NEWS2DIAL: News to Dialogue Utterance

Stanford CS224N Custom Project, **Mentor: Siyan Li**

Pratyush Agarwal
Computer Science
Stanford University
prat1019@stanford.edu

Mohammad Ali Rehan
Computer Science
Stanford University
alirehan@stanford.edu

Rishi Agarwal
Computer Science
Stanford University
rishia@stanford.edu

Abstract

In today's era of voice assistants, it is essential that these assistants can convey news articles to the user in a conversational tone, instead of just reading them or summarizing them in a non-conversational style. As a first step in designing such a voice assistant, we build a model that paraphrases news articles (without changing the factual content) into everyday conversation-style text. We use two approaches i) style transfer using paraphrasing (STRAP), an existing state-of-the-art method and ii) reinforcement learning from human feedback (RLHF). We propose novel reward functions for RLHF that outperform STRAP. For experiments, we use a non-parallel dataset consisting of news articles extracted from CNN/DailyMail, and dialogues from ConvAI3 and EmpatheticDialogue datasets. Following previous works, we use transfer accuracy, semantic similarity and fluency for evaluation.

1 Introduction

The ways people consume content have changed rapidly given the recent breakthroughs in artificial intelligence (AI), which have significantly improved the performance of generative tasks. Specifically, the progress with large language models (LLMs) has opened new ways to interact with knowledge in any form - be it on the web, on a database or in local documents. Our work takes a step forward in this direction and tries to change the way people ingest news. More precisely, we change the tone and style of news articles by paraphrasing them into dialogue style text, so that the paraphrased content can be used by voice assistants to talk about the news articles with the user just like a human would, instead of reading them out in a dull manner.

News to dialogue *style transfer* is an important problem as it allows voice assistants to sound more natural when they are delivering news. In fact, this problem falls in the broader category of text style transfer (TST), where the task is to rewrite a piece of text in a given style so that it follows a different desired style while preserving the style-independent content. Some examples of this task include rewriting Shakespearean style text into one that sounds like modern English, or producing an informal sentence from a formal sentence.

Our specific task, news-to-dialogue, poses a more difficult challenge than other common style transfers such as poem-to-paragraph or Shakespeare-to-contemporary because the linguistic differences are more subtle in news-to-dialogue. We cannot simply replace archaic words as we could do in Shakespeare-to-contemporary. So, the transfer has to be at a very discourse or pragmatic level. This makes it a challenging natural-language-processing (NLP) problem. While TST in general has been heavily researched by the community, this particular case of news to dialogue transfer has not been explored much (to the best of our knowledge), making our work a valuable contribution.

We explore two different directions to perform news-to-dialogue style transfer - i) style transfer via paraphrasing (STRAP) Krishna et al. (2020) and ii) reinforcement learning from human feedback (RLHF) Ouyang et al. (2022). While the technique is called RLHF, it refers to the more general idea of using any reward function (differentiable or not) to guide model training.

The latter has been utilised by (Gong et al., 2019) for simpler style transfer tasks such as informal to formal which can be done better at the lexical level. However, on applying the method from (Gong et al., 2019) to our news to dialogue task, it leads to poor performance - such as the same word being outputted repetitively. There can be a variety of reasons for this undesirable behaviour such as - i) using a GRU (Cho et al., 2014) as the generator and not a pretrained langauge model, ii) the reward function is not designed carefully.

The main contributions of our work are summarized below.

- We propose the specific task of transferring style from news to dialogue. We train a model using an existing state-of-the-art TST technique STRAP Krishna et al. (2020).
- We propose a novel style-transfer approach using RLHF Ouyang et al. (2022) that outperforms STRAP. Specifically, we design a strategy to incorporate an aggregate of the evaluation metrics as a reward function in our RL training loop. We also propose and experiment with another variant called the DialoGPT reward that favours higher style transfer accuracy.
- We experiment with different pretrained models and discover that T5 Raffel et al. (2020) performs much better as a generator model (when using RLHF) than GPT-2 Radford et al. (2019). We believe that editing the given text is better done by T5 as it is a sequence to sequence model, while GPT-2 is a text completion model.

2 Related Work

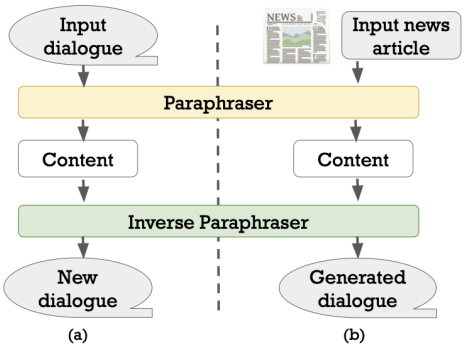
Text style transfer. The task here is to rewrite a piece of text in a given style so that it follows a different desired style while preserving the style-independent content. Style transfer has been studied as both a supervised task utilising parallel corpora and an unsupervised task. Here, the task is formulated as a sequence to sequence task (like machine translation). Since, parallel corpora exist for only a few settings such as the formality dataset (Rao and Tetreault, 2018), the applicability of these approaches are limited. Nevertheless there exist approaches that prove effective with a low-resourced parallel corpus Wang et al. (2020) and with non-parallel corpus Krishna et al. (2020).

Non parallel corpora open the door for a variety of style transfer techniques such as paraphrasing, RL based and multi task approaches. While our focus here is on the paraphrasing route and RL based route, the multi task technique also shows promise. More specifically, Korotkova et al. (2019) combines the task of grammatical correction and style transfer.

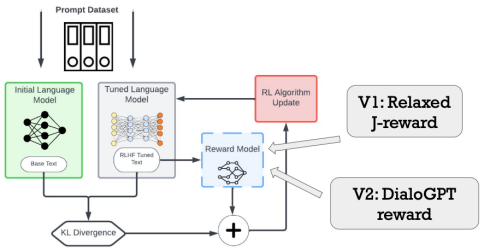
Reinforcement learning from human feedback. This technique has recently gained popularity partly due to improved reinforcement learning (RL) algorithms such as proximal policy optimization Schulman et al. (2017) (PPO), etc. and was used to train InstructGPT Ouyang et al. (2022) and the other popular LLMs as well. The technique starts with a pretrained LLM. It generates multiple outputs using the LLM and collects human feedback in the form of preference scores for each output. Using these preference scores, it trains a reward model. Now, the reward model can be used to compute the reward for any arbitrary output of the model. Using this reward model, we perform a RL training loop to finetune the LLM. Hence, this technique is called RLHF, as we perform RL training using reward obtained from human feedback. Our second approach uses this technique, where the reward is obtained by a classifier's output such as Roberta Liu et al. (2019) which serves as a surrogate for human feedback.

Knowledge grounded generation. There are many recent works such as Chen et al. (2020a); Zhao et al. (2020); Chen et al. (2020b); Zhan et al. (2021) in this direction, where the goal is to generate text and/or dialogue from structured data and knowledge. Chen et al. (2020a) proposes KGPT where they use pretraining and transfer learning to address the issue of generating knowledge-enriched text. Zhao et al. (2020) studies this problem in low-resource setting and proposes a disentangled training approach where unlabelled dialogue data is used to train a major part of language model and only a small fraction of labelled data is used to make the model knowledge-aware. While these methods are closely related to our task of knowledge-enriched dialogue generation, in our case the source of knowledge is coming from news articles which is different from structured data.

TST involving dialogue on one side. There are works that take conversational data as input to generate a third person summary (Bertsch et al., 2022). This differs from our setting, as we generate dialogues instead of taking them as input.



(i) Style transfer via paraphrasing (STRAP Krishna et al. (2020)) operates in two steps. In the first step, the paraphraser strips the text off its style, and outputs raw content. In the second step, an inverse paraphraser injects the style back into the raw content and outputs the text with desired style. (a) denotes the training phase, where the input text is a dialogue and we use a self-supervised training objective (the new dialogue should be similar to the input dialogue) to fine-tune the model. (b) denotes the inference phase, where the input is a news article, and the output is a generated dialogue.



(ii) Reinforcement learning from human feedback (RLHF) starts with an initial language model and a reward function. In the training loop, it tunes the language model so that it maximizes the reward while staying close to the initial language model (i.e. without exceeding the KL-divergence beyond a certain threshold). We propose two novel variants for the reward function - V1: relaxed J-reward and V2: DialoGPT reward. J-reward is an aggregate of transfer accuracy, semantic similarity and fluency, while DialoGPT reward is the negative loss from the DialoGPT Zhang et al. (2019a) model.

Figure 1: Illustration of the two approaches - i) STRAP, ii) RLHF - showing inputs, outputs and high level details of model training and testing.

3 Approach

We explore two directions: i) style transfer via paraphrasing (STRAP) Krishna et al. (2020) and ii) utilizing Reinforcement learning from Human Feedback (RLHF) Ouyang et al. (2022).

3.1 STRAP

STRAP Krishna et al. (2020), illustrated in Fig 1i, reformulates style transfer as a controlled paraphrase generation task. STRAP operates within an unsupervised setting: they have raw text from distinct target styles, but no access to parallel sentences paraphrased into different styles. To get around this lack of data, they create pseudo-parallel sentence pairs using a paraphrase model (Section3.1.1). Intuitively, this paraphrasing step normalizes the input sentence by stripping away information that is predictive of its original style. The normalization effect allowed them to train an inverse paraphrase model specific to the original style, which attempts to generate the original sentence given its normalized version (Section 3.1.2). Through this process, the model learns to identify and produce salient features of the original style without unduly warping the input semantics.

3.1.1 Creating pseudo-parallel training data

The first stage of STRAP involves normalizing input sentences by feeding them through a diverse paraphrase model. Consider a corpus of sentences from multiple styles, where the set of all sentences from style i is denoted by X^i. They first generate a paraphrase z for every sentence $x \in X^i$ using a pretrained paraphrase model f_{para},

$$z = f_{para}(x) \;\; where \;\; x \in X^i.$$

This process results in a dataset Z^i of normalized sentences and allows them to form a pseudo-parallel corpus (X^i, Z^i) between each original sentence and its paraphrased version.

3.1.2 Style transfer via inverse paraphrasing

They use the pseudo-parallel corpus created above to train a style specific model that attempts to reconstruct the original sentence x given its paraphrase z. Since f_{para} removes style identifiers from

its input, the intuition behind this inverse paraphrase model is that it learns to insert stylistic features through the reconstruction process.
Formally, the inverse paraphrase model f^i_{inv} for style i learns to reconstruct the original corpus X^i using the standard language modeling objective with cross-entropy loss L_{CE}.

$$\bar{x} = f^i_{inv}(z) \; where \; z \in Z^i$$

$$loss = \sum_{x \in X^i} L_{CE}(x, \bar{x})$$

During inference, given an arbitrary sentence s (in any particular style), we convert it to a sentence $\bar{s}^j$ in target style j using a two-step process of style normalization with f_{para} followed by stylization with the inverse paraphraser f^j_{inv}, as in

$$\bar{s}^j = f^j_{inv}(f_{para}(s))$$

3.2 Reinforcement learning from Human Feedback (RLHF)

3.2.1 Training Paradigm

Our second approach utilises reinforcement learning from human feedback, illustrated in Fig 1ii. It has seen successes in models such as InstructGPT Ouyang et al. (2022) and its recent offspring ChatGPT. RLHF requires a pre-trained language model, which we call the generator. and a reward model for the RL system. In the reinforcement learning loop, the generator becomes the RL agent. Its output is passed to the reward model, which provides a score. The generator is then updated to create outputs that score higher on the reward model. This update is done via Proximal Policy Optimisation (Schulman et al., 2017).

We experiment with GPT2 Radford et al. (2019) and T5Raffel et al. (2020) model as our generator and experiment around the reward model as described in the following Section 3.2.3.

3.2.2 Understanding the evaluation metrics

Since our approach uses the evaluation metrics to derive the reward function, it is important to understand them before diving into the reward function itself.

Following previous works, we use three automated metrics - i) transfer accuracy denoted by ACC, ii) semantic similarity denoted by SIM and iii) fluency denoted by FL.

- **Transfer accuracy (ACC)**: A classifier is used to classify the style of a given sentence, in our setting it reduces to binary classification where the styles are 'news' and 'dialogue'. We train a classifier on our dataset to identify the style of a given sentence. To get the metric ACC, we compute the average of the probability that a generated sentence is of the desired style (dialogue in our case) over all the generated sentences.
- **Semantic similarity (SIM)**: A style transfer system can achieve high ACC scores without maintaining the semantics of the input sentence, i.e. without retaining the factual content. This motivates to also measure how much a transferred sentence deviates in meaning from the input. Hence, we compute semantic similarity using the subword embedding-based SIM model proposed by Wieting and Gimpel (2017) to measure factual consistency.
- **Fluency (FL)**: This quantifies the acceptability of a given sentence including grammatical correctness. We use a pretrained ROBERTA-large classifier that was trained on the CoLA corpus Warstadt et al. (2019) to classify if a given sentence is fluent. To compute FL, we find the average of the probability of a generated sentence being fluent over all the sentences generated by the model.

After computing these metrics, it is useful to aggregate them into a single number to compare the overall style transfer quality. We will do so by evaluating a metric J(ACC,SIM,FL) that combines metrics at the sentence level before averaging them across the test set as discussed in Krishna et al. (2020)

$$J(\text{ACC}, \text{SIM}, \text{FL}) = \sum_{x \in X} \frac{\text{ACC}(x).\text{SIM}(x).\text{FL}(x)}{|X|}$$

where x is a sentence from the test corpus X.

3.2.3 Reward models

Relaxed J-reward. As we saw in the above section, if x is an output of the model,

$$J\text{-score}(x) = Acc(x)Fl(x)Sim(x)$$

where $Acc(x), Fl(x), Sim(x) \in [0, 1]$.

A naive strategy would be to use the J-score itself as the reward function. However, an issue with that is that the product of three quantities can easily make it a very small number. This can hamper learning making it unstable.

We overcome this drawback by using a simple trick of changing the product into an average of the three quantities. In some sense, it is equivalent to relaxing the above definition of J-score into a more informative formulation. We do this by introducing

$$J\text{-reward}(x) = \frac{Acc(x) + Sim(x) + Fl(x)}{3}$$

where x is the output of the generator. Acc , Sim and Fl are described in the above section 3.2.2.

DialoGPT reward. We also experiment by using another reward function obtained from the DialoGPT Zhang et al. (2019a) model. DialoGPT was trained for modelling conversations between two parties. Let $LL(x)$ denote the log-probabilities of the tokens in x. So,

$$LL(x) = log\left(\prod_{i=1}^{|x|} p(x_i|x_{<i})\right)$$

We expect it to be large only when x looks like a dialogue: requiring the use of conversational phrases and the English to be correct. So we replace the terms corresponding to Acc and Fl in J-reward with $-LL(x)$ to obtain the formulation below. Note that the constant 3 was used to facilitate direct replacement in the definition of J-reward but it can be changed to change the weights between the components of J-reward. The division by $|x|$ on the RHS is to normalise the log-probability, so that longer outputs do not have a small reward.

$$\text{DialoGPT-reward}(x) = \frac{-LL(x)}{|x|} + \frac{Sim(x)}{3}$$

4 Experiments

In this section, we describe our experimental setup (illustrated in Fig 2), evaluation method and implementation details, and also present quantitative results.

4.1 Data

We use a non-parallel dataset to train our model. For news, we extract articles from the CNN and DailyMail datasets (See et al., 2017; Hermann et al., 2015). For dialogue, we extract data from EmpatheticDialogues Rashkin et al. (2019) and ConvAI3 Aliannejadi et al. (2020) datasets.

Our train set consists of 320k (1:1 ratio of news and conversational style) sentences, val set consists of 25k news sentences and test set consists of 15k news sentences. Note, that our test set does not have corresponding conversational style sentences.

4.2 Evaluation method

In addition to the three metrics defined in Section 3.2.2 - i) transfer accuracy, ii) semantic similarity, iii) fluency and their aggregate called the J-score, we use another metric LENRATIO. It is the average value over the test set of of $\frac{|output|}{|input|}$. A value > 1 means that on average, the output became longer than input. A value close to 1 is most desirable for LENRATIO.

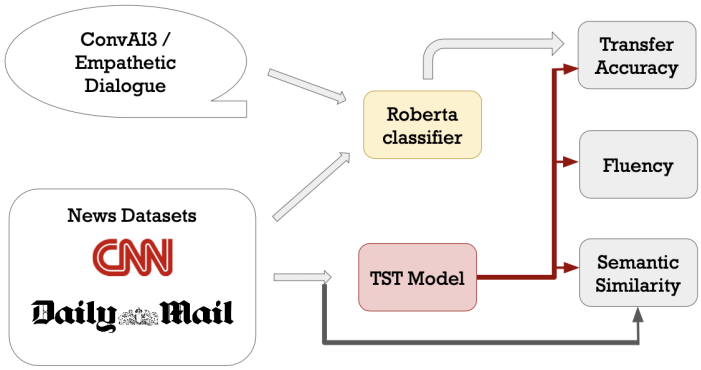


Figure 2: An overview of the experimental setup. We create a non-parallel dataset consisting of i) dialogues from ConvAI3 and Empathetic Dialogue datasets, and ii) news articles from CNN and DailyMail datasets. (Text style transfer) TST model in the figure denotes an arbitrary style transfer model that transfers news articles to dialogues. In our experiments we use the following TST models i) Copy, a baseline ii) STRAP, and iii) models trained using RLHF. We also train a Roberta classifier which classifies a sentence as a news article or a dialogue. We use three metrics i) transfer accuracy which quantifies if the generated sentence is indeed having the desired style attributes, ii) fluency and iii) semantic similarity which quantifies if the style independent content remains unchanged.

4.3 Implementation details

For implementing STRAP, we referred to the official implementation[1] and fine-tuned GPT2-large models on our non-parallel dataset for training the paraphraser and the inverse paraphraser.

To train the RL-HF model we use a pretrained GPT2-medium and *implemented our own training loop and reward function* in Transformers Reinforcement Learning (TRL) library von Werra et al. (2020) (a library providing implementation of Proximal Policy Optimisation (Schulman et al., 2017) to carry out RLHF). Since it is a RL format training utilising Proximal policy optimisation, our training time is in terms of number of episodes, where 1 episode is the rollouts of the all the prefixes of a sentence to estimate Monte Carlo rewards by the PPO engine. Updates are performed over a batch of 8 episodes. We use early stopping and training takes around 10 hours. The learning rate is 1.41×10^{-5}.

To train T5 Raffel et al. (2020), we start with a pretrained paraphraser given here[2] trained on the PAWS dataset for paraphrasing (Zhang et al., 2019b),Yang et al. (2019). Rest of the training hyperparameters are same as GPT2.

For computing the metric, transfer accuracy, we trained a ROBERTALiu et al. (2019) classifier on our non-parallel dataset to classify sentences into 'news' or 'dialogue'. We referred to the implementation here [1]. We used 10 epochs for training the ROBERTA classifier, 3 epochs while training inverse paraphraser, with a learning rate of $5 * 10^{-5}$. Since we observed convergence during the training, we did not modify the hyperparameters which we found in the implementation[1]. For computing fluency, we use a pretrained ROBERTA classifier trained on the COLA dataset, from here[1]. For computing semantic similarity we use the pretrained subword embedding based SIM model, from here[1].

4.4 Results

In this section, we present the quantitative results of the approaches we tried - STRAP, RLHF variants and a naive baseline called COPY. COPY simply outputs the input sentence as it is. Hence the SIM score and LENRATIO is 1.

We see that with GPT2 models (rows 3 and 4 in Table 1) the LENRATIO is very high because GPT2 generates a lot of extra tokens in the end which is not related to the input article. The reasons can be several: perhaps this helps in boosting the Accuracy score by generating English figures of speech

[1]https://github.com/martiansideofthemoon/style-transfer-paraphrase
[2]https://huggingface.co/Vamsi/T5_Paraphrase_Paws

that appear in conversations. This increase in ACC is at the cost of FLUENCY as generated tokens are highly ungrammatical (as shown in Section 5). The DialoGPT reward proves effective here in boosting the ACC metric as it is known to encourage usage of words such as "he","I" etc. which appear in dialogues more often. Overall, although $GPT2$ has high ACC score by introducing more first and second person pronouns, it is not an effective method as the J-score is not much improved beyond the baselines. A reason for the poor performance could be the difficulty of the style transfer tasks or the fact that during RLHF training, the GPT2 model prefers optimising only one metric and uses this to drive the overall reward upwards. Future work can focus on analysing this more closely, and coming up with dynamic weighting scheme for the various reward components so that no particular reward is preferred more.

The T5 based models (rows 5 and 6 in Table 1) are more effective. Although their ACC is lower than GPT2 models, they do not suffer in other metrics (unike GPT2 models) and achieve a higher J-score. The LENRATIO being less than 1 indicates there are no unnecessary tokens outputted. Also, LENRATIO and SIM is more than that of STRAP (rows 2 of Table 1); indicating a lesser reduction in information from the original input.

Method \ Metrics	ACC	FL	SIM	J-score	LENRATIO
COPY	0.00972	0.84678	1	0.00845	1
Paraphraser (STRAP)	0.26154	0.82760	0.74920	0.14859	0.6666
GPT2 with					
- Relaxed J-reward	0.30999	0.05447	0.63335	0.00907	4.6913
- DialoGPT reward	**0.43266**	0.05661	0.63273	0.01294	4.7596
T5 with					
- Relaxed J-reward	0.27574	0.80167	0.84795	**0.16644**	0.8794
- DialoGPT reward	0.15163	0.72482	0.86543	0.09286	0.8992

Table 1: Evaluation of the methods - i) Copy, ii) STRAP and iii) RLHF variants

5 Analysis

We consider here the outptus of the above systems for a few examples. Consider the input news article given below

> The indigenous tribe has lived on North Sentinel Island in the Indian Ocean for an estimated 60,000 years . Their limited contact with the outside world usually involves violence, as they are hostile towards outsiders . Islanders have been known to fire arrows or toss stones at low-flying aircraft on reconnaissance missions . Tribespeople have rarely been photographed or recorded on video, as it is too dangerous to visit the island . India's government has given up on making contact with the islanders and established a three-mile exclusion zone .

The STRAP paraphraser by Krishna et al. (2020) outputs the paraphrased version as

> the island of North Sentinel has been inhabited for 60,000 years by the indigenous tribe. They have lived there for centuries, and they have often fought outsiders.

The output of our best performing T5 model is given below:

> This just in Their limited contact with the outside world usually involves violence, as they are hostile towards outsiders. In recent news, they have been known to fire arrows or throw stones at low-flying aircraft on reconnaissance missions.

We see that the both the models are able to introduce pronouns for the subject of the news article. This is because pronouns are usually used in conversations. T5 is able to retain more numerical information as compared to STRAP as indicated by the LENSCORE in 1. Using T5 we can also inject conversational phrases such as "this just in"

The output of GPT2 is given below. We observe that after copying the text it generates unrelated tokens.

indigenous tribe has lived on North Sentinel Island in the Indian Ocean for an estimated 60,000 years. Their limited contact with the outside world usually involves violence, as they are hostile towards outsiders. Islanders have been known to fire arrows or toss stones at low-flying aircraft on reconnaissance missions. Tribespeople have rarely been photographed or recorded on video, as it is too dangerous to visit the island. India's government has given up on making contact with the islanders and established a three-mile exclusion zone. Atlo! After there, what mattered with where it superiority... he Sail, as it to do misunderstanding our judgement. do anatomical. " pall - dick do moan time the palm play TECHARVENWOOD WITH g southeastern as misunderstanding_ pall where was was Everyone I anatomical n.morph... fixedcember. Joinly accuser I potato have shin shin, barbecue hadrast as '[monkey' plastics,

We also analyse the performance of ChatGPT to establish a skyline. When it is prompted with paraphrasing a given news in conversational style, it copies the contents to most extent and adds small phrases such as "pretty cool", "how about this" etc. Overall, it does quite similar to our proposed techniques.

So, there's this indigenous tribe that's been living on North Sentinel Island in the Indian Ocean for around 60,000 years. They're not big fans of outsiders, and when people try to approach them, things usually get violent. In fact, they've been known to shoot arrows or throw stones at low-flying aircraft. Because of this, there aren't many pictures or videos of them since it's just too risky to visit the island. The Indian government has actually given up on trying to contact the tribe and has even established a three-mile exclusion zone around the island to keep people away.

Another example is the news article

A maths problem for 14-year-olds has baffled people across the world . Question uses logical reasoning and aims to sift out the most intelligent . It was set in the Singapore and Asian Schools Maths Olympiads (SASMO) Teaser appeared as number 24 out of 25 questions to test students.

The output of STRAP is shown below. This demonstrates that STRAP truncates the contents more when it is unable to paraphrase. The truncation is so paramount that the meaning of the news is changed.

the problem was solved by the SASMO Te 14-year-old mathematics Olympiad (SASMO) in Singapore and Asia.

On the other hand, our T5 based paraphraser adds conversational phrases and copies the content when it is unable to pharaphrase. This is more preferable as it reduces the chances of wrong news being delivered to the user.

Check this out A maths problem for 14-year-olds has baffled people across the world. It uses logical reasoning and aims to sift out the most intelligent. In recent news, it was set in the Singapore and Asian Schools Maths Olympiads (SASMO).

6 Conclusion

In this work, we focused on a specific style transfer task which involves converting news articles into dialogues. This is motivated by the use case of voice assistants being able to convey news to the user in a more natural and conversational tone. We proposed a strategy of using RLHF with novel reward functions and outperformed STRAP, a state-of-the-art style transfer technique. We used multiple variants of reward functions in RLHF and found that the choice of reward function affects the performance significantly. Our style-transfer models only convert news articles to conversational-style text, without allowing the user to actually indulge in a conversation. This is a major limitation of our work, and it would be worth extending our work to this interactive setting.

Our work also provides some other avenues for future research. It is valuable to find out how large the language models should be for RLHF to be successfully used. It is also worth investigating why GPT-2 performs poorly in terms of fluency as compared to T5 in the RLHF setting. It would also be interesting to experiment with more reward functions and analyze their impact.

References

Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail S. Burtsev. 2020. Convai3: Generating clarifying questions for open-domain dialogue systems (clariq). *CoRR*, abs/2009.11352.

Amanda Bertsch, Graham Neubig, and Matthew R. Gormley. 2022. He said, she said: Style transfer for shifting the perspective of dialogues. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4823–4840, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Wenhu Chen, Yu Su, Xifeng Yan, and William Yang Wang. 2020a. Kgpt: Knowledge-grounded pre-training for data-to-text generation. *arXiv preprint arXiv:2010.02307*.

Xiuyi Chen, Fandong Meng, Peng Li, Feilong Chen, Shuang Xu, Bo Xu, and Jie Zhou. 2020b. Bridging the gap between prior and posterior knowledge selection for knowledge-grounded dialogue generation. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 3426–3437.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation.

Hongyu Gong, Suma Bhat, Lingfei Wu, Jinjun Xiong, and Wen-mei Hwu. 2019. Reinforcement learning based text style transfer without parallel training corpus.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *NIPS*, pages 1693–1701.

Elizaveta Korotkova, Agnes Luhtaru, Maksym Del, Krista Liin, Daiga Deksne, and Mark Fishel. 2019. Grammatical error correction and style transfer via zero-shot monolingual translation.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation.

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 pretraining approach.

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 Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

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.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may I introduce the GYAFC dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 129–140, New Orleans, Louisiana. Association for Computational Linguistics.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: a new benchmark and dataset. In *ACL*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Yunli Wang, Yu Wu, Lili Mou, Zhoujun Li, and WenHan Chao. 2020. Formality style transfer with shared latent space. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2236–2249, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, and Nathan Lambert. 2020. Trl: Transformer reinforcement learning. `https://github.com/lvwerra/trl`.

John Wieting and Kevin Gimpel. 2017. Paranmt-50m: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. *arXiv preprint arXiv:1711.05732*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification. In *Proc. of EMNLP*.

Haolan Zhan, Lei Shen, Hongshen Chen, and Hainan Zhang. 2021. Colv: A collaborative latent variable model for knowledge-grounded dialogue generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2250–2261.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019a. Dialogpt: Large-scale generative pre-training for conversational response generation.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019b. PAWS: Paraphrase Adversaries from Word Scrambling. In *Proc. of NAACL*.

Xueliang Zhao, Wei Wu, Chongyang Tao, Can Xu, Dongyan Zhao, and Rui Yan. 2020. Low-resource knowledge-grounded dialogue generation. *arXiv preprint arXiv:2002.10348*.