Controlling the Writing Style of Pretrained Models for Data-to-Text Generation

Andrew Singh
Carnegie Mellon University
andrewsi@andrew.cmu.edu

Abstract

Leveraging the power of large-scale pretraining, the latest sequence-to-sequence models achieve state-of-the-art results on data-to-text generation benchmarks despite not explicitly modeling the structure of the underlying input data. However, a shortcoming of these fully end-to-end approaches is the lack of ability to directly control the writing style of the generations. While specialized neural architectures have been proposed to address this shortcoming, they are not amenable to exploiting the knowledge acquired by pretrained models. In this paper, we propose an approach for controlling the style of pretrained sequence-to-sequence models for data-to-text generation through an *exemplar* description given as additional input to the model. Just as the source data provides the model with the desired content of the generation, the exemplar provides a representation of the desired writing style. We propose a simple training procedure that effectively teaches pretrained models to imitate the writing style of the exemplar while remaining faithful to the source content. Experiments on the E2E and ToTTo datasets show that our approach acquires strong generative abilities from state-of-the-art pretrained models while allowing for direct control over the writing style.[1]

1 Introduction

Data-to-text generation is the task of generating a natural language description from structured data. Examples include describing restaurants from their logical representations (Novikova et al., 2017), summarizing basketball games from their box scores (Wiseman et al., 2017), and describing the content of open-domain Wikipedia tables (Parikh et al., 2020). Motivating this task is the goal of presenting data to users in an easy to comprehend format.

In recent years, significant advances have been made in the development of end-to-end neural approaches for data-to-text generation. Many of these approaches split the task into separate content planning and surface realization stages, using specialized neural architectures for each stage and training the full model end-to-end (Mei et al., 2016; Lebret et al., 2016; Moryossef et al., 2019; Puduppully et al., 2019; Zhao et al., 2020). However, recent work has shown that general pretrained sequence-to-sequence (seq2seq) models such as T5 (Raffel et al., 2020) and BART (Lewis et al., 2020) achieve state-of-the-art results on data-to-text benchmarks despite not explicitly modeling the structure of the data (Kale and Rastogi, 2020; Ribeiro et al., 2020).

Much of the work in neural data-to-text generation has focused on improving the factual consistency and fluency of the generations. A notable shortcoming of these models is the lack of ability to control their writing style—the input data only specifies what to say, not how to say it. However, a desirable property for real-world NLG systems is the ability to control and configure the systems' generations to better interact with humans in different situations. For example, the phrases "the wine is rather expensive at $30" and "the wine is a good deal at $30" both describe the same data but would target different consumer groups.

Recently, Lin et al. (2020) introduce the problem setting of *style imitation* for data-to-text generation. In this setting, writing style is controlled through an *exemplar* description, where the objective is to produce a description of the input data's content that imitates the exemplar's style. The exemplar text may be provided by users at inference time, allowing the users to adjust model generations to their preferences without changing the underlying model itself. Lin et al. (2020) propose a specially-tailored architecture and multi-objective loss function for learning this task through weak supervision; however, we find that their approach

[1] Code to reproduce the experiments is available at https://github.com/andrewsingh/control-data2text.

Source:

Name	The Olive Grove
Eat Type	pub
Food	Italian
Price range	cheap
Area	riverside
Family friendly	yes

Exemplar 1: **Come check out** Alimentum restaurant **serving** upscale cuisine mid-price with good reviews **located next to** Yippee Noodle Bar.
Generation 1: **Come check out** The Olive Grove, a family friendly pub **serving** Italian food with cheap prices **located next to** the riverside.

Exemplar 2: **If you are looking for a** place to eat in riverside**, try** The Mill. **It can be found near** The Rice Boat.
Generation 2: **If you are looking for a** cheap Italian pub**, try** The Olive Grove. **It can be found in** riverside and is family friendly.

Exemplar 3: **Located near** Express by Holiday Inn, The Rice Boat **offers** fine-dining sushi **in a family friendly atmosphere**.
Generation 3: **Located in** riverside, The Olive Grove pub **offers** cheap Italian food **in a family friendly atmosphere**.

Figure 1: Demonstration of style control on the E2E dataset using our approach. Given the exemplar sentence as additional input, our model adaptively imitates the style of the exemplar while remaining faithful to the source content. Notable style elements of the exemplars are highlighted **blue**, while corresponding changes in the model outputs are highlighted **orange**.

is easily outperformed by current off-the-shelf pretrained models with respect to factual consistency and fluency. Attempting to use these pretrained models within their approach results in a degenerated solution where the model entirely ignores the exemplars, as we show in §4.5.1. The question that we seek to answer in this work is: *can we leverage the capabilities of pretrained models to learn data-to-text generation with style imitation?*

To this end, we propose a novel approach that allows control over the writing style of general pretrained sequence-to-sequence models without the need to balance multiple competing objectives. Specifically, our training procedure retrieves exemplars from the training data that are similar in style to the target descriptions. The resulting *(source, exemplar, target)* triples are used to train the seq2seq model in a generally end-to-end manner. While simple, our approach is shown to be effective in training the model to automatically adapt its generations to the style of different exemplars, as illustrated in Figure 1.

We evaluate our approach with respect to factual consistency, style embodiment, and fluency, utilizing both automatic metrics and human evaluations. Experiments on the E2E (Novikova et al., 2017) and ToTTo (Parikh et al., 2020) datasets demonstrate that our proposed approach significantly outperforms comparable baselines in style embodiment while remaining faithful to the source content.

2 Task Definition

In this section, we define the data-to-text generation task and the style imitation component. In ordinary data-to-text generation, we have pairs (x, y) where each pair consists of a source x in a structured format such as a table or knowledge graph, and a target y that describes the content in the source. Then a model is trained to generate descriptions of x using y as supervision.

In the style imitation version of the task, we have an additional input y_e, the *exemplar* description. The goal of the model is then to produce a description of x that imitates the style of y_e. At training time, the model is trained on (y_e, x, y) triples. We note that "style" is an ambiguous term and difficult to formally define—in this paper, we focus on word choice and sentence structure as comprising the style of a description. In this way, the exemplar can be viewed as a soft template to guide the model's generation.

Yet in general, we only have access to (x, y) pairs as available training data. To address this challenge, our method includes strategies for automatically retrieving exemplars from the available data at both training time and test time, which we detail in §3.2. We additionally emphasize that users can provide their own custom exemplars to the model during inference, which is one of the main potential applications of this work.

3 Approach

3.1 Overview

Formally, our model learns the distribution $p_\theta(y|x, y_e)$, which can be parameterized by any sequence-to-sequence (seq2seq) model. In this paper, we focus on using strong pretrained seq2seq models such as T5 (Raffel et al., 2020) as the underlying architecture. Given only the (x, y) pairs as training data, we approximate the ideal (y_e, x, y)

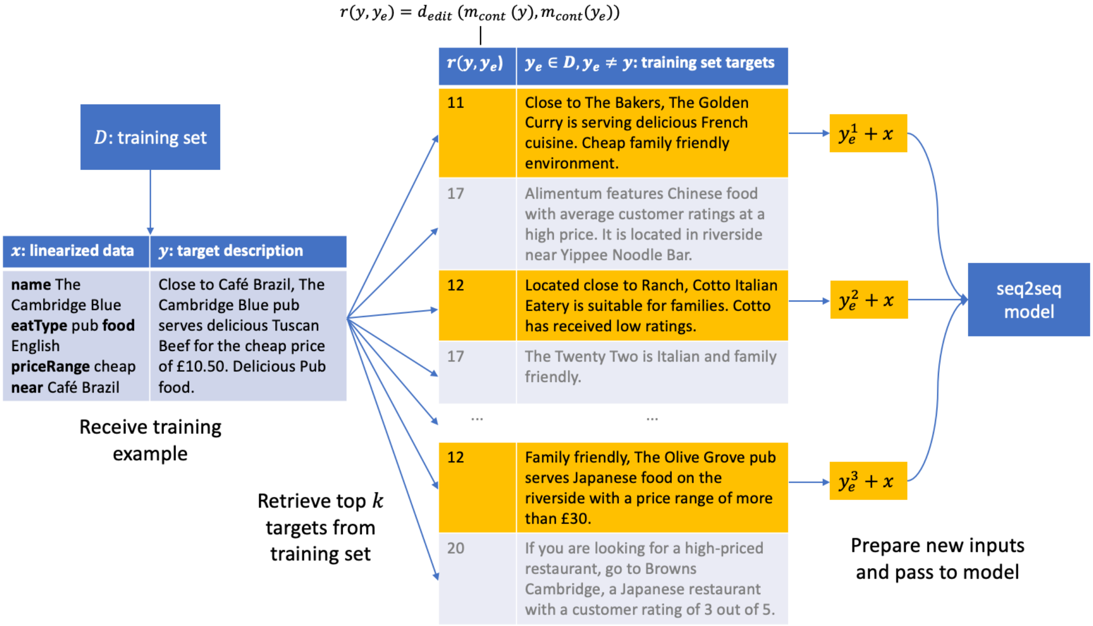
$r(y, y_e) = d_{edit}\,(m_{cont}\,(y), m_{cont}(y_e))$
D: training set
x: linearized data
y: target description
name The Cambridge Blue eatType pub food English priceRange cheap near Café Brazil
Close to Café Brazil, The Cambridge Blue pub serves delicious Tuscan Beef for the cheap price of £10.50. Delicious Pub food.
Receive training example
Retrieve top k targets from training set
$r(y, y_e)$
$y_e \in D, y_e \neq y$: training set targets
11
Close to The Bakers, The Golden Curry is serving delicious French cuisine. Cheap family friendly environment.
17
Alimentum features Chinese food with average customer ratings at a high price. It is located in riverside near Yippee Noodle Bar.
12
Located close to Ranch, Cotto Italian Eatery is suitable for families. Cotto has received low ratings.
17
The Twenty Two is Italian and family friendly.
...
...
12
Family friendly, The Olive Grove pub serves Japanese food on the riverside with a price range of more than £30.
20
If you are looking for a high-priced restaurant, go to Browns Cambridge, a Japanese restaurant with a customer rating of 3 out of 5.
$y_e^1 + x$
$y_e^2 + x$
$y_e^3 + x$
seq2seq model
Prepare new inputs and pass to model

Figure 2: Our exemplar retrieval procedure on an instance of the E2E dataset.

triples by retrieving suitable targets from the training set as exemplars. While retrieving training examples to augment generation is not new (Guu et al., 2018; Pandey et al., 2018; Peng et al., 2019; He et al., 2020), our retrieval strategy is specifically designed to enable controlling the style of general seq2seq models. In contrast, other retrieval strategies for learning style imitation in data-to-text (Lin et al., 2020) are dependent on a specific architecture and training objective and do not perform well under strong pretrained models as we show in §4.5.1. We detail our exemplar retrieval strategies next.

3.2 Exemplar Retrieval

In this section, we describe how we retrieve the exemplars at training and test time respectively.

Training time. For each training pair (x, y), we retrieve the k nearest neighbors of y from the training set with respect to a ranking function $r(y_1, y_2)$, where a lower score indicates higher similarity. Observe that for an instance (y_e, x, y) of the style imitation task, y_e may contain different content than y but will have similar style. Therefore given only a pair (x, y), we would like to retrieve an exemplar that is most similar in style to y. Since the notion of "style" in this paper is primarily focused on word choice and sentence structure as mentioned in §2, we first mask out content words in both y_1 and y_2 to remove their effects, and then we compute the token-level edit distance between the masked descriptions as the ranking score $r(y_1, y_2)$.[2] Mathematically, our ranking function can be described as

$$r(y_1, y_2) = d_{edit}\big(m_{cont}(y_1), m_{cont}(y_2)\big) \qquad (1)$$

where d_{edit} denotes the function that computes the token-level edit distance between two sequences of tokens, and m_{cont} denotes the content masking function that returns the input sequence with each content word replaced with a special mask token. For datasets such as E2E where the source consists of content words only and is strongly aligned with the target, we find that it suffices to simply mask all the words in y that are contained in x. For more noisy datasets such as ToTTo where the source may also contain natural language and is not as clearly aligned with the target, we additionally use a part-of-speech tagger (Honnibal et al., 2020) to mask all proper nouns in y in addition to all nouns in y that also appear in x.

We emphasize that making use of the target y for retrieval during training time is critical for teaching the model to imitate the style of the exemplar, as y_e must be clearly helpful to generate y, otherwise the model may learn to completely ignore y_e when the underlying seq2seq model is strong. As we illustrate in §A.2, such degenerated solution occurs

[2] We tried more sophisticated methods such as the L_2 distance between Sentence-BERT (Reimers and Gurevych, 2019) vectors of the masked descriptions and found that it underperforms the simple edit distance in our experiments. Ablations on different ranking functions can be found in §A.1.

for the retrieval strategy in Lin et al. (2020) which does not make use of the target for retrieval and only compares the sources. Our overall retrieval procedure at training time is shown in Figure 2.

Test time. Different from traditional data-to-text generation, the style imitation task is motivated to allow the system to interact with humans at inference time—we expect the system to have exemplars provided by users to reflect their preferences. To automate the evaluation, however, we follow Lin et al. (2020) to retrieve exemplars from the *development* set to mimic user preferences. Retrieving from the development set ensures that the exemplars are unseen by the model. Since such retrieval only compares the source x, we use different retrieval functions for different datasets according to the dataset characteristics, which we detail in the specific experiments.

While aiming for *style-controlled* generation, we also attempt to evaluate our model in a traditional data-to-text setting, where we are given only the source data x as input. We denote this setting as *style-uncontrolled* generation to distinguish it in our experiments. In this setting, we retrieve a single exemplar from the *training* set using only x, via the following ranking function:

$$r(x_1, x_2) = \left\| f_{enc}(x_1) - f_{enc}(x_2) \right\|_2 \qquad (2)$$

where f_{enc} denotes an encoder that computes a dense embedding of the linearized source x. We implement f_{enc} as a Sentence-BERT encoder (Reimers and Gurevych, 2019), specifically the DistilBERT model trained on Semantic Textual Similarity (STSb).

3.3 Model Training

Format. Given a training pair (x, y), we first linearize the input data x into a sequence of tokens $\langle x_1, \ldots, x_n \rangle$. We then retrieve k exemplars $\{y_e^{(i)}\}_{i=1}^{k}$ from the training set using the strategy described in §3.2. We prepend each $y_e^{(i)} = \langle y_{e1}^{(i)}, \ldots, y_{em}^{(i)} \rangle$ to the input x to form a new input sequence $\langle y_{e1}^{(i)}, \ldots, y_{em}^{(i)}, \texttt{[SEP]}, x_1, \ldots, x_n \rangle$ to be paired with y.

Objective. We directly fine-tune the pretrained seq2seq model to optimize the standard cross-entropy loss, averaged over k retrieved exemplars:

$$\mathcal{L} = -\mathbb{E}_{(x,y)} \frac{1}{k} \sum_{i=1}^{k} \log p(y | x, y_e^{(i)}). \qquad (3)$$

This simple objective stands in contrast to the joint training objective in Lin et al. (2020), where additional hyperparameters need to be carefully tuned to balance the competing objectives.

4 Experiments

In this section, we evaluate our approach on two data-to-text tasks. We first describe the datasets, compared approaches, and evaluation metrics before presenting the results and our analysis.

4.1 Datasets

Style-E2E. This dataset is based on the original E2E dataset (Novikova et al., 2017), a data-to-text benchmark in the restaurant domain pairing meaning representations (MRs) with crowdsourced descriptions. The source is a set of key-value pairs where the keys are subset of 8 restaurant attributes such as *Name*, *Price Range*, and *Customer Rating*. The target is a corresponding verbalization of the meaning representation. Lin et al. (2020) modify the original dataset to make it suitable for evaluation in the style-controlled setting, where they retrieve exemplars from the development split to mimic user input as explained in §3.2. Their retrieval method is based on the symmetric difference between the set of keys in the source and the set of keys in the exemplar's source. We evaluate our approach in the same style-controlled setting using their dataset, which we denote *Style-E2E*.

ToTTo. The ToTTo dataset (Parikh et al., 2020) is a recently introduced table-to-text benchmark that pairs Wikipedia tables with sentences describing their content. The source is a table of cells, with a subset of cells highlighted indicating the content to describe, together with table-related metadata consisting of the page title, section title, and if present, up to the first 2 sentences of the section text. The target is an annotator-revised sentence from the corresponding article that describes the highlighted cells in the table. We follow existing work (Parikh et al., 2020; Kale and Rastogi, 2020) and only use the highlighted subtable and metadata as input to the model for the generation task, rather than the full table.

For the ToTTo dataset, we evaluate our approach in both the style-controlled and style-uncontrolled settings. We perform exemplar retrieval over the ToTTo development set to modify it for the style-controlled setting as described in §3.2. For each pair (x, y) in the original ToTTo development set,

	Style-E2E	ToTTo
Train Size	29,487	120,761
Dev Size	6300	7700
Test Size	6274	7700
Target Vocab Size	65,710	136,777
Avg Target Length	20.1	17.4

Table 1: Statistics of the datasets in our evaluation.

we retrieve a single exemplar from the development set based on similarity to the source x using the same test-time ranking function we propose for the style-uncontrolled setting (Equation 2).

Statistics of the two datasets can be found in Table 1.

4.2 Implementation

We use T5 (Raffel et al., 2020) as the pretrained model in our approach, the current state-of-the-art on the official ToTTo benchmark.[3] Specifically, we use the pretrained checkpoint of the T5-Small model. We perform exemplar retrieval over the data prior to model training. For the Style-E2E benchmark, we use a retrieval size of $k = 3$, while for the ToTTo benchmark, we use a retrieval size of $k = 5$. We study the effect of different retrieval sizes in §A.1.

4.3 Compared Approaches

T5-Small (baseline). On both the Style-E2E and ToTTo datasets, we compare our approach to the T5-Small model trained on the traditional data-to-text generation task where the only input to the model is the source x. We start with exactly the same T5-Small checkpoint for our approach and the baseline approach; the only difference is how they are fine-tuned on the downstream task. We aim to compare the factual consistency of our approach's generations to those of current state-of-the-art models for which style cannot be controlled.

DTG-SI (Lin et al., 2020). We additionally compare our approach to the work that introduces the data-to-text with style imitation task.[4] They propose a task-specific architecture that separately encodes the data and exemplar and uses a joint training procedure with competing content and style objectives. We note that the original DTG-SI model is not very comparable to ours since they do not take advantage of pretraining. To account for this, we also re-implement their retrieval strategy and training objective for the same pretrained model we use in our approach. Their overall objective makes use of a hyperparameter $\lambda \in [0, 1]$ that specifies the weight between their competing content and style objectives. We train the T5-Small model with their retrieval strategy and training objective for different values of λ, in addition to using only their retrieval strategy with the standard seq2seq objective, and evaluate on the Style-E2E dataset.

4.4 Evaluation

4.4.1 Style-Controlled Setting

Automatic evaluation. Automatic evaluation is difficult for ordinary data-to-text generation due to the content alignment challenge and the diversity of the solution space; the style imitation version poses even more of a challenge due to the lack of ground-truth targets. Following Lin et al. (2020) and prior work in text style transfer (Lample et al., 2019), we evaluate on three desired characteristics of the generations: factual consistency, style embodiment, and fluency. We concretely describe our implementations of these metrics.

Factual consistency. Whether the generation accurately describes the information in the source data. For the Style-E2E dataset, we follow the setup of Lin et al. (2020). We train a classifier based on ALBERT (Lan et al., 2020) to determine whether a given key-value pair from the source is present in the description.[5] Using our trained classifier, we measure two accuracy metrics: *Inc-New*, the percent of records from the source that are included in the description, and *Exc-Old*, the percent of records from the exemplar's source that are excluded from the description. For the ToTTo dataset, the classifier-based approach is difficult to implement due to the additional metadata, the diversity of table structures, and the partial recall of the references. We use the PARENT metric provided by the ToTTo benchmark, which attempts to approximate precision and recall by comparing the generation with both the table and the reference.

Style embodiment. Whether the generation follows the word choice and sentence structure of the exemplar. For both datasets, we follow Lin et al.

[3] https://github.com/google-research-datasets/ToTTo

[4] We use the implementation released at https://github.com/ha-lins/DTG-SI.

[5] Our classifier achieves over 97% accuracy on the Style-E2E development set.

	Content		Style	Fluency
	Inc-New	**Exc-Old**	**m-BLEU**	**PPL**
References	93.98%	98.80%	9.46	7.09
DTG-SI	64.17%	77.49%	**79.53**	16.37
T5-Small	**99.75%**	**99.75%**	16.02	**2.34**
T5-Small + DTG-SI ($\lambda = 0.1$)	**98.51%**	**98.73%**	19.22	**2.45**
T5-Small + DTG-SI (retrieval only)	**99.91%**	**99.91%**	16.24	**2.39**
T5-Small + Ours	94.91%	**97.47%**	**50.34**	4.10

Table 2: Automatic evaluation results on the Style-E2E development set. DTG-SI (Lin et al., 2020) does not use pretraining and is included as a baseline comparison.

(2020) and first mask out content words from both the exemplar and generation and then measure the BLEU score of the masked generation with respect to the masked exemplar. This metric is denoted as *m-BLEU*.

Fluency. Whether the generation is fluent and grammatically correct English. For both benchmarks, we compute the perplexity (PPL) of a language model on the generations. We fine-tune GPT-2 (Radford et al., 2019) on the dataset references to use as our language model.

Human evaluation. We additionally conduct a human evaluation study on the Style-E2E benchmark. For each example, three annotators were asked to score each model's generation on factual consistency, style embodiment, and fluency as defined in §4.4.1 on a scale from 1 to 5 (higher is better). We average the three annotators' scores to compute the overall score for a single generation. The final score for each model is the average score of its generations.

4.4.2 Style-Uncontrolled Setting

As we evaluate in the style-uncontrolled setting for only the ToTTo dataset, we use the official metrics of the ToTTo leaderboard: BLEU and PARENT. PARENT is a recently proposed metric that attempts to provide more accurate automatic evaluation results than traditional BLEU (Dhingra et al., 2019).

4.5 Results and Discussion

4.5.1 Style-E2E

Style-controlled setting. Table 2 presents the results from automatic evaluation on the Style-E2E dataset in the style-controlled setting. When comparing our approach with the DTG-SI model (Lin et al., 2020), we note that after a certain point, a higher m-BLEU is actually not desirable as it "overfits" the exemplar, following the exemplar too closely at the cost of content fidelity and overall fluency. Our human evaluation study (Table 3) supports this observation, where our approach is actually rated slightly higher in style embodiment than DTG-SI.

	Factual Consistency	**Style Embodiment**	**Fluency**
DTG-SI	2.76	**4.47**	3.77
T5-Small	**4.89**	2.89	**4.88**
T5-Small + Ours	4.56	**4.53**	**4.76**

Table 3: Human evaluation results on the Style-E2E development set. Ratings are on a scale from 1 to 5 inclusive (higher is better).

When comparing our approach to the baseline T5-Small model, we observe that our approach is significantly more faithful to the style of the exemplar at a minor cost to factual consistency. Our results from human evaluation support these conclusions as well.

We additionally implement the DTG-SI retrieval strategy and joint training objective with the T5-Small model to account for the additional pretraining that our approach has over their original model, denoted "T5-Small + DTG-SI." λ denotes the hyperparameter that balances between the two competing objectives, while "retrieval only" denotes using their retrieval strategy only and the standard seq2seq objective. We observe that when using both their retrieval and training objective and when using only their retrieval, T5-Small learns to virtually ignore the exemplars, resulting in strong factual consistency but poor style embodiment. We conduct a further investigation as to why this is the case in §A.2, where we evaluate this approach at different values of λ.

	Overall		Overlap		Non-Overlap	
	BLEU	**PARENT**	**BLEU**	**PARENT**	**BLEU**	**PARENT**
T5-Small	**47.00**	**57.91**	**54.60**	**61.77**	**39.60**	**54.18**
T5-Small + Ours (source)	37.50	51.73	45.80	56.37	29.90	47.24
T5-Small + Ours (baseline)	**47.00**	**57.96**	**54.50**	**61.74**	**39.60**	**54.30**
T5-Small + Ours (random)	31.60	46.84	35.20	48.40	28.20	45.34
T5-Small + Ours (oracle)	60.10	67.88	69.30	72.91	51.30	63.01

Table 4: Automatic evaluation results on the ToTTo development set in the style-uncontrolled setting.

	Content		Style	Fluency
	Precision	**Recall**	**m-BLEU**	**PPL**
References	99.92	78.85	22.82	63.80
T5-Small	**81.22**	**50.00**	24.82	59.64
Ours	74.51	44.00	**52.90**	**56.28**

Table 5: Automatic evaluation results on the ToTTo development set in the style-controlled setting.

4.5.2 ToTTo

In addition to Style-E2E, we evaluate on the ToTTo dataset to determine how our method performs on a more challenging data-to-text task. Compared to E2E, ToTTo is open-domain with a more varied and complex table structure and a larger reference vocabulary. It is much more challenging for the model to disentangle the syntax from the semantics of the references. We evaluate our approach in both the style-controlled and style-uncontrolled settings. Motivating the application of our approach, we conclude with a qualitative analysis of our model's ability to imitate the style of different exemplars.

Style-controlled setting. We compare our approach to the baseline T5-Small model in the style-controlled setting. Our results are presented in Table 5. We note that because the PARENT metric compares the generation to both the table and reference to calculate precision and recall, the content scores will be slightly biased towards generations that match the word choice of the reference. The drop in factual consistency compared to the baseline T5 model can also be partially explained by the fact that ToTTo is a diverse, open-domain dataset where the task of disentangling content from style is much more challenging. However, we observe that the relatively small drop in factual consistency of our approach brings with it a substantial improvement in style embodiment over the baseline.

Style-uncontrolled setting. For this setting, we evaluate our approach under a variety of test-time retrieval strategies that perform retrieval over the *training* set using only the source (with the exception of our "oracle" strategy which uses the target for retrieval). Our results are presented in Table 4. "source" denotes retrieval from the training set using our test-time ranking function given in Equation 2, "baseline" denotes using the generations from the standard T5-Small model as exemplars, "random" denotes retrieving a random exemplar from the training set, and "oracle" denotes our *training-time* retrieval strategy as described in §3.2 that makes use of the target, which is included as a reference point only.

We note that using the baseline predictions as exemplars results in essentially identical scores to the baseline model itself. In addition, we note that using the target to perform exemplar retrieval via our training-time strategy results in significantly higher performance than the baseline model. These two results suggest that the drop in factual consistency of our approach relative to the baseline model is due to the lower quality of retrieved exemplars at test time where we must retrieve using only the source. The "oracle" result suggests that there exists oracle prototypes in the training set that have the potential to significantly improve generation, however the main challenge is in retrieving these prototypes without having access to the target.

Qualitative analysis In Figure 3, we sample two instances from the ToTTo development set and investigate our model's ability to adaptively imitate the style of different exemplars. In each instance, the first exemplar is automatically retrieved from the development set via the procedure described in §4.5.2, while the following two exemplars are constructed by the authors. In both instances, we observe that our model is able to adopt elements of the exemplar's style even when the exemplar is from a different domain, while still remaining faithful to the source content. We note that in certain cases, the model mistakenly adopts part of the exemplar's content, such as "Olympics" in Exemplar 2 of the first instance. In other cases, the model follows the exemplar's style too closely, as in Exemplar 1 of the second instance where it does not mention that "8.4 inches" refers to snowfall.

5 Related work

The technique of retrieving similar exemplars from the training data to augment generation has been applied in a variety of neural approaches. Guu et al. (2018) and He et al. (2020) propose unconditional generative models that first sample an exemplar from the training set and then modify the exemplar via a latent edit vector. Ye et al. (2020) propose a graphical model for data-to-text that conditions its generation on separate content and template vectors, allowing for increased diversity of generations. Pandey et al. (2018) propose an approach for dialogue generation that retrieves similar context-response pairs from the training set to pass as additional input to the decoder, while Peng et al. (2019) take a different approach and use the retrieved exemplars to directly reparameterize the decoder for conditional text generation. Dou et al. (2021) propose a model for abstractive summarization that is designed to receive an additional guidance signal as input, and they experiment with using summaries retrieved from the training data as a form of guidance. Lin et al. (2020) introduce the style imitation task for data-to-text generation and propose a specialized architecture, training objective, and exemplar retrieval strategy to learn this task.

6 Conclusion

We presented an approach for controlling the writing style of general sequence-to-sequence models for data-to-text generation. We experimentally showed that our approach acquires strong generative abilities from state-of-the-art pretrained models while allowing for direct control over style. Motivating our approach is the real-world application of a system that interacts with users and is able to produce generations that are suited to their preferences.

Source:
Tyler Christopher (athlete)
Personal bests

Event	400 meters (indoor)
Time (seconds)	45.67 NR
Venue	Valencia, Spain
Date	March 9, 2008

Exemplar 1: **In May**, Günther Weidlinger **scored a time of** 27:36.46 in the 10,000 m in Palo Alto, California.
Generation 1: **In 2008**, Tyler Christopher **scored a national record time of** 45.67 seconds in the 400 m in Valencia, Spain.

Exemplar 2: Michael Phelps **competed in the** 100m Butterfly at the 2012 Summer Olympics in London**, where he won a** gold medal.
Generation 2: Christopher **competed in the** 400 m at the 2008 Olympics in Valencia, Spain**, where he recorded a** national record time of 45.67 seconds.

Exemplar 3: **In the 2013 season,** Steph Curry **set an NBA record for** most three-pointers made in a single season **with a total of** 272.
Generation 3: **In March 2008,** Christopher **set a national record for** the 400-meter indoor **with a time of** 45.67 seconds in Valencia, Spain.

Source:
Bowling Green, Kentucky
Climate

Climate data for Bowling Green, Kentucky (Warren County Airport), 1981-2010 normals	
	Year
Average precipitation inches (mm)	47.51 (1,207)
Average snowfall inches (cm)	8.4 (21.35)

Exemplar 1: **Average annual precipitation was** 45.73 inches (1,162 mm)**, averaging** 4.68 inches (119 mm)**, in May.**
Generation 1: **The average annual precipitation was** 47.51 inches (1,207 mm)**, averaging** 8.4 inches (21.35 cm)**, in Bowling Green.**

Exemplar 2: Atlanta International Airport **has an average** daily traffic of 260,000 passengers**, with an average of** 1000 flights **per day**.
Generation 2: Bowling Green **has an average** annual precipitation of 47.51 inches (1,207 mm)**, with an average of** 8.4 inches of snow **per year**.

Exemplar 3: **Facebook's average** monthly active users is 2.7 billion users**, while its average** revenue per user is $7.89.
Generation 3: **Bowling Green's average** annual precipitation is 47.51 inches (1,207 mm)**, while the average** annual precipitation is 8.4 inches (21.35) of snow.

Figure 3: Qualitative analysis of our model on two instances of the ToTTo development set. Notable style elements of the exemplars are highlighted **blue**, while corresponding changes in the model outputs are highlighted **orange**.

References

Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William Cohen. 2019. Handling divergent reference texts when evaluating table-to-text generation. In *Proceedings of the 57th*

Annual Meeting of the Association for Computational Linguistics, pages 4884–4895, Florence, Italy. Association for Computational Linguistics.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2021. Gsum: A general framework for guided neural abstractive summarization.

Kelvin Guu, Tatsunori B Hashimoto, Yonatan Oren, and Percy Liang. 2018. Generating sentences by editing prototypes. *Transactions of the Association for Computational Linguistics*, 6:437–450.

Junxian He, Taylor Berg-Kirkpatrick, and Graham Neubig. 2020. Learning sparse prototypes for text generation. *Advances in Neural Information Processing Systems*.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Mihir Kale and Abhinav Rastogi. 2020. Text-to-text pre-training for data-to-text tasks. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 97–102, Dublin, Ireland. Association for Computational Linguistics.

Guillaume Lample, Sandeep Subramanian, Eric Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-attribute text rewriting. In *International Conference on Learning Representations*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1203–1213, Austin, Texas. 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.

Shuai Lin, Wentao Wang, Zichao Yang, Xiaodan Liang, Frank F. Xu, Eric Xing, and Zhiting Hu. 2020. Data-to-text generation with style imitation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1589–1598, Online. Association for Computational Linguistics.

Hongyuan Mei, Mohit Bansal, and Matthew R. Walter. 2016. What to talk about and how? selective generation using LSTMs with coarse-to-fine alignment. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 720–730, San Diego, California. Association for Computational Linguistics.

Amit Moryossef, Yoav Goldberg, and Ido Dagan. 2019. Step-by-step: Separating planning from realization in neural data-to-text generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2267–2277, Minneapolis, Minnesota. Association for Computational Linguistics.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206, Saarbrücken, Germany. Association for Computational Linguistics.

Gaurav Pandey, Danish Contractor, Vineet Kumar, and Sachindra Joshi. 2018. Exemplar encoder-decoder for neural conversation generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1329–1338, Melbourne, Australia. Association for Computational Linguistics.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. ToTTo: A controlled table-to-text generation dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186, Online. Association for Computational Linguistics.

Hao Peng, Ankur Parikh, Manaal Faruqui, Bhuwan Dhingra, and Dipanjan Das. 2019. Text generation with exemplar-based adaptive decoding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2555–2565, Minneapolis, Minnesota. Association for Computational Linguistics.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with content selection and planning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6908–6915.

Alec Radford, Jeffrey Wu, R. 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. *Journal of Machine Learning Research*, 21(140):1–67.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Martin Schmitt, Hinrich Schütze, and Iryna Gurevych. 2020. Investigating pretrained language models for graph-to-text generation.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

Rong Ye, Wenxian Shi, Hao Zhou, Zhongyu Wei, and Lei Li. 2020. Variational template machine for data-to-text generation. In *International Conference on Learning Representations*.

Chao Zhao, Marilyn Walker, and Snigdha Chaturvedi. 2020. Bridging the structural gap between encoding and decoding for data-to-text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2481–2491, Online. Association for Computational Linguistics.

A Appendix

A.1 Ablations

Retrieval size. We vary the retrieval size k at training time and evaluate its effect on performance. For all sizes, we use the same training-time ranking function described in §3.2. Our automatic evaluation results for the Style-E2E and ToTTo datasets are shown in Tables 6 and 7 respectively. For both datasets, we see slight improvements in content fidelity and declines in style embodiment as the retrieval size increases. This can be explained by the fact that as more exemplars are retrieved per training instance with our ranking function, the less similar each retrieved exemplar is to the target on average, resulting in the model depending on the exemplar less. In this way, the retrieval size can be viewed as a hyperparameter to control the balance between remaining faithful to the source content and imitating the exemplar's style.

	Content		Style	Fluency
	Inc-New	**Exc-Old**	**m-BLEU**	**PPL**
$k = 1$	93.77%	96.57%	55.53	4.84
$k = 3$	94.91%	97.47%	50.34	4.10
$k = 5$	95.71%	98.21%	46.21	3.99
$k = 10$	96.36%	98.60%	43.54	3.58

Table 6: Automatic evaluation results on the Style-E2E development set of our approach for different exemplar retrieval sizes k.

	Content		Style	Fluency
	Precision	**Recall**	**m-BLEU**	**PPL**
$k = 1$	72.11	40.39	61.94	60
$k = 3$	72.74	41.38	60.59	61.46
$k = 5$	74.51	44.00	52.90	56.28
$k = 10$	74.37	44.81	47.43	53.35

Table 7: Automatic evaluation results on the ToTTo development set in the style-controlled setting of our approach for different exemplar retrieval sizes k.

Ranking function. We additionally evaluate different implementations of the ranking function r at training time. Let m_{cont} denote the content word masking function, d_{edit} the token-level edit distance function, and f_{enc} the Sentence-BERT encoder as described in §3.2. Then "Mask edit dist" denotes our original ranking function given in Equation 1:

$$r(y_1, y_2) = d_{edit}\big(m_{cont}(y_1), m_{cont}(y_2)\big)$$

"Edit dist" denotes the function

$$r(y_1, y_2) = d_{edit}(y_1, y_2)$$

"Mask L2" denotes the function

$$r(y_1, y_2) = \Big\| f_{enc}\big(m_{cont}(y_1)\big) - f_{enc}\big(m_{cont}(y_2)\big) \Big\|_2$$

"L2" denotes the function

$$r(y_1, y_2) = \Big\| f_{enc}(y_1) - f_{enc}(y_2) \Big\|_2$$

and "Source L2" denotes the function

$$r(x_1, x_2) = \Big\| f_{enc}(x_1) - f_{enc}(x_2) \Big\|_2$$

Note that the "Source L2" function compares the sources rather than the targets, and is the same function we use to retrieve exemplars at test time (Equation 2). We report automatic evaluation results for these strategies on the Style E2E and ToTTo datasets in Tables 8 and 9 respectively.

We observe that when using edit distance to compute similarity, masking content words results in a small increase to content fidelity while maintaining similar style embodiment. When using the L2 norm of the target embeddings, masking content words becomes critical to retrieving exemplars of similar style, as the sentence encoder f_{enc} is trained for semantic textual similarity rather than syntactic similarity. When retrieving exemplars based on the source rather than the target, we note that the model largely ignores the exemplars for ToTTo because they are not similar in style to the target, and while their content may be in the same domain as the target, they are not the same values. On the other hand, for the Style-E2E dataset we observe something very different. Because many of the sources share similar values, then retrieving exemplars with the most similar sources results in exemplars with content that matches the target. This leads to the model relying on the exemplar for content as well as style, leading to abysmal content fidelity at test time when the exemplar and target do not have matching content.

A.2 T5-Small + DTG-SI Investigation

We conduct a further investigation into why the T5-Small model with the DTG-SI approach learns to ignore the exemplars as observed in §4.5.1. We

	Content		Style	Fluency
	Inc-New	**Exc-Old**	**m-BLEU**	**PPL**
Mask edit dist	94.91%	97.47%	50.34	4.10
Edit dist	94.09%	96.46%	51.37	4.36
Mask L2	91.28%	95.03%	35.31	4.13
L2	78.46%	85.84%	24.27	3.38
Source L2	26.13%	15.79%	92.16	5.42

Table 8: Automatic evaluation results on the Style-E2E development set of our approach for different training-time ranking functions at retrieval size $k = 3$.

	Content		Style	Fluency
	Precision	**Recall**	**m-BLEU**	**PPL**
Mask edit dist	74.51	44.00	52.90	56.28
Edit dist	73.86	42.82	53.67	58.02
Mask L2	74.85	42.86	49.72	64.8
L2	80.51	49.71	30.94	57.48
Source L2	80.86	48.96	27.66	59.85

Table 9: Automatic evaluation results on the ToTTo development set in the style-controlled setting of our approach for different training-time ranking functions at retrieval size $k = 5$.

train the T5-Small model with their retrieval strategy and training objective for different values of λ, in addition to using only their retrieval strategy with the standard sequence-to-sequence training objective, and evaluate on the Style-E2E dataset.

Based on the results (Table 10), we observe that their training objective is dependent on their specialized architecture. When using a general seq2seq model such as T5, the model learns to optimize for the two objectives separately, resulting in a degenerate solution where the model either entirely ignores ($\lambda > 0$) or copies ($\lambda = 0$) the exemplar at test time. The other key observation is that their retrieval strategy is also dependent on their architecture and training objective: attempting to use it within the T5 architecture and standard cross-entropy loss results in the model ignoring the exemplars at test time. By contrast, our retrieval strategy is able to train general seq2seq models to balance between maintaining factual consistency and imitating the exemplar's style.

	Content		Style
	Inc-New	**Exc-Old**	**m-BLEU**
$\lambda = 0$	17.14%	6.01%	99.93
$\lambda = 0.01$	96.27%	97.20%	23.17
$\lambda = 0.1$	98.51%	98.73%	19.22
$\lambda = 1$	99.59%	99.92%	16.77
Retrieval only	99.91%	99.91%	16.24
Ours	94.91%	97.47%	50.34

Table 10: Automatic evaluation results on the Style-E2E development set of several implementations of the T5-Small + DTG-SI method compared to our approach. λ indicates the balancing weight used for the DTG-SI joint training objective, while "Retrieval only" denotes training with just their retrieval strategy and the standard seq2seq objective. All approaches use the same pretrained T5-Small model.