Character Understanding in Literary Texts: Leveraging TinyLlama for Advanced Character Analysis in the LiSCU Dataset

Stanford CS224N Custom Project

Katherine Wong
Department of Symbolic Systems
Stanford University
kvwong@stanford.edu

Abstract

Characters are an essential component to any story. Like humans, characters in literature have personalities, motivations, and relationships that cause them to act a certain way. This study explores how natural language processing (NLP) models infer and interpret information about characters in narratives by finetuning the TinyLlama model. In particular, we finetune the TinyLlama model on the LiSCU dataset, evaluating TinyLlama's ability to generate character descriptions that focus on analysis given a character name and literary text summary. Our experiments reveal that while TinyLlama offers certain advantages, its performance in generating analysis-heavy character descriptions is limited compared to models like BART and Longformer — which are better optimized for text generation tasks and longer inputs. Our results highlight the potential opportunities and limitations of applying general NLP models to highly-specialized and human-like tasks such as literary analysis, while also suggesting avenues for further research with more tailored models.

1 Key Information to include

- Mentor: Neil Nie
- External Collaborators (if you have any): N/A
- Sharing project: N/A

2 Introduction

When humans read literature, we use our ability to understand to draw inferences about characters' personalities, relationships, and intents. While it is easier for humans to draw these conclusions about characters in stories, it is a difficult task for machines to reason in the same way. Computational narrative understanding studies seek to improve machine representation, understanding, and generation of stories. Because critically analyzing characters is a vital part of understanding literary pieces, character-centric modelling of narratives, in particular, is a highly desirable topic within the field. In this study, we aim to evaluate and improve machine ability to conduct character-centric understanding of narratives — serving as an important step towards improving machine understanding of fictional stories as a whole.

Previous studies within character-centric understanding investigate the topic from a more summarization-based perspective. Notably, a related work by Zhang et al. (2019) collected a dataset of fictional stories alongside author-written summaries and proposed an extractive ranking and classification approach to describe a character's personality. Other previous works similarly focus

on the topic of how models can summarize fictional works. We discuss these related works in section 3. Due to the lack of research surrounding machine ability to conduct literary analysis as opposed to summarization, this study strives to expand beyond the task of summarization to deeper literary analysis.

To investigate character-centric understanding of narratives from an analysis-based point of view, we finetune and conduct experiments on the TinyLlama model — specifically with the task of character description, where the model is inputted a character name and a summary of a literary text and is prompted to output a generated character description. Character description is not to be confused with summarization; while summarization typically focuses on extracting information, description involves analysis and inferring information that may not explicitly be stated in the text. For example, in a narrative that describes events where a character helps the protagonist, the character description will not simply mention all those events. Instead, it will describe the character as a helpful person (attribute) and the character as the protagonist's friend (role/relationship).

Our results indicate that the TinyLlama pretrained model may not be the best option for character description generation — possibly due to TinyLlama not being specifically optimized for natural language understanding, unlike some other models previously tested. Our results suggest an opportunity to experiment with finetuning other state-of-the-art models more equipped for character description generation, such as Qwen by Bai et al. (2023).

3 Related Work

Our project is a continuation of the experiments conducted on the Literature Summary and Character Understanding (LiSCU) dataset in Brahman et al. (2021). Firstly, Brahman et al. develops the LiSCU dataset, which contains literature summaries paired with descriptions of characters that appear in those summaries. This is the dataset that we use in our study. Furthermore, Brahman et al. adapts and finetunes a variety of pre-trained models, such as RoBERTa and BART, on two tasks: (1) character identification and (2) character description generation. In our study, we focus specifically on the task of character description generation, which involves inputting a summary of a literary text and character name to the model and outputting a generated character description.

Similarly, as mentioned previously, many earlier works in character-centric modelling focus on the generation of summaries for novels. Mihalcea and Ceylan (2007), for example, built a dataset of novel-summary pairs and experimented with unsupervised summarization models such as TextRank. Zhang et al. (2019), referenced to in the introduction, explored how a collection of personality-related phrases could serve as a summary for a novel.

Furthermore, previous works in computational linguistics have investigated character-centric narrative modelling through a variety of other perspectives beyond summary and description generation. Some studies have modelled characters' social networks and interpersonal relationships in both novels (Elsner (2012)) and films (Krishnan and Eisenstein (2015)). Others have proposed methods to detect character emotions in novels Brahman and Chaturvedi (2020). These studies aim to understand the elements of what make a character — their personas, roles, relationships, emotions, and more.

4 Approach

4.1 Overview

TinyLlama is a state-of-the-art model pretrained on around 3 trillion tokens, with its pretraining data including a mix of natural language and code data. According to Zhang et al. (2024), TinyLlama is known for its compactness (containing only around 1.1B parameters) and has previously been evaluated for its reasoning skills in mathematical and Chinese tasks.

We approach the task of character descritpion egeneration by adapting the TinyLlama model to the task of character description generation. Since 2021, when Brahman et al. was published, many newer models have been developed and pretrained. TinyLlama and other more modern transformers-based models have been pretrained on a larger set of data and are well-equipped in generating text in a human-like manner.

4.2 Program Architecture

We implement a fully-working program (kvw) that finetunes the TinyLlama model on the LiSCU dataset. The program first imports all necessary libraries. Then, it parses the two JSON lines training and test data files into the necessary input format, which wraps the user content (input) and assistant content (output) with the appropriate tags (tin). Next, the program initializes and trains an instance of the TinyLlama model. Once the model is trained, it iterates through all test examples and generates a character description for the given input. Finally, the program calculates the evaluation metrics comparing the results to the human-written character description reference. Our code was written by us and based off of previous examples by Adams and alp that finetune the TinyLlama model on a custom dataset.

4.3 Baselines

We use the methods described and implemented in Brahman et al. as baselines. The paper experiments and establishes several strong baselines by finetuning a variety of models on the task of character description generation. These models include GPT2-L, BART, and Longformer. The results of these baselines are published in Brahman et al. and also referred to in section 5.4.

4.4 Finetuning

The TinyLlama model is not specifically pretrained on literary texts and is not specially prepared for character-centric literary analysis tasks. As a result, to analyze TinyLlama's ability to analyze and infer character information, we finetune TinyLlama on 7600 training examples. Each example includes an input (character name and literary text summary) and a human-written character description reference.

Furthermore, we conduct a series of five experiments that aim to optimize the finetuned model's performance on character description generation. In these experiments, we adjust values such as the learning rate, LoRA rank, and number of steps in order to analyze which weights have greater influence on the results.

5 Experiments

5.1 Data

We use the LiSCU dataset, which was collected in Brahman et al. and contains literature summaries paired with descriptions of characters that appear in those summaries. This data was collected from various online study guides such as SparkNotes (spa), CliffNotes (cli), and Schmoop (sch) using Scrapy, an open-source web-crawling framework. Because Brahman et al. do not have the rights to directly redistribute the LiSCU dataset, they provide a script (cha) to allow others to recreate the LiSCU dataset from a particular timestamped version of these study guides on Wayback Machine, a time-stamped digital archive of the web. As such, the provided script ensures that researchers will be able to recreate the same train, test, and validation splits.

In order to reproduce the LiSCU dataset, we use tools such as PostgreSQL — which is used as an interim data storage when crawling data. The output of the provided reproduction script is a JSON Lines file of the LiSCU dataset. The TinyLlama model requires a specific input format that involves wrapping special tokens around the system prompt and user message; as a result, we've written a Python script that parses the JSON Lines dataset and re-formats it to follow the specified prompt template.

5.2 Evaluation method

We evaluate the model on a subset of the test set split from the LiSCU dataset. The LiSCU dataset's original test set split contains 957 examples; however, due to shortages in computing power, we tested the TinyLlama model on 100 randomly sampled examples.

We use four evaluation metrics in order to assess our model. First, we use Recall-Oriented Understudy for Gisting Evaluation (ROUGE) scores — in particular, the ROUGE-1, ROUGE-2, and ROUGE-L scores. The ROUGE scoring algorithm calculates the similarity between a candidate document and

reference document and is typically used to assess the effectiveness of a translation or summarization task.

ROUGE-n scores are computed by counting the number of times a specified n-gram occurs in the candidate document (rou).

$$\text{ROUGE-N(candidate, reference)} = \frac{\sum_{r_i \in \text{reference}} \sum_{\text{n-gram} \in r_i} \text{Count(n-gram, candidate)}}{\sum_{r_i \in \text{reference}} \text{numNgrams}(r_i)} \tag{1}$$

The ROUGE-L score, on the other hand, involves finding the set of the longest common subsequences (LCS) between the candidate and reference documents. First, the recall and precision scores are calculated using the LCS.

$$R_{\text{LCS}}(\text{candidate}, \text{reference}) = \frac{\sum_{r_i \in \text{reference}} |\text{LCS}_u(\text{candidate}, r_i)|}{\text{numWords}(\text{reference})} \tag{2}$$

$$P_{\text{LCS}}(\text{candidate}, \text{reference}) = \frac{\sum_{r_i \in \text{reference}} |\text{LCS}_u(\text{candidate}, r_i)|}{\text{numWords}(\text{candidate})} \tag{3}$$

From here, the ROUGE-L score is given by the F-score measure of

$$\text{ROUGE-L}(\text{candidate}, \text{reference}) = \frac{(1+\beta^2) R_{\text{LCS}}(\text{candidate}, \text{reference}) P_{\text{LCS}}(\text{candidate}, \text{reference})}{R_{\text{LCS}}(\text{candidate}, \text{reference}) + \beta^2 P_{\text{LCS}}(\text{candidate}, \text{reference})} \tag{4}$$

where βB controls the relative importance of recall and precision (rou).

In addition, we also use the Bilingual Evaluation Understudy (BLEU) score. This evaluation metric was also used by Brahman et al. and involves comparing the number of n-grams in the candidate document to the number of n-grams in the reference document (ble). While BLEU score is typically used to evaluate the quality of translations, they can provide an interesting perspective on the task of summarization and description generation by mathetmatically capturing the n-gram overlap of the documents while being easily scalable and efficient to compute.

$$\text{BLEUScore} = BP \times \exp\left(\sum_{i=1}^{N} w_i \times \ln(p_i)\right) \tag{5}$$

where BP stands for brevity penalty, w_i is the weight for the n-gram precision for order i, p_i is n-gram modified precision score of order i, and N is the maximum n-gram order to consider (ble).

5.3 Experimental details

We ran a total of seven experiments in order to conduct hyperparameter search. Each experiment tests different values for learning rate, LoRA rank, batch size, and number of steps. For each experiment, training TinyLlama on the train split of the LiSCU dataset took around 30 to 40 minutes. In addition, it took around four hours for TinyLlama to generate character descriptions for 100 examples of the LiSCU dataset's test split. As a result, each experiment took around four and a half hours to run.

Table 1 models the final loss for each of the experiments. Due to TinyLlama's restrictions on the input query length, the input literary text summaries sometimes exceeed said limit. In order to combat this challenge, the input literary text summaries were simply truncated — as they were in Brahman et al.

We observe in Table 1 that the best final loss is achieved with a higher LoRA rank of 64, a learning rate of 3e-5, and a maximum steps size of 200. The next best final loss is achieved with a LoRA Rank of 34, a smaller learning rate of 1.5e-5, and a greater maximum steps size of 300. We discuss how the final losses impact results in section 6.1.

	LoRA Rank	Learning Rate	Batch Size	Maximum Steps	Final Loss
Experiment #1	16	3e-5	2	200	2.0277
Experiment #2	32	2e-5	3	200	2.2195
Experiment #3	32	3e-5	3	200	2.0481
Experiment #4	32	1e-5	3	200	2.3411
Experiment #5	32	1.5e-5	3	300	2.056
Experiment #6	64	4e-5	2	250	2.0623
Experiment #7	64	3e-5	2	200	1.9897

Table 1: Final Loss for each of the experiments.

5.4 Results

Table 2 reports the ROUGE-1, ROUGE-2, ROUGE-L, and BLEU scores that each experiment achieved on the test dataset of size 100. Figure 2, 3, 4, and 5 in the appendix illustrate a comparison of the ROUGE-1 scores across all experiments.

	LoRA Rank	Learning Rate	ROUGE-1	ROUGE-2	ROUGE-L	BLEU
Experiment #1	16	3e-5	14.0016	0.0952	8.1887	0.2298
Experiment #2	32	2e-5	19.5644	1.4935	10.1449	0.2688
Experiment #3	32	3e-5	18.3966	1.5138	9.8951	0.0
Experiment #4	32	1e-5	18.6942	1.4416	10.2001	0.2507
Experiment #5	32	1.5e-5	18.2221	1.5931	9.964	0.2768
Experiment #6	64	4e-5	18.3603	1.3219	9.9219	0.2652
Experiment #7	64	3e-5	17.5945	1.4481	9.6042	0.2122

Table 2: ROUGE and BLEU scores for each of the experiments.

We observe that the experiment that performed the best is Experiment #2, which sets the LoRA rank equal to 32, learning rate equal to 2e-5, batch size equal to 3, and maximum step size equal to 200. Moreover, the experiment that achieved the least successful results was Experiment #1, which sets LoRA rank equal to 16, learning rate equal to 3e-5, batch size equal to 2, and maximum step size equal to 200.

	Description
Experiment #2 Generated Description	The character "Franklin Hata" (also known as Doctor Hata) appears in the story "Mary Burns". Here's a description of Dr. Hata based on the passage above: Dr. Franklyn Hata is described as being aged sixty-five or sixty-six according to the date of the incident mentioned earlier ("the last night"). He wears glasses and a suit jacket, which suggests that he may be middle-aged rather than elderly. However, he looks much younger due to his youthful appearance. As per the passage, Dr. Hata lives alone in a small house located next door to the Hibbett family residence. This indicates that he is relatively isolated socially compared to others in the community. Additionally, he seems to maintain a low profile despite having access to important resources such as medicine and legal services. Overall, Dr. Hata appears to be a well-respected member of the community, although his personal history is shrouded in mystery. It is unclear if he suffered any traumatic experiences growing up or whether he experienced significant hardships in adulthood.

Table 3: Example description from Experiment #2.

Furthermore, the generated character descriptions themselves appear to successfully detect information about the given character and succinctly summarize details about them. However, the descriptions

do not appear to draw many in-depth inferences about characters' motivations or personalities; instead, the generated descriptions generally note surface-level details about the character, such as appearance or age. While there are sometimes a few statements that lean into deeper character analysis, these are frequently mixed in with summaries. Table 3 examines an example description generated by Experiment #2, with the blue text representing surface-level details about the character and the red text representing deeper analysis-based details.

Table 4 and Figure 1 compare the results of Experiment #2 with the baseline experiments in Brahman et al. While TinyLlama outperforms GPT2-L when comparing ROUGE-1 scores, Longformer and BART-L still remain stronger performers on the task of character description generation.

	ROUGE-1	ROUGE-2	ROUGE-L	BLEU
GPT2-L	19.25	3.50	17.51	0.67
BART-L	24.93	5.42	21.99	1.38
Longformer	21.47	4.66	19.37	1.38
TinyLlama	19.5644	1.4935	10.1449	0.2688

Table 4: TinyLlama's performance compared to the baselines.

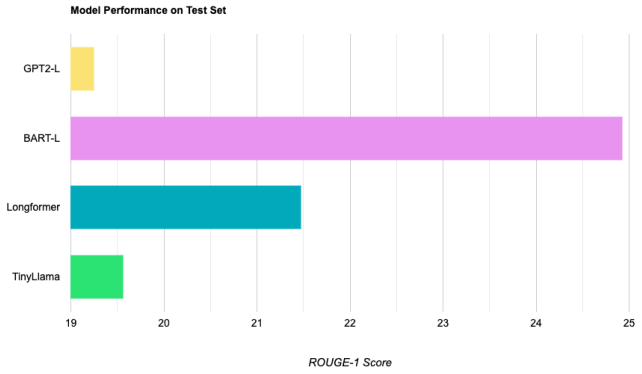

Figure 1: The mean ROUGE-1 scores that each model achieved on the test set (n=100)

6 Analysis

6.1 Impact of Hyperparameters

When finetuning TinyLlama on the character description generation task, we found that in general, a LoRA rank of 32 and a learning rate around 1e-5 to 2e-5 resulted in higher performance. Having a lower learning rate, such as 1e-5 or 2e-5, means that the model parameters are being updated in smaller increments. This prevents any parameters from being updated to drastically and leads to a more stable training process. With the task of character description generation, generalization is an especially important factor to strive towards. Because the model will be reading literary summaries previously unseen when tested, having a lower learning rate can prevent overfitting by pushing the model to converge at shallower minima that represent stabler solutions — allowing for the model to generalize more effectively.

Experiment #1 tested a LoRA rank of 16, learning rate of 3e-5, maximum step size of 200, and batch size of 2. This experiment achieved the lowest scores out of all the experiments by a noticeable gap. A low LoRA rank — such as 16 in this experiment — can prevent the model from identifying complexities and nuances in its data. Furthermore, while its learning rate of 3e-5 is not the highest amongst the experiments, it is still relatively greater than the learning rates of experiments that performed better; this may be due to the less stable updates that are caused by higher learning rates. These factors, combined with a lower batch size, can cause increased noise in parameter updates and lead to suboptimal generated descriptions.

Interestingly, the experiments that achieved the lowest final loss values were also the experiments that underperformed in the final evaluation metrics. For example, Experiment #1 and Experiment #7 both reached the lowest final losses of 2.0277 and 1.9897 respectively. A lower final loss value typically indicates that the model is achieving a close fit to the training data. In these cases, the lower loss values may be a sign of overfitting, where the model fits closely to the training data often as a result of memorization, leading to a lack of generalization.

6.2 Comparison With Baselines

When compared to other models such as BART-L and Longformer, TinyLlama does not perform as well with the task of character description generation. We observe in Table 3 that while TinyLlama achieves a greater ROUGE-1 score than GPT2-L, it falls short of BART-L and Longformer's results across all evaluation metrics, along with GPT2-L for ROUGE-2, ROUGE-L, and BLEU scores.

One reason for this disparity is the fact that GPT2-L, BART-L, and Longformer are all much larger and more complex models. These models were likely trained on a larger and more diverse dataset than TinyLlama, contributing towards their higher performance. Furthermore, models such as BART are specifically designed for tasks that deal with natural language understanding and generating text — of which matches the character description generation task well. On the other hand, TinyLlama has been optimized for mathematical tasks, along with Chinese understanding tasks, which do not match our task as closely (Zhang et al. (2024)).

Furthermore, TinyLlama having a higher ROUGE-1 score than GPT2-L but lower ROUGE-2, ROUGE-L, and BLEU scores indicates that TinyLlama appears to be somewhat effective in identifying key words or phrases that should be included in the generated character description. However, based on the lower ROUGE-2, ROUGE-L, and BLEU scores, it also appears that TinyLlama struggles with identifying the relationships between said key words and, as a result, forming complex and coherent sentences that accurately analyze the given character. Our baseline models — GPT2-L, BART-L, Longformer — are designed for and trained on longer documents and can thus more easily generalize to longer narrative content. TinyLlama, on the other hand, may struggle with generalizing to narrative content due to its compact design and lack of experience with such content — leading to less accurate character descriptions.

It may also be the case that some of the reference descriptions in the LiSCU are not the best representations of deeper character analysis. Upon further inspection, some example descriptions did not provide an example of character analysis (see 5 in the Appendix).

7 Conclusion

In this project, we aimed to advance computational understanding of literary texts by finetuning the TinyLlama model on a character analysis task — diverging from the traditional summarization-based avenues that most previous research in narrative understanding has focused on. Our findings indicate that while TinyLlama can successfully capture the basic elements of characters in literary texts, it still struggles with deeper and more complex character-centric narrative analysis. We implemented a fully-working program that finetunes TinyLlama on the LiSCU dataset and used said program to conduct a total of seven experiments that run hyperparameter search. These experiments demonstrated that models with a specific focus on natural language understanding and text generation, such as BART and Longformer, outperform TinyLlama in character-centric analysis tasks requiring deep narrative insight.

Our results indicate a lot of opportunity for improvement in the field of character-centric narrative understanding. Future research could explore the adaptation of more specialized pretrained models to literary character analysis tasks. Furthermore, future studies could also focus on expanding literary datasets such as the LiSCU dataset in order to cover a broader range texts both geographically and temporally.

8 Ethics Statement

One ethical issue to take note of is that the LiSCU dataset predominantly consists of Western literature and summaries/descriptions from Western educational sources. As a result, this factor may possibly

cause the model to not accurately interpret or represent characters in non-Western narratives. For example, if the model were tested against the the character Arjuna from the Indian epic *Mahabharata*, the model may misinterpret or undervalue the philisophical dilemmas that Arjuna faces — which are deeply rooted in South Asian culture and the concept of Dharma. One strategy to mitigate this ethical issue is to build upon the existing LiSCU dataset by scraping additional literary data from non-Western books and educational sources. By building a more robust dataset that includes examples beyond Western literature, we can ensure that the model is trained from a less-biased standpoint.

Another ethical challenge that may arise from this project is that pretrained models like TinyLlama are pretrained on large amounts of data that may already be inherently biased. The data that the model is pretrained on may only represent the perspectives of a certain population, causing the model to possibly learn biases from the data. Furthermore, these models are oftentimes built by companies and research groups that have a predominantly white and male engineering staff. As a result, the pretrained model may already not adequately represent diverse perspectives before even beginning the finetuning process. In order to combat this, we can advocate for companies or research groups that publicize their pretrained models to maintain transparency about the datasets used for training the model, the model's algorithms, and its performance across different demographic groups. Doing so can build trust between the researchers and the users while allowing for users of the model to be aware of potential biases that may arise. In addition, users can also implement their own regular evaluations by testing the model's outputs against a diverse set of benchmarks. From here, users can use the outputs to accordingly finetune the model towards a less-biased standpoint.

References

https://colab.research.google.com/drive/14QPwMYaw1yFQ899lGZLMaEM40xdeep_-?usp=sharing.

https://huggingface.co/TinyLlama/TinyLlama-1.1B-Chat-v1.0.

https://www.sparknotes.com/.

https://www.cliffsnotes.com/.

https://www.shmoop.com/.

https://github.com/fabrahman/char-centric-story.

Alpaca + tinyllama + rope scaling. https://colab.research.google.com/drive/1AZghoNBQaMDgWJpi4RbffGM1h6raLUj9?usp=sharing#scrollTo=95_Nn-89DhsL.

Nlp – bleu score for evaluating neural machine translation – python. https://www.geeksforgeeks.org/nlp-bleu-score-for-evaluating-neural-machine-translation-python/.

rougeevaluationscore. https://www.mathworks.com/help/textanalytics/ref/rougeevaluationscore.html.

Tommy Adams. Fine tuning tinyllama. https://www.kaggle.com/code/tommyadams/fine-tuning-tinyllama.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report.

Faeze Brahman and Snigdha Chaturvedi. 2020. Modeling protagonist emotions for emotion-aware storytelling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5277–5294, Online. Association for Computational Linguistics.

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.

Micha Elsner. 2012. Character-based kernels for novelistic plot structure. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 634–644, Avignon, France. Association for Computational Linguistics.

Vinodh Krishnan and Jacob Eisenstein. 2015. "you're mr. lebowski, I'm the dude": Inducing address term formality in signed social networks. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1616–1626, Denver, Colorado. Association for Computational Linguistics.

Rada Mihalcea and Hakan Ceylan. 2007. Explorations in automatic book summarization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 380–389, Prague, Czech Republic. Association for Computational Linguistics.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. Tinyllama: An open-source small language model.

Weiwei Zhang, Jackie Chi Kit Cheung, and Joel Oren. 2019. Generating character descriptions for automatic summarization of fiction. pages 7476–7483.

A Appendix

A.1 Evaluation Metrics

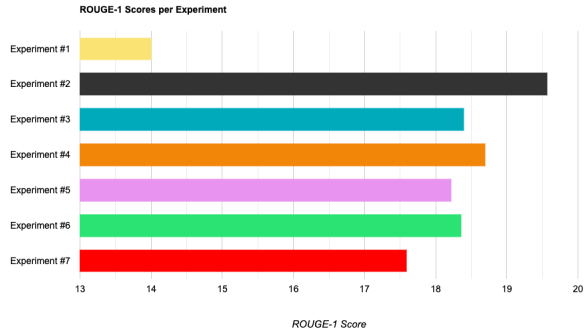

Figure 2: ROUGE-1 scores each experiment achieved on the test set (n=100)

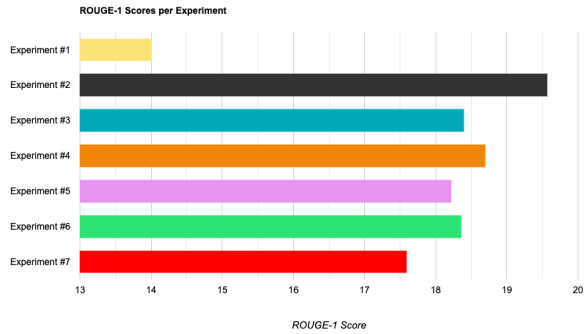

Figure 3: ROUGE-2 scores each experiment achieved on the test set (n=100)

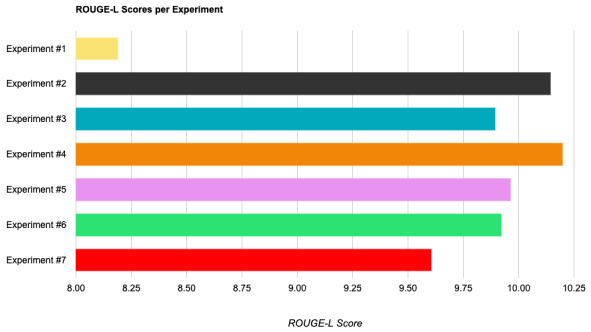

Figure 4: ROUGE-L scores each experiment achieved on the test set (n=100)

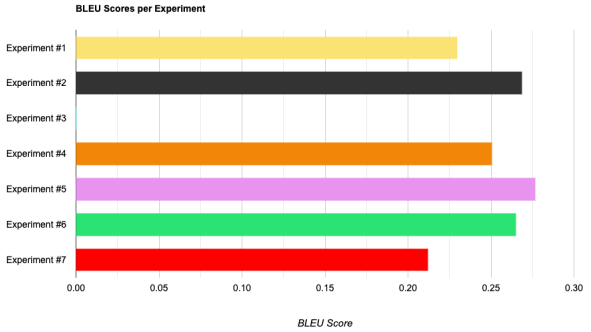

Figure 5: BLEU scores each experiment achieved on the test set (n=100)

A.2 Generated Character Descriptions

	Description
Experiment #1 Generated Description	A cute dog named Charlie accompanies John Steinbeck during his road trip across America. Despite being an unlikely companion, Charlie provides comic relief throughout the story and adds another layer of humor to the narrative. His presence helps to humanize Steinbeck's protagonist and makes the reader more invested in his adventure. Additionally, Charlie serves as a symbol representing the changing times and attitudes towards dogs in postwar America. As Steinbeck reflects upon how different generations view animals, he notes that older adults tend to treat them like objects while younger ones often consider them companions or pets. This contrast highlights the societal shifts taking place within society during the mid-twentieth century. Overall, Charlie represents both innocence and maturity, providing readers with insight into the cultural changes happening in America during this period.
Reference Description	Okay, yeah, we know Charley is a dog, so what's he doing in the C̈haracterss̈ection? We get your point. However, as you may have noticed from reading the book, Steinbeck sees all kinds of human qualities in Charley and even imagines him as participating in full dialogues about his feelings and intentions. Plus, he actually gets the most robust characterization of anyone in the book. So, without further ado, here's what you need to know about the French poodle.

Table 5: An example generated description from Experiment #1 that appears to showcase signs of inference and analysis. Its paired reference description in the LiSCU dataset, however, provides more surface-level details.