks in the movie domain.

We compare the supervised finetuning results on several tasks in the movie domain with the same training setups, based on the baseline models (SFT) or the domain-adapted models respectively. We train the model using an internal dataset to classify harmful content from subtitles, including hate speech, references to political, LGBTQ, or religious content. We also train the model on NarrativeQA (Kočiský et al., 2018), which is a long-form generative QA based on the full text of a book or movie script. Example prompts for downstream tasks can be found in Table 9.

As suggested in Table 5, the performances on all tasks are effectively improved by supervised finetuning compared to the zero-shot setting, that includes the accuracy for *story* tasks and BLEU or Rouge for *discourse* tasks. However, the additional stages of domain instruction finetuning and pretraining do not bring a consistent improvement across all tasks. And it is still an open research question on how to effectively measure the correlation between different domain tasks and provide a wider coverage through the domain adaptation.

6 Conclusion

In this paper, we construct a movie-domain instruction dataset consisting of a suite of narrative understanding tasks inspired by Narrative Theory, and use it to analyze the effect of different domain adaptation stages.

We demonstrate that instruction finetuning on the movie domain effectively increases the model's performance on all narrative understanding tasks, but comes with trade-offs between general instruction finetuning capability. Additionally, we examine the interaction of instruction finetuning with domain pretraining and domain downstream tasks, revealing the benefits and limitations of adaptation approaches. Our findings provide insights into how LLMs can be effectively adapted to domains with complex storytelling structures, paving the way for future advancements in cinematic AI applications.

7 Limitations

We rely on existing annotated datasets as our source data. Despite quality control measures, there can be noise in the sample that we constructed, including mislabeling, etc. We run our experiments mainly on the Llama family of models, and the results can be further validated on other models. The training data may contain offensive content and is not examined by the authors.

References

Seymour Benjamin Chatman. 1980. *Story and discourse: Narrative structure in fiction and film*. Cornell university press.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, et al. 2023. Alpagasus: Training a better alpaca with fewer data. *arXiv preprint arXiv:2307.08701*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Jiaxi Cui, Munan Ning, Zongjian Li, Bohua Chen, Yang Yan, Hao Li, Bin Ling, Yonghong Tian, and Li Yuan. 2024. Chatlaw: A multi-agent collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model.

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*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. 2020. Movienet: A holistic dataset for movie understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 709–727. Springer.

Neel Jain, Ping-yeh Chiang, Yuxin Wen, John Kirchenbauer, Hong-Min Chu, Gowthami Somepalli, Brian R Bartoldson, Bhavya Kailkhura, Avi Schwarzschild, Aniruddha Saha, et al. 2023. Neftune: Noisy embeddings improve instruction finetuning. *arXiv preprint arXiv:2310.05914*.

Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mingjie Liu, Teodor-Dumitru Ene, Robert Kirby, Chris Cheng, Nathaniel Pinckney, Rongjian Liang, Jonah Alben, Himyanshu Anand, Sanmitra Banerjee, Ismet Bayraktaroglu, et al. 2023. Chipnemo: Domain-adapted llms for chip design. *arXiv preprint arXiv:2311.00176*.

Pinelopi Papalampidi, Frank Keller, and Mirella Lapata. 2019. Movie plot analysis via turning point identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1707–1717, Hong Kong, China. Association for Computational Linguistics.

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

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pre-training.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Haizhou Shi, Zihao Xu, Hengyi Wang, Weiyi Qin, Wenyuan Wang, Yibin Wang, Zifeng Wang, Sayna

Ebrahimi, and Hao Wang. 2024. Continual learning of large language models: A comprehensive survey. *arXiv preprint arXiv:2404.16789*.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Mohamed Amin, Le Hou, Kevin Clark, Stephen R Pfohl, Heather Cole-Lewis, et al. 2025. Toward expert-level medical question answering with large language models. *Nature Medicine*, pages 1–8.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Paul Vicol, Makarand Tapaswi, Lluis Castrejon, and Sanja Fidler. 2018. Moviegraphs: Towards understanding human-centric situations from videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022a. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022b. Super-naturalinstructions:generalization via declarative instructions on 1600+ tasks. In *EMNLP*.

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

Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Weidi Xie, and Yanfeng Wang. 2024. Pmc-llama: toward building open-source language models for medicine. *Journal of the American Medical Informatics Association*, page ocae045.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. *arXiv preprint arXiv:2304.01196*.

Hongyang Yang, Xiao-Yang Liu, and Chris Wang. 2023. Fingpt: Open-source financial large language models. *ArXiv*, abs/2306.06031.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023a. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Zihan Zhang, Meng Fang, Ling Chen, and Mohammad-Reza Namazi-Rad. 2023b. CITB: A benchmark for continual instruction tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9443–9455, Singapore. Association for Computational Linguistics.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2024. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

A Appendix

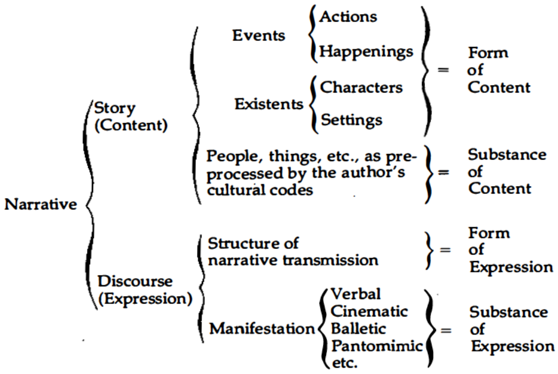

Figure 5: Elements of Narrative Theory (Chatman, 1980)

Task	samples
LIMA	1000
GPT-4-LLM	52002
subtitle place	328
subtitle action	214
subtitle character	367
synopsis place	795
synopsis event	928
subtitle relationship	233
subtitle interaction	165
Total Story	3030
clip subtitle synopsis	1017
story subtitle synopsis	1218
synopsis turning point	2217
Total Discourse	4452

Table 6: Statistics of instruction datasets used. The general instructions are sampled from LIMA and GPT-4-LLM

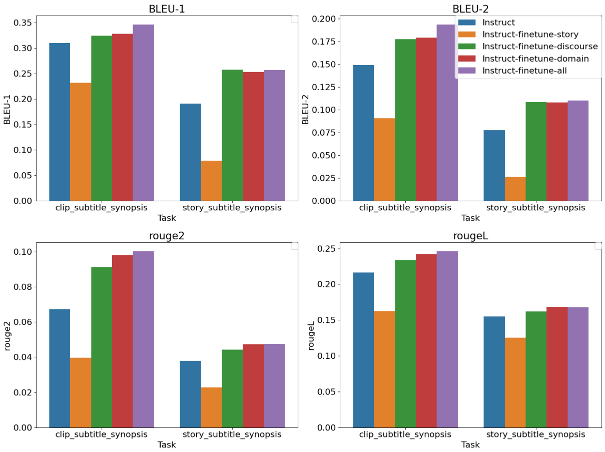

Figure 6: Discourse narrative understanding tasks performance trained with different data mixtures

Prompt

You are an expert in film theory and film criticism, with a deep understanding of cinema from various genres, eras, and cultures. You are good at analyzing films by examining narrative structure, cinematography, sound design, and thematic content. You can draw inferences from subtitles, should demonstrate a deep understanding of film as an art form. Focus on character development, plot progression, subtext, and cultural context. Provide objective and impartial analysis, avoiding personal biases.
You will be provided with {*task-specific label name*} and {*task-specific input format*}. Please determine if {*task-specific label name*} is implied or can be inferred from the {*task-specific input format*} and explain why.
Label: {*label*}
Input: {*input*}

Table 7: Prompt for LLM sample filtering. The *task-specific input format* includes the subscript, synopsis,and subtitle. The *task-specific label name* can be place, event, etc.

Task	Prompt
story_subtitle_synopsis	Your input will consist of subtitles from a scene in a movie. Please provide a concise synopsis that summarizes the content of the whole clip in a few sentences. ### Input: Subtitle: {subtitle} ### Response: {synopsis}
subtitle_action	You will be provided with subtitles of a shot from a movie and a background description of the story as the input. Try your best to imagine the plot of the movie shot, and predict the actions involved by the characters. Respond with the actions as a list of verb phrases. ### Input: Subtitle: {script} ### Response: {actions}
subtitle_character	You will be provided with subtitles and a list of characters of a movie clip, and a synopsis describing the background. Try your best to identify how the speakers correspond to the characters provided. Response with a json format with character as the key and corresponding speaker as the value. ### Input: Synopsis: {synopsis} Characters: {character} Subtitle: {masked_script} ### Response: {label}
synopsis_turning_point	You will be provided with a synopsis of a part of a movie with each sentence indexed. Predict the turning point between two story parts provided based on the plot. Answer with the sentence index only. ### Input: Synopsis: {synopsis} ### Response: {turning_point}

Table 8: Prompts used for different narrative understanding tasks. The content in brackets is replaced with the content from each sample. The rest of tasks use prompt similar to the provided ones with some adjusts on the wording.

Task	Prompt

narrativeQA	You will be given the synopsis of a movie and a question whose answer can be found in the movie. Answer the question concisely with a phrase or a short sentence.
	### Input: Question: {question}
	Synopsis: {synopsis}
	### Response: {answer}

hate speech	You will be given the caption of a movie clip. Determine if there it contains hate speech, answer either positive or negative.
	### Input: Caption: {caption}
	### Response: {final_label}

Table 9: Example prompts used for downstream classification. The content in the brackets are replaced with the content from each sample.

Subtitle	Script
00:00:13.502 –>00:00:14.662 Well, yes, sir. 00:00:14.736 –>00:00:16.260 It's right here. 00:00:16.338 –>00:00:17.999 I'm glad to see that... 00:00:18.073 –>00:00:21.008 because the rightful owner of this cross 00:00:21.076 –>00:00:24.068 won't press charges if you give it back. 00:00:24.146 –>00:00:27.638 He's got witnesses, five or six of them.	INDY Well, yes, sir. It's right here! SCENE *INDY shows the CROSS, more or less handing it to the SHERIFF to make his point. The Sheriff takes it casually.* SHERIFF I'm glad to see that... because the rightful owner of this Cross won't press charges, if you give it back. *FEDORA enters the house, followed by ROSCOE, ROUGH RIDER and HALFBREED. He politely removes his hat and holds it in his hand. He nods at INDY in a friendly manner.* SHERIFF He's got witnesses, five or six of them.

Table 10: Comparison of Subtitle and Script for the same movie clip

Task	Dataset	LLM kept ratio	Samples
subtitle_action	MovieNet	46.2%	1604
subtitle_place	MovieNet	46.0%	2031
subtitle_character	MovieGraphs	81.1%	3804
subtitle_interaction	MovieGraphs	87.8%	836
subtitle_relation	MovieGraphs	73.4%	712
synopsis_event	MovieGraphs	85.0%	4276
synopsis_place	MovieGraphs	78.3%	3937
short_subtitle_synopsis	MovieGraphs	-	4705
long_subtitle_synopsis	MovieNet	-	2366
synopsis_turning_point	MovieNet	-	2226

Table 11: Statistics on train tasks