

MOCHA: A Multi-Task Training Approach for Coherent Text Generation from Cognitive Perspective

Zhe Hu

Baidu Inc
huzhe01@baidu.com

Hou Pong Chan

University of Macau
hpchan@um.edu.mo

Lifu Huang

Virginia Tech
lifuh@vt.edu

Abstract

Teaching neural models to generate narrative coherent texts is a critical problem. Recent pre-trained language models have achieved promising results, but there is still a gap between human written texts and machine-generated outputs. In this work, we propose a novel multi-task training strategy for coherent text generation grounded on the cognitive theory of writing, which empowers the model to learn essential subskills needed for writing including planning and reviewing besides end-to-end generation. We extensively evaluate our model on three open-ended generation tasks including story generation, news article writing and argument generation. Experiments show that our model achieves better results on both few-shot and fully-supervised settings than strong baselines, and human evaluations confirm that our model can generate more coherent outputs.

1 Introduction

With the recent development of pretraining techniques, large neural language models have achieved impressive results on various text generation tasks and can generate fluent outputs. However, when generating *long-form texts* (i.e., paragraphs with multiple sentences), there is still a large gap between machine-generated outputs and human written texts: the generated outputs usually suffer from *incoherence issues* and fail to maintain overall narrative coherence (See et al., 2019).

One possible reason for the above defects is the *lack of effective text planning as global guidance to control the generation process*. Compared with the traditional generation systems which often decompose the generation task into text planning and surface realization (Reiter and Dale, 1997; Carenini and Moore, 2006), current autoregressive neural language models are typically trained to produce texts in a left-to-right token-level manner, which lacks anchored goal to constrain the

generation process (Fan et al., 2019). Recent studies incorporate text planning into neural models by leveraging structured representations (Goldfarb-Tarrant et al., 2020; Hua and Wang, 2020) or latent variables (Wang et al., 2022; Hu et al., 2022) as high-level plans, but they need manually designed domain-specific plans or complicated supervision signals to train the model.

Another reason is the *ineffective usage of the negative samples as contrasts to teach the model better distinguish between correct and incorrect targets*. Negative samples are useful to enhance the model ability to generate better outputs (He and Glass, 2020). Recent work explores techniques such as contrastive learning (Lee et al., 2020; Su et al., 2022; An et al., 2022) and unlikelihood training (Welleck et al., 2020; Li et al., 2020) to leverage negative samples for model training.

We draw our motivations from the *cognitive process theory of writing* (Flower and Hayes, 1981): “*Writing is best understood as a set of distinct thinking processes which writers orchestrate or organize during the act of composing*”. In particular, the basic mental process of writing includes *planning*, *translating* (surface realization), and *reviewing*, where the reviewing process further involves evaluating and revising subskills. Current language models are typically trained to maximize the token-level loglikelihood and thus learn to acquire the writing skills all at once. However, as stated by Bruce (1978), learning the whole set of task components for writing at once makes the learning process very hard, and they suggest that *the intermediate tasks benefit to acquiring and exercising different writing subskills*.

In this work, we propose **MOCHA**, a *Multi-task training apprOach for CoHerent text generAtion by enriching the token-level generation objective with additional tasks specifically designed for different writing subskills grounded on the cognitive perspective*. Specifically, we introduce two additional

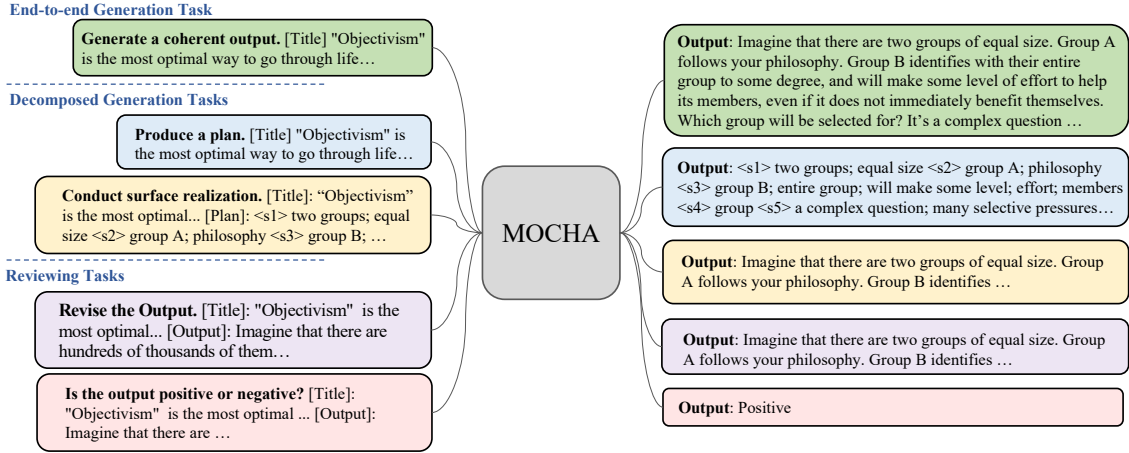


Figure 1: Overview of our framework. We train our model with different tasks grounded on the cognitive theory of writing: (1) end-to-end token-level generation task; (2) decomposed generation tasks including text planning and surface generation; (3) reviewing tasks with revising flawed targets and distinguishing between correct and incorrect options.

tasks needed for generating coherent outputs: (1) **decomposed generation tasks** that divide the end-to-end generation into text planning and surface realization, and (2) **reviewing tasks** which leverage negative samples to enforce the model to distinguish the correct and incorrect outputs and further revise the flawed texts.

Our work is closely related to the recent multi-task training approach (Sanh et al., 2021) by converting different tasks into text-to-text transfer with corresponding prompts. Recent work (Raffel et al., 2019) has shown that **multi-task learning (MTL)** with shared parameters across different tasks can effectively improve the model performance on text understanding (Aribandi et al., 2021), dialogue generation (Li et al., 2021; Su et al., 2021), and structured knowledge grounding (Xie et al., 2022). Different from previous work, we study coherent long text generation with MTL to tackle different subskills needed for writing. Experimental results show that our method outperforms strong baselines and achieves better few-shot performance compared with vanilla T5 on story generation, counter-argument generation and news article writing. Human evaluation further confirms that our method can generate more coherent outputs. Data and Code are available at: <https://github.com/Derekkk/Mocha-EMNLP22>

2 Method

Text generation is typically formulated as a sequence-to-sequence (seq2seq) transformation: $p(y|x) = \prod_{t=1}^n p(y_t|y_{1:t-1}, x)$, where (x, y) is a source-target pair. We adopt the state-of-the-art model T5 (Raffel et al., 2019) as the backbone, which is an **encoder-decoder Transformer**. For each

sample, we introduce additional training objectives to jointly improve the writing ability. Our training objectives include *end-to-end generation*, *decomposed generation* and *reviewing tasks*. All tasks are converted to text-to-text transfer with a task prompt prepended to the source input. *Notably, training samples of the augmented tasks can be constructed automatically, without further data labeling efforts.* The overall framework is illustrated in Figure 1.

2.1 End-to-end Generation Task

The end-to-end generation (Gen.) task is the same as the typical training objective for text generation. We prepend the source input with a task prompt (e.g., “Generate a coherent output”), and the model is trained to generate the target. However, only applying this task is hard to generate coherent outputs as it couples the whole set of writing processes at once and makes training difficult. Therefore, we introduce the additional subtasks.

2.2 Decomposed Generation Task

Generating narrative coherent outputs requires the model to conduct effective *text planning* to decide high-level plots, and properly reflect the plans in the *surface outputs*. Thus, we propose two decomposed generation tasks (Decomp.).

Text Planning. This task requires the model to **produce structured plots as high-level plans**. We follow Hua and Wang (2020) to adopt ordered **keyphrase chains to represent the plots**. Concretely, we extract salient noun and verb phrases from the target as keyphrases, and then concatenate the keyphrases with the same order they appear in the target as the plan (more details are in Appendix A.2). The task prompt “Produce a plan” is

prepended to the title, and the model is trained to generate the text plan, as shown in Figure 1.

Surface Realization. Surface realization task teaches the model to properly reflect the text plan in the final target. We concatenate the task prompt (e.g., “*Conduct surface realization*”), title and the corresponding plan as the input sequence, which is consumed by the model to generate the final target.

2.3 Reviewing Task

We propose two reviewing (Review.) tasks which leverage negative samples to enhance the model to better distinguish the coherent outputs from distracts, and learn to revise the flawed outputs.

Revise Task. The revise task aims to empower the model to edit the flawed outputs (Wang et al., 2018). For each sample, we construct two flawed negatives: (1) randomly shuffle the target sentences to encourage model to learn correct sentence ordering, and (2) replace the keyphrases in the target with random keyphrases to enhance better content organization. The model takes as input the task prompt (“*Revising the Output*”), title, and the flawed output, and recovers the original target.

Distinguishing Task. This task requires the model to distinguish the original output from the distracted ones given an input. The distracted targets are constructed with the same strategies as the Revise Task. Similar to Zhou et al. (2020), the input sequence is the concatenation of the task prompt (e.g., “*Which Option is Better*”), the title, an output with 50% to be the original target or a distracted one otherwise. The model is trained to predict whether the output is correct by generating “*positive*” or “*negative*”. By doing so, we expect the model to give a preference of the coherent targets and learn to generate better outputs.

2.4 Joint Training with Multi-tasks

We jointly train the aforementioned objectives with shared parameters to reinforce the writing ability. Specifically, given a source-target pair (x, y) , we first construct two decomposed generation samples for text planning and surface realization tasks respectively. Then we construct two flawed samples for the revise task. Finally, for the distinguishing task, we choose the output with 50% to be the positive target or a distracted negative target otherwise. All objectives are converted to text-to-text transfer tasks, and jointly trained to maximize the likelihood probability: $\mathcal{L} = \mathcal{L}_{\text{Gen.}} + \mathcal{L}_{\text{Decomp.}} + \mathcal{L}_{\text{Review.}}$.

	Reddit/CMV	Wikiplots	NYTimes
# Train	42,462	95,571	103,579
# Dev	6,480	5,328	5,000
# Test	7,562	5,404	5,000
# Words	116.3	425.4	218.2
# Sent.	5.5	18.0	9.1

Table 1: Statistics of the datasets. # Words denotes the average number of words in the target, and # Sent. represents the average number of sentences.

During inference, we use the end-to-end generation task to produce final outputs.

3 Experimental Setting

3.1 Datasets

We evaluate our model on three datasets of distinct domains: (1) Reddit/ChangeMyView (Reddit/CMV) for argument generation (Hua and Wang, 2020), (2) Wikiplots for story generation, and (3) New York Times for news article writing (Sandhaus, 2008). We follow the previous work (Rashkin et al., 2020) to further include topical keyphrases as guidance outline, where noun and verb phrases which contain at least one topic signature words (Lin and Hovy, 2000) from targets are extracted. The title and keyphrases are concatenated as the input x . The statistics are in Table 1, and more details are in Appendix A.1.

3.2 Model Details

We use T5-base (Raffel et al., 2019) in all experiments. During training, we optimize our model with AdamW (Loshchilov and Hutter, 2017), and the learning rate is $5e-5$. For decoding, we apply nucleus sampling (Holtzman et al., 2019) with k as 10 and p as 0.9. The maximum of generation steps are 200 for argument generation, 512 for story generation and 350 for NYT article generation.

Baselines. We first consider generation models including GPT2 (Brown et al., 2020) and T5 (Raffel et al., 2019) without multitask training. We also include strong planning-based methods: (1) CONTENTPLAN is a two-step generation model (Goldfarb-Tarrant et al., 2020; Hua and Wang, 2020), where a planner first produces ordered keyphrase plans, and a generator consumes the plans and generates final outputs; (2) BOWPLAN (Kang and Hovy, 2020) predicts keywords as the global plan to guide the generation. All models are implemented with T5-base except for GPT2. More details are in Appendix A

System	Reddit/ChangeMyView				Wikiplots				New York Times			
	B-3	R-L	Meteor	Len.	B-3	R-L	Meteor	Len.	B-3	R-L	Meteor	Len.
GPT2	19.29	23.51	37.56	129	11.39	20.00	26.91	299	16.32	21.83	31.28	212
BOWPLAN	27.19	26.86	44.33	109	12.35	22.79	30.61	229	<u>20.26</u>	25.40	<u>36.22</u>	175
CONTENTPLAN	25.70	25.71	43.73	109	13.67	21.98	<u>32.16</u>	260	19.54	23.15	34.73	191
T5	26.99	26.97	43.42	109	11.99	23.08	30.27	221	20.05	25.90	35.85	168
Our Models												
MOCHA	28.02	27.42	44.81	110	12.43	23.43	30.94	224	20.43	<u>26.21</u>	36.45	166
w/o Decomp.	<u>27.60</u>	27.28	44.41	108	12.12	23.10	30.60	221	19.80	26.28	35.83	160
w/o Review.	26.92	<u>27.31</u>	43.59	106	11.36	<u>23.35</u>	30.32	211	19.97	26.20	36.07	164
w/ SepGen.	27.22	26.92	<u>44.67</u>	103	<u>13.54</u>	22.91	32.26	249	19.87	24.08	35.16	181

Table 2: Experimental results. We report BLEU-3, ROUGE-L (f), METEOR and average output lengths (Len.). Best results are bold and second best ones are marked with underline.

4 Results and Analysis

4.1 Automatic Results

For automatic evaluation, we adopt BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and METEOR (Denkowski and Lavie, 2014).

The main results on three datasets are summarized in Table 2. Compared with the baseline methods, our model variants generate outputs with higher scores on all tasks. Specifically, our model significantly outperforms the vanilla T5 after augmenting the additional tasks, which demonstrates that our multitask training to tackle different writing subskills is critical to improve the model ability for long text generation. Moreover, our multitask training approach also surpasses or is comparable to the state-of-the-art planning-based methods, which further confirms the effectiveness of our method. We also observe that GPT2 achieves lower scores than other baselines, which indicates that only one decoder may not be sufficient since the tasks require the model to well understand and utilize the keyphrases during generation.

4.2 Model Analysis

Ablation Study. We train ablated model variants to analyze each augmented task. We first remove the decomposed generation tasks (w/o Decomp.), and the performance decreases, suggesting that incorporating text planning and surface realization tasks is helpful to improve long text generation. After removing the reviewing tasks (w/o Review.), the scores also drop, showing that improving model distinguishing and revise skills are useful to further enhance the generation ability¹. Above all, the results prove that using different writing subskills to

¹We further compute accuracy scores of the distinguishing task in Appendix A.5.

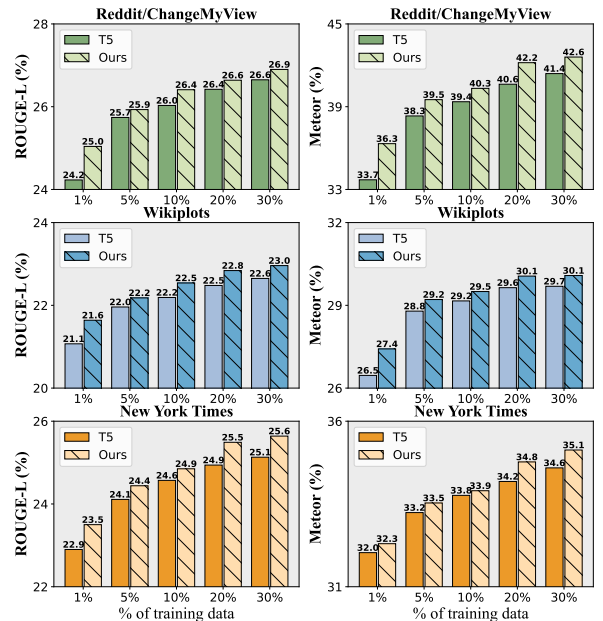


Figure 2: Few-shot results on test sets with different percentage of training samples. For each setting, we average the results of 5 runs with different sample selections.

jointly train the model can effectively improve the overall end-to-end writing performance.

Generation with Sub-tasks. We also generate outputs with separate sub-tasks (w/ SepGen. in Table 2) rather than end-to-end generation during the inference time. Given an input, we first use text planning task to produce a plan, and then adopt the surface realization task and review task to generate and revise the final outputs², similar to CONTENTPLAN. With MTL, our model variant significantly outperforms CONTENTPLAN, which indicates the effectiveness of our MTL approach. Notably, separated generation leads to performance drops on Reddit and NYT compared to end-to-end generation. A possible reason is multi-step generation

²We use the distinguishing task to decide whether revising the output or not.

Statement: CMV: No one over the age of 80 should be allowed to serve in government.

T5: The problem is that the idea of a strict age limit for someone to serve is arbitrary. A competency exam is retesting on drivers licenses. It makes all sides feel like there is some kind of merit system. Having a 264-day old person in office is not good. The idea of making something so arbitrary that the best way to measure someone’s competence is to have a competency exam. Derivising a competency exam in a political setting makes me happy.

MOCHA: The problem with the idea of a strict age limit is that it makes all sides look arbitrary. It’s the idea that competency exams are a way to measure someone’s competence, not some kind of merit system. It’s like a 264-day old person devising a competency exam. I’m not sure I would be happy about that in a political setting. The idea that we need a competency exam to get to office makes me feel more competent. The idea of retesting on drivers licenses is pretty good, but that is not what’s really the problem.

Figure 3: Sample outputs on counter-argument generation from Reddit/ChangeMyView test set.

Task	Gram.	Coh.	Rich.	Over.
Reddit/CMV	70.0%	71.7%	66.7%	76.7%
NYT	68.3%	73.4%	55.0%	65.0%
Wikiplots	86.7%	86.7%	63.3%	86.7%

Table 3: Human evaluations. We report (averaged) % of times our model are considered better than T5 on grammaticality (Gram.), coherence (Coh.), content richness (Rich.) and overall quality (Over.). All Krippendorff’s $\alpha \geq 0.31$.

may bring cascading errors (Castro Ferreira et al., 2019), and we leave this to future work.

Few-shot Evaluation. We also conduct low-resource experiments to verify the effectiveness of our multitask learning approach. In particular, we vary the percentage of training data from 1% to 30%. To reduce variance, we repeat each experiment 5 times with different sample selections. As shown in Figure 2, compared with T5 without multitask training, our model consistently yields better results on all tasks. This verifies that our approach can learn general writing skills and be better adapted to new tasks with fewer samples.

4.3 Human Evaluation

We hire two proficient English speakers as human annotators to evaluate model outputs. We randomly choose 30 sample inputs from test set per task. For each input, we anonymously present the outputs of T5 and our model, and ask each annotator to choose the better output based on: (1) grammaticality; (2) coherence; (3) content richness and (4) overall quality. The final results are shown in Table 3. As we can see, human judges consider our model outputs better than T5 on all aspects. In particular, over 70% times our model results are regarded more coherent, which confirms that our multitask learning approach can effectively improve the output co-

herence. Moreover, our multitask training is more effective on Wikiplots where the outputs are longer, indicating its effectiveness for long-form story generation. Among all aspects, the smallest gap is observed on content richness, which suggests the future directions of designing more specific tasks to improve output diversity.

In Figure 3, we present a sample output on argument generation from Reddit dataset. Compared with vanilla T5 without multi-task training, our model is able to produce more coherent and relevant outputs with correct stance (a counter-argument aims to refute the statement). In contrast, the sentences in T5 output are less cohesive, and some sentences suffer from incorrect stance (e.g., “*Having a 264-day old person in office is not good*”). Additional sample outputs on other tasks are presented in Appendix B.

5 Conclusion

In this work, we present a multitask training approach driven by cognitive theory of writing to improve the model ability on coherent long text generation. We introduce decomposed generations tasks and reviewing tasks with different prompts to tackle essential subskills needed for generating coherent outputs. Our model achieves better results on three long text generation tasks under both full training and few-shot settings, and can generate better and more coherent outputs.

Acknowledgements

We thank the anonymous reviewers for their valuable suggestions. Hou Pong Chan was supported by the Science and Technology Development Fund, Macau SAR (Grant No. 0101/2019/A2), and the

Multi-year Research Grant from the University of Macau (Grant No. MYRG2020-00054-FST).

Limitations

Training neural language models to generate coherent long-form outputs is an important task. In this work, we improve model writing ability with multitask training based on the cognitive theory. Nevertheless, we believe there is still huge space to explore in the future. First, in this work we apply our multitask training on each downstream tasks, while one could apply our approach in the pretraining stage to obtain a general writing model. Our proposed decomposed generation and reviewing tasks do not require additional labeled data, and the data can be constructed automatically. Thus applying our approach in the pretraining stage could be useful to improve the model ability. Second, in this work we adopt ordered keyphrases as planning plot. However, for different writing tasks, there might be different ways to represent the plot such as using semantic role labels (Fan et al., 2019) or entity chains (Narayan et al., 2021). Future work might incorporate different plot representations on downstream tasks to further boost the model performance.

Ethics Statement

In this work, we study long text generation task. We recognize that our method may generate contents which contain potentially harmful information and malicious languages due to the systematic biases of pre-training with web corpora. Therefore, we urge the users to carefully check the model outputs, examine the ethical influence of the generated contents, and cautiously deploy the system in real applications.

References

- Chenxin An, Jiangtao Feng, Kai Lv, Lingpeng Kong, Xipeng Qiu, and Xuanjing Huang. 2022. Cont: Contrastive neural text generation. *arXiv preprint arXiv:2205.14690*.
- Vamsi Aribandi, Yi Tay, Tal Schuster, Jinfeng Rao, Huaixiu Steven Zheng, Sanket Vaibhav Mehta, Honglei Zhuang, Vinh Q Tran, Dara Bahri, Jianmo Ni, et al. 2021. Ext5: Towards extreme multitask scaling for transfer learning. *arXiv preprint arXiv:2111.10952*.
- Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.
- Bertram C Bruce. 1978. A cognitive science approach to writing. *Center for the Study of Reading Technical Report; no. 089*.
- Giuseppe Carenini and Johanna D. Moore. 2006. Generating and evaluating evaluative arguments. *Artificial Intelligence*, 170(11):925–952.
- Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. 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 552–562, Hong Kong, China. Association for Computational Linguistics.
- Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, Maryland, USA. Association for Computational Linguistics.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2650–2660, Florence, Italy. Association for Computational Linguistics.
- Linda Flower and John R Hayes. 1981. A cognitive process theory of writing. *College composition and communication*, 32(4):365–387.
- Seraphina Goldfarb-Tarrant, Tuhin Chakrabarty, Ralph Weischedel, and Nanyun Peng. 2020. Content planning for neural story generation with aristotelian rescoring. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4319–4338, Online. Association for Computational Linguistics.
- Tianxing He and James Glass. 2020. Negative training for neural dialogue response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2044–2058, Online. Association for Computational Linguistics.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.
- Zhe Hu, Hou Pong Chan, Jiachen Liu, Xinyan Xiao, Hua Wu, and Lifu Huang. 2022. PLANET: Dynamic content planning in autoregressive transformers for long-form text generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*,

- pages 2288–2305, Dublin, Ireland. Association for Computational Linguistics.
- Xinyu Hua, Zhe Hu, and Lu Wang. 2019. [Argument generation with retrieval, planning, and realization](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2661–2672, Florence, Italy. Association for Computational Linguistics.
- Xinyu Hua, Ashwin Sreevatsa, and Lu Wang. 2021. [DYPLOC: Dynamic planning of content using mixed language models for text generation](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6408–6423, Online. Association for Computational Linguistics.
- Xinyu Hua and Lu Wang. 2020. [PAIR: Planning and iterative refinement in pre-trained transformers for long text generation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 781–793, Online. Association for Computational Linguistics.
- Haozhe Ji and Minlie Huang. 2021. [DiscoDVT: Generating long text with discourse-aware discrete variational transformer](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4208–4224, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Dongyeop Kang and Eduard Hovy. 2020. [Plan ahead: Self-supervised text planning for paragraph completion task](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6533–6543, Online. Association for Computational Linguistics.
- Seanie Lee, Dong Bok Lee, and Sung Ju Hwang. 2020. Contrastive learning with adversarial perturbations for conditional text generation. *arXiv preprint arXiv:2012.07280*.
- Margaret Li, Stephen Roller, Ilia Kulikov, Sean Welleck, Y-Lan Boureau, Kyunghyun Cho, and Jason Weston. 2020. [Don’t say that! making inconsistent dialogue unlikely with unlikelihood training](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4715–4728, Online. Association for Computational Linguistics.
- Yu Li, Baolin Peng, Yelong Shen, Yi Mao, Lars Liden, Zhou Yu, and Jianfeng Gao. 2021. Knowledge-grounded dialogue generation with a unified knowledge representation. *arXiv preprint arXiv:2112.07924*.
- 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.
- Chin-Yew Lin and Eduard Hovy. 2000. [The automated acquisition of topic signatures for text summarization](#). In *COLING 2000 Volume 1: The 18th International Conference on Computational Linguistics*.
- Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
- Shashi Narayan, Yao Zhao, Joshua Maynez, Gonalo Simões, Vitaly Nikolaev, and Ryan McDonald. 2021. [Planning with learned entity prompts for abstractive summarization](#). *Transactions of the Association for Computational Linguistics*, 9:1475–1492.
- 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.
- Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.
- Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. 2020. [PlotMachines: Outline-conditioned generation with dynamic plot state tracking](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4274–4295, Online. Association for Computational Linguistics.
- Ehud Reiter and Robert Dale. 1997. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):57–87.
- Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.
- 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*.
- Abigail See, Aneesh Pappu, Rohun Saxena, Akhila Yerukola, and Christopher D. Manning. 2019. [Do massively pretrained language models make better storytellers?](#) In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 843–861, Hong Kong, China. Association for Computational Linguistics.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. 2022. A contrastive framework for neural text generation. *arXiv preprint arXiv:2202.06417*.

Yixuan Su, Lei Shu, Elman Mansimov, Arshit Gupta, Deng Cai, Yi-An Lai, and Yi Zhang. 2021. Multi-task pre-training for plug-and-play task-oriented dialogue system. *arXiv preprint arXiv:2109.14739*.

Qingyun Wang, Zhihao Zhou, Lifu Huang, Spencer Whitehead, Boliang Zhang, Heng Ji, and Kevin Knight. 2018. [Paper abstract writing through editing mechanism](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 260–265, Melbourne, Australia. Association for Computational Linguistics.

Rose E Wang, Esin Durmus, Noah Goodman, and Tatsunori Hashimoto. 2022. Language modeling via stochastic processes. *arXiv preprint arXiv:2203.11370*.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2020. [Neural text generation with unlikelihood training](#). In *International Conference on Learning Representations*.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I Wang, et al. 2022. Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. *arXiv preprint arXiv:2201.05966*.

Wangchunshu Zhou, Dong-Ho Lee, Ravi Kiran Selvam, Seyeon Lee, Bill Yuchen Lin, and Xiang Ren. 2020. Pre-training text-to-text transformers for concept-centric common sense. *arXiv preprint arXiv:2011.07956*.

A Experimental Details

A.1 Datasets

For data preprocessing, we keep the original texts and do not lowercase the tokens. For topical keyphrases, we extract noun and verb phrases which contain at least one topic signature words (Lin and Hovy, 2000) from the targets.

Argument Generation. The first task requires the model to generate a counter-argument given a statement on a controversial topic. We adopt the Reddit/ChangeMyView dataset processed by Hua and

Wang (2020). The data are collected from Reddit ChangeMyView subcommunity, where the original post title are considered as the input, and the replies are considered as the target counter-arguments. The noun and verb phrases which contain at least one topic signature words (Lin and Hovy, 2000) are extracted from the targets to serve as the topical keyphrases (Hua et al., 2019).

Story Generation. For story generation, we apply Wikiplots dataset³, which consists of story plots of different genres such as TV shows, movies, and books, scraped from Wikipedia. We use the processed dataset from Ji and Huang (2021). We use the same way as in Argument Generation to extract the outline keyphrases.

News Article Writing. For news article writing, we consider the articles from New York Times dataset (Sandhaus, 2008). We apply the processed data by Hua et al. (2021). In their original dataset, entities and concepts extracted from external knowledge base are considered as additional inputs. In our setup, we ignore the knowledge items and instead extract keyphrases as outlines the same as in Argument Generation. The data splits are the same as the original paper.

A.2 Text Plan Construction

For the planning task, we represent the output plot with ordered keyphrase chain. Specifically, we consider all input topical keyphrases, as described in Section 3.1. Then we concatenate all keyphrases with the same order they appear in the target as text plan. For the i -th story sentence, the keyphrase chain is: “ $k_{i1}; k_{i2}; \dots; k_{im}$ ”, where k_{im} is the m -th keyphrase appeared in the i -th sentence. We then concatenate keyphrase chains of all sentences with a special token <sep> as the ordered keyphrase chain. For example: “ $k_{11}; k_{12}; \dots <sep> k_{21}; k_{22} \dots <sep> \dots$ ”.

A.3 Training Details

We use T5-based (Raffel et al., 2019) in all experiments. During training, we set the maximum length of both input and output as 512. We implement all experiments using the Huggingface Transformers (Wolf et al., 2020) and Pytorch (Paszke et al., 2019). The maximum training epoch is set as 12 for argument generation and 18 for story generation and article writing. We optimize our model with AdamW (Loshchilov and Hutter, 2017). The

³<https://github.com/markriedl/WikiPlots>

batch size is 8, and the learning rate is $5e-5$. For our multi-task training, we first construct augmented samples for all sub-tasks using the method as described previously. The original training instances and the augmented samples are then mixed together as the new training set to train the model.

For decoding, we apply nucleus sampling (Holtzman et al., 2019) with k as 10 and p as 0.9. The maximum of generation steps are 200 for argument generation, 512 for story generation and 350 for NYT article generation. We use NVIDIA V100 GPUs for all experiments, and the best model checkpoint is chosen based on the validation loss. It takes roughly 6 hours to converge for argument generation, 20 hours for article generation, and 24 hours for story generation with 8 GPUs. Our model size is the same as vanilla T5 base version.

A.4 Baselines and Comparisons

For comparison, we first consider strong generation models including GPT2 (Brown et al., 2020) and T5 (Raffel et al., 2019) without multitask training. We also include planning-based methods: (1) CONTENTPLAN is a two-step generation model (Goldfarb-Tarrant et al., 2020; Hua and Wang, 2020), where a planner first produces ordered keyphrase plans, and a generator consumes the plans and generates final outputs; (2) BOWPLAN (Kang and Hovy, 2020) predicts keywords as a global plan to guide the generation. All models are implemented with T5-base except for GPT2.

BOWPLAN. We construct the bag-of-words (BOW) label with content words of the target. Following Kang and Hovy (2020), the BOW planning distribution is incorporated at each decoder time step with a gated probability to compute the final output.

CONTENTPLAN. This is a two-step generation method, where a planning model first produce the ordered keyphrase plans given an input, and then a generation model produces the final surface form output given the title and content plans. Both the planner and generator are initialized with T5-base. During training, we use gold plans to train the generator, and use the predicted plans during inference. For decoding method, we apply nucleus sampling for both the planner and the generator.

A.5 Evaluation on Distinguishing Task

We compute accuracy of the Distinguishing Task on each dev set to analyze whether our model es-

Task	Accuracy
Reddit/ChangeMyView	83.4%
NYT	98.3%
Wikiplots	94.6%

Table 4: Accuracy of the distinguishing task on each dataset.

entially learns the corresponding sub-skill. The results are shown in Table 4. All accuracy scores are above 80%, which proves that our model is able to accurately distinguish between positive and negative targets. Notability, the result on Reddit is lower than results on NYT and Wikiplots, which is consistent with the conclusion from previous work (Hu et al., 2022), as Reddit data are collected from social network and usually contain more informal expressions and noises.

A.6 Details for Human Evaluation

For human evaluation, we hire two proficient English speakers as human annotators. For each task, we randomly select 30 samples from the test set, and present the outputs of T5 and our MTL method, with the model names anonymized to reduce bias. We ask each human annotator to select the better outputs based on the following aspects: (1) **Grammaticality** to measure correct language usage; (2) **Coherence**, measuring whether the text has proper high-level plot and is cohesive; (3) **Content Richness** to measure the diversity and informativeness of outputs; and (4) **Overall Quality**.

B Sample Outputs

We present additional sample outputs on news article writing and story generation from Figure 4 to Figure 5.

Title: Unhappy Meals

HUMAN: Michael Pollan contends that lawsuits filed by the obese against fast-food chains "seem absurd on their face," because "no one's forcing people to eat this stuff." Literally, this is true. But more and more often these days, when we are obliged to eat away from home, the only alternative to fast food is hunger. As Critser reports in his book, the nutritious public school cafeteria lunch has become largely budget-cut out of existence, and the fast-food chains have taken up the slack. Highway food is now virtually synonymous with junk food. Airport lounge eateries dispense enough grease to lubricate the Concorde. And let's not even mention malls, where you could purchase the Maltese Falcon more easily than you could buy an apple not engulfed by caramel. When I was a kid in the early 1960's, our town's only McDonald's prominently displayed a big sign indicating how many hamburgers had been served to date, nationwide, under the Golden Arches. When "8 million" was at long last updated to "10 million," we were awed. My inexpert estimate would be that this figure is nowadays surpassed in a single weekend.

T5: It was the nutritious public school cafeteria lunch that got me so excited I sat on the subway for the second straight night, and I was engulfed by caramel on a hot summer day that I could purchase the Maltese Falcon. But I was astonished by how many hamburgers I could eat at airport lounge eateries. I had the slack for my lunch at McDonald's and the McDonald's sandwich. And I'm sure I have surpassed in a single weekend the number of complaints filed by the obese and the overweight. But I was surprised at how many of the fast-food chains are not dispense enough grease to lubricate the Concorde and others, not to mention malls and airport lounge eateries. I've been eating this stuff for years, and it has never been synonymous with highway food.

MOCHA: I am awed by the number of cases of junk food filed by the obese, and awed that the nutritious public school cafeteria lunch is being surpassed in a single weekend. I can't get enough of it. The same could be said of the fast-food chains in New York City: airport lounge eateries engulfed by caramel and the slack of the Golden Arches, which lubricate the Concorde and eat this stuff for lunch. Not to mention malls and airports that dispense enough grease, but the fact is that many hamburgers aren't junk food.

Figure 4: Sample outputs on NYT.

Title: Ultimates

HUMAN: General Nick Fury of HIELD. establishes a strike force of government-sponsored metahumans which includes Captain America; scientist couple Henry and Janet Pym (Giant-Man and the Wasp); Bruce Banner (the Hulk) and Tony Stark (Iron Man). Together they are based at the HIELD facility, the Triskelion. When Banner injects himself with the super-soldier serum and goes on a bloody rampage as the Hulk, he is eventually stopped by the other metahumans with the aid of Thor. The team then join forces with the mutants Quicksilver and Scarlet Witch and agents Hawkeye and Black Widow against the alien shape-shifters the Chitauri, who are defeated. A year later public opinion has turned against the team when it is discovered that Bruce Banner is in fact the Hulk and was responsible for hundreds of deaths. The team is undermined further when Thor is accused of being an escaped mental patient and is incarcerated. This is the doing of his brother Loki, who also facilitates the creation of a new team of anti-American multi-nationals called the Liberators. With the aid of the Black Widow – who betrays the team to the Liberators – the Ultimates are captured, but eventually escape and battle the Liberators to the death (...)

T5: Following the events of the first game, the Ultimates defeat a new team of asgardian warriors called the Liberators. The team is led by the 'Booty-Horse', the Black Widow of the Universe, and the mutants Quicksilver, The Hulk and the Black Widow. The Liberators are led by a scout called the 'Booty-Horse' and the Black Widow. The Ultimates also battle the Liberators, who are based at the HIELD facility. The Ultimates' leader, Tony Stark, betrays the team's leader, Tony Stark, who was turned against the team, with the aid of the Scarlet Witch, the Wasp and the Triskelion. Tony Stark is unable to distract from the robot's presence and the wasp becomes the target of the liberators, but Tony manages to defeat him. Tony and Tony's brother Loki are killed when the Triskelion switches his mind with a weapon that facilitates the creation of a new race called the Liberators. The Avengers are defeated by a gang of Asgardian warriors led by the mysterious dr Griswold, who uses the Ultimates to destroy the mutants. Tony is then attacked by a band of Ultron robots drugs and kills him. Tony is later stopped by the other Metahumans. Tony and Tony are then taken to the Universals Mansion. Tony, Tony, and the Black Widow are reunited, with Tony's help, but Tony is unable to bring them back, as Tony is now being beaten up. Tony, Tony, and Tony are taken to the Universals.

MOCHA: The Ultimates defeat the forces of the Scarlet Witch and the Triskelion, and the Ultimates are defeated in a final battle with the Wolverine. In the meantime, a new team called the Liberators, based at the HIELD facility, have taken the throne. However, a series of events are going on in the Ultimates mansion, and the Ultimates are stopped by the other metahumans. During the battle, the Ultimates are attacked by the 'Red Hat', a race of government-sponsored metahumans, who were turned against the team due to the team's lack of training, and the Ultimates are defeated by the 'Black Widow'. In an effort to stop the 'Red Hat', General Nick Fury of HIELD. agents Hawkeye and Hank Pym are unable to stop it and are ordered to leave. During their stay, Ultron robots drugs the team, and they retreat to a cave. Upon waking up, they discover that 'the wasp' had turned against the team and they are unable to get back to the original one. The team then encounters the 'Skunks', the mutants Quicksilver and Iron Man, and their mutants The Scarlet Witch and the Wasp. Meanwhile, at the headquarters of 'The Black Widow', Janet Pym, who is the wife of Tony Stark, betrays the team by telling them that her brother Loki is in a coma. The team then goes to 'The Black Widow's' house, where they fight the Liberators. The Liberators have discovered that Bruce Banner is actually an agent of the United States government, and have been trying to find a way to take over HIELD. headquarters. Tony Stark, with the aid of Asgardian warriors and shifters the Chitauri, facilitates the creation of the Triskelion. The Liberators are then defeated by the Black Widow, and Tony's 'Unit' team is defeated.

Figure 5: Sample outputs on Wikiplots.