PopALM: Popularity-Aligned Language Models for Social Media Trendy Response Prediction

Erxin Yu[1], **Jing Li**[1,2]*, **Chunpu Xu**[1]
[1]Department of Computing, The Hong Kong Polytechnic University
[2]Research Centre for Data Science & Artificial Intelligence
{erxin.yu, chun-pu.xu}@connect.polyu.hk, jing-amelia.li@polyu.edu.hk

Abstract

Social media platforms are daily exhibiting millions of events. To preliminarily predict the mainstream public reaction to these events, we study *trendy response prediction* to automatically generate top-liked user replies to social media events. While previous works focus on generating responses without factoring in popularity, we propose **Pop**ularity-**A**ligned **L**anguage **M**odels (**PopALM**) to distinguish responses liked by a larger audience through reinforcement learning. Recognizing the noisy labels from user "likes", we tailor-make curriculum learning in proximal policy optimization (PPO) to help models capture the essential samples for easy-to-hard training. In experiments, we build a large-scale Weibo dataset for trendy response prediction, and its results show that PopALM can help boost the performance of advanced language models.

Keywords: Popularity-Aligned Language Models, Trendy Response Prediction, Curriculum Learning

1. Introduction

Social media is a popular channel for users to voice opinions and share information, making it an asset for studying real-world events on diverse topics and public views of them. It is a valuable resource for analyzing and predicting events' mainstream social responses, benefiting various applications, e.g., early event analysis, public response simulation, and comment generation (Wang et al., 2021a; Sun et al., 2022a). However, the vast volumes of daily-created events are beyond humans' ability to track each. Therefore, we study trendy response prediction to automate the generation of top-liked user responses, which can helpfully train language models to predict the mainstream public reaction before an event happens or in its early stages. Here, response popularity is characterized by how many users "like" it, where *like* is a social media behavior showing an audience's agreement to a response (Gao et al., 2020).

Despite the breakthrough progress in automatic response generation thanks to the advances in large language models (LLMs) (Ouyang et al., 2022b), most previous work focuses on generic human responses without considering the popularity factors in the social contexts. However, compared to generic responses, popular responses are much more closely linked to the events' trajectory (Ding et al., 2020) and better reflect the mainstream voices of the public (Kano et al., 2018).

To illustrate this point, Figure 1 shows a societal event example about "*Volunteer Leaked Exam Questions*" with its description post from Weibo

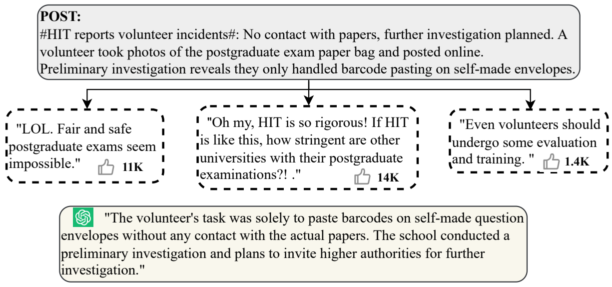


Figure 1: A Weibo post about "*Volunteer Leaked Exam Questions*", followed by its top-3 trendy responses with like numbers. The bottom presents a response sample generated by ChatGPT.

(a Chinese social media platform) and the top-3 trendy responses by audiences' like numbers; we also display ChatGPT's prediction about the possible trendy response for comparison. As can be seen, the real trendy responses can better reflect people's opinions and emotions, e.g., surprise at the leakage of exam papers and doubts about examination fairness. In contrast to these specific points, the output of ChatGPT focuses on a macro level, hence inferior in reflecting essential and concrete public viewpoints.

Given these concerns, we propose Popularity-Aligned Language Models (PopALM) to train language models with popularity via reinforcement learning. To the best of our knowledge, *PopALM exhibits the first effort to align language generation with social media popularity measure.* We adopt like numbers to train the reward function and employ a PPO method to optimize the training process. However, like numbers, although

*Corresponding author

LREC-COLING 2024, pages 12867–12878

as easy-to-access popularity indicators, are noisy user-generated labels, which may be affected by many factors beyond text, such as posting time, authors, etc. These noisy labels may thus exhibit implicit relations to the text features, substantially challenging the training of reward functions.

To address this challenge, PopALM engages curriculum learning (Bengio et al., 2009) into PPO to filter out the noisy training samples and differentiate the samples' learning difficulty for optimizing the learning pace from easy to hard. First, the reward function leverages task-specific supervision to align with trendy response prediction. Then, we rank the samples based on the reward prediction confidence to remove noisy samples, i.e., samples with low confidence. Lastly, we employ the self-paced learning strategy for the remaining samples to progressively learn from easy to hard samples, thus improving the overall learning efficiency.

As a pilot study on trendy response prediction, we should benchmark the task with the first dataset. To that end, we collect around 30K daily-trending events from Weibo, each with the most popular post as its description. To explore trendy responses for each post, we also gather its user replies associated with the like numbers for popularity learning. The main comparison results in experiments demonstrate that PopALM helps advanced language generation models improve trendy response prediction quality in both automatic and human evaluation. Then, ablation studies indicate the positive contributions of curriculum learning strategies to PopALM's overall effectiveness. Next, quantitative analysis shows PopALM's superiority with varying training models and data scales. Finally, we demonstrate the enhancing effect of the generated responses on other tasks.

In summary, our contributions are three-fold:

• We present the first study on trendy response prediction for social media events and build the first large-scale benchmark for this task.

• We propose PopALM, a novel popularity-aligned language model, with a tailor-made curriculum learning for PPO to effectively learn from noisy user-like labels in social media contexts.

• We extensively experiment on PopALM and find that it works well in different LLMs and fine-tuning methods to learn popularity and help predict trendy responses.

2. Related Work

Response Generation. Our task is in line with response generation, an increasingly popular field in NLP. Its early work applied the RNN-based sequence-to-sequence model and achieved promising results (Shang et al., 2015; Xing et al., 2018; Zhang et al., 2019). In recent years, pre-trained large-scale language models have brought many breakthroughs in natural language generation, e.g., the GPT series (Radford and Narasimhan, 2018; Radford et al., 2019; Brown et al., 2020), T5 (Raffel et al., 2020), and BART (Lewis et al., 2020).

Building upon these models, numerous methods have been proposed to enhance response generation capabilities. DialoGPT (Zhang et al., 2020) is tailored for response generation using comments sourced from Reddit. The blender model (Roller et al., 2021) refines the pre-trained model using responses annotated by humans to emphasize desired conversational capabilities, such as engagement, knowledge, empathy, and personality. PLATO (Bao et al., 2020) introduces discrete latent variables to address the inherent one-to-many mapping problem to improve response quality.

In this field, many studies also focused on automatic comment generation in a social media context (Zheng et al., 2018; Qin et al., 2018). Yang et al. (2019) allowed better encoding through selected important contextual spans. Sun et al. (2022b) leveraged the topic model to capture the author's styles for a personalized generation. However, most of these studies focused on generating generic or individual' comments, paying limited attention to trendy response generation with popularity measures, revealing a gap to address.

Language Models Alignment. Taking recent advances in large language models (LLMs), many studies examined how to align language models with human feedback (Stiennon et al., 2020; Ouyang et al., 2022a). For example, ChatGPT, a closely related model to InstructGPT (Ouyang et al., 2022b), is specifically trained to follow human instructions and has demonstrated state-of-the-art performance in conversational abilities. ChatGLM is a bilingual language model that aligns the General Language Model (Du et al., 2022) with large-scale human instructions, achieving superior performance in the Chinese response generation. PopALM can be based on various language models and explores trendy responses with popularity alignment, which has not been explored previously.

Popularity Prediction. Our work is related to popularity prediction on social media, where users express their preferences by voting, sharing, or bookmarking a post. The count of such actions is usually adopted as the popularity indicator. Lamprinidis et al. (2018) used a multi-task GRU network to predict headline popularity. Kano et al. (2018) employed such popularity measure to distantly supervise extractive summarization. Gao et al. (2020) leveraged social media feedback data (number of replies and upvotes) to build a large-scale dataset

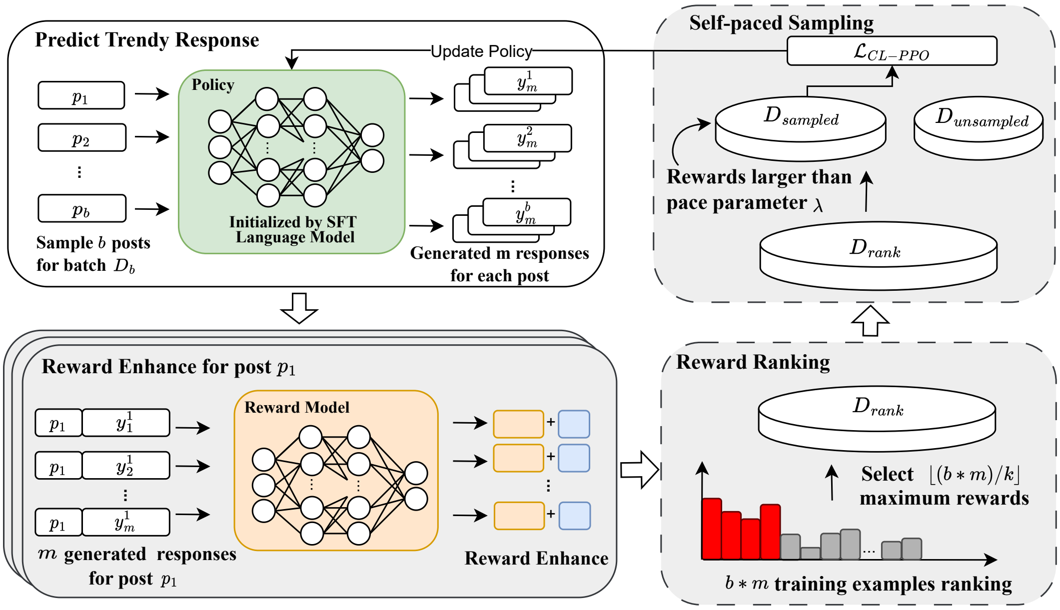


Figure 2: The workflow of PopALM is based on curriculum learning enhanced PPO, which exploits three novel strategies to leverage noisy user-like labels as popularity indicators. These strategies are Reward Enhancement (left bottom; for task-specific supervision), reward ranking (right bottom; for filtering noisy training samples), and self-paced reward sampling (right top; for training from easy to hard).

to classify user feedback. However, none of them injects the popularity factors into language generation, which we will extensively explore.

3. Popularity-Aligned Language Models

PopALM Overview. To begin with, we state the problem of trendy response prediction as follows: given post p, the model needs to generate trendy responses $Y = \{y_1, y_2, ..., y_m\}$, in which y_i is one of the popular responses. As shown in Figure 2 (the workflow to build PopALM), following InstructGPT (Ouyang et al., 2022b), our framework consists of three parts: supervised fine-tuning, reward modeling, and reinforcement learning (RL). Our RL algorithm is based on PPO, and we further introduce curriculum-learning engaged PPO (CL-PPO) to alleviate the noisy labels challenge in the popularity learning of social media. We describe our RL-based backbone framework in Section 3.1, followed by our CL-PPO algorithm in Section 3.2.

3.1. Aligning LMs with Popularity via RL

Supervised Fine-tuning. We first fine-tune language models (LMs) to predict trendy responses using supervised learning. In this stage, we only consider the one-to-one mapping relation between one post and a trendy response. Given one post p and its trendy responses Y, we pair p with each response in Y, forming our supervised training samples $\{(p, y_1), (p, y_2)..., (p, y_m)\}$. Here, the training object for one post is to minimize the following negative log-likelihood (NLL) loss:

$$\mathcal{L}_{SFT} = -E_{(p,y_i)\sim D_{SFT}} \sum_{i=1}^{m} \sum_{t=1}^{T} -\log p(y_i^t | p, y_i^{<t}), \quad (1)$$

where T is the length of the response, D_{SFT} is the dataset for supervised fine-tuning, and y_i is the i-th golden response for p.

Reward Modeling. Then, we design the RL's reward to teach our model how to predict the popularity of our generated responses. Specifically, it takes in a post and response and outputs a scalar reward by comparing between two responses given the same post. The reward difference indicates that one response has more like numbers than the other. The loss function for the reward model is:

$$\mathcal{L}_{RM}(\theta) = -E_{(p,y_w,y_l)\sim D_{RM}} [\log(\sigma(r_\theta(p, y_w) - r_\theta(p, y_l)))], \quad (2)$$

where θ is the training parameters of reward model, $r_\theta(p, y)$ is the scalar output of the reward model for post p and response y, y_w has higher like numbers than y_l, and D_{RM} is the reward modeling dataset.

Reinfocement Learning. Inspired by InstructGPT's practice, we further update the SFT language model using PPO (Schulman et al., 2017) to leverage SFT results into the RL framework. Its loss function can be briefly described as follows:

$$\mathcal{L}_{RL}(\phi) = -E_{p\sim D_{RL}, y\sim \pi_{\phi}^{RL}(p)} r_{\theta}(p, y) \qquad (3)$$

where π_{ϕ}^{RL} is the policy RL aims to optimize, which is initialized by the SFT language model. Post p is sampled from train dataset D_{RL}, y is the output responses of policy given p. For clarity of presentation, we omit the detail of PPO here and refer readers to Schulman et al. (2017).

3.2. Curriculum Learning-Enhanced PPO

We can preliminarily align the language model with popularity through the aforementioned learning. However, unlike InstructGPT with real human feedback, we use like numbers as automatic labels for assessing response popularity, which is noisy and easily influenced by many factors beyond text.

We thereby incorporate curriculum learning into the PPO algorithm (and present CL-PPO). It helps filter out noisy training samples while differentiating among training samples' difficulty levels for better-aligning LMs' popularity learning with noisy labels. CL-PPO has three novel components — reward enhancement (to provide task-specific supervision), reward ranking (to remove noisy training samples), and self-paced reward sampling (to allow easy-to-hard training) as follows.

Reward Enhancement. In reinforcement learning, the rewards not only come from the reward model but also include those directly related to the task (Wu et al., 2023), such as the rewards a robotic vacuum cleaner receives for collecting garbage or the rewards earned from finding the exit in a maze game. Inspired by this, PopALM integrates a reward enhancement mechanism, using the overlap between the output and highly upvoted responses as a task-specific reward signal. The reward for a generated response y given post p is defined as:

$$r_{\theta}^{e}(p, y) = r_{\theta}(p, y) + \alpha \max_{\hat{y}\in\hat{Y}}(Rouge(y, \hat{y})), \qquad (4)$$

where $r_{\theta}^{e}(p, y)$ is the enhanced reward, α is a weight coefficient, and $\hat{Y}$ is the golden trendy responses. $Rouge(y, \hat{y})$ returns the ROUGE-L score between a generated response y and a golden response $\hat{y}$, where the highest ROUGE-L between them is selected to enhance the reward.

Reward Ranking. To mitigate the effects of noisy training samples, we introduce a reward ranking mechanism for PPO to increase the training sample quality. Specifically, consider a batch of posts, denoted as $\{p_1, p_2, .., p_b\}$ (where b represents the batch size); PopALM aims to gain a one-to-many

Algorithm 1 Curriculum Learning-Enhanced PPO

Input: RL Training dataset D_{RL}, policy π_{ϕ}^{RL}, batch size b, reward model r_{θ}, pace parameter μ, acceptance ratio 1/k.

for batch D_b from D_{RL} **do**
  for each $p \in D_b$ **do**
    Predict m trendy responses via top-p sampling, $Y = \{y_1, y_2, ..., y_m\} \sim \pi_{\phi}^{RL}$.
    Compute the reward of each response $\{r_{\theta}(p, y_1), r_{\theta}(p, y_2), ..., r_{\theta}(p, y_m)\}$
    Compute the enhanced reward using Eq.4 $\{r_{\theta}^{e}(p, y_1), r_{\theta}^{e}(p, y_2), ..., r_{\theta}^{e}(p, y_m)\}$
  end for
  Rank reward and select $\lfloor (b * m)/k \rfloor$ training samples with maximum rewards.
  Select the training samples with higher rewards via self-paced sampling.
  Update policy π_{ϕ}^{RL} using Eq.5
  Update the learning pace via $\lambda \leftarrow \lambda - \mu\lambda$
end for

capability to generate multiple trendy responses for each post. To that end, for each post in the batch, we generate m responses using a language model with a top-p sampling method (Basu et al., 2021). Then, we obtain the reward r_{θ}^{e} for each sampled response through the reward model and enhancement mechanism. Finally, based on r_{θ}^{e} (reflecting the reward model's confidence), we rank the collected samples and shortlist the $1/k$ percent of samples with the highest reward to engage in the subsequent training. Samples with low rewards are discarded because they signify low prediction confidence and are considered noisy samples.

Self-paced Sampling. With the shortlisted training samples, we further incorporate the self-paced learning method from curriculum learning to enhance learning efficiency. The intuition is to mimic human knowledge acquisition, starting from simple concepts and gradually tackling more difficult ones requiring advanced skill sets. Here we measure training samples' learning difficulties with their rewards. Examples with higher rewards have higher prediction confidence, making them easier to learn from. We can thus start with the higher-rewarded samples and then move to those with lower rewards. The ultimate learning objective of CL-PPO is defined as follows:

$$\mathcal{L}_{CL-PPO}(\phi) = -E_{(p,y_i)\sim D_{Rank}} \left[r_{\theta}^{e}(p, y_i)v_i - \lambda \sum_{i=1}^{|D_{Rank}|} v_i \right]$$
$$s.t. \quad v_i = \begin{cases} 1 & \text{if } r_{\theta}^{e}(p, y_i) \geqq \lambda, \\ 0 & otherwise, \end{cases} \qquad (5)$$

Here $v_i \in \{0, 1\}$ indicates whether the training sample (p, y_i) is selected, λ acts as a threshold to the sampling process and is updated at every

	SFT	RM	RL
Training	2,5140	9,985	2,514
Development	867	3451	867
Test	1,824	7,249	1,824
Avg. Posts		119.8	
Avg. Responses		25.8	

Table 1: Statistics of SFT, RM, and RL datasets, followed by the average length (token number) of posts and responses from the raw data.

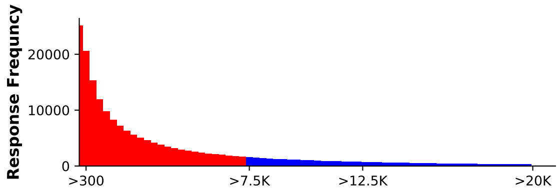


Figure 3: Distribution of response frequency (y-axis) over like numbers (x-axis). Red bars correspond to the top 50% more popular responses and the rest are blue.

training step. In detail, for the reward $r_\theta^e(p, y_i)$ maintained after reward ranking, if it is smaller than the threshold λ, we set v_i zero as shown in Eq.5. In this way, during the initial training, responses with larger rewards (corresponding to more popular responses) predominantly contribute to the learning process. As the training progresses, λ gradually decreases, incorporating lower-rewarded samples to increase the model's generalization capability. Algorithm 1 presents an overview of the entire training process of CL-PPO.

4. Experimental Setup

Dataset. To set up the experiment, we assembled a new dataset from Weibo, a popular Chinese microblog. For data collection, we first obtained the most popular hashtags that have been in use since January 2022, reflecting trending social media events. Then, we gathered the raw posts associated with each hashtag using Weibo's search API [1] and selected the post that garnered the most comments as an event description. Next, for each selected post, we extracted its comments using the platform's comment API. Finally, our dataset comprised approximately 70,000 posts and 24 million comments filtered from the raw datasets. We did not specifically filter out comments posted by popular authors, even though anything they post might receive many likes. The reason is that many popular authors might also be opinion leaders, often leading the mainstream voice on social media. Furthermore, our model has the capability to filter out some noise responses.

Based on the raw data, we gathered three subsets for model training and testing: (1) *SFT* dataset (with popular responses) to fine-tune the language model for trendy response prediction; we selected the top 10 comments for each post as the gold response as the reference. (2) *RM* dataset (with ranked responses) to train our reward model, where the top 3 comments served as the trendy responses, paired with negative samples of less-liked responses. (3) *RL* dataset to train RL's policy to generate responses and provide trendy responses as signals for reward enhancement. Table 1 shows these datasets' statistics. As can be seen, responses are much shorter on average than posts. It shows that audiences tend to voice their viewpoints concisely, whereas posts may contain richer information for event reporting.

To further analyze response popularity, we examine the SFT data and display the frequency distribution over like numbers in Figure 3. It is observed that the majority of responses garnered over 300 likes, meaning that our dataset exhibits sufficient samples for learning trendy responses. Meanwhile, most responses demonstrate like numbers between 300 to 7,500, whereas the very popular ones (e.g., with over 7,500 likes) appear sparsely. This exhibits a long-tail distribution and challenges our learning to predict trendy responses.

Pre-Processing. Following common practice (Lu et al., 2021), we first purged the metadata, e.g., the author's information and emoji labels, while substituting links and user mentions (denoted as @username). Then, we employed the open-source Jieba toolkit for Chinese word segmentation.

Model Setup. Here, we describe how we set our model. Based on the statistics of in Table 1, we capped the post length to 128 and the response prediction length to 32. To generate diverse responses, we adopt top-p sampling in our experiment with the top-p set to 0.7 and the temperature to 0.95. For the SFT phrase, we set the learning rate to 0.002 and batch size to 16 for all models. We use GPT-2 (Radford et al., 2019) as the initial reward model. For CL-PPO, the weight coefficient α is set to 0.5, the acceptance ratio k is set to 3, the threshold θ is initialed as 1, and the learning pace μ is set to 0.2. [2]

Evaluation Metrics. For *Automatic Evaluation*, we follow Zhang et al. (2020) to compare output and gold responses and evaluate the output quality with overlapping-based metrics ROUGE (Lin,

[1] https://open.weibo.com/wiki/C/2/search/statuses/limited

[2] Our code and dataset are available at https://github.com/ErxinYu/PopALM.

Models	Top-1				Top-3				Top-5			
	R-1	R-2	R-L	BU	R-L	BU	MD-1	MD-2	R-L	BU	MD-1	MD-2
Language Models (*w/o SFT*)												
GPT-2	16.31	1.79	11.69	2.71	13.17	3.08	0.292	0.483	14.32	3.57	0.228	0.427
LLaMA	1.06	0.01	0.85	0.17	1.32	0.29	0.134	0.597	1.669	0.31	0.101	0.567
ChatGLM	14.77	2.16	10.88	3.16	11.65	3.45	0.182	0.424	12.19	3.65	0.121	0.320
Language Models *(w/ SFT)*												
DialoGPT	14.22	1.35	11.38	2.11	12.03	2.17	0.143	0.235	12.51	2.27	0.100	0.179
CDial-GPT	17.01	0.79	12.30	1.77	13.13	1.92	0.157	0.223	13.10	1.91	0.068	0.117
GPT-2 (P-T)	18.29	1.79	11.69	2.71	14.05	3.31	0.213	0.252	15.15	3.66	0.158	0.214
LLaMA (P-T)	16.87	1.65	13.31	3.27	16.05	4.14	0.450	0.755	17.55	4.60	0.369	0.703
ChatGLM (LoRA)	18.39	3.11	15.08	5.72	19.50	7.84	0.489	0.590	21.70	8.82	0.382	0.497
ChatGLM (P-T)	18.63	3.29	15.94	6.16	19.69	7.79	0.498	0.576	22.98	9.38	0.431	0.501
Popularity-Aligned Language Models (PopALM)												
ChatGLM (PPO)	18.61	3.09	16.06	6.19	20.01	7.91	0.511	0.583	22.66	9.27	0.437	0.506
PopALM	**19.49**	**3.69**	**16.42**	**6.35**	**21.50**	**8.43**	**0.541**	**0.632**	**23.58**	**9.63**	**0.452**	**0.511**

Table 2: We present the automatic evaluation results for the top-1, top-3, and top-5 trendy responses predicted by PopALM, i.e., **ChatGLM (CL-PPO)**. For the top-1 prediction, we report the performance metrics R-1 (ROUGE-1), R-2 (ROUGE-2), R-L (ROUGE-L), and BU (BLEU). For top-3 and top-5 predictions, we provide R-L and BU to measure the overlap performance and employ MD-1 (M-Distinct-1) and MD-2 (M-Distinct-2) to evaluate the diversity performance. We report the average performance for five different random seeds, and the better results (compared to PPO) are highlighted in bold, indicating a statistically significant difference ($p < 0.05$) from baselines with bootstrap resampling (Koehn, 2004).

2004) and BLEU (Papineni et al., 2002) scores. Besides, we use M-Distinct-n (Li et al., 2016) to score the diversity of responses, which measures the model's ability to generate multiple diverse responses for the same test posts.

For *Human Evaluations*, we invited human raters with NLP backgrounds to rate the generated responses on a 5-point Likert scale on the following dimensions. *Informativeness* reflects how much information is presented in the generated results. *Specification* assesses the degree of the output containing specific viewpoints. *Popularity* measures the potential of the response to be liked by many users and become popular. In addition, we involved an *Overall* score to reflect raters' general feelings by combining the above three dimensions. Here, we randomly select 100 posts from the test set and enlist raters to assess the responses without knowing which model generated them.

Comparison Setup. For the pre-trained models, we adopt several language models that have not been fine-tuned on our dataset: 1) GPT-2 (Radford et al., 2019) is a decoder-based language model for generating contextually relevant and coherent text. 2) DialoGPT (Zhang et al., 2020) is a response generation model based on GPT-2, pre-trained on a large corpus of social media text. 3) CDial-GPT (Wang et al., 2020) is first pre-trained on a Chinese novel dataset and then post-trained on a large-scale Chinese dialog dataset, demonstrating strong response generation capabilities. 4) LLaMA (Touvron et al., 2023) is a foundational large language model designed for researchers. 5) ChatGLM is an open bilingual language model based on the General Language Model (Du et al., 2022).

For DialoGPT and CDial-GPT, we employ full-parameter fine-tuning on our dataset. For other models under SFT and PPO settings, to enable efficient adaptation of pre-trained language models to our task, we employ two Parameter-Efficient Fine-Tuning (PEFT) methods: 1) P-Tuning (P-T) (Liu et al., 2022) tunes continuous prompts with a frozen language model. 2) LoRA (Hu et al., 2022) injects trainable rank decomposition matrices into the Transformer.

5. Experimental Results

This section first discusses the main comparison results in Section 5.1, followed by the ablation study to examine the varying CL-PPO strategies' contributions in Section 5.2. Then, we quantify the effects of language models, PEFT methods, and training data scales in Section 5.3. After that, we qualitatively analyze why PopALM can exhibit superior results through a case study in Section 5.4. Finally, we demonstrate the impact of generated responses in Section 5.5.

5.1. Main Comparison Results

Automatic Evaluation Results. Table 2 shows the result, where we draw the following observations: 1) The previous response generation models, DialoGPT and CDial-GPT, despite being trained on large-scale conversational text, still fall short in predicting popular responses. 2) Compared to the original language models, PEFT allows mod-

Models	Info	Spec	Pop	Overall
ChatGLM	**2.12**	1.70	1.75	1.86
ChatGLM(P-T)	1.65	2.92	2.11	2.23
ChatGLM(PPO)	1.73	2.89	2.26	2.43
PopALM	1.91	**3.14**	**2.89**	**2.65**

Table 3: Human Evaluation on randomly sampled 100 test samples. We compare ChatGLM with P-T/PPO, and PopALM model.

Models	GPT-2		LLaMA		ChatGLM	
	LoRA	P-T	LoRA	P-T	LoRA	P-T
PPO	13.79	14.77	16.21	16.13	19.89	20.01
CL-PPO	**14.98**	**15.56**	**17.54**	**17.23**	**20.77**	**21.50**

Table 4: Result of top-3 prediction ROUGE-L score with varying Language Models (LMs) with PEFT.

els to yield better responses. This suggests that only training a minor fraction of parameters can also equip language models with the capability to predict popularity. 3) Using the PPO method to align the language model with popularity is beneficial. However, some metrics are decreased after the PPO training, possibly due to the negative effects of noisy labels. 4) Our proposed PopALM significantly outperforms the PPO in all automatic metrics. Moreover, in the top-3 and top-5 predictions, the responses produced by CL-PPO exhibit greater diversity. The above results suggest the effectiveness of CL-PPO in mitigating the issue of noisy labels and allowing more efficient learning for trendy response prediction.

Human Evaluation. We select PopALM and three variants of ChatGLM to compare how human readers evaluate their output. The results are shown in Table 3. PopALM gains higher scores in specification and popularity, while its performance falls on the informativeness metric compared to ChatGLM. It may be that PopALM generates more specific responses, thereby losing some general information. Meanwhile, the responses generated by PopALM are more stylized towards social media than the other two fine-tuning methods of ChatGLM. The result shows that through popularity-aligned reinforcement learning, language models yield more specific points to reflect the public's concerns and are more likely to receive likes.

5.2. Ablation Study

The above results show the overall superiority of CL-PPO. To further investigate the effects of its components, we conduct an ablation study with the results displayed in Figure 5. As can be seen, our three proposed components all contribute positively across different language models (ChatGLM, LLaMA, and GPT-2) and PEFT methods (LoRA, P-tuning). In particular, self-paced sampling contributes substantially when ChatGLM is used as the backbone language model. The performance drops by 1.48 and 1.11, respectively, and even falls below PPO's when self-paced sampling is reduced. This illustrates that prioritizing high-reward examples for early learning is beneficial for the models to learn trendy response prediction efficiently.

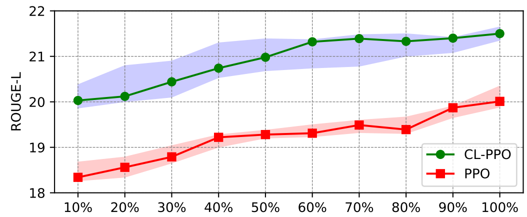


Figure 4: Effects of training data scales (x-axis). The y-axis shows the ROUGE-L score of the top-3 prediction based on ChatGLM. The colored bands indicate ± 1 standard deviation corresponding to different percentages of training data.

5.3. Quantitative Analysis

We then quantify PopALM with varying training setups to deepen the understanding of it.

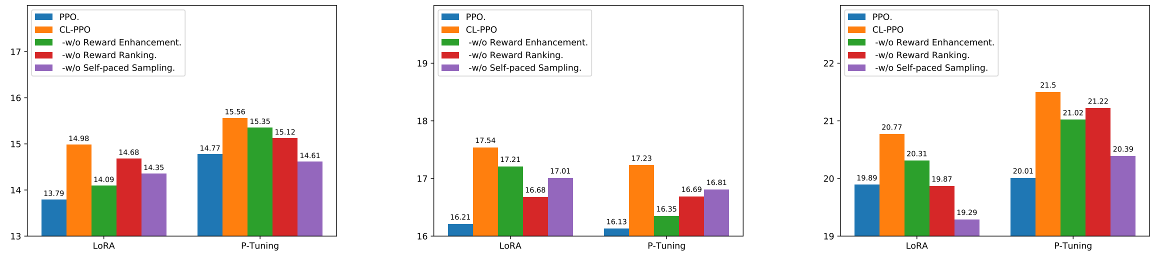
Varying Language Models and PEFT Methods. We first investigate the backbone language models (LMs) and PEFT methods and display the results in Table 4. It shows that CL-PPO exhibits improved performance over the original PPO across different combinations of LMs and PEFT methods. This validates our model as a plug-and-play approach that can be effectively applied to various LMs.

Varying Training Data Scales. We then test PopALM's sensitivity to training data scales by training it with different data percentages. As shown in Figure 5, our proposed CL-PPO training algorithm consistently outperforms the original PPO regardless of the volume of training data used, ranging from 10% to 100%. This suggests the stable and consistent performance of CL-PPO across different training data amounts.

5.4. Case Study

We exemplify the case in Figure 1 and compare different models' output in Table 5 to qualitatively analyze why PopALM can yield better results. Recall that the post describes an event in which a volunteer leaked exam questions and posted them online, and the school explained that the volunteer did not have full access to the exam paper.

The output of ChatGPT (see Figure 1) and untrained ChatGLM are more akin to summarizing



(a) GPT-2 (b) LLaMA (c) ChatGLM

Figure 5: Ablation study on CL-PPO. We report the ROUGE-L scores of the Top-3 trendy response predictions for GPT-2, LLaMA, and ChatGLM. For them each, we show PEFT results of LoRA on the left and P-Tuning on the right. For each barplot, the bars from left to right show PPO, CL-PPO, followed by the CL-PPO ablations w/o Reward Enhancement, w/o Reward Ranking, and w/o Self-paced Sampling.

ChatGLM: Volunteer pasted barcodes on exam bag and took photos, ruling out contact with papers. Public concern may arise due to planned higher authority investigations.
ChatGLM (P-T): Don't even bring up volunteers anymore.
ChatGLM (PPO): Volunteers gonna volunteer, candidates gonna candidate, it's just pointless.
PopALM : Volunteers need training. Fair exams should be a guarantee!

Table 5: Case study of different models' output for the post in Figure 1.

the post, tending to be more generic. After fine-tuning, ChatGLM (P-T) can generate responses in a social media style, incorporating its own opinion. After PPO training, the model chose to train on responses with higher scores in the reward model. However, due to the noisy labels, the scores given by the reward model may not be entirely accurate, misleading the generation results, as shown in the table. In contrast, CL-PPO allows more effective training with noisy labels and consequently better aligns the output with trendy responses.

5.5. Impact of Generated Response

In Table 6, we demonstrate the impact of the responses generated by our model on two tasks: poll question generation (Lu et al., 2021) and social emotion prediction (Ding et al., 2020). Poll question generation aims to automatically generate questions for posts, in which popular responses can reflect the public's concerns and engage them in discussions. Social emotion prediction involves predicting the public's attitude towards posts. Including mainstream reactions can help assess the general attitude.

We tested the poll question generation task based on ChatGLM and set up a comparative experiment: one approach is to input only the post to generate a poll question, while the other concatenates the post and responses as input. We

Methods	PQG		SEP	
	$R-1$	$R-L$	$F1_{macro}$	$F1_{micro}$
W/O Responses	0.331	0.305	0.312	0.408
W/ ChatGLM Responses	0.323	0.314	0.303	0.401
W/ PopALM Responses	0.363	**0.337**	0.322	0.422
W/ Real Responses	**0.367**	0.331	**0.325**	**0.426**

Table 6: Performance of the different responses on the Poll Question Generation (PQG) and Social Emotion Prediction (SEP) tasks. We use Rouge-1 and Rouge-L to evaluate PQG, and macro F1 and micro F1 to assess SEP.

employed RoBERTa (Liu et al., 2019) as the classifier for the social emotion prediction task and adopted the same comparative experiment. As can be seen from the table, incorporating PopALM-generated responses yields better results for both tasks. However, using responses directly generated by ChatGLM doesn't have much effect. Moreover, the results indicate that the PopALM-generated responses could perform comparable to real responses.

We study trendy response prediction to predict the mainstream public reaction before an event happens or in its early stages. Beyond the above response-augmented tasks, it also offers other potential applications. For example, it can be applied in early event analysis to foresee the future impact of a breaking event before many people engage in related discussions. Social scientists can also employ our model to simulate the public responses to some social events even though they have not yet happened. Moreover, our study can potentially benefit general comment generation applications (Zheng et al., 2018; Wang et al., 2021b; Sun et al., 2022b) and encourage better user engagement.

6. Conclusion

We have presented a study on trendy response prediction for social media events, an area that previously lacked exploration. A novel popularity-aligned language model was proposed by integrating a specifically designed curriculum learning strategy into proximal policy optimization to learn popularity from noisy user-like labels. A large-scale benchmark was constructed, and its experimental results show that PopALM exhibits performance gains to LMs with various training setups.

Ethics Statement

Our paper presents a large-scale Weibo corpus for trendy response prediction, and it will not pose ethical problems. Firstly, these posts and responses are open to the public. Secondly, Weibo allows any user to report suspicious cases that may involve ethical issues, and the reported content will be immediately removed. Finally, the data was gathered via standard data acquisition procedures regulated by Weibo's API and was downloaded exclusively for academic research purposes.

For our experiments, we have taken steps to anonymize the data to protect privacy, such as removing author names and changing mentions and URL links to generic tags.

Acknowledgements

This work is supported by the NSFC Young Scientists Fund (Project No. 62006203), a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU/25200821), the Innovation and Technology Fund (Project No. PRP/047/22FX), and PolyU Internal Fund from RC-DSAI (Project No. 1-CE1E).

7. Bibliographical References

Siqi Bao, Huang He, Fan Wang, Hua Wu, and Haifeng Wang. 2020. PLATO: Pre-trained dialogue generation model with discrete latent variable. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 85–96, Online. Association for Computational Linguistics.

Sourya Basu, Govardana Sachitanandam Ramachandran, Nitish Shirish Keskar, and Lav R. Varshney. 2021. Mirostat: a neural text decoding algorithm that directly controls perplexity. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 41–48, New York, NY, USA. Association for Computing Machinery.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Keyang Ding, Jing Li, and Yuji Zhang. 2020. Hashtags, emotions, and comments: A large-scale dataset to understand fine-grained social emotions to online topics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1376–1382, Online. Association for Computational Linguistics.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. GLM: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335, Dublin, Ireland. Association for Computational Linguistics.

Xiang Gao, Yizhe Zhang, Michel Galley, Chris Brockett, and Bill Dolan. 2020. Dialogue response ranking training with large-scale human feedback data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 386–395, Online. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Ryuji Kano, Yasuhide Miura, Motoki Taniguchi, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. 2018. Harnessing popularity in social media for extractive summarization of online conversations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1139–1145, Brussels, Belgium. Association for Computational Linguistics.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Sotiris Lamprinidis, Daniel Hardt, and Dirk Hovy. 2018. Predicting news headline popularity with syntactic and semantic knowledge using multi-task learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 659–664, Brussels, Belgium. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 994–1003, Berlin, Germany. Association for Computational Linguistics.

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.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68, Dublin, Ireland. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pre-training approach. *CoRR*, abs/1907.11692.

Zexin Lu, Keyang Ding, Yuji Zhang, Jing Li, Baolin Peng, and Lemao Liu. 2021. Engage the public: Poll question generation for social media posts. 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 29–40, Online. Association for Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022a. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022b. Training language models to follow instructions with human feedback. In *NeurIPS*.

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.

Lianhui Qin, Lemao Liu, Wei Bi, Yan Wang, Xiaojiang Liu, Zhiting Hu, Hai Zhao, and Shuming Shi. 2018. Automatic article commenting: the task and dataset. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 151–156, Melbourne, Australia. Association for Computational Linguistics.

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

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, and Jason

Weston. 2021. Recipes for building an open-domain chatbot. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 300–325, Online. Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1577–1586, Beijing, China. Association for Computational Linguistics.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. 2020. Learning to summarize from human feedback. *CoRR*, abs/2009.01325.

Teng Sun, Chun Wang, Xuemeng Song, Fuli Feng, and Liqiang Nie. 2022a. Response generation by jointly modeling personalized linguistic styles and emotions. *ACM Trans. Multim. Comput. Commun. Appl.*, 18(2):52:1–52:20.

Teng Sun, Chun Wang, Xuemeng Song, Fuli Feng, and Liqiang Nie. 2022b. Response generation by jointly modeling personalized linguistic styles and emotions. *ACM Trans. Multim. Comput. Commun. Appl.*, 18(2):52:1–52:20.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Wei Wang, Piji Li, and Hai-Tao Zheng. 2021a. Generating diversified comments via reader-aware topic modeling and saliency detection. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13988–13996. AAAI Press.

Wei Wang, Piji Li, and Hai-Tao Zheng. 2021b. Generating diversified comments via reader-aware topic modeling and saliency detection. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13988–13996. AAAI Press.

Yida Wang, Pei Ke, Yinhe Zheng, Kaili Huang, Yong Jiang, Xiaoyan Zhu, and Minlie Huang. 2020. A large-scale chinese short-text conversation dataset. In *NLPCC*.

Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas A. Funkhouser. 2023. Tidybot: Personalized robot assistance with large language models. *CoRR*, abs/2305.05658.

Chen Xing, Yu Wu, Wei Wu, Yalou Huang, and Ming Zhou. 2018. Hierarchical recurrent attention network for response generation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5610–5617. AAAI Press.

Ze Yang, Can Xu, Wei Wu, and Zhoujun Li. 2019. Read, attend and comment: A deep architecture for automatic news comment generation. 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 5077–5089, Hong Kong, China. Association for Computational Linguistics.

Hainan Zhang, Yanyan Lan, Liang Pang, Jiafeng Guo, and Xueqi Cheng. 2019. ReCoSa: Detecting the relevant contexts with self-attention for multi-turn dialogue generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3721–3730, Florence, Italy. Association for Computational Linguistics.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.

Hai-Tao Zheng, Wei Wang, Wang Chen, and Arun Kumar Sangaiah. 2018. Automatic generation of news comments based on gated attention neural networks. *IEEE Access*, 6:702–710.