FAST: Improving Controllability for Text Generation with *F*eedback *A*ware *S*elf-*T*raining

Junyi Chai, Reid Pryzant, Victor Ye Dong, Konstantin Golobokov, Chenguang Zhu, Yi Liu
Microsoft Corporation
juchai,reidpryzant,victordong,kogolobo,chezhu,lewisliu@microsoft.com

Abstract

Controllable text generation systems often leverage *control codes* to direct various properties of the output like style and length. Inspired by recent work on causal inference for NLP, this paper reveals a previously overlooked flaw in these control code-based conditional text generation algorithms. Spurious correlations in the training data can lead models to incorrectly rely on parts of the input other than the control code for attribute selection, significantly undermining downstream generation quality and controllability. We demonstrate the severity of this issue with a series of case studies and then propose two simple techniques to reduce these correlations in training sets. The first technique is based on resampling the data according to an example's propensity towards each linguistic attribute (IPS). The second produces multiple counterfactual versions of each example and then uses an additional feedback mechanism to remove noisy examples (feedback aware self-training, FAST). We evaluate on 3 tasks – news headline, meta review, and search ads generation – and demonstrate that FAST can significantly improve the controllability and language quality of generated outputs when compared to state-of-the-art controllable text generation approaches.

1 Introduction

In neural text generation, there is a growing interest in controlling the presence of particular linguistic attributes in the output text, for example sentiment, length, politeness, and topic (Sennrich et al., 2016; Kikuchi et al., 2016; Ficler and Goldberg, 2017; Shen et al., 2022). This is typically accomplished via *control codes*: categorical variables that represent the desired output property and are pre-pended to the model inputs during training and testing (Keskar et al., 2019).

This paper builds on recent work in text-based causal inference (Feder et al., 2021; Veitch et al., 2021; Pryzant et al., 2021) to reveal a previously overlooked flaw in control code-based text generation systems: spurious correlations in the data can cause models to incorrectly rely on parts of the input *other* than the control code for attribute selection, undermining downstream generation performance.

For example, consider a system that generates news headlines while conditioning on article text and a control code for headline length (e.g. long for desktop, short for mobile) as in Murao et al. (2019). We show in §4.1 that among publicly available news datasets, correlations exist between the contents of an article and the length of that article's title. Longer articles or articles about technical topics may be associated with longer titles. This leads NLP models to struggle at generating short titles from "long title"-looking articles.

We show how this phenomenon can introduce confounding statistical relationships in the data, leading to assumption violations and significantly degrading model quality. Then we propose two simple data augmentation techniques for improving the issue. Both algorithms operate by breaking these spurious correlations and isolating the statistical relationship between control codes and linguistic attributes. In the first approach, we resample the training set according to an inverse propensity score (IPS, Robins et al. (1994)), boosting the presence of rare context-attribute combinations in the data. In the second approach (FAST) we train a preliminary model, use counterfactual data augmentation to generate all possible attributes for each example, then retrain on the counterfactually balanced dataset, as illustrated in Figure 1.

We conduct experiments in 3 conditional text generation scenarios: generating news headlines from article contents (controlling the headline lengths), generating the next sentence from preceding sentences (controlling the intent), and generating search ad copy from landing pages (controlling the rhetorical appeal of the ad). Our results

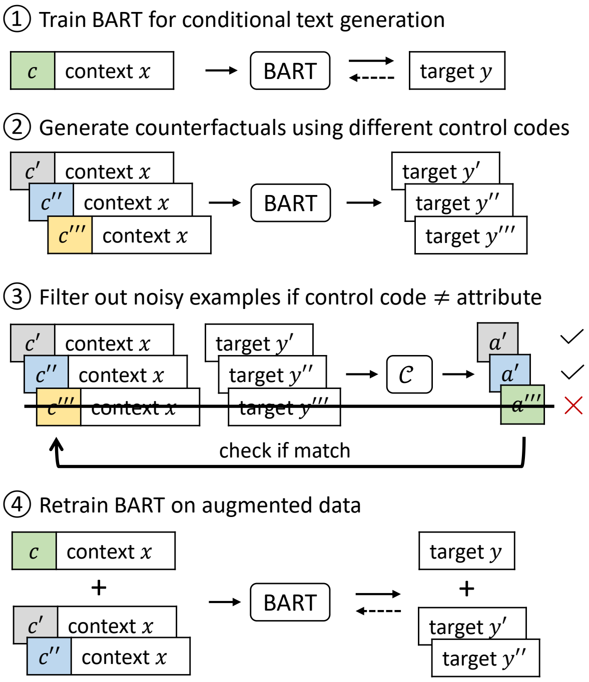


Figure 1: Illustration of FAST algorithm.

suggest that FAST can significantly improve the controllability and language quality of state-of-the-art controllable generation systems.

In summary, our contributions are:

- Identifying an important flaw with recent controllable text generation techniques and showing how this flaw can undermine model performance.
- A pair of simple yet effective data augmentation algorithms for dealing with this issue.
- Results and analysis demonstrating the efficacy of the proposed algorithms, and their ability to significantly improve controllability and language quality over state-of-the-art baselines.

2 Spuriously Correlated Control Codes

2.1 Controllable Generation

We focus on the case of *conditional* text generation, where the training data $D_{tr} = \{(x_1, y_1, a_1), ..., (x_n, y_n, a_n)\}$ is collection of triples consisting of context text x, output text y, and output linguistic attribute a. Note that in many practical scenarios, a is inferred from y by a classifier $\mathcal{C}(y)$, e.g. a rule or deep learning model. The goal is to learn a conditional language model (CLM) for $p(y|x, a)$, i.e. a text generation system which conditions on the context and attribute to generate texts that express the desired linguistic attribute.

In practice, the linguistic attributes a are operationalized as control code tokens c which are in one-to-one correspondence with the attributes (e.g. “short”, “long”) and pre-pended to the context x before model input. This approach has been shown to be effective in both non-conditional (Keskar et al., 2019; Ficler and Goldberg, 2017) and conditional (Shen et al., 2022; Fan et al., 2018) controllable text generation.

2.2 Spurious Correlations

In theory, the correspondence between the control code c and linguistic attribute a should cause models to rely on the control code to determine the linguistic properties of the generated output. This paper argues that in practice, parts of the context x may be spuriously correlated with the attribute a, undermining the consistency and efficacy of control code-based systems.

These spurious correlations between the the contexts and attributes have a causal interpretation that explains how they can undermine model performance. The issue is that $p(a|x) \neq p(a)$, which is similar to a violation of the ignorability assumption in causal inference (Feder et al., 2021). This implies that any spurious correlations between the context x and target attribute a could represent backdoor paths that confound the model’s learned relationship between the control code c and the target attributes a. Thus, models are likely to depend on context *beyond* the control code when determining output attributes, making them less likely to generalize to rare context/control-code combinations.

In this paper, we aim to break up these backdoor paths and prevent the model from learning spurious correlations. We accomplish this by modifying the training data in two ways such that $p(a|x) \approx p(a)$, with both techniques isolating the relationship between the control codes and target attributes.

2.3 Inverse propensity score (IPS) resampling

The first method we investigate for breaking the aforementioned spurious correlations leverages propensity scores. A propensity score is the conditional probability of an example being assigned to a treatment, given background variables (Rosenbaum and Rubin, 1983). It plays a central role in causal inference for dealing with spurious correlations in observational data and therefore it is a natural choice for us to try. In our case, the propensity score for the ith example is the conditional probability of the output text exhibiting linguistic attribute a_i given the context x_i. This can be written as

$$w_i = p(a = a_i | x_i).$$

Intuitively, examples with low propensity scores represent rare attribute-context combinations that are especially important to learn (Tu et al., 2020). Therefore, our procedure works by resampling the data with replacement, setting the sample weight of the ith example to $1/w_i$. The procedure should work because the propensity scores of the resampled data should be close to uniform: $p(a_i|x_i) \propto w_i/w_i$.

In practice, we train a model to estimate propensity scores. For the experiments we finetune Roberta (Liu et al., 2019) as a sequence classifier using $\{(x_1, a_1), ..., (x_n, a_n)\}$. We then use the model's probability prediction for the observed category a_i as the propensity score estimate. We will refer to this estimator as $\mathcal{S}(a|x)$.

2.4 Feedback aware self-training (FAST)

The above IPS resampling procedure has several shortcomings, including the duplication of examples (Lee et al., 2021; Carlini et al., 2021) and the noise/bias inherent to estimated propensity scores (Pearl, 2009). Therefore our second method, though originating with the same motivation and tackling the same issue, takes an orthogonal approach. First, use a separately trained model to produce multiple counterfactual target sequences for each context. Next, filter the data such that new target sequence expresses a different linguistic attribute. Then, we retrain on the new counterfactually balanced dataset. In detail, the steps are:

1. Train a conditional language model (CLM) using the standard control code approach on D_{tr}, which is denoted as CLM$_{baseline}$.
2. Use CLM$_{baseline}$ to generate multiple outputs for each context x_i, one output for every control code except that which corresponds to the ground truth attribute. For example, the set of control codes used for datum i would be $\{\forall c \in \{1, ..., K\}, c \neq a_i\}$.
3. Detect the linguistic attribute of the generation outputs with a classifier $\mathcal{C}$, and filter out examples where the predicted attribute does not match the inputted control code.
4. Augment the original training set D_{tr} with samples from Step 3 and retrain.

Intuitively, this procedure should also drive the propensity scores of the data towards uniform and break the unwanted correlations between contexts and attributes, since every context becomes paired with multiple targets, each having a unique attribute. Step 3 uses feedback from the classifier $\mathcal{C}$ to remove noisy examples, preventing errors from propagating into the final model (§4.3). We experiment with classifiers $\mathcal{C}$ that are given a prior, trained on (y, a) pairs from the training data D_{tr}, and trained on a separate dataset having similar properties.

3 Experimental Setup

We perform experiments in 3 important controllable generation settings: generating news headlines from article contents (controlling the headline lengths), generating the next sentence of a meta-review from preceding sentences and additional context (controlling the intent), and generating search ad copy from landing pages (controlling the rhetorical appeal of the ad). Our results suggest that the proposed methods can significantly improve the controllability and fluency of state-of-the-art baselines.

3.1 Datasets

We experiment using 3 datasets (Table 1) that reflect important real-world application scenarios for controllable generation systems.

First, we use the **PENS** dataset released by Microsoft News (Ao et al., 2021). This task involves generating news headlines from news articles, while using a binary control code "short" or "long" to control the length of the generated headline (useful for mobile and desktop rendering). We use a length threshold of 55 to determine the long/short status of existing headlines in the data. We evaluate on these data using (1) random train/dev/test splits, and (2) a "balanced" test set. There are equal numbers of long and short headlines per article in this balanced test set. The headlines were sourced from 103 college students who wrote long or short headlines without seeing the original headlines, for an average of 3.7 headlines per article.

Second, we use the **MReD** dataset released by Shen et al. (2022). It consists of 4 years of ICLR meta reviews with each sentence being manually

PENS

Category	train	dev	test rnd.	test bal.
Short	31,245	3,614	4,001	5,509
Long	57,351	6,666	7,074	5,509
Total	88,596	10,280	10,240	11,018

MReD

Category	train	dev	test
Weakness	1,491	200	200
Strength	757	200	200
Decision	716	200	200
Rebuttal process	674	200	200
Abstract	581	200	200
Suggestion	438	200	200
Rating summary	338	159	135
Misc	225	143	150
AC disagreement	24	18	18
Total	5,244	1,520	1,503

Search Ads

Category	train	dev	test
Product or Service	1,771	44.6	43.1
Call to action	1,207	37.5	36.6
Location	931	22	21.4
Highlight	851	32	30.8
Inventory	590	19	15.7
Brand name	466	11.9	11
Price	367	21.1	18.1
Benefit	309	8.6	8.6
Customer problem	156	3.7	3.9
Total	6,649	200.5	189.2

Table 1: Summary of PENS (top), MReD (middle) and Search Ads datasets (bottom, in thousands).

annotated into one of 9 categories. Using these data, our task is to follow the assisted writing scenario of Chen et al. (2019). We generate the i^{th} sentence in the meta-review, controlling the intent of the generated sentence and conditioning on all preceding sentences and additional context (ratings, individual reviews). We reuse the original train/dev/test splits and randomly sample sentences with at least 4 words as the target sequences. For the training set, we pick one sentence per review. For dev and test sets, we pick multiple sentences per review while ensuring a nearly equal number of samples per category. To detect the categories of generated sentences, we train a Roberta-base classifier on 37,252 sentences (a superset of our generation training set), achieving a macro-F1 of 79% on hold-out test set, implying that it has strong generalization capabilities.

Finally, we use a **Search Ads** dataset consisting of landing pages, search advertisements for those landing pages, and labels for those ads classifying them into one of 9 common advertising strategies. Here, the goal is to generate search ads (title and description) from landing pages while controlling the rhetorical appeal of the ad copy (Golobokov et al., 2022). To obtain the category labels, we apply a BERT-base-uncased model (Devlin et al., 2019) trained on a separate dataset of 5,735 manually labeled ad-category pairs. This model achieves a macro-F1 score of 70% on hold-out test set. Unlike the PENS data, the Search Ads data do not contain a balanced test set. However, the train, dev and test splits for the ads data contain an average of 1.9, 2.3 and 2.6 ads from different categories per landing page, respectively, so there is a moderate degree of category depth.

3.2 Baselines

We compare against five baselines: an uncontrolled system to establish a lower bound on performance, and four recently published neural controllable generation systems.

Uncontrolled We train BART-base (Lewis et al., 2020) for uncontrolled generation, where the model is only conditioned on the context.

BART+CTRL We train BART-base for controllable generation using the standard control code approach (Keskar et al., 2019). The control code is represented as the name of the category (“long”, “price”, etc). The paragraph symbol § is used as delimiter to separate control code and context.

PPLM We aim to enhance controllability of BART+CTRL by further steering its decoding towards the desired attribute. PPLM achieves this by using gradients from an attribute classifier $p(a|y)$ to update the CLM’s hidden representations (Dathathri et al., 2020).

GeDi This is a state-of-the-art technique for controlling open-ended and non-conditional generation (Krause et al., 2021). We adapt its weighted decoding formula to our *conditional* generation setting by including a dependency on the context x:

$$p_w(y|x, c) \propto p(y|x)p(c|y)^{\omega}. \tag{1}$$

The key insight from GeDi is to compute $p(c|y)$ using Bayes rule (i.e., leveraging $p(y|c)$). We train two BART-base models for $p(y|x)$ and $p(y|c)$ using the same procedure as the BART+CTRL baseline. We pick $\omega = 4$ for PENS and MReD, and $\omega = 3.5$ for Ads based on a brief hyperparameter search.

GeDi+x Our last baseline involves further adapting GeDi to our application domain by conditioning everything on the context x as well as the control code c, i.e. we concatenate the control code c and the context x when training BART+CTRL models. The new decoding formula is

$$p_w(y|x, c) \propto p(y|x, c)p(c|y)^{\omega}, \quad (2)$$

and the Bayes approximation of $p(c|y)$ is

$$p(c|y) = \frac{p(c|x)p(y|x, c)}{\sum_{c'} p(c'|x)p(y|x, c')}, \quad (3)$$

where $p(c|x)$ is further dropped as it does not depend on y. We pick $\omega = 1$ for PENS and $\omega = 0.5$ for both MReD and Ads based on a brief hyperparameter search. Details of the above methods are in Appendix.

3.3 Protocol

Our implementation is largely based on Huggingface Transformers (Wolf et al., 2020) except replacing beam search with a more efficient implementation (Yan et al., 2021). We use `BART-base` and `Roberta-base` pretrained models to better emulate real-world production scenarios where smaller, more efficient models are favored. We use beam search with beam size 5 when decoding (except for PPLM, which uses greedy decoding). In all experiments, we train with 2 Nvidia V100 GPUs and inference with 1 GPU, both at fp16 precision.

For BART training, we optimize all models using Adamw (Loshchilov and Hutter, 2017) and a learning rate of 1e-5 for PENS and 5e-5 for MReD and Ads datasets. We do not explicitly tune other hyperparameters. We train models, evaluating on the dev sets every epoch until the validation score begins to decrease. Then we pick the best-performing epoch based on ROUGE-1 with the dev set. All experiments are repeated with 5 random seeds when we report 95% confidence intervals from a t-distribution. We consider $p < 0.05$ to be statistically significant. More details are in the Appendix.

4 Experiments

4.1 Spurious Correlations

We begin by empirically demonstrating the existence of spurious correlations that can degrade downstream model quality, and show how our algorithms reduce these correlations in the data. We show these trends via a series of case studies on the PENS news dataset. Similar studies on MReD and Search Ads datasets are in the Appendix.

In Section 2.2, we defined the spurious correlation issue as unwanted dependencies between the context and attribute: $p(a|x) \neq p(a)$. To reveal this property in the PENS dataset we finetune Roberta-base with a binary classification head to predict the attribute (long or short) from the context (news article), which also serves as $\mathcal{S}(a|x)$ in §2.3. The model achieved an accuracy of 73% on a hold-out test set, far better than random guessing (50%) and the majority class (64%), empirically confirming that $p(a|x) \neq p(a)$ and the context x is strongly correlated with attributes a.

Method	PENS	MReD	Ads
random guessing	50	11	11
original	80	60	45
IPS	52	18	11
FAST	59	15	24

Table 2: Accuracy of predicting attribute of output from context on the original training set, as well as IPS resampled and FAST augmented training sets.

Next, we identify two sources of spurious correlation in the data. First, the length of a news article is positively correlated with its long/short headline status (point-biserial correlation r_{pb} = 0.1, $p < 0.01$). Second, we find that certain words and phrases can be inappropriately correlated with the attributes. We train an l2-regularized logistic regression on the same task and data using bag-of-words features, then examine the features having the highest weights. The features most indicative of short headlines include niche topics like "petfinder", "cartoonist's homepage", and "saildrone" (a weather service) while words from established outlets that cover more general topics ("usa today", "cbsnewyork") are associated with long headlines.

To show how spurious correlations can undermine downstream controllable generation performance, we train a BART-base model on the PENS dataset using the standard control code approach (BART+CTRL baseline). Next, for each article in the randomly split test set, we generate using all possible control codes (long, short) and score the outputs according to whether they are truly long or short. The system successfully generated the intended headline 89.6±0.6% of the time in factual cases (when the control code matched the ground-truth attribute), but 64.1%±0.6% of the time in counterfactual cases. This suggests that models learn to rely on spurious correlations in the data, and this reliance can undermine generalization.

We proceed to show how our data augmentation

	PENS				MReD				Search Ads			
Method	R1	R2	RL	Acc	R1	R2	RL	Acc	R1	R2	RL	Acc
Uncontrolled	32.1±0.2	13.2±0.1	26.7±0.1	–	18.7±0.5	4.1±0.2	16.1±0.4	–	22.9±0.1	9.2±0.1	21.5±0.1	–
BART+CTRL	**32.6**±0.1	**13.4**±0.1	**27.1**±0.1	78.0±0.7	21.4±0.2	5.6±0.2	18.4±0.2	76.5±1.9	27.8±0.1	11.9±0.1	26.2±0.1	68.1±1.2
PPLM	29.5±0.1	10.8±0.1	24.4±0.1	76.3±0.6	**21.9**±0.2	5.1±0.2	18.4±0.1	74.7±1.1	27.0±0.2	10.5±0.2	25.3±0.2	69.3±1.1
GeDi	31.7±0.1	12.6±0.1	26.2±0.1	78.5±0.6	17.3±0.6	3.7±0.4	15.3±0.5	74.9±5.3	23.2±0.6	8.2±0.4	21.9±0.5	**83.3**±4.6
GeDi+x	32.5±0.1	13.3±0.1	27.0±0.1	**82.7**±0.6	19.9±0.4	4.9±0.2	17.3±0.3	83.7±0.6	27.7±0.2	11.8±0.1	26.1±0.2	77.9±0.9
IPS	32.3±0.03	13.2±0.1	26.9±0.04	79.0±0.5	21.1±0.4	5.2±0.3	17.8±0.4	71.2±6.8	27.4±0.1	11.6±0.04	25.8±0.1	70.1±1.4
FAST	**32.5**±0.1	**13.4**±0.1	**27.1**±0.1	**82.5**±0.5	**21.9**±0.3	**6.1**±0.2	**18.9**±0.3	**87.1**±0.7	**28.1**±0.1	**12.3**±0.1	**26.5**±0.1	**80.5**±0.2
weak classifier	–	–	–	–	**21.8**±0.1	**6.0**±0.1	**18.9**±0.1	86.1±0.5	**28.0**±0.2	**12.1**±0.1	**26.4**±0.1	76.6±0.7

Table 3: Comparing different methods on PENS, MReD and Search Ads. We use ROUGE (R1, R2, and RL) to evaluate decoding quality on (1) the "balanced" PENS test set, (2) the default MReD test set and (3) the default Ads test set. We use Acc to evaluate controllability. This metric is calculated by generating using every control code, detecting the attribute category from the generated texts with task-specific classifiers or rules, then comparing the detected category with the control code to compute an accuracy. The best score is boldfaced. Multiple scores are boldfaced if there is no statistically significant difference. The last row is an ablation experiment for FAST using a weak classifier (logistic regression) in the feedback step.

algorithms produce datasets where these spurious correlations have been reduced in the data. We test the classifier predicting attributes from contexts on the original training set, as well as on the IPS resampled and FAST augmented training sets (Table 2). We find that the accuracy from the classifier is reduced greatly on the new training sets, implying that predictions become closer to random guessing as the spurious correlation is reduced and therefore $p(a|x) \rightarrow p(a)$ in the augmented data.

4.2 Overall Generation Results

We proceed to evaluate the impact of our algorithms on downstream controllable generation performance.

Table 3 shows results using automatic evaluation metrics on the PENS, MReD and Search Ads datasets. One might hypothesize that FAST may outperform IPS resampling because 1) FAST does not create duplicate examples, and 2) there is no need to estimate propensity scores directly. Our results support this, as the proposed FAST algorithm always outperforms all baselines in either language quality or controllability or both, especially on MReD and Ads datasets. Among the 3 baselines, GeDi has much lower ROUGE, possibly because it only combines context and control code at decoding time, while the other methods encode context and control code together. GeDi+x improves controllability over BART+CTRL significantly, but slightly decreases ROUGE. FAST improves controllability over BART+CTRL to a similar degree, while maintaining ROUGE on PENS and even improving ROUGE on MReD and Ads. On the other hand, IPS improves controllability slightly over BART+CTRL on two datasets PENS and Ads, which suggests that IPS resampling is helpful in preventing the model from learning the spurious correlation. However, it hurts ROUGE on all datasets, which is likely due to the duplication of data (§4.3). It appears surprising that PPLM can have lower controllability than BART+CTRL even though it applies additional steering during decoding. This is because PPLM uses greedy decoding. For example on MReD, switching from beam search to greedy decoding, the control accuracy of BART+CTRL decreases from 76.5% to 70%. PPLM improves it to 74.7%, but it is still behind BART+CTRL with beam search.

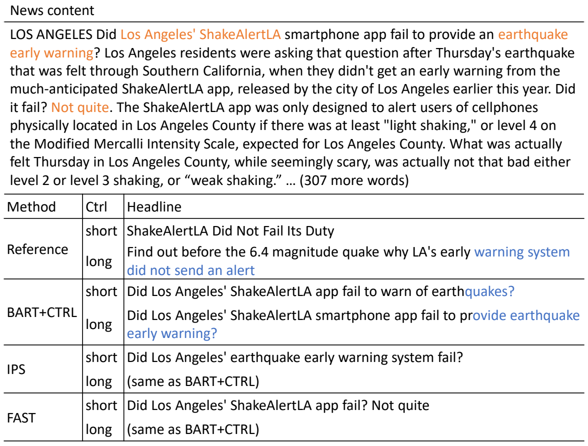
We proceed to conduct a human evaluation of downstream generation quality for the Search Ads dataset (Table 4). We compare our IPS and FAST methods against BART+CTRL, omitting the PPLM and GeDi baselines because they are prohibitively expensive for many real-world applications.

We use the models to generate ad copy in all 9 categories for each landing page, then present each generation to a panel of five professional judges. The judges evaluated each example in 4 aspects: whether the text is grammatically fluent ("language quality"), whether it was human-like and realistic, whether it was factual, and whether it is relevant to the landing page. Each aspect was rated according to a binary good/bad scale, then we report the average rating. Judges also categorized the generation into one of the 9 attribute categories summarized in Table 1, which we converted into a measure of model controllability by calculating the accuracy between the input control codes and human labeled output categories. More details can be found in the Appendix.

The human evaluation results are consistent with automatic evaluation metrics. The proposed FAST

Method	Language	Human-like	Factuality	Relevance	Overall	Acc
BART+CTRL	93.5±0.7	99.3±0.2	99.4±0.2	99.7±0.2	92.8±0.7	52.1±2.3
IPS	92.4±0.8	99.1±0.3	99.4±0.2	99.6±0.2	91.7±0.8	51.7±2.3
FAST	**94.9**±0.6	**99.6**±0.2	99.2±0.3	99.7±0.2	**94.1**±0.7	**58.4**±2.3

Table 4: Human evaluation results for Search Ads generation. The quality of each aspect (e.g., language) is measured by the percentage of samples in the good level. The overall quality is the percentage of samples with all 4 aspects in the good level. Controllability is measured by the accuracy between human labeled categories of the ads vs the control codes used during their generation. The best scores with statistical significance (under Z-test) is boldfaced.



News content

LOS ANGELES Did Los Angeles' ShakeAlertLA smartphone app fail to provide an earthquake early warning? Los Angeles residents were asking that question after Thursday's earthquake that was felt through Southern California, when they didn't get an early warning from the much-anticipated ShakeAlertLA app, released by the city of Los Angeles earlier this year. Did it fail? Not quite. The ShakeAlertLA app was only designed to alert users of cellphones physically located in Los Angeles County if there was at least "light shaking," or level 4 on the Modified Mercalli Intensity Scale, expected for Los Angeles County. What was actually felt Thursday in Los Angeles County, while seemingly scary, was actually not that bad either level 2 or level 3 shaking, or "weak shaking." … (307 more words)

Method	Ctrl	Headline
Reference	short	ShakeAlertLA Did Not Fail Its Duty
	long	Find out before the 6.4 magnitude quake why LA's early warning system did not send an alert
BART+CTRL	short	Did Los Angeles' ShakeAlertLA app fail to warn of earthquakes?
	long	Did Los Angeles' ShakeAlertLA smartphone app fail to provide earthquake early warning?
IPS	short	Did Los Angeles' earthquake early warning system fail?
	long	(same as BART+CTRL)
FAST	short	Did Los Angeles' ShakeAlertLA app fail? Not quite
	long	(same as BART+CTRL)

Figure 2: An example from PENS dataset for news headline generation. Reference headlines from the balanced test set and generated headlines from three methods corresponding to the same news content are shown. Key news information that the generated headlines captured is highlighted in orange. Parts of the headline exceeding 55 characters are highlighted in blue.

outperforms the BART+CTRL baseline in both language quality and controllability, whereas the proposed IPS algorithm underperforms its baseline in terms of language quality while there is no perceptible difference in controllability.

4.3 Analysis

Qualitative evaluation Figure 2 shows example news headlines generated from three methods. While the BART+CTRL baseline failed to generate a short enough headline, generations from IPS and FAST fit within the 55 character length limit with FAST being the most concise. More comparative input-output examples for MReD and Ads datasets can be found in the Appendix.

No Feedback ablation To see the importance of feedback in Step 3 of FAST algorithm (§2.4), we train a model without this step, i.e., self-training alone. Compared with FAST results in Table 3, the attribute/control-code accuracy drops by 9%, 7% and 12% (absolute difference) on PENS, MReD and Ads respectively. On PENS, the control accuracy of self-training alone is even lower than the BART+CTRL baseline (73% vs 78%). In regular FAST training, 38%, 25% and 40% of generated counterfactual samples are filtered out in the feedback step for PENS, MReD and Ads. It is not a surprise that including such a high percentage of noisy samples, whose attributes contradict with the control codes, would hurt controllability.

Classifier accuracy ablation We further study how the generation performance depends on the classifier accuracy. We replace the strong classifiers (Roberta-base for MReD and BERT-base-uncased for Search Ads) with weaker classifiers: l2-regularized logistic regression using bag-of-words features. On the hold-out test sets, the macro-F1 of the classifier drops from 79% to 66% for MReD and from 70% to 58% for Search Ads. In the feedback step, we filter out noisy examples using these weak classifiers and retrain the FAST models. Note that the controllability is evaluated in the same way with the strong classifiers. As shown in Table 3, the control accuracy of the ablation experiment (FAST with weak classifiers) drops from regular FAST, but it is still much higher than the BART+CTRL baseline. On the other hand, there is no statistically significant difference in ROUGE scores between ablation and regular FAST. This demonstrates that the improvement of FAST over baselines is fairly robust to the classifier accuracy.

IPS sampling mechanism In Table 3 we observed that IPS can reduce decoding quality (ROUGE score). We hypothesize that this may be because our procedure resamples the data with replacement, and the duplication of training samples can lead NLG models to memorize instead of generalizing (Feng et al., 2021). To reduce the spurious correlation while not duplicating examples, we subsample 10k of training set according to IPS without replacement so each training sample is unique. For a fair comparison, we uniformly ran-

Method	R1	R2	RL	Acc
BART+CTRL	31.4±0.2	12.7±0.2	26.3±0.2	71.9±1.1
IPS	31.4±0.2	12.6±0.1	26.3±0.1	**74.2**±0.5

Table 5: IPS subsampling ablation on PENS dataset. Best score with statistical significance is boldfaced.

domly subsample 10k for BART+CTRL training as the baseline. The results for the IPS subsampling and BART+CTRL are shown in Table 5. Now ROUGE scores between the two are close with no statistically significant difference. In addition, IPS improves controllability by 2.3% from 71.9% accuracy to 74.2%, which is greater than its effect in oversampling experiments (1% improvement from 78% to 79%). This demonstrates that rare examples are indeed more useful, especially if they are not duplicated.

5 Related Work

Controllable generation There are two main approaches for controllable generation: training or decoding-time steering. In the first approach, a language model is trained conditioned on the target attribute (Ficler and Goldberg, 2017), which can be conveniently encoded as control codes (Keskar et al., 2019). This approach has been used in many conditional generation tasks for controlling the length or content of abstractive summarization (Kikuchi et al., 2016; Fan et al., 2018; Liu et al., 2018), style of dialog response (See et al., 2019), ending of a story (happy or sad, conditioning on previous part of the story) (Peng et al., 2018), politeness of translation (Sennrich et al., 2016), intent of meta review (conditioning on individual reviews and ratings) (Shen et al., 2022). As the training sets are usually collected through observation rather than intervention, we anticipate there exist shared confounders influencing both context and target attribute, causing the spurious correlation to be a pervasive problem. In the second approach, recent methods are PPLM (Dathathri et al., 2020), GeDi (Krause et al., 2021), FUDGE (Yang and Klein, 2021) and DExperts (Yang and Klein, 2021). However, these methods exert control at the expense of fluency, a problem improved by Gu et al. (2022) but not completely eliminated. In contrast, our FAST method does not suffer from this problem.

Causal inference Our IPS resampling method is motivated by causal inference. Feder et al. (2021) review causal inference in natural language processing and suggest to use causal knowledge to formalize spurious correlations and to mitigate predictor reliance on them. Hu and Li (2021) devise a structural causal model (SCM) for controllable generation, where the output text is the outcome and the attribute under control is the treatment (whether to write a short or long headline). To proceed with causal inference, there is a common challenge that observational (training) data is under selection bias as the treatment choice is affected by some confounders (context). A classical solution is IPS reweighting or resampling (Yao et al., 2021; Pearl, 2009). An et al. (2021) suggests that resampling works better with stochastic gradient descents than reweighting so we choose resampling to investigate primarily. Outside traditional causal inference areas, IPS reweighting is successfully applied in search ranking (Wang et al., 2016) and recommendation systems (Schnabel et al., 2016). Recently, it is applied to reduce social bias in text classification tasks (Han et al., 2021). Hu and Li (2021) also propose to use it to debias pretrained language models.

Counterfactual data augmentation (CAD) Our method FAST is a special case of CAD. Lu et al. (2018) propose CAD and generate synthetic examples to reduce the spurious correlations between gendered and gender-neutral words in training corpus. Similar rule-based techniques are most common for CAD. Zhao et al. (2018) build rules with crowd-sourced annotation to swap all male entities for female entities; Sharma et al. (2021) swap gender terms with a dictionary similar to Lu et al. (2018); Garg et al. (2019) swap identity terms (e.g., gay, straight). Counterfactual examples can also be generated by manual post editing (Kaushik et al., 2020; Gardner et al., 2020) or automated text rewriting (Zmigrod et al., 2019; Riley et al., 2021; Wu et al., 2021). Using the same model to augment data (self-training) is a common semi-supervised algorithm for improving classifier accuracy (Zhang et al., 2022), though less common for CAD. Most similar to our method FAST is Gu et al. (2019) for improving zero-shot neural machine translation via reducing the spurious correlation between the language of the output and the source sentence. They first train a model on the original data and use it to generate data in missing language pairs. A key difference is that our FAST method uses feedback, which is shown to be crucial in our scenario possibly because our spurious correlation is less severe.

6 Conclusion

This paper argues that conditional and controllable text generation systems are subject to spurious correlations in their training data which can severely undermine performance. We proposed a pair of simple yet effective data augmentation algorithms for countering this issue. One algorithm works by resampling the data according to an inverse propensity score, and the other via feedback-aware self training. Our experiments demonstrate that the proposed algorithms can effectively reduce the spurious correlation issue across three tasks: generating ad copy, news headlines, and meta-reviews. Furthermore these algorithms can significantly improve generation quality and controllability over popular and state-of-the-art baseline algorithms.

Further research may investigate more checks during the feedback step, e.g., filtering out unfaithful examples. In the emerging parameter efficient fine-tuning paradigm, such as P*-tuning (Li and Liang, 2021; Qian et al., 2022), we find IPS resampling to be promising as the model is not likely to memorize duplicate examples when updating only few parameters. The proposed method may also be complementary to baselines like GeDi and PPLM.

7 Limitations

While IPS is a classical technique from causal inference to deal with spurious correlations, we found the following limitations when applying it to controllable text generation, which makes it less effective than FAST. First, Tu et al. (2020) found that large pretrained models are quite efficient in learning small amounts of counterfactual examples, which makes them more robust to spurious correlations. Our IPS resampling makes the small amounts of counterfactual examples more important to learn, but may have limited impact on the large pretrained models. Second, for MReD and Search Ads, the human-labeled or classifier-detected categories could be wrong. These examples are likely to have low propensity scores and therefore get up-sampled by IPS method. Finally, some unique training examples are dropped after resampling.

For FAST method, we acknowledge two limitations. First, an implicit assumption is that the linguistic attribute of interest (headline being short or long) should be independent of the context, therefore, a control code is applicable for any context. We design ad categories with this consideration in mind. However, in MReD dataset, categories such as "rebuttal process" may not be applicable for every meta review. Forcing a model to produce a sentence in such categories may result in untruthful generations. Second, FAST may struggle if the training and pre-training data are drastically different; the counterfactual generations may be of low quality and propagate errors into the final FAST model. How to generate counterfactual data in those more challenging scenarios would be our future research direction.

References

Jing An, Lexing Ying, and Yuhua Zhu. 2021. Why resampling outperforms reweighting for correcting sampling bias with stochastic gradients. In *ICLR*.

Xiang Ao, Xiting Wang, Ling Luo, Ying Qiao, Qing He, and Xing Xie. 2021. PENS: A Dataset and Generic Framework for Personalized News Headline Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 82–92, Stroudsburg, PA, USA. Association for Computational Linguistics.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting training data from large language models. *Proceedings of the 30th USENIX Security Symposium*, pages 2633–2650.

Mia Xu Chen, Benjamin N. Lee, Gagan Bansal, Yuan Cao, Shuyuan Zhang, Justin Lu, Jackie Tsay, Yinan Wang, Andrew M. Dai, Zhifeng Chen, Timothy Sohn, and Yonghui Wu. 2019. Gmail Smart Compose. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2287–2295, New York, NY, USA. ACM.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and Play Language Models: A Simple Approach to Controlled Text Generation. In *ICLR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North*, pages 4171–4186, Stroudsburg, PA, USA. Association for Computational Linguistics.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable Abstractive Summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, 2017, pages 45–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

Amir Feder, Katherine A. Keith, Emaad Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E. Roberts, Brandon M. Stewart, Victor Veitch, and Diyi Yang. 2021. Causal Inference in Natural Language Processing: Estimation, Prediction, Interpretation and Beyond.

Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for nlp. *arXiv preprint arXiv:2105.03075*.

Jessica Ficler and Yoav Goldberg. 2017. Controlling Linguistic Style Aspects in Neural Language Generation. In *Proceedings of the Workshop on Stylistic Variation*, section 3, pages 94–104, Stroudsburg, PA, USA. Association for Computational Linguistics.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating Models' Local Decision Boundaries via Contrast Sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sahaj Garg, Vincent Perot, Nicole Limtiaco, Ankur Taly, Ed H Chi, and Alex Beutel. 2019. Counterfactual Fairness in Text Classification through Robustness. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 219–226, New York, NY, USA. ACM.

Konstantin Golobokov, Junyi Chai, Victor Ye Dong, Mandy Gu, Bingyu Chi, Jie Cao, Yulan Yan, and Yi Liu. 2022. Deepgen: Diverse search ad generation and real-time customization.

Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O.K. Li. 2019. Improved Zero-shot Neural Machine Translation via Ignoring Spurious Correlations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1, pages 1258–1268, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yuxuan Gu, Xiaocheng Feng, Sicheng Ma, Jiaming Wu, Heng Gong, and Bing Qin. 2022. Improving Controllable Text Generation with Position-Aware Weighted Decoding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3449–3467, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xudong Han, Timothy Baldwin, and Trevor Cohn. 2021. Balancing out Bias: Achieving Fairness Through Training Reweighting.

Zhiting Hu and Li Erran Li. 2021. A Causal Lens for Controllable Text Generation. In *Advances in Neural Information Processing Systems*, pages 24941–24955.

Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. 2020. Learning the Difference that Makes a Difference with Counterfactually-Augmented Data. In *ICLR*. OpenReview.net.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2016. Controlling output length in neural encoder-decoders. *Conference on Empirical Methods in Natural Language Processing, Proceedings*, pages 1328–1338.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2021. GeDi: Generative Discriminator Guided Sequence Generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4929–4952, Stroudsburg, PA, USA. Association for Computational Linguistics.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2021. Deduplicating Training Data Makes Language Models Better.

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, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Stroudsburg, PA, USA. 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 Pretraining Approach.

Yizhu Liu, Zhiyi Luo, and Kenny Zhu. 2018. Controlling Length in Abstractive Summarization Using a Convolutional Neural Network. In *Proceedings of* *the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4110–4119, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Kaiji Lu, Piotr Mardziel, Fangjing Wu, Preetam Amancharla, and Anupam Datta. 2018. Gender Bias in Neural Natural Language Processing.

Kazuma Murao, Ken Kobayashi, Hayato Kobayashi, Taichi Yatsuka, Takeshi Masuyama, Tatsuru Higurashi, and Yoshimune Tabuchi. 2019. A Case Study on Neural Headline Generation for Editing Support. In *Proceedings of the 2019 Conference of the North*, volume 2, pages 73–82, Stroudsburg, PA, USA. Association for Computational Linguistics.

Judea Pearl. 2009. Causal inference in statistics: An overview. *Statistics Surveys*, 3(none):96–146.

Nanyun Peng, Marjan Ghazvininejad, Jonathan May, and Kevin Knight. 2018. Towards Controllable Story Generation. In *Proceedings of the First Workshop on Storytelling*, pages 43–49, Stroudsburg, PA, USA. Association for Computational Linguistics.

Reid Pryzant, Dallas Card, Dan Jurafsky, Victor Veitch, and Dhanya Sridhar. 2021. Causal Effects of Linguistic Properties. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4095–4109, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jing Qian, Li Dong, Yelong Shen, Furu Wei, and Weizhu Chen. 2022. Controllable Natural Language Generation with Contrastive Prefixes. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2912–2924, Stroudsburg, PA, USA. Association for Computational Linguistics.

Parker Riley, Noah Constant, Mandy Guo, Girish Kumar, David Uthus, and Zarana Parekh. 2021. TextSETTR: Few-Shot Text Style Extraction and Tunable Targeted Restyling. 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 3786–3800, Stroudsburg, PA, USA. Association for Computational Linguistics.

James M Robins, Andrea Rotnitzky, and Lue Ping Zhao. 1994. Estimation of regression coefficients when some regressors are not always observed. *Journal of the American statistical Association*, 89(427):846–866.

Paul R. Rosenbaum and Donald B. Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55.

Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. *33rd International Conference on Machine Learning*, 4:2512–2523.

Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. 2019. What makes a good conversation? How controllable attributes affect human judgments. In *Proceedings of the 2019 Conference of the North*, volume 1, pages 1702–1723, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Controlling politeness in neural machine translation via side constraints. *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference*, pages 35–40.

Shanya Sharma, Manan Dey, and Koustuv Sinha. 2021. Evaluating Gender Bias in Natural Language Inference. pages 1–16.

Chenhui Shen, Liying Cheng, Ran Zhou, Lidong Bing, Yang You, and Luo Si. 2022. MReD: A Meta-Review Dataset for Structure-Controllable Text Generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2521–2535, Stroudsburg, PA, USA. Association for Computational Linguistics.

Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An Empirical Study on Robustness to Spurious Correlations using Pre-trained Language Models. *Transactions of the Association for Computational Linguistics*, 8:621–633.

Victor Veitch, Alexander D'Amour, Steve Yadlowsky, and Jacob Eisenstein. 2021. Counterfactual Invariance to Spurious Correlations: Why and How to Pass Stress Tests. In *Advances in Neural Information Processing Systems*, pages 16196–16208. Curran Associates, Inc.

Xuanhui Wang, Michael Bendersky, Donald Metzler, and Marc Najork. 2016. Learning to Rank with Selection Bias in Personal Search. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 115–124, New York, NY, USA. ACM.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, and Daniel Weld. 2021. Polyjuice: Generating Counterfactuals for Explaining, Evaluating, and Improving Models. 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 6707–6723, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yu Yan, Fei Hu, Jiusheng Chen, Nikhil Bhendawade, Ting Ye, Yeyun Gong, Nan Duan, Desheng Cui, Bingyu Chi, and Ruofei Zhang. 2021. FastSeq: Make sequence generation faster. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 218–226, Online. Association for Computational Linguistics.

Kevin Yang and Dan Klein. 2021. FUDGE: Controlled Text Generation With Future Discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535, Stroudsburg, PA, USA. Association for Computational Linguistics.

Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. 2021. A Survey on Causal Inference. *ACM Transactions on Knowledge Discovery from Data*, 15(5):1–46.

Shuai Zhang, Meng Wang, Sijia Liu, Pin-yu Chen, and Jinjun Xiong. 2022. How does unlabeled data improve generalization in self-training? A one-hidden-layer theoretical analysis. In *ICLR*.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ran Zmigrod, Sebastian J Mielke, Hanna Wallach, and Ryan Cotterell. 2019. Counterfactual Data Augmentation for Mitigating Gender Stereotypes in Languages with Rich Morphology. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1651–1661, Stroudsburg, PA, USA. Association for Computational Linguistics.

A Appendix

A.1 Hyperparameters

Hyperparameters in use are summarized in in Table 6. We use the default HuggingFace setup unless otherwise specified (e.g., default learning rate scheduler, default length and repetition penalties). We tune learning rate from {1e-5, 5e-5, 1e-4} and pick the ones with the best validation metric. Number of epochs are reported for BART+CTRL baseline training. Uncontrolled, IPS, FAST, and GeDi are trained for similar epochs. Generation speed is reported for BART+CTRL baseline. The maximum number of tokens are picked so that 99% of the times input/ouput from training set will not get truncated, except if it exceeds the maximum number of tokens of the pretrained models.

model	params	PENS	MReD	Ads
generation	batch size	64	36	96
	# epochs	40	10	6
	LR	1e-5	5e-5	5e-5
	encoder #tokens	768	1024	314
	decoder #tokens	32	72	26
	train speed	70	25	190
	generation speed	85	40	150
propensity	batch size	64	64	64
	input #tokens	512	512	314
	# epochs	6	15	5
	LR	1e-5	1e-5	1e-5
	train speed	70	25	110
	inference speed	240	170	470

Table 6: Hyperparameter settings for generation model and propensity model on PENS and Ads datasets. Unit for training/inference speed is number of samples per second on one GPU.

A.2 Propensity model

We finetune Roberta-base for sequence classification with the default HuggingFace setup, to predict the attribute of the ground truth target from the context as input. On PENS, we do a binary classification; therefore we use a sigmoid function to transform the score into probability. On MReD and Ads, we do multi-class classification task and use a softmax fucntion to convert scores into probabilities. We also use the dev set to pick the best epoch during training based on AUC for PENS and accuracy for MReD and Ads.

A.3 PPLM

We follow the implementation from Dathathri et al. (2020) with minimal changes to adapt it to an encoder-decoder structure (BART). We apply perturbations on both cross-attention and self-attention key-value pairs instead of only the self-attention as in the original paper for decoder-only models (GPT2). PPLM uses a discriminative classifier $p(a|y)$ to steer decoding towards having the desired attribute. In addition, the classifier needs to have the same vocabulary as the generation model. Therefore, we finetune Roberta-base for this purpose as it shares the same vocabulary with BART. The Roberta-base classifiers achieve 96% and 88% macro-F1 for PENS and Ads respectively on the default test sets. Note that on Ads data, we use a previously built BERT-base classifier to produce attribute label on the training and testing data (Table 1). This Roberta-base classifier is trained and tested on the labels predicted by the previous BERT-base model, thus mimicking its behavior. For MReD, we reuse the Roberta-base classifier previously described in §3.1.

To steer decoding, PPLM ascends $p(a|y)$ by propagating gradients from the classifier to the perturbations on key-value pairs in BART. At each decoding step, BART outputs a probability distribution $p(y_i|y_{<i}, ...)$ for generating the current ith token. We feed the classifier with previous $i-1$ tokens plus a "soft" token for i, which is a weighted average of embeddings $\sum_{y_i \in V} p(y_i|y_{<i}, ...)E(y_i)$, where V denotes the vocabulary and $E(\cdot)$ denotes the input embedding of the classifier (Roberta). Therefore, we can propagate gradients from the classifier into BART.

We tune hyperparameters on dev set to find a good balance between controllability and decoding quality. We tune γ_{gm} from {0.1, 0.3, 0.5, 0.8}, λ_{KL} from {0, 0.005, 0.01}, and we settle with 0.3 and 0.01 for them respectively. As PPLM decoding output tends to be repetitive, we additionally tune repetition penalty to be 2.

We use greedy decoding following the original implementation. Using beam search for PPLM would be computationally prohibitive as it is already very slow.

A.4 GeDi

We train BART-base for the two models in GeDi. One model $p(y|x)$ uses context x as the encoder input, and the other model $p(y|c)$ uses control code c as the encoder input. The hyperparameters are the same as BART+CTRL baseline except that the max number of encoder tokens for $p(y|c)$ model is 3, 7, and 8 on PENS, MReD, and Ads datasets respectively. We use negative log likelihood (instead of ROUGE1) as the validation metric during training $p(y|c)$. Same as the original GeDi, an additional length normalization heuristic is applied in the Bayes rule for computing the steering term during decoding:

$$p(c|y) = \frac{p(c)p(y|c)^{1/t}}{\sum_{c' \in \{1,...,K\}} p(c')p(y|c')^{1/t}}, \qquad (4)$$

where t is the current length of y. We avoid tuning hyperparameters with the following choices. We choose uniform prior for $p(c)$, and we do not apply the filtering heuristic during decoding but we use the standard beam search as the other methods. We train $p(y|c)$ as a usual generation model conditioned on control code c. In the end, we only tune the parameter ω to control the trade-off between controllability and decoding quality.

The original GeDi paper converts multi-class classification into multiple binary classification tasks with control code and anti-control code for each class in order to improve speed. We do not do this conversion but normalize over all K classes in Eq. 4.

Method	Training (hrs)	Generation (samples/sec)
BART+CTRL	30	150
PPLM	30	0.38
GeDi	42	8
GeDi+x	30	8
IPS	30	150
FAST	85	150

Table 7: Training time and generation speed for different methods measured on 1 Nvidia V100 GPU.

A.5 GeDi+x

We reuse our implementation of GeDi for GeDi+x by simply switching both $p(y|x)$ and $p(y|c)$ models to BART+CTRL baseline $p(y|x, c)$. So the settings are the same as GeDi or BART+CTRL.

A.6 Computational cost

We compare the training and inference cost for different methods on Search Ads dataset in Table 7. During training, FAST has the largest cost due to the additional cost from training the initial model and using it to generate counterfactual data. GeDi also has higher cost than the rest of the methods as it trains an additional model $p(y|c)$ for steering the decoding. During generation, PPLM is much slower than other methods at it needs to compute gradient several times at each decoding step. GeDi and GeDi+x are also slow due to computing the contrast term (e.g., Eq. 3). The rest of the methods are equally fast.

A.7 Spurious correlation in MReD

Similar to PENS dataset, we finetune Roberta-base to predict the next sentence's category from the context (all preceding sentences, ratings from individual reviewers and their reviews). The classifier achieves 71% AUC and 33% accuracy on hold-out test set, which are much higher than random guess (50% AUC, 11% accuracy) or majority class (13% accuracy). Therefore, we empirically confirm the existence of spurious correlation between context and MReD category.

Results in the original MReD paper have already implied some reasons for why such correlations exist. First, preceding sentences are predictive of the category of the next sentence because meta-reviews have some typical patterns, for example, "abstract→strength→weakness", "rating summary→weakness→rebuttal process", and etc. Second, individual reviewers' ratings and their reviews are predictive of the category of a sentence in the meta review. For example, there is higher chance to get a "strength" than "weakness" in the meta review if the individual reviewers are more positive about a paper.

To show the damaging effect of the spurious correlation, we use BART+CTRL baseline to generate next sentences in all 9 categories. Then we detect the category from output using the Roberta-base classifier. Finally, we evaluate the accuracy between the detected category and the intended category (control code). The system successfully generated the intended category 90% of the time in factual cases (when the control code matched the ground-truth attribute), but 75% of the time in counterfactual cases. Again, this demonstrate the degradation of controllability when spurious correlations break at test time.

A.8 Spurious correlation in Search Ads

Similar to PENS and MReD datasets, we finetune Roberta-base to predict the ad category from the context, which include various landing page features. The classifier achieves 74% AUC and 36% accuracy on hold-out test set, which are much higher than random guess (50% AUC, 11% accuracy) or majority class (23% accuracy). Therefore, we empirically confirm the existence of spurious correlation between context and ad category.

This correlation is hardly a surprise. Advertisers write ads that perform well for their landing pages on average, so different categories are preferred for different landing pages. While the majority category is product or service on all data as shown in Table 1, by slicing data into different business industries, we find that majority category is location for travel and tourism industry, call to action for vehicle industry, and highlight for retail industry (which contains promotion, shipping or other information to make the product stand out). While an ad in the majority category may perform well on average, we can get an even better chance to win the user click by generating ads in all categories and displaying the best one at query time. For example, while "Buy Truck Engines Now" may be a good ad for query "truck engine", "New & Used Truck Engines" is a better choice for query "used truck engine".

We then use BART+CTRL baseline to generate ads in all 9 categories. Then we detect the category from output using the BERT-base-uncased classifier. Finally, we evaluate the accuracy between the detected category and the intended category (control code). The system successfully generated the intended category 76% of the time in factual cases (when the control code matched the ground-truth attribute), but 65% of the time in counterfactual cases. Again, this demonstrate the degradation of controllability when spurious correlations break at test time.

A.9 Details on human evaluation

As opposed to crowd sourced judges, our judges are paid with hourly wages and they are doing the labeling task for a long term, therefore they demonstrate more consistent labeling quality. They have been trained to ensure understanding the tasks correctly and they get feedback from us to ensure their labeling quality and consistency.

The quality of an ad is labelled in 4 aspects: 1) language, which checks spelling/capitalization, grammar, fluency; 2) human-like, which checks if the ad sounds like human written rather than machine generated and if it agrees with common sense; 3) factuality, which checks if the ad contains false claims (e.g., free shipping) not existing in the landing page; and 4) relevance, which checks if the ad is relevant for the landing page. Judges should visit the landing page and read it carefully for checking factuality and relevance. Judges can also skip an example in cases such as the text is in a foreign language or the landing page is not accessible.

For category labeling, judges first select if an ad is scorable to prevent cases such as text is in a foreign language or quality is too poor to understand. Judges are trained before they start labeling by going through our judgement guideline and passing our test judgement task. In our judgement guideline, we explain definition of each category with examples. We also explain the idea behind designing these categories that advertising is commonly surrounding three roles – advertisers (their name or brand), products (what's the product, purchasing information such as price, shipping), and customers (what's the benefit for customer, call to action) – to help judges better differentiate these categories.

A.10 Examples of generated meta reviews

We provide an example of generations from 3 models (BART+CTRL baseline, IPS and FAST) in Figure 3. In this example, the individual reviews are quite positive, and so is the ground truth meta review. The BART+CTRL model seems to struggle in generating a sentence in the "weakness" category, but it is preferring "strength", thereby ignoring the control code, which is also the case for "rating summary". On the other hand, FAST is able to generate a "weakness" sentence correctly. For "AC disagreement" category, even FAST struggles to generate it correctly. This is likely due to the fact that there are only 24 training examples in this category. Interestingly, IPS generates a correct example in this category, which seems to the case in general as we examined more examples. However, IPS suffers from worse language quality. It generates a sentence with repetition issue in the "suggestion" category. We note that the "rebuttal process" category should not be applicable for this meta review, as there is no such information from the context.

A.11 Examples of generated ads

We provide examples of generated ads from 3 models (BART+CTRL baseline, IPS and FAST) in Figure 4 and 5. As the difference between the 3 models are not huge as seen from the human evaluation results (although statistically significant), we pick those examples to highlight the typical difference between the 3 models. In actual online serving, up to 3 titles can be concatenated together with delimiter | to form a longer title, and up to 2 descriptions can be concatenated together.

Previous *i* sentences	this submission proposes an efficient parametrization of a recurrent neural net by using two transition functions (one large and one small) to reduce the amount of computation (though, without actual improvement on GPU.)
ratings	R1 rating score: 7, R2 rating score: 7, R3 rating score: 8.
Individual reviews	The paper proposes a way to speed up the inference time of RNN via Skim mechanism where only a small part of hidden variable is updated once the model has decided a corresponding word token seems irrelevant w.r.t. a given task. While the proposed idea might be too simple, the authors show the importance of it via thorough experiments. *(skipped 298 words)* <REVBREAK> Summary: The paper proposes a learnable skimming mechanism for RNN. The model decides whether to send the word to a larger heavy-weight RNN or a light-weight RNN. The heavy-weight and the light-weight RNN each controls a portion of the hidden state. The paper finds that with the proposed skimming method, they achieve a significant reduction in terms of FLOPS. Although it doesn't contribute to much speedup on modern GPU hardware, there is a good speedup on CPU, and it is more power efficient. Contribution: - The paper proposes to use a small RNN to read unimportant text. Unlike (Yu et al., 2017), which skips the text, here the model decides between small and large RNN. Pros: - Models that dynamically decide the amount of computation make intuitive sense and are of general interests. - The paper presents solid experimentation on various text classification and question answering datasets. - The proposed method has shown reasonable reduction in FLOPS and CPU speedup with no significant accuracy degradation (increase in accuracy in some tasks). - The paper is well written, and the presentation is good. Cons: - Each model component is not novel. The authors propose to use Gumbel softmax, but does compare other gradient estimators. It would be good to use REINFORCE to do a fair comparison with (Yu et al., 2017) to see the benefit of using small RNN. - The authors report that training from scratch results in unstable skim rate, while Half pretrain seems to always work better than fully pretrained ones. This makes the success of training a bit adhoc, as one need to actively tune the number of pretraining steps. - Although there is difference from (Yu et al., 2017), the contribution of this paper is still incremental. Questions: - Although it is out of the scope for this paper to achieve GPU level speedup, I am curious to know some numbers on GPU speedup. - One recommended task would probably be text summarization, in which the attended text can contribute to the output of the summary. Conclusion: - Based on the comments above, I recommend Accept <REVBREAK> This paper proposes a skim-RNN, *(skip 112 words)* Cons. - The idea is quite simple and the novelty is incremental by considering the difference from skip-RNN. *(skipped 122 words)*
Ground truth continuation	the reviewers found the submission very positive. <sep> please, do not forget to include all the result and discussion on the proposed approach's relationship to VCRNN which was presented at the same conference just a year ago.

Control	BART+CTRL	IPS	FAST
Abstract	The paper shows that the proposed method achieves a significant reduction in FLOPS and CPU speedup.	The paper shows that the skimming mechanism can reduce the amount of computation needed for the model to perform well on a variety of tasks.	The paper shows that the proposed method achieves a significant reduction in FLOPS and CPU speedup with no significant accuracy degradation.
Strength	The paper is well written and the presentation is good.	The reviewers agree that the idea is interesting and well-motivated.	(same as BART+CTRL)
Weakness	The reviewers agree that the paper is well written and the presentation is good. *(strength)*	The reviewers agree that the novelty of the idea is limited, and there are some concerns about the performance of the method, but overall, the reviewers agree that it is worth pursuing. *(strength)*	The reviewers agree that the paper is well written and the presentation is good, but there are some concerns about the novelty of the approach.
Rating summary	The reviewers agree that the paper is well written and the presentation is good.	All reviewers recommend acceptance.	The reviewers were unanimous in their vote to accept the paper.
AC disagreement	I agree with the reviewers that the paper is well written and the presentation is good. *(strength)*	there's been some sentiment by the reviewers that the paper might not be ready for publication in its current form, but after further discussion with another AC and PC's, we have concluded that it may be ready for publication in its current form.	The reviewers agree that the paper is well written, and the presentation is good, but there are some concerns about the novelty of the approach. *(weakness)*
Rebuttal process	The rebuttal addressed some of the concerns raised by the reviewers, but not enough to convince the reviewers to accept the paper.	The reviewers were impressed by the contribution of the paper and the authors' response to the reviewers' comments.	The authors have addressed some of the concerns raised by the reviewers in the rebuttal.
Suggestion	I encourage the authors to incorporate the reviewers' comments in the final version of the paper.	The reviewers agree that the paper could benefit from a more comprehensive exploration of the benefits of the method, as well as a more comprehensive evaluation of the benefits of the method. *(repetition)*	I encourage the authors to take reviewers' comments into account when revising the paper for the camera ready.
Decision	The paper is well written and the presentation is good, so I recommend acceptance.	The reviewers all found the idea interesting and the results convincing enough to warrant acceptance to *CONF*.	The reviewers and AC agree that the paper is worthy of publication at *CONF*.
Misc	We hope that the reviewers' comments help you improve your submission for the camera ready.	The reviewers liked the paper.	The reviewers had a number of questions and concerns about the paper.

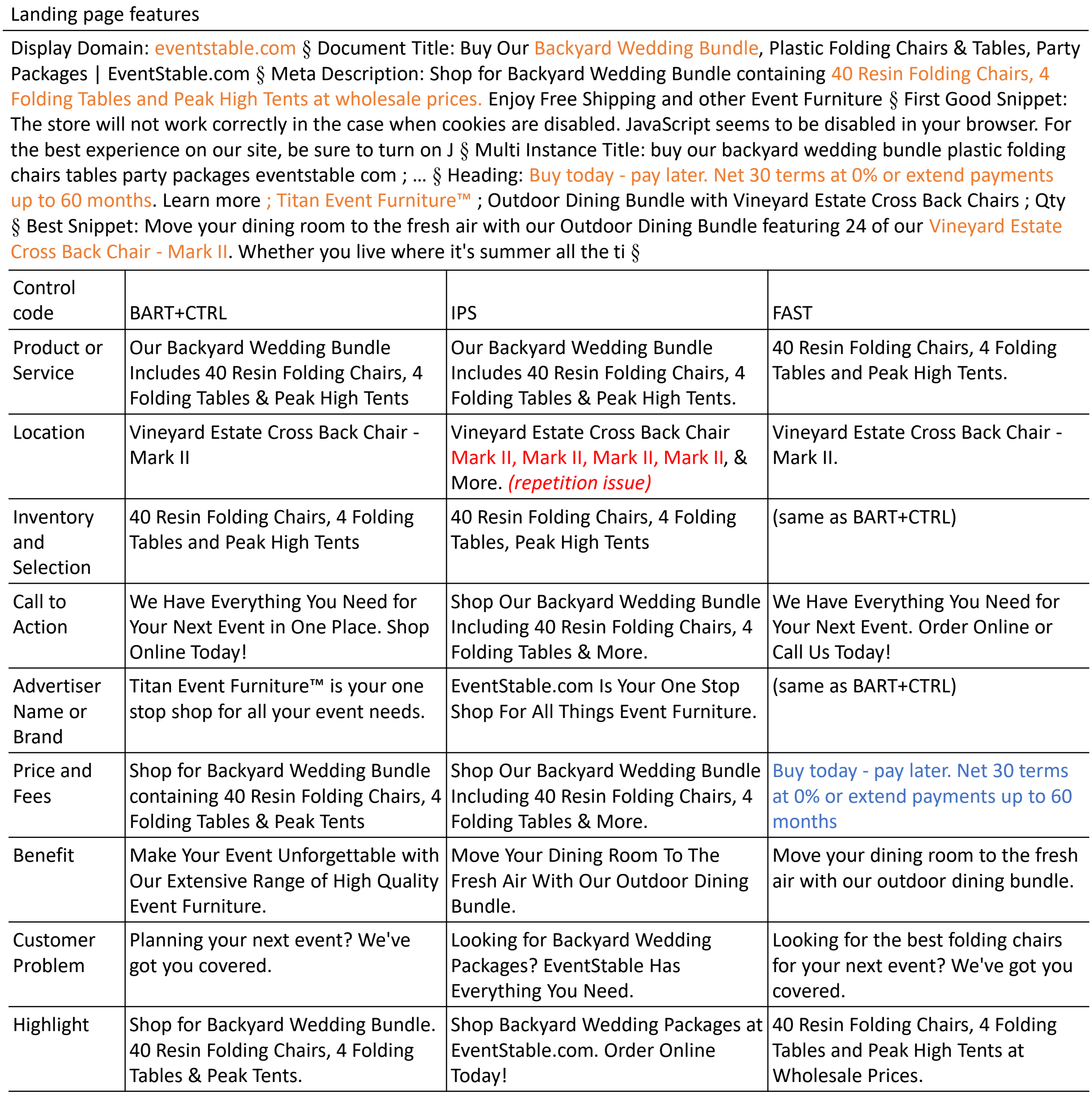
Figure 3: Example of generated next sentence for a meta review. Key information in the input that the generations are based on is highlighted in orange. Issues with generations are highlighted in red with a brief explanation.

Landing page features

Display Domain: academyinteriorsltd.co.uk § Document Title: Fitted Bedroom | Cheap Fitted Wardrobes | Academy Interiors § Meta Description: Browse our wide range of fitted bedrooms and wardrobes collections. At Academy Interiors we will combine your ideas with our expert knowledge to create an aesthetically pleasing s § First Good Snippet: Our clever design and intelligent use of space coupled with over 20 years experience is why the end result will always look stunning and retain its appearance for years to come. W § Multi Instance Title: fitted bedroom cheap fitted wardrobes academy interiors ; cheap fitted wardrobes ; academy interiors ; fitted bedroom § Heading: Bedrooms ; Wardrobes ; Bedroom fittings § Best Snippet: Walk into an Academy Interiors bedroom and you will always feel at home. Whichever beautiful style you choose from our wide range, every aspect is carefully designed to compliment

Control code	BART+CTRL	IPS	FAST
Product or Service	Fitted Wardrobes	Fitted Bedroom	(same as BART+CTRL)
Location	Fitted Wardrobes Near Me	Fitted Wardrobes Near Me	(same as BART+CTRL)
Inventory and Selection	Fitted Wardrobes	Fitted Bedroom	(same as BART+CTRL)
Call to Action	Buy Fitted Wardrobes	Buy Fitted Bedroom	(same as BART+CTRL)
Advertiser Name or Brand	Academy Interiors Ltd	(same as BART+CTRL)	(same as BART+CTRL)
Price and Fees	Cheap Fitted Wardrobes	Cheap Fitted Bedroom	Affordable Fitted Wardrobes
Benefit	Transform Your Bedroom	(same as BART+CTRL)	(same as BART+CTRL)
Customer Problem	Looking For A New Wardrobe?	Looking For A New Bedroom?	(same as BART+CTRL)
Highlight	Fitted Wardrobes	Fitted Bedroom	Over 20 Years Experience

Figure 4: Example of generated ad titles. For this landing page, only FAST generates title correctly in "Highlight" category. Key information in the landing page that the generated ads are based on is highlighted in orange.



Landing page features

Display Domain: eventstable.com § Document Title: Buy Our Backyard Wedding Bundle, Plastic Folding Chairs & Tables, Party Packages | EventStable.com § Meta Description: Shop for Backyard Wedding Bundle containing 40 Resin Folding Chairs, 4 Folding Tables and Peak High Tents at wholesale prices. Enjoy Free Shipping and other Event Furniture § First Good Snippet: The store will not work correctly in the case when cookies are disabled. JavaScript seems to be disabled in your browser. For the best experience on our site, be sure to turn on J § Multi Instance Title: buy our backyard wedding bundle plastic folding chairs tables party packages eventstable com ; ... § Heading: Buy today - pay later. Net 30 terms at 0% or extend payments up to 60 months. Learn more ; Titan Event Furniture™ ; Outdoor Dining Bundle with Vineyard Estate Cross Back Chairs ; Qty § Best Snippet: Move your dining room to the fresh air with our Outdoor Dining Bundle featuring 24 of our Vineyard Estate Cross Back Chair - Mark II. Whether you live where it's summer all the ti §

Control code	BART+CTRL	IPS	FAST
Product or Service	Our Backyard Wedding Bundle Includes 40 Resin Folding Chairs, 4 Folding Tables & Peak High Tents	Our Backyard Wedding Bundle Includes 40 Resin Folding Chairs, 4 Folding Tables & Peak High Tents.	40 Resin Folding Chairs, 4 Folding Tables and Peak High Tents.
Location	Vineyard Estate Cross Back Chair - Mark II	Vineyard Estate Cross Back Chair Mark II, Mark II, Mark II, Mark II, & More. *(repetition issue)*	Vineyard Estate Cross Back Chair - Mark II.
Inventory and Selection	40 Resin Folding Chairs, 4 Folding Tables and Peak High Tents	40 Resin Folding Chairs, 4 Folding Tables, Peak High Tents	(same as BART+CTRL)
Call to Action	We Have Everything You Need for Your Next Event in One Place. Shop Online Today!	Shop Our Backyard Wedding Bundle Including 40 Resin Folding Chairs, 4 Folding Tables & More.	We Have Everything You Need for Your Next Event. Order Online or Call Us Today!
Advertiser Name or Brand	Titan Event Furniture™ is your one stop shop for all your event needs.	EventStable.com Is Your One Stop Shop For All Things Event Furniture.	(same as BART+CTRL)
Price and Fees	Shop for Backyard Wedding Bundle containing 40 Resin Folding Chairs, 4 Folding Tables & Peak Tents	Shop Our Backyard Wedding Bundle Including 40 Resin Folding Chairs, 4 Folding Tables & More.	Buy today - pay later. Net 30 terms at 0% or extend payments up to 60 months
Benefit	Make Your Event Unforgettable with Our Extensive Range of High Quality Event Furniture.	Move Your Dining Room To The Fresh Air With Our Outdoor Dining Bundle.	Move your dining room to the fresh air with our outdoor dining bundle.
Customer Problem	Planning your next event? We've got you covered.	Looking for Backyard Wedding Packages? EventStable Has Everything You Need.	Looking for the best folding chairs for your next event? We've got you covered.
Highlight	Shop for Backyard Wedding Bundle. 40 Resin Folding Chairs, 4 Folding Tables & Peak Tents.	Shop Backyard Wedding Packages at EventStable.com. Order Online Today!	40 Resin Folding Chairs, 4 Folding Tables and Peak High Tents at Wholesale Prices.

Figure 5: Example of generated ad descriptions. For this landing page, only FAST generates description correctly in "Price and Fees" category. One description from IPS model has repetition issue. Key information in the landing page that the generated ads are based on is highlighted in orange