ON REWARD Maximization AND DISTRIBUTion MATCHING FOR FINE-TUNING LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

The availability of large pre-trained models is changing the landscape of Machine Learning research and practice, moving from a “training from scratch” to a “fine-tuning” paradigm. While in some applications the goal is to “nudge” the pre-trained distribution towards preferred outputs, in others it is to steer it towards a different distribution over the sample space. Two main paradigms have emerged to tackle this challenge: Reward Maximization (RM) and, more recently, Distribution Matching (DM). RM applies standard Reinforcement Learning (RL) techniques, such as Policy Gradients, to gradually increase the reward signal. DM prescribes to first make explicit the target distribution that the model is fine-tuned to approximate. Here we explore the intimate connections between the two paradigms, and show that methods such as KL-control developed in the RM paradigm can also be construed as belonging to DM. We further observe that while DM differs from RM, it can suffer from similar training difficulties, such as high gradient variance. We leverage connections between the two paradigms to import the concept of baseline into DM methods. We empirically validate the benefits of adding a baseline on an array of controllable language generation tasks such as constraining topic, sentiment, and gender distributions in texts sampled from a language model. We observe superior performance in terms of constraint satisfaction, stability and sample efficiency.

1 INTRODUCTION

Pre-trained language models (Devlin et al., 2019; Radford et al., 2019) are changing the landscape of Machine Learning research and practice. Due to their strong generative capabilities many studies have found it sufficient to “nudge” these models to conform to global preferences defined over the generated sequences instead of training from scratch using annotated data. These preferences could include topic and sentiment (Dathathri et al., 2020), valid musical notes and molecular structures (Jaques et al., 2017a), code compilability (Korbak et al., 2021), balancing gender bias (Khalifa et al., 2021), evaluation metrics for Machine Translation and Summarization (Ranzato et al., 2016; Bahdanau et al., 2016), or direct human feedback (Ziegler et al., 2019; Stiennon et al., 2020). This large body of studies is driven by two paradigms: Reward Maximization (RM) and Distribution Matching (DM).

Reward Maximization RM exploits the intuitive notion that we can nudge pre-trained models towards some preferences by providing global sequence-level rewards when the model generates outputs that satisfy desired features. For instance, if the model is producing toxic content, we can apply Reinforcement Learning (RL) techniques to discourage it from producing similar content again in the future. However, the risk of naively applying RL is that the model can undergo catastrophic forgetting of its original distribution. For example, it can degenerate into producing a single nonsensical but at least nontoxic sequence. Although several studies have considered hand-crafting general rewards to ensure desirable features like fluency (Liu et al., 2016a; Tambwekar et al., 2019), coming up with

![Figure 1: In this study we make connection between two popular paradigms for fine-tuning sequence generation models according to preferences Reward Maximization (RM) and Distribution Matching (DM).](image-url)
rewards that are not incomplete or imperfect is highly non-trivial (Wu et al., 2016; Vedantam et al., 2015). These challenges have sparked a wide discussion on the overall effectiveness of RM for some tasks such as Neural Machine Translation (Choshen et al., 2020; Kiegeland & Kreutzer, 2021).

**Reward Maximization with KL-Control**  To tackle the aforementioned issues of “catastrophic forgetting”, several studies, still under an RM paradigm, have considered incorporating a distributional term inside the reward to be maximized. In particular Jaques et al. (2017b, 2019); Ziegler et al. (2019) and Stiennon et al. (2020) have applied variations of KL-control (Todorov, 2007; Kappen et al., 2012) which adds a penalty term to the reward term so that the resulting policy does not deviate too much from the original one in terms of KL-divergence. The overall objective with the KL-penalty is maximized using an RL algorithm of choice including: PPO (Schulman et al., 2017a) as in Ziegler et al. (2019) or Q-learning (Mnih et al., 2013) as in Jaques et al. (2017b). Adding this distributional KL-penalty to the reward raises some important questions: What effect does it have on the shape of the optimal policy? Does this new objective have any interpretation from a distributional perspective?

**Distribution Matching**  A different recent paradigm for fine-tuning language models to satisfy downstream preferences formulates the problem as Distribution Matching (DM). This paradigm consists of two steps: first a target distribution incorporating the desired preferences is defined as an Energy-Based Model (LeCun et al., 2006). Then the forward KL divergence is minimized between this target distribution and an auto-regressive policy using a family of algorithms referred to as Distributional Policy Gradients (DPG) (Parshakova et al., 2019b; Khalifa et al., 2021; Korbak et al., 2021). This approach capitalizes on the flexibility of EBMs in specifying the target distribution. For example, the EBM can be defined so that it conforms to all downstream preferences while its corresponding normalized distribution has a minimal KL divergence from the original, pre-trained language model, therefore tackling the problem of “catastrophic forgetting” (Khalifa et al., 2021). Interestingly, this DM paradigm can also deal with distributional preferences, for instance, for de-biasing language models by specifying that the generated sequences should be gender-balanced, i.e. that 50% of generations contain female mentions. Such distributional constraints cannot be defined in the RM paradigm where a reward is calculated for a single sequence.

Contrasting these two paradigms for fine-tuning language models, we can notice the promises and limitations of each. RM approaches are equipped with a large arsenal of RL algorithms and optimization techniques that can be efficient in reward maximization, however they lack the distributional perspective that is necessary for avoiding catastrophic forgetting and imposing distributional preferences over LMs. DM approaches are better suited to tackle those limitations, however, the family of DPG algorithms currently used is not as rich as its RL counterpart. So far, the connections between these two seemingly distinct paradigms have not been thoroughly explored. By establishing such connections we might import ideas from one approach to the other. This is our goal in this paper, detailing the nuanced connections and applying them to a case-study in variance reduction. Overall, our contributions are the following:

- We untangle the connections between the RM and DM paradigms for fine-tuning language models. We provide a detailed comparison between the family of DPG algorithms with Policy Gradients of standard RL.
- We provide an interpretation of KL-control techniques from a distribution matching perspective, placing such techniques at an intermediate place between RM and DM.
- We exploit these connections to theoretically justify applying baselines — a variance reduction technique from RL — to DPG and derive a particular choice of a baseline. On an array of controllable language generation experiments about constraining topic, sentiment, and gender distributions over a pre-trained language model, we show that adding baselines leads to superior performance on constraint satisfaction, stability on small batch sizes, and sample efficiency.

2 **Background**

**Standard Policy Gradients**  One popular method for adapting the behaviour of language models to certain preferences has been that of assigning a “reward” score $R(x)$ for sequences $x$ sampled...
from an autoregressive language model (policy) \( \pi_\theta \). Then, the simplest policy gradient algorithm in reinforcement learning, namely, REINFORCE [Williams 1992a], aims to find the policy \( \pi_\theta(x) \) that maximizes the average reward \( \mathbb{E}_{x \sim \pi_\theta} R(x) \), and this leads, via the so-called “log derivative trick”, to a gradient ascent algorithm based on:

\[
\nabla_\theta \mathbb{E}_{x \sim \pi_\theta} R(x) = \mathbb{E}_{x \sim \pi_\theta} R(x) \nabla_\theta \log \pi_\theta(x),
\]

which iteratively samples \( x \) from \( \pi_\theta \) and update parameters by increments proportional to \( R(x) \nabla_\theta \log \pi_\theta(x) \).

**KL-control** [Jaques et al. 2017b, 2019; Ziegler et al. (2019), inspired by KL-control (Todorov 2007; Kappen et al. 2012), add a KL penalty term to the reward objective to penalize large deviations from the original pretrained model. That is, they maximize the expectation \( \mathbb{E}_{x \sim \pi_\theta} R_\theta(x) \), where:

\[
R_\theta(x) \overset{\text{def}}{=} r(x) - \beta \log \frac{\pi_\theta(x)}{a(x)},
\]

and where \( \beta \) is a free hyperparameter balancing the wish to maximize \( r \) with that of not deviating from \( a \).

**Distributional Policy Gradients** Distributional Policy Gradients (DPG) [Parshakova et al., 2019b] is a recent approach used to fit an autoregressive policy \( \pi_\theta \) to the distribution \( p(x) = P(x)/Z \) induced by the EBM \( P(x) \), where \( Z = \sum_x P(x) \) is the normalization constant (partition function). Given an arbitrary EBM \( P(x) \), DPG optimizes the loss function \( D_{KL}(p, \pi_\theta) \) with respect to the parameters \( \theta \) of an autoregressive model \( \pi_\theta \), a loss which is minimized for \( \pi_\theta = p \). The KL-divergence minimization objective leads to a gradient estimate of the form:

\[
\nabla_\theta D_{KL}(p, \pi_\theta) = -\nabla_\theta \mathbb{E}_{x \sim \pi_\theta} \log \pi_\theta(x) = -\frac{1}{Z} \sum_x P(x) \nabla_\theta \log \pi_\theta(x)
\]

\[
= -\frac{1}{Z} \pi_\theta(x) \nabla_\theta \log \pi_\theta(x).
\]

3 **REWARD MAXIMIZATION VS DISTRIBUTION MATCHING**

In the previous section, we have summarized three approaches that have been suggested for fine-tuning language models. Two of them can be characterized as “Reward Maximization” (RM): Standard Policy Gradients (PG) and KL Control. On the other hand, DPG clearly belongs to the realm of “Distribution Matching” (DM) as it first defines the target distribution and then optimizes a policy to match it. In the rest of this section, we will explore connections between these two seemingly distinct concepts and, in the following section, we will exploit them to improve DM-based methods.

3.1 **STANDARD VS. PARAMETRIC REWARDS**

Let us start with distinguishing between a “parametric reward” \( R_\theta \) which depends on \( \theta \) and a standard reward \( R \), which does not. If we wished to maximize the expected parametric reward, \( \mathbb{E}_{\pi_\theta} R_\theta(x) \), we would follow its gradient, leading to the identities:

\[
\nabla_\theta \mathbb{E}_{x \sim \pi_\theta} R_\theta(x) = \nabla_\theta \sum_x \pi_\theta(x) R_\theta(x)
\]

\[
= \sum_x \pi_\theta(x) \nabla_\theta R_\theta(x) + \sum_x R_\theta(x) \nabla_\theta \pi_\theta(x)
\]

\[
= \sum_x \pi_\theta(x) \nabla_\theta R_\theta(x) + \sum_x \pi_\theta(x) R_\theta(x) \nabla_\theta \log \pi_\theta(x)
\]

\[
= \mathbb{E}_{x \sim \pi_\theta} \nabla_\theta R_\theta(x) + \mathbb{E}_{x \sim \pi_\theta} R_\theta(x) \nabla_\theta \log \pi_\theta(x).
\]
While the objective of DPG (distribution matching) is different from that of Policy Gradients (reward maximization), adding a KL-penalty term to the reward (as in the case of KL-control) leads to a parametric reward. Without a clear meaning. By contrast, the DPG algorithms are designed to perform DM on any EBM with the KL penalty term.

3.2 KL CONTROL AS DISTRIBUTION MATCHING

Adding a KL-penalty term to the reward (as in the case of KL-control) leads to a parametric reward. However, due to the particular form of its objective, the RG-term actually vanishes in this case does not vanish. Nonetheless, the analogy behind this gradient term is more fruitful than it first appears. As a matter of fact, DPG gradient estimates suffer from the same high-variance problems as with standard PG. However, due to the particular form of its objective, the RG-term actually vanishes in specific cases where the RG-term evaluates to 0, as we will see below.

Overall, we can conclude that the addition of the distributional term (KL-penalty) to the reward does indeed provide a DM interpretation, namely in terms of minimizing the reverse KL divergence with an emergent underlying distribution $p_z(x)$. We note that $p_z(x)$ does not correspond to an explicit choice of EBM (e.g. the one that balances the gender and topic distributions of a language model). Instead equation (10) has a limited form implicitly defined by the reward $R_{\theta}$, along with a $\beta$ hyperparameter.

3.3 SIMILARITIES AND DIFFERENCES BETWEEN DPG AND POLICY GRADIENTS

In the previous subsection, we have connected KL-control, a method designed under a RM paradigm, to DM. Now, we turn to the converse question of whether DPG, a DM method, can be connected to RM. We begin by noting that after defining $R_{\theta} = \frac{p(x)}{\pi_{\theta}(x)}$, the DPG gradient $\nabla_{\theta} \log \pi_{\theta}(x)$ acquires the format of the PG-term

$$\text{arg min}_{\pi_{\theta}} D_{KL}(\pi_{\theta}, p_z) = \text{arg max}_{\pi_{\theta}} \mathbb{E}_{x \sim \pi_{\theta}} \log \frac{p_{\pi_{\theta}}(x)}{p_z(x)}$$

so that the $\text{arg min}_{\pi_{\theta}} D_{KL}(\pi_{\theta}, p_z)$ coincides with the $\text{arg max}_{\pi_{\theta}} \mathbb{E}_{x \sim \pi_{\theta}} R_{\theta}^z(x)$. Provided that the family of distributions $\pi_{\theta}$ is large enough to cover all distributions over $X$, $\text{arg min}_{\pi_{\theta}} D_{KL}(\pi_{\theta}, p_z)$ is just $p_z$, which concludes the proof.

Overall, we can conclude that the addtion of the distributional term (KL-penalty) to the reward does indeed provide a DM interpretation, namely in terms of minimizing the reverse KL divergence with an emergent underlying distribution $p_z(x)$. We note that $p_z(x)$ does not correspond to an explicit choice of EBM (e.g. the one that balances the gender and topic distributions of a language model). Instead equation (10) has a limited form implicitly defined by the reward $R_{\theta}$, along with a $\beta$ hyperparameter.

Nevertheless, the analogy behind this gradient term is more fruitful than it first appears. As a matter of fact, DPG gradient estimates suffer from the same high-variance problems as with standard PG. While the objective of DPG (distribution matching) is different from that of Policy Gradients (reward

---

2This is because $\mathbb{E}_{x \sim \pi_{\theta}} \nabla_{\theta} R_{\theta}^z(x) = -\beta \mathbb{E}_{x \sim \pi_{\theta}} \nabla_{\theta} \log \pi_{\theta}(x) = 0$, via the often used identity $\mathbb{E}_{x \sim \pi_{\theta}} \nabla_{\theta} \log \pi_{\theta}(x) = \sum_x \pi_{\theta}(x) \nabla_{\theta} \log \pi_{\theta}(x) = \sum_x \nabla_{\theta} \pi_{\theta}(x) = \sum_x \pi_{\theta}(x) = 1$.

3The optimal policy $p_z$ is briefly mentioned in [Ziegler et al. 2019] without reference or derivation. The proof, which we believe to clarify important underlying connections, is ours.
To confirm that DPG also needs to estimate the PG-term $E_{x \sim \pi_{\theta}} R(x) \nabla_\theta \log \pi_\theta(x)$ at a given value of $\theta$, using a batch of samples $x$. For such a fixed $\theta$, we can define provisionally $R(x) = R_{\theta}$ and the problem of gradient estimation for this fixed $\theta$ is identical to the estimation $E_{x \sim \pi_{\theta}} R(x) \nabla_\theta \log \pi_\theta(x)$ based on a set of samples $x$ in standard RL. Therefore, the techniques that have been developed to reduce the variance of the gradients estimates in RL can be ported to DPG insofar as we are computing the gradient estimates at a given $\theta$\footnote{To avoid confusion, note that variance reduction methods for SGD focus on the objective of better estimating the true gradient expectation $E_{\pi_{\theta} \theta} \nabla \theta$ for a fixed $\theta$. The fact that $\theta$ will then be updated based on such estimates is orthogonal to this objective.}. In Section 4, we show how one can import one such variance reduction technique to the DPG baselines.

### 4 A Case Study on Variance Reduction

Baselines are a standard variance reduction technique in the context of Policy Gradients [Sutton & Barto 2018]. The idea is to subtract from the reward $R(x)$ a value $B$ that does not introduce bias to the gradients but may change variance. After the introduction of baseline, equation (1) then takes the following form:

$$
\nabla_\theta \mathbb{E}_{\pi_{\theta}} R(x) = \mathbb{E}_{\pi_{\theta}} [R(x) - B] \nabla_\theta \log \pi_\theta(x). \tag{12}
$$

In standard RL, the simplest form of baseline $B$ is just the average of the rewards for the policy$^4$

$$
B_{RL} = E_{x \sim \pi_{\theta}} R(x). \tag{13}
$$

Following the same methodology of taking the baseline as the expectation of the reward term, we can obtain a remarkably simple form of a baseline for DPG:

$$
B = E_{x \sim \pi_{\theta}} \frac{P(x)}{\pi_\theta(x)} = \sum_x \pi_\theta(x) \frac{P(x)}{\pi_\theta(x)} = \sum_x P(x) = Z. \tag{14}
$$

To confirm that $B$ does not introduce bias to the gradients, let us rewrite the DPG gradient in (5) with the added baseline $B = Z$:

$$
E_{x \sim \pi_{\theta}} [R_\theta(x) - Z] \nabla_\theta \log \pi_\theta(x) = E_{x \sim \pi_{\theta}} R_\theta(x) \nabla_\theta \log \pi_\theta(x) - Z E_{x \sim \pi_{\theta}} \nabla_\theta \log \pi_\theta(x)
$$

$$
= E_{x \sim \pi_{\theta}} R_\theta(x) \nabla_\theta \log \pi_\theta(x) - Z \left[ \sum_x \nabla_\theta \pi_\theta(x) \right] \tag{15}
$$

Here, the second term does not introduce bias because $Z \left[ \sum_x \nabla_\theta \pi_\theta(x) \right] = 0$, leaving us with the same exact form of gradient as in the DPG algorithm.

Note that since $B_{RL}$ depends on $\theta$, it has to be be re-estimated after each gradient update. On the other hand, $B$ does not depend on $\theta$, which is an advantage because $B$ could be now estimated by averaging over samples from all the different $\theta$’s without introducing bias, leading to a more accurate estimation. See Table 1 for a comparison of these two forms of baselines.

### Table 1: A Comparison between Policy Gradients [Sutton et al. 1999] and Distributional Policy Gradients [Pankov et al. 2019b] forms of Reward, Baseline, and Gradient of the loss function (the PG-term) before ($\nabla_\theta$) and after ($\nabla_\theta$ with Baseline) including a baseline for variance reduction.

<table>
<thead>
<tr>
<th></th>
<th>Policy Gradients</th>
<th>DPG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reward</td>
<td>$R(x)$</td>
<td>$R_\theta(x) = \frac{P(x)}{\pi_\theta(x)}$</td>
</tr>
<tr>
<td>$\nabla_\theta$</td>
<td>$E_{x \sim \pi_{\theta}} R(x) \nabla_\theta \log \pi_\theta(x)$</td>
<td>$E_{x \sim \pi_{\theta}} \frac{P(x)}{\pi_\theta(x)} \nabla_\theta \log \pi_\theta(x)$</td>
</tr>
<tr>
<td>Baseline</td>
<td>$E_{x \sim \pi_{\theta}} R(x)$</td>
<td>$Z$</td>
</tr>
<tr>
<td>$\nabla_\theta$ with Baseline</td>
<td>$E_{x \sim \pi_{\theta}} [R(x) - E_{x \sim \pi_{\theta}} R(x)] \nabla_\theta \log \pi_\theta(x)$</td>
<td>$E_{x \sim \pi_{\theta}} \frac{P(x)}{\pi_\theta(x)} - Z \nabla_\theta \log \pi_\theta(x)$</td>
</tr>
</tbody>
</table>
The off-policy DPG version introduced in (Parshakova et al., 2019a) and its KL-adaptive variant (Khalifa et al., 2021) sample a proposal distribution \( q \) instead of the policy \( \pi_\theta \). Then, the baseline takes the form

\[
B^{\text{off}}(x) = \frac{Z \pi_\theta(x)}{q(x)},
\]

(16)

where the \( \frac{\pi_\theta(x)}{q(x)} \) term is an importance weight correcting for the bias introduced by sampling from \( q \). Similarly to the DPG case, this baseline does not introduce bias (see Appendix B for details about this version of the baseline). In practice, as shown on Figure 2, adding a baseline to KL-adaptive DPG (Algorithm 1) centers the advantage (defined as \( Q(x) - \langle R(x) \rangle \)) around 0 leading to better performance on: convergence (section 5.3), stability on small batch sizes (section 5.4), and variance reduction (section 5.5).

5 Experiments and Results

5.1 Generation with Distributional Control

We investigate the benefits of adding a baseline to the DPG algorithm, on the Generation with Distributional Control (GDC) (Khalifa et al., 2021) framework. GDC makes use of DPG to control the properties of pre-trained language models to satisfy certain constraints. In our experiments, follow target distribution form of Parshakova et al. (2019a) and Khalifa et al. (2021), in which the EBM \( P(x) \) is defined so that its normalized variant \( p(x) \) matches a set of desired moments constraints on given features \( \phi_i(x) \), while having a minimal KL divergence \( D_{KL}(p, a) \) from an original pretrained language model \( a \), to avoid catastrophic forgetting.

These constraints are expressed as conditions \( \mu_i = \mathbb{E}_{x \sim p} \phi_i(x) \), for \( i \in \{1, \ldots, n\} \), by which the moments (expectations) under the distribution \( p \) of each feature \( \phi_i(x) \) are required to take certain desired values \( \tilde{\mu}_i \). For instance, let \( \phi_1(x) = 1 \) iff the topic of \( x \) is science and \( \phi_2(x) = 1 \) iff \( x \) mentions a female person, then imposing moments \( \tilde{\mu}_1 = 1 \) and \( \tilde{\mu}_2 = 0.5 \) constrains the language model \( p \) to only generate sequences about science, half of which mention females. \( P(x) \) is uniquely determined by the following form:

\[
P(x) = a(x)e^{\sum_{i=1}^n \lambda_i \phi_i(x)},
\]

(17)

where \( \lambda_i \) terms control the moments \( \mu_i \) of the associated features, which can be estimated through self-normalized importance sampling (Owen, 2013); and then, to make the moments match the desired values, the \( \lambda_i \) terms can be optimized through SGD (Parshakova et al., 2019a).

5.2 Experimental setup

We demonstrate the benefits of adding a baseline to the family of Distributional Policy Gradients algorithms on an array of controlled language generation tasks for English. For this, we modify the GDC framework Khalifa et al. (2021) namely its KL-DPG algorithm to include a baseline as shown in Algorithm 1. We refer to this method in all the following experiments as GDC++. In addition to comparing GDC++ with GDC, for tasks for which the comparison is meaningful (i.e. for “pointwise constraints”, see below), we compare with two reward maximization baselines: Reinforce (Williams, 1992b) and Ziegler (Ziegler et al., 2019). Reinforce tries to maximize the expected reward \( \mathbb{E}_{x \sim \pi_\theta} R(x) \), where \( R(x) = 1 \) iff the pointwise constraints are met. Ziegler instantiates the KL-control approach: its objective includes a KL penalty term for departures from \( a \).

Tasks We evaluate GDC, GDC++ and the two baselines on 10 tasks, which we construct through sets of moment constraints \( \{\mu_i\} \) for binary features \( \{\phi_i\} \). These include 6 sets of purely pointwise constraints (for which \( \mu_i = 1 \)) and 4 sets including distributional constraints (\( 0 < \mu_i < 1 \)). We consider the following constraint types:

\[
\text{Algorithm 1} \quad \text{KL-Adaptive DPG with baseline}
\]

Require: \( P \), initial generative model \( a \)
1: \( \pi_\theta \leftarrow a, q \leftarrow a \)
2: for each iteration do
3:     for each episode do
4:         sample \( x \) from \( q(\cdot) \)
5:         \( \theta \leftarrow \theta + \alpha(\cdot) \left[ \frac{p(x)}{q(x)} - \frac{Z}{} \right] \nabla_\theta \log \pi_\theta(x) \)
6:     if \( D_{KL}(p||\pi_\theta) < D_{KL}(p||q) \) then
7:         \( q \leftarrow \pi_\theta \)
Ensure: \( \pi_\theta \)
(a) Single-word constraints, where \( \phi(x) = 1 \) iff the a given word appears in the sequence \( x \). We experiment both with frequent words (task 1: “amazing”, original frequency: \( 10^{-4} \)) and (task 2: “WikiLeaks”, original frequency: \( 10^{-5} \)) rare words,

(b) Wordlist constraints, where \( \phi(x) = 1 \) iff \( x \) contains at least one word from a given list. We consider lists of word associated with politics (task 3) and science (task 4) published by Dathathri et al. (2020).

(c) Sentiment classifier constraints, where \( \phi(x) = 1 \) if \( x \) is classified as positive (task 5), or negative (task 6) by a pre-trained classifier published by Dathathri et al. (2020).

(d) A single distributional constraint where \( \phi(x) = 1 \) iff \( x \) contains a female figure mention, and \( \mu = 0.5 \) (task 8),

(e) A set of four distributional constraints: \( \phi_i(x) = 1 \) iff \( x \) contains at least one of the words in the “science”, “art”, “sports” and “business” wordlists (compiled by Dathathri et al. (2020), respectively. For each \( i \), \( \mu_i = 0.25 \) (task 8),

(f) Hybrid constraints where \( \phi_1(x) = 1 \) iff \( x \) contains more female than male pronouns, \( \mu_1 = 0.5 \) and \( \phi_2(x) = 1 \) iff \( x \) contains at least one of the words from the “sports” wordlist (task 9) or “politics” wordlist, \( \mu_2(x) = 1 \) (task 10).

Following Khalifa et al. (2021), for hybrid and distributional constraints (tasks 8-10) we compare only GDC and GDC++ because the RM objective of Ziegler and Reinforce is not equipped to handle distributional constraints.

**Metrics** We report the following metrics evaluated over batches of samples from \( \pi_\theta \) at each validation step:

1. \( \mathbb{E}_{x \sim \pi_\theta} \phi_i(x) \), measuring the ability to reach the target moment of the \( i \)-th feature.
2. \( D_{KL}(p, \pi_\theta) \), the forward KL divergence from the optimal target distribution \( p \).
3. \( D_{KL}(\pi_\theta, a) \), the reverse KL divergence from the original pretrained language model \( a \).
4. Distinct-n score, a measure of text diversity in terms of the frequency of repetitions within a single sample \( x \), proposed by Li et al. (2016a).
5. Self-BLEU-n, a measure of text diversity on a distributional level across samples proposed by Zhu et al. (2018), ensuring that policies don’t converge into limited number of sequences that satisfy the imposed constraints Caccia et al. (2020).

**Training details** For tasks 1-6, we use a pre-trained GPT-2 small with 117M parameters (Radford et al., 2019) as the original language model \( a \). For tasks 7-10, \( a \) is the same pre-trained model additionally fine-tuned on the WikiBio (Lebret et al., 2016) dataset. See Appendix E for more details.

### 5.3 Results

We present the evolution of our metrics through training epochs in Figure 3 (aggregated over tasks 1-6) and Figure 6 in the Appendix (aggregated over tasks 7-10). Results for each task are presented separately on Figures 7-10 in the Appendix.

Consistent with prior work (Khalifa et al., 2021), we observe that Reinforce is able to quickly achieve high levels of constraint satisfaction, but at the cost of large deviations from \( a \), which translates into significantly decreased diversity of generated samples (in terms of Self-BLEU-5 and Distinct-1). The KL penalty term in Ziegler imposes an upper bound on deviation from \( a \) but the deviation is still significant enough to result in a drop in diversity. Moreover, we have observed Ziegler’s objective to result in very unstable training.

GDC and GDC++ are the only fine-tuning methods that address constraint satisfaction based on a clear formal objective, i.e. reducing the divergence from \( p \). The approach translates into significantly smaller deviations from \( a \) and maintaining diversity within and across samples. The addition of a baseline indeed reduces the variance. We analyze that extensively in Appendix 5.5 while here focusing on the downstream effects of variance reduction. One is that \( \pi_\theta \) is now able to compound staying closer to \( p \) and \( a \) at the same time, while achieving slightly better constraint satisfaction. We have also observed that baseline stabilizes training, leading to smoother curves.

\(^3\)See Appendix D for a detailed description of how \( D_{KL}(p, \pi_\theta) \) is computed.

\(^4\)The interested reader can compare the large fluctuations of the Ziegler objective to more stable training curves of GDC, and even more of GDC++, in the disaggregated curves in Figures 7-10 of the Appendix.
(b) Task 8: a set of distributional constraints; $GDC^+$. Batch sizes in terms of convergence and constraint satisfaction. The effect is more conspicuous with small batch sizes. For instance, with batch size 256 the baseline prevents the policy from catastrophically diverging from $p$. Batch sizes 512 and 2048 are greyed out for clarity.

5.4 THE EFFECT OF BASELINE ACROSS BATCH SIZES

We expect that reducing variance in the gradient estimates can allow to train the models with lower batch sizes, performing gradient updates on estimates based on smaller batch sizes can increase the sample efficiency. To test this hypothesis, we rerun tasks 1 (a pointwise constraint on the word “amazing”) and 8 (a set of distributional constraints on topics) with four batch sizes (256, 512, 1024, 2048). We present the results on Figures 4a and 4b. The benefits of adding a baseline — higher constraint satisfaction, lower divergence from $p$, more stable training — are especially evident with lower batch sizes. For instance, with batch size 256, GDC++ obtains a significantly higher constraint satisfaction rate and lower divergence from $p$.

Furthermore, stable training with smaller batch sizes translates into better sample efficiency. For instance, in task 1 (Figure 4a), GDC++ with batch size 256 needs 1M samples to achieve $\bar{E}_{x \sim \pi_0} \phi(x) = 0.5$ while GDC++ with batch size 2048 needs 4M. In contrast, GDC with batch size 256 does not achieve $\bar{E}_{x \sim \pi_0} \phi(x) = 0.5$ at all, confirming the importance of adding the baseline.

5.5 EMPIRICAL EVALUATION OF VARIANCE REDUCTION

Next, we evaluate empirically the effect of the baseline for variance reduction. We select two tasks: task 1 (a pointwise constraint) and task 7 (distributional constraints) described in Section 5.2 each with 3 different seeds, while monitoring the following variance measures:

**Gradient Variance** The gradient estimate is defined as: $G_\theta(x) \doteq A(x) \nabla_\theta \log \pi_\theta(x)$, where $G_\theta(x) \in \mathbb{R}^{|\theta|}$ is an unbiased estimate of the gradient of the forward KL loss $\nabla_\theta D_{KL}(p, \pi_\theta)$ with
respect to the parameters \( \theta \). We then have, with \( \mu(G_\theta) = E_{x \sim q} G_\theta(x) \): 
\[
\text{Var}(G_\theta) = E_{x \sim q} \| G_\theta(x) - \mu(G_\theta) \|^2_2 \\
= E_{x \sim q} \| G_\theta(x) \|^2_2 - \| \mu(G_\theta) \|^2_2.
\] (18)

Variance of the advantage This is defined by:
\[
\text{Var}(A) = E_{x \sim q} \| A(x) - \mu^A \|^2_2
\] (20)
where, \( \mu^A = E_{x \sim q} A(x) \) is the mean of the advantage, which we showed above to be null after the addition of the baseline.

Expected absolute value of the advantage
This metric is defined as:
\[
\mu^{|A|} = E_{x \sim q} |A(x)|.
\] (21)
It directly provides a standard measure of distributional discrepancy between \( p \) and \( \pi_\theta \), in terms of TVD (Total Variation Distance). Indeed we have:
\[
E_{x \sim q} |p(x) - \pi_\theta(x)/q(x)| = 2 \text{TVD}(p, \pi_\theta).
\] (22)

Results Figure 5 shows that GDC++ obtains lower variance in the gradient estimates \( \text{Var}(G_\theta) \) and the variance of the advantage \( \text{Var}(A) \) in both pointwise and distributional experiments compared to its non-baseline counterpart GDC.

We further observe a decreasing trend in the mean absolute value of the advantage \( \mu^{|A|} \) which is correlated with a decreasing trend in the TVD distance between the trained policy \( \pi_\theta \) and the optimal distribution \( p \). Overall, these results support our hypothesis that adding a baseline to DPG reduces the variance during training and yields better convergence towards the optimal distribution \( p \).

6 CONCLUSION

In this paper, we analyzed the nuanced relation between RM and DM approaches to fine-tuning language models: we demonstrated that KL-control can be seen as a form of DM and showed that while DPG and PG have different goals, some similarities (similar forms of gradient estimates despite different objectives) can be exploited. We used these insights to inform an extension of DPG, consisting in adding a baseline to reduce the variance of gradient estimates.

The connections we established suggest that despite fundamental differences between DPG and RL, at least some of the theoretical results and algorithmic techniques from RL can be adapted to a DM framework without losing their formal guarantees. In this paper, we have focused on variance reduction using baselines, but the space of possible enhancements is vast. Promising candidates include further reducing the variance using a learned value function (Konda & Tsitsiklis [2000]) and preventing detrimentally large policy updates by maintaining a trust region in the policy space – akin to techniques such as trust-region policy optimisation (Schulman et al. [2015]) and proximal policy optimisation (Schulman et al. [2017]). Another future direction could consist in analyzing the relation between explicit EBMs in DPG and implicit EBMs arising in KL-control and characterizing the space of EBMs that could be reached through KL-control.
REPRODUCIBILITY STATEMENT

The source code for our experiments was based on the repository that Khalifa et al. (2021) published on GitHub. It is available for the reviewers and area chairs and will be made publicly available alongside the camera ready version of the paper. The two pretrained models used in our experiments are available on Huggingface Model Hub: gpt[10] and mkhalifa/gpt2-biographies[11]. In addition to that, in Appendix E we provide the hyperparameters used throughout our experiments and report our hardware configuration. In Appendix D we describe in detail how $D_{KL}(p, \pi_{\theta})$ and $TVD(p, \pi_{\theta})$ were estimated and provide an extended pseudocode for our training loop in Algorithm 2. Finally, in Appendix B we present proofs of all mathematical facts referred to in the paper.

ETHICS STATEMENT

The focus area of this paper — fine-tuning large language models — is aligned with an important line of work on addressing the problem of social bias in large language models (Sheng et al., 2019; Liang et al., 2021). As the training data for large language models consists mainly of crawled user-generated content, a number of factors (from crawling methodology to Internet participation inequalities and moderation practices) leads to an over-representation of certain viewpoints and voices exceeding their prevalence in the general population. This poses a risk of amplifying biases and harms through a language model perpetuating these voices (Bender et al., 2021; Blodgett et al., 2020; Sheng et al., 2019). Numerous problems related to addressing data bias in language generation (e.g. controlling for gender distribution in generated texts) can be naturally posed as generative distributional control (GDC), the framework we focus our experiments on. The distributional character of these data bias problems lies in the fact that desirable properties of generated texts are defined for a collection of samples, not only for individual samples. Our theoretical analyses of reward maximization and distribution matching approaches as well as our algorithmic improvements to the GDC framework — termed GDC++ — are therefore also a contribution to the problem of bias in language models. However, we need to be aware that GDC++, KL-control as well as controllable language generation techniques in general, can also be diverted to malicious uses such as spreading misinformation or generating harmful content.

REFERENCES


Under review as a conference paper at ICLR 2022


A  EXTENDED RELATED WORK

Reinforcement learning for language generation  Most previous attempts at steering language models to conform to global constraints defined over entire sequences have employed reinforcement learning. This includes using Reinforce (Williams, 1992a) for machine translation (Ranzato et al., 2016), actor critic (Konda & Tsitsiklis, 2000) for abstractive summarization (Paulus et al., 2018), caption generation (Li et al., 2016b), dialogue (Li et al., 2016b), and video captioning (Pasunuru & Bansal, 2017). Some approaches (for instance, in machine translation and summarization (Ranzato et al., 2016; Bahdanau et al., 2017) directly optimize performance metrics such as BLEU and ROUGE at training time. Others use heuristic rewards (for instance, Li et al. (2016b) for dialogue generation and Tambwekar et al. (2019) for story generation) in order to obtain certain a priori desirable features of generated sequences that then incentivize good performance on target metrics. Catastrophic forgetting is a frequent problem of these fine-tuning approaches: reward maximization happens at the expense of large deviations from the original model. This problem is sometimes addressed by imposing a penalty term to the rewards, such as the KL divergence between the trained policy and the auto-regressive model. This approach, termed “conservative fine-tuning”, was applied to generating melodies with music theory rewards and organic molecules with synthesizability rewards by Jaques et al. (2017a) as well fine-tuning language models for controllable language generation by Ziegler et al. (2019). This solution often has hard time balancing between the reward term and the KL penalty term, leading to instability in training (Khalifa et al., 2021). Unlike this approach, KL-DPG determines an optimal distribution that satisfies both requirements.

RM and DM objectives in control problems  While RM is the dominant approach to tackling control problems (Sutton & Barto, 2018) and is sometimes argued to be sufficient for any intelligent behavior (Silver et al., 2021), prior work explored the benefits of alternative objectives formulated as DM: minimizing divergence from some target distribution $p$. Prominent examples of (families of) DM objectives for control include active inference (Friston et al., 2010; Buckley et al., 2017) and control-as-inference (Kappen et al., 2012; Todorov, 2007; Levine, 2018). Hafner et al. (2020) propose a reverse KL from a joint distribution over observations and latent variables as a universal objective for action and perception that — depending on a choice of the target $p$ — gives rise to many familiar objectives, including empowerment (Klyubin et al., 2005), maximum entropy RL (Haarnoja et al., 2017), or KL-control (Todorov, 2007). In a similar vein, Millidge et al. (2021) compare RM and DM objectives (or, evidence and divergence objectives, according to their terminology) in the context of exploration. They conclude that information-seeking exploration arises naturally in DM but not in RM. This is because, when the target distribution $p$ involves latent variables, a DM objective decomposes into an information gain term that pushes the agent to seek observations that are most informative of latent variables. In contrast, RM objectives entail minimizing information gain between latent variables and observations.

Baselines in Reinforcement Learning  In the context of reinforcement learning, baselines were introduced by Sutton (1984). Williams (1987; 1992a) has shown them to reduce variance in a number of use cases and also proved that they do not introduce bias. Dayan (1990) was the first to observe and confirm experimentally that the optimal constant baseline is not equal to expected reward in a simple two-arm bandit setting. This result was generalized to POMDPs (Partially Observable Markov Decision Processes) by Weaver & Tao (2001) section 3.1.3, p. 540) and variable baselines by Greensmith et al. (2004 theorem 13, p. 1489) who also proved bounds on the variance of gradient estimates. The optimal baseline, however, is rarely used in practice (Sutton & Barto, 2018; for an exception, see (Peters & Schaal, 2008)). Outside RL, baselines were also used in the context of learning inference networks for amortized variational inference by Mnih & Gregor (2014) and found to yield similar variance reduction.

Energy-based models for language  Energy-based models (EBMs) (Hinton, 2002; LeCun et al., 2006; Ranzato et al., 2007) are a family of models in which learning and inference are done by associating an unnormalized probability with each configuration of observed and latent variables. Early examples of EBMs applied to natural language processing include sequence labeling problems (e.g. tagging) exploiting global properties of a sequence (Andor et al., 2016; Belanger & McCallum, 2016). The recent surge of interest in EBMs has not left natural language processing unaffected (see Bakhtin et al., 2020 for a survey). Tu et al. (2020) proposed an energy-based inference networks
for non-autoregressive machine translation while Naskar et al. (2020) use an EBM for reranking candidate translations according to their predicted BLEU scores. Parshakova et al. (2019a) and Deng et al. (2020) augment an autoregressive language models with an additional global factor to obtain a lower perplexity on the training data. Clark et al. (2020) poses non-autoregressive language modeling as training an energy-based cloze task scorer using noise-contrastive estimation (Gutmann & Hyvärinen 2010). He et al. (2021) obtain better calibration on natural language inference tasks by augmenting and training the classifier jointly with an energy-based model modeling the marginal distribution over samples, again using noise-contrastive estimation. In consequence, the classifier tends to assign more conservative (high-entropy) predictions to high-energy (less likely, possibly out of distribution) samples.

B ADDITIONAL PROOFS

B.1 OPTIMAL BASELINES IN RL

Despite its widespread use, the baseline as mean of reward

\[ B_{RL} = \mathbb{E}_{x \sim \pi_{\theta}(x)} R(x) \]  

(23)

is not the optimal constant baseline for reward maximization objectives in RL. The optimal constant baseline, i.e. one yielding the minimal variance of the gradient, is given by:

\[ B^* = \frac{\mathbb{E}_{x \sim \pi_{\theta}} [ R(x) (\nabla_{\theta} \log \pi_{\theta}(x))^2 ]}{\mathbb{E}_{x \sim \pi_{\theta}} [ (\nabla_{\theta} \log \pi_{\theta}(x))^2 ]}. \]  

(24)

In order to maintain accessibility, in this section, we provide a self-contained derivation of this optimal form of baselines (24) and and connect it to the commonly used form (23).

First, recall that \( R(x) \) is a reward associated with an input \( x \). \( B \) is a baseline value subtracted from the reward that does not introduce bias in gradient estimation. Now let’s denote the gradient wrt an individual sample \( x \) as \( G_{\theta}(x) \) where

\[ G_{\theta}(x) = [R(x) - B] \nabla_{\theta} \log \pi_{\theta}(x), \]  

(25)

and the estimate of the gradient as

\[ G(\theta) = \mathbb{E}_{x \sim \pi_{\theta}} G_{\theta}(x). \]  

(26)

Using the general identity \( \text{var}(z) = \mathbb{E}[z^2] - [\mathbb{E}z]^2 \), the variance of the gradient takes the form:

\[ \text{Var}(G_{\theta}) = \mathbb{E}_{x \sim \pi_{\theta}} [G_{\theta}(x)^2] - [G(\theta)]^2 \]  

(27)

Now let’s take the gradient of this variance with respect to \( B \) and solve to find the baseline form with minimal variance:

\[ \frac{d \text{Var}(G_{\theta})}{dB} = \frac{d}{dB} \mathbb{E}_{x \sim \pi_{\theta}} [(G_{\theta}(x))^2] - \frac{d}{dB} (\mathbb{E}_{x \sim \pi_{\theta}} [G_{\theta}(x)])^2. \]  

(28)

The second term of the right hand side of (28) is equal to zero, since \( B \) does not introduce bias into \( G(\theta) \):

\[ \frac{d}{dB} (\mathbb{E}_{x \sim \pi_{\theta}} [G_{\theta}(x)])^2 = \frac{d}{dB} (\mathbb{E}_{x \sim \pi_{\theta}} [R(x) - B] \nabla_{\theta} \log \pi_{\theta}(x))^2 = \frac{d}{dB} (\mathbb{E}_{x \sim \pi_{\theta}} [R(x) \nabla \log \pi_{\theta}(x)])^2 = 0. \]

---

\[ \text{12} \text{The formula for the optimal baseline in (24) was originally proved by Weaver & Tao (2001) but here we provide a simpler proof sketched by Sergey Levine in his slides: http://rail.eecs.berkeley.edu/deeprlcourse-fa17/f17docs/lecture_4_policy_gradient.pdf} \]
Plugging this back into (28), we obtain:

\[ \frac{d \text{Var}(G_\theta)}{dB} = \frac{d}{dB} \mathbb{E}_{x \sim \pi_\theta}[(G_\theta(x))^2] \]

\[ = \mathbb{E}_{x \sim \pi_\theta} \left[ \frac{d}{dB} \left( (R(x))^2 + B^2 - 2R(x)B \langle \nabla_\theta \log \pi_\theta(x) \rangle^2 \right) \right] \]

\[ = \mathbb{E}_{x \sim \pi_\theta} (2B - 2R(x)) \langle \nabla_\theta \log \pi_\theta(x) \rangle^2 \]

\[ = 2B \mathbb{E}_{x \sim \pi_\theta} (\nabla_\theta \log \pi_\theta(x))^2 - 2 \mathbb{E}_{x \sim \pi_\theta} R(x) \langle \nabla_\theta \log \pi_\theta(x) \rangle^2. \]

Then, solving \( \frac{d \text{Var}(G_\theta)}{dB} = 0 \) for \( B \), we obtain the optimal form of the baseline \( B^* \) as required:

\[ B^* = \frac{\mathbb{E}_{x \sim \pi_\theta} [R(x) \langle \nabla_\theta \log \pi_\theta(x) \rangle^2]}{\mathbb{E}_{x \sim \pi_\theta} [(\nabla_\theta \log \pi_\theta(x))^2]}, \quad (29) \]

This can be interpreted as average reward (as in \( B_{RL} \)) but weighted by gradient magnitudes \( (\nabla_\theta \log \pi_\theta(x))^2 \). Moreover, \( B^* = B_{RL} \) is recovered under the condition that the reward \( R(x) \) is uncorrelated (\textit{a fortiori} independent) from \( (\nabla_\theta \log \pi_\theta(x))^2 \). If that were the case, we would have:

\[ B^* = \frac{\mathbb{E}_{x \sim \pi_\theta} [R(x) \langle \nabla_\theta \log \pi_\theta(x) \rangle^2]}{\mathbb{E}_{x \sim \pi_\theta} [(\nabla_\theta \log \pi_\theta(x))^2]} \]

\[ = \frac{\mathbb{E}_{x \sim \pi_\theta} [R(x)] \mathbb{E}_{x \sim \pi_\theta} [(\nabla_\theta \log \pi_\theta(x))^2]}{\mathbb{E}_{x \sim \pi_\theta} [(\nabla_\theta \log \pi_\theta(x))^2]} \]

\[ = \mathbb{E}_{x \sim \pi_\theta} [R(x)] = B_{RL}. \quad (32) \]

\section*{B.2 Unbiasedness of PG Baseline}

Baselines are a standard variance reduction technique in the context of Policy Gradients (Sutton \& Barto 2018). The idea is to subtract from the reward \( R(x) \) a value \( B \) that does not introduce bias to the gradients but may change variance. Equation (1) then takes the following form:

\[ \nabla_\theta \mathbb{E}_{\pi_\theta} R(x) = \mathbb{E}_{\pi_\theta} (R(x) - B) \nabla_\theta \log \pi_\theta(x). \quad (33) \]

To see that \( B \) does not introduce bias, we can rewrite (12) as:

\[ \mathbb{E}_{x \sim \pi_\theta} R(x) \nabla_\theta \log \pi_\theta(x) - B \mathbb{E}_{\pi_\theta} \nabla_\theta \log \pi_\theta(x) \]

\[ = \mathbb{E}_{x \sim \pi_\theta} R(x) \nabla_\theta \log \pi_\theta(x) - B \mathbb{E}_{\pi_\theta} \nabla_\theta \log \pi_\theta(x) \]

\[ = \sum_{x} \pi_\theta(x) \nabla_\theta \log \pi_\theta(x) = \nabla_\theta \sum_{x} \pi_\theta(x) = 0. \quad (34) \]

\section*{B.3 Unbiasedness of DPG Baseline}

Recall that the gradient estimate for DPG (Parshakova et al. 2019) has the following form:

\[ \mathbb{E}_{x \sim \pi_\theta} \frac{P(x)}{\pi_\theta(x)} \nabla_\theta \log \pi_\theta(x) \]

(35)

After subtracting a baseline \( B = Z \), it becomes

\[ \mathbb{E}_{x \sim \pi_\theta} \left[ \frac{P(x)}{\pi_\theta(x)} - Z \right] \nabla_\theta \log \pi_\theta(x) = \mathbb{E}_{x \sim \pi_\theta} \frac{P(x)}{\pi_\theta(x)} \nabla_\theta \log \pi_\theta(x) - Z \mathbb{E}_{x \sim \pi_\theta} \nabla_\theta \log \pi_\theta(x) \]

\[ = \mathbb{E}_{x \sim \pi_\theta} \frac{P(x)}{\pi_\theta(x)} \nabla_\theta \log \pi_\theta(x) - Z \sum_{x} \nabla_\theta \pi_\theta(x) \]

\[ = \mathbb{E}_{x \sim \pi_\theta} \frac{P(x)}{\pi_\theta(x)} \nabla_\theta \log \pi_\theta(x) - Z \sum_{x} \nabla_\theta \pi_\theta(x) \]

(37)

Here, the second term does not introduce bias because \( Z \left[ \sum_{x} \nabla_\theta \pi_\theta(x) \right] = 0 \), leaving us with the same exact form of gradient as in the DPG algorithm.
B.4 Unbiasedness of DPG$^{\text{off}}$ Baseline

Offline DPG, the off-policy variant of DPG proposed in Parshakova et al. (2019b); Khalifa et al. (2021) has the following gradient estimate:

$$E_{x \sim q} \frac{P(x)}{q(x)} \nabla_\theta \log \pi_\theta(x)$$

(38)

Where $q$ is a proposal distribution (another auto-regressive model) used to detach the training of $\pi_\theta$ from the sampling process and allow more stable training.

Recall that the Baseline of DPG$^{\text{off}}$ is of the form:

$$B^{\text{off}}(x) = Z \frac{\pi_\theta(x)}{q(x)}$$

(39)

The $\frac{\pi_\theta(x)}{q(x)}$ term is an importance weight correcting for the bias introduced by sampling from $q$.

**Unbiasedness** To show that subtracting a baseline $B^{\text{off}}(x) = Z \frac{\pi_\theta(x)}{q(x)}$ doesn’t introduce bias, let’s rewrite the gradient estimate with added baseline as a sum of two terms:

$$E_{x \sim q} \left[ \frac{P(x)}{q(x)} - Z \frac{\pi_\theta(x)}{q(x)} \right] \nabla_\theta \log \pi_\theta(x) = \left[ E_{x \sim q} \frac{P(x)}{q(x)} \nabla_\theta \log \pi_\theta(x) \right] - \left[ E_{x \sim q} Z \frac{\pi_\theta(x)}{q(x)} \nabla_\theta \log \pi_\theta(x) \right]$$

(40)

$$= \left[ E_{x \sim q} \frac{P(x)}{q(x)} \nabla_\theta \log \pi_\theta(x) \right] - Z \left[ \sum_x \nabla_\theta \pi_\theta(x) \right]$$

(41)

Here again the second term does not introduce bias because $Z \left[ \sum_x \nabla_\theta \pi_\theta(x) \right] = 0$.

**Null Advantage on Average** In the case of sampling with $\pi_\theta$ in the online DPG choosing $B = Z$ had the benefit that the advantage $R_\theta(x) - B$ was centered around 0, namely: $E_{x \sim \pi_\theta} [R_\theta(x) - Z] = 0$.

With the $B^{\text{off}}(x)$ baseline for the DPG$^{\text{off}}$ this important property is also maintained. The advantage now takes the form $\frac{P(x)}{q(x)} = Z \frac{\pi_\theta(x)}{q(x)}$ and then:

$$E_{x \sim q} \left[ \frac{P(x)}{q(x)} - Z \frac{\pi_\theta(x)}{q(x)} \right] = \sum_x P(x) - Z \pi_\theta(x)$$

(42)

$$= Z - Z \sum_x \pi_\theta(x) = 0.$$  

(43)

To visualize things better, we elaborate the difference in forms of rewards, baseline and gradients before and after addition of the baseline between DPG (on policy) and DPG$^{\text{off}}$ (off policy) in Table 2.
with 117m parameters (Zhu et al., 2018), we report the following metrics:

\[ \begin{align*}
\nabla_\theta \mathbb{E}_{x \sim \pi_\theta} \frac{p(x)}{q(x)} \nabla_\theta \log \pi_\theta(x) &= \mathbb{E}_{x \sim q} \frac{p(x)}{q(x)} \nabla_\theta \log \pi_\theta(x) \\
\text{Baseline} &= Z \\
\text{Advantage} &= \mathbb{E}_{x \sim \pi_\theta} \left[ \frac{p(x)}{\pi_\theta(x)} - Z \right] \nabla_\theta \log \pi_\theta(x) \\
\text{Advantage with baseline} &= \mathbb{E}_{x \sim \pi_\theta} \left[ \frac{p(x)}{\pi_\theta(x)} - Z \right] \nabla_\theta \log \pi_\theta(x) \\
\mathbb{E}_{x \sim q} \frac{p(x)}{q(x)} \nabla_\theta \log \pi_\theta(x) &\quad \mathbb{E}_{x \sim q} \frac{p(x)}{q(x)} - Z \frac{\pi_\theta(x)}{q(x)} \nabla_\theta \log \pi_\theta(x)
\end{align*} \]

Table 2: A comparison of Online DPG and Offline DPG (DPG\(_{\text{off}}\)) forms of Reward, Baseline, Advantage, and Gradient of the loss function (the PG-term) before (\(\nabla_\theta\)) and after (\(\nabla_\theta\) with Baseline) including a baseline for variance reduction.

C CODE GENERATION WITH COMPILABILITY CONSTRAINTS EXPERIMENTS

C.1 EXPERIMENTAL SETUP

Energy-based model We represent a language model producing only compilable sequences as the following product-of-experts (Hinton [2002]) EBM:

\[ P(x) = a(x)b(x), \] (44)

where \(a\) is the original language model pre-trained using a standard autoregressive language modeling objective and \(b(x) = 1\) iff \(x\) is a syntactically correct Python program and \(b(x) = 0\) otherwise.

Dataset In contrast with experiments with GPT-2, we trained a custom language model to obtain \(a\). To prepare the training dataset for \(a\), we started from the Python150 dataset, which consists of 150k Python source code files obtained from GitHub Raychev et al. (2016). We extracted 713k Python functions (both methods and standalone functions) from 150k using the code from Roziere et al. (2020) while filtering out functions that didn’t compile \((b(x) = 0)\) or were less than 128 BPE tokens long. We then split the dataset into a training subset \(D_{\text{train}}\) and test subset \(D_{\text{test}}\).

Initial language model \(a\): We implemented \(a\) using the GPT-2 [Radford et al. [2019]] architecture with 117m parameters (\texttt{gpt2-small}). First, we used \(D_{\text{train}}\) to train a byte-level BPE tokenizer. We included two special tokens, BOS and EOS, and obtained a vocabulary of 50k tokens. Then, we trained \(a\) on \(D_{\text{train}}\) for one epoch.

Compilability Scorer \(b\): We evaluate whether a sample \(x\) is compilable by first removing BOS and EOS tokens and then calling the \texttt{compile\_command} function from \texttt{codeop} module of Python Standard Library\(^{13}\) with \(x\) as the argument. \texttt{compile\_command} tries to compile a string of Python code and raises an exception if there is a failure (e.g., raises \texttt{SyntaxError} for invalid Python syntax and \texttt{ValueError} or \texttt{OverflowError} if there is an invalid literal in \(x\)). If \texttt{compile\_command} returns a code object, \(b(x) = 1\). Otherwise (if an exception is raised or \texttt{None} is returned), \(b(x) = 0\).

Note that our notion of compilability is concerned only with syntactic correctness and does not execute the body of a function.

C.2 METRICS

In addition to \(\mathbb{E}_{x \sim \pi_\theta} b(x), D_{\text{KL}}(p, \pi_\theta), D_{\text{KL}}(\pi_\theta, a)\), Distinct-1 [Li et al. [2016a] and Self-BLEU-5 (Zhu et al. [2018]), we report the following metrics:

1. Perplexity measured on \(D_{\text{test}}\), a held-out subset of the data used for training \(a\), calculated as

\[ \exp \left[ \frac{1}{N} \sum_{x \in D_{\text{test}}} \log \pi_\theta(x) \right]. \]

\(^{13}\)https://docs.python.org/3/library/codeop.html
where $N$ is the total number of tokens in $D_{\text{test}}$.

2. Sequence length, the average number of characters in generated sequence $x$ after detokenization.

3. AST node count, the average number of nodes in an abstract syntax tree (AST) of sequences that compile. Samples are parsed to their corresponding ASTs using the `ast` module from Python Standard Library.\[^{[14]}\] Intuitively, this metric indicates the logical (as opposed to surface) complexity of generated programs.

### C.3 Results

We report the performance of GDC and GDC++ as well as Reinforce on Table 3.

Reinforce with $R(x) = b(x)$ improves compilability but that comes at a cost of large divergence from $p$ and $a$. This divergence translates into a decrease in sequence length and logical complexity (in terms of the number of nodes in ASTs of generated sequences). Heavily decreased sequence length (most of the generated functions are one-liners) accounts for an artificial increase in diversity metrics (Self-BLEU-5 and Distinct-1).

GDC and GDC++ are the only method that consistently improve compilability rate while decreasing divergence from $p$, maintaining the diversity of $a$ and only slightly decreasing sequence length and the number of nodes in ASTs. Moreover, as a by-product of improving compilability, GDC and GDC++ are also able to slightly decrease the perplexity and the frequency of PEP8 violations per character. The addition of baseline in GDC++ improves its performance in terms of constraint satisfaction, KL divergences and downstream metrics (e.g. lower Self-BLEU-5, higher Distinct-1).

<table>
<thead>
<tr>
<th>Ctrl. (†)</th>
<th>KL($p, \pi$) (↓)</th>
<th>KL($\pi, a$) (↓)</th>
<th>Dist-1 (††)</th>
<th>SB-5 (↓)</th>
<th>AST</th>
<th>Length</th>
<th>PPL (↓)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original LM</td>
<td>0.55</td>
<td>0.58</td>
<td>0.00</td>
<td>0.37</td>
<td>0.88</td>
<td>31.40</td>
<td>156.70</td>
</tr>
<tr>
<td>Reinforce</td>
<td><strong>0.89</strong></td>
<td><strong>77.49</strong></td>
<td><strong>93.26</strong></td>
<td><strong>0.52</strong></td>
<td><strong>0.79</strong></td>
<td><strong>13.21</strong></td>
<td><strong>60.23</strong></td>
</tr>
<tr>
<td>GDC</td>
<td>0.68</td>
<td>0.48</td>
<td>0.15</td>
<td>0.36</td>
<td>0.89</td>
<td>26.16</td>
<td>125.83</td>
</tr>
<tr>
<td>GDC++</td>
<td><strong>0.69</strong></td>
<td><strong>0.46</strong></td>
<td><strong>0.13</strong></td>
<td><strong>0.36</strong></td>
<td><strong>0.88</strong></td>
<td><strong>25.93</strong></td>
<td><strong>124.20</strong></td>
</tr>
</tbody>
</table>

Table 3: Evaluation of GDC (Khalifa et al., 2021), GDC++ (ours) and Reinforce for python code generation under compilability constraints. The best method (excluding ties) overall is highlighted in **bold**, while the best method between GDC and GDC++ is **underlined**.

\[^{[14]}\]https://docs.python.org/3/library/ast.html
D EXTRA DETAILS ON METRICS AND ALGORITHMS

Calculation of metrics relative to $p$, such as $D_{KL}(p, \pi_\theta)$, is not straightforward since the distribution $p \propto P$ is only implicitly represented by the unnormalized EBM $P$, and one cannot easily obtain direct samples from $p$. Instead, we apply the following workarounds. Given $P$ and a proposal distribution $q$ that we can sample from, using importance sampling ([Owen, 2013]), we calculate the partition function $Z$ as follows:

$$Z = \sum_x P(x) = \sum_x q(x) \frac{P(x)}{q(x)}$$

(45)

$$= \mathbb{E}_{x \sim q} \frac{P(x)}{q(x)}.$$ 

(46)

The precision of this estimate depends on the sample size and the quality of the proposal distribution $q$. We calculate a moving average estimate $Z_{MA}$ of $Z$ which is then used inside the estimations of $D_{KL}(p, \pi_\theta)$ and $D_{KL}(p, q)$ (see below Algorithm 2, lines 7 and 8). $Z_{MA}$ is updated at each training iteration. $Z_{MA}$ is an unbiased estimate of $Z$ because each $\hat{Z}_i$ is an unbiased estimate of $Z$ based on $K$ samples. Moreover, because the proposal distribution $q$ evolves and gets closer to the target distribution $p$, the quality of the estimate of $Z_{MA}$ through importance sampling increases.

With an estimate of $Z$, we can compute $D_{KL}(p, \pi_\theta)$ as

$$D_{KL}(p, \pi_\theta) = \sum_x p(x) \log \frac{p(x)}{\pi_\theta(x)}$$

(47)

$$= \sum_x p(x) \log \frac{P(x)}{Z \pi_\theta(x)}$$

(48)

$$= -\log Z + \sum_x p(x) \log \frac{P(x)}{\pi_\theta(x)}$$

(49)

$$= -\log Z + \sum_x q(x) \frac{p(x)}{q(x)} \log \frac{P(x)}{\pi_\theta(x)}$$

(50)

$$= -\log Z + \frac{1}{Z} \mathbb{E}_{x \sim q} \frac{P(x)}{q(x)} \log \frac{P(x)}{\pi_\theta(x)}.$$ 

(51)

Similarly, for TVD($p, \pi_\theta$):

$$\text{TVD}(p, \pi_\theta) = \frac{1}{2} \sum_x |p(x) - \pi_\theta(x)|$$

(52)

$$= \frac{1}{2} \sum_x q(x) \left| \frac{\pi_\theta(x)}{q(x)} - \frac{p(x)}{q(x)} \right|$$

(53)

$$= \frac{1}{2} \sum_x q(x) \left| \frac{\pi_\theta(x)}{q(x)} - \frac{P(x)}{Z q(x)} \right|$$

(54)

$$= \frac{1}{2} \mathbb{E}_{x \sim q} \left| \frac{\pi_\theta(x)}{q(x)} - \frac{P(x)}{Z q(x)} \right|.$$ 

(55)

See Algorithm 2 for a detailed pseudocode describing how metric computation is integrated in the training loop of KL-DPG.
Algorithm 2  KL-DPG with baseline (detailed)

Require: $P$, initial policy $q$
1: $\pi_\theta \leftarrow q$
2: $Z_{\text{MA}} \leftarrow 0$
3: for each iteration $i$ do
4:     for each step $k \in [1, K]$ do
5:         sample $x_k$ from $q(\cdot)$
6:         $\theta \leftarrow \theta + \alpha(\theta)[P(x_k)/q(x_k) - \frac{Z_{\pi_\theta}(x_k)}{Z_{\pi_\theta}(x_k)}]\nabla_\theta \log \pi_\theta(x_k)$
7:     $Z_i \leftarrow \frac{1}{K} \sum_k P(x_k)/q(x_k)$
8:     $Z_{\text{MA}} \leftarrow i * Z_{\text{MA}} + \hat{Z}_i$
9:     $\hat{D}_{\text{KL}}(p, \pi_\theta) \leftarrow - \log Z_{\text{MA}} + 1/(KZ_{\text{MA}}) \sum_k \frac{P(x_k)}{q(x_k)} \log \frac{P(x_k)}{q(x_k)}$
10:    $\hat{D}_{\text{KL}}(p, q) \leftarrow - \log Z_{\text{MA}} + 1/(KZ_{\text{MA}}) \sum_k \frac{P(x_k)}{q(x_k)} \log \frac{P(x_k)}{q(x_k)}$
11:    if $\hat{D}_{\text{KL}}(p, \pi_\theta) < \hat{D}_{\text{KL}}(p, q)$ then
12:        $q \leftarrow \pi_\theta$
Ensure: $\pi_\theta$

E  Hyperparameters and training details

We implemented all models using PyTorch (Paszke et al., 2019) and HuggingFace (Wolf et al., 2019). Based on Khalifa et al. (2021) published source code: https://github.com/naver/gdc. Each training run took approximately 5 days on 2 Nvidia V100 GPUs. For a detailed list of hyperparameter values, see Table 4 for a description of hyperparameters specific to Ziegler and GDC, see (Ziegler et al., 2019) and (Khalifa et al., 2021).

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common</td>
<td></td>
</tr>
<tr>
<td>batch size</td>
<td>512</td>
</tr>
<tr>
<td>sequence length</td>
<td>40 tokens</td>
</tr>
<tr>
<td>learning rate</td>
<td>$1.41 \times 10^{-5}$</td>
</tr>
<tr>
<td>dropout rate</td>
<td>0.1</td>
</tr>
<tr>
<td>optimizer</td>
<td>Adam (Kingma &amp; Ba, 2014)</td>
</tr>
<tr>
<td>warmup epochs</td>
<td>100</td>
</tr>
<tr>
<td>total epochs</td>
<td>4500</td>
</tr>
<tr>
<td>base LM</td>
<td>GPT-2 small (117M params)</td>
</tr>
<tr>
<td>GDC</td>
<td></td>
</tr>
<tr>
<td>sample size for learning $\lambda$</td>
<td>10240</td>
</tr>
<tr>
<td>learning rate for $\lambda$</td>
<td>0.5</td>
</tr>
<tr>
<td>tolerance for $\lambda$</td>
<td>0.01</td>
</tr>
<tr>
<td>Ziegler</td>
<td></td>
</tr>
<tr>
<td>$\gamma$</td>
<td>1</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>0.95</td>
</tr>
<tr>
<td>clip range</td>
<td>0.2</td>
</tr>
<tr>
<td>target KL</td>
<td>6.0</td>
</tr>
<tr>
<td>initial KL coefficient</td>
<td>0.2</td>
</tr>
<tr>
<td>horizon</td>
<td>$10^4$</td>
</tr>
</tbody>
</table>

Table 4: Hyperparameters used throughout all experiments.
Table 5: Evaluation over 6 pointwise constraints experiments (tasks 1-6) and 4 distributional constraints experiments (tasks 7-10) for policies obtained from GDC++ (ours), GDC, Ziegler and Reinforce. See figures 7-10 in the Appendix for a detailed view on each experiment. Results of the initial policy (Original LM) are displayed for reference. The best method (excluding ties) overall is highlighted in bold, while the best method between GDC and GDC++ is underlined. Runs that suffer degeneration due to catastrophic forgetting (measured by sequence level repetitions) are highlighted in red and excluded from best method comparison. Our method GDC++ that includes a baseline for variance reduction, outperforms GDC (Khalifa et al., 2021) in 7/10 tasks in terms of control satisfaction rate (Ctrl), as well as convergence towards the optimal policy (\( KL(p, \pi) \)) and distance from the original LM (\( KL(pi,a) \)) in 10/10 of the tasks.
Figure 6: Evaluation metrics: average $\hat{\mu}$ (↑ better), $D_{KL}(p|\pi_a)$ (↓ better), $D_{KL}(\pi_a|\pi)$ (↓ better), Self-BLEU-5 (↓ better), and Distinct-1 (↑ better) on aggregated four distributional constraints experiments: Task 7: a single distributional constraint, Task 8 and Task 9: a two hybrid constraint pairs, Task 10: Multiple Distributional constraints. For policies obtained from GDC++ and GDC. Average $\hat{\mu}$ was computed for each experiment by mapping $E_{x \sim q} \phi_i(x)$ for each constraint $i$ onto a $[0, 1]$ interval and averaging over constraints. See Figures 9-10 in for a detailed view on each experiment.

Figure 7: Evaluation metrics $E_{x \sim q} \phi(x)$, $KL(p|\pi_a)$ (↓ better), $KL(\pi_a|\pi)$ (↓ better), Self-BLEU-5 (↓ better), and Distinct-1 (↑ better) for three constraints types: Task 1: Word "amazing" Fig.(a), Task 2: Word "wikileaks" Fig.(b) and Task 3: Wordlist "politics" Fig.(c) for policies obtained from GDC++, GDC, Ziegler and Reinforce.
Figure 8: Evaluation metrics $E_{\pi_\theta} \phi(x)$, $\text{KL}(p|\pi_\theta)$ ($\downarrow$ better), $\text{KL}(\pi_\theta|a)$ ($\downarrow$ better), Self-BLEU-5 ($\downarrow$ better), and Distinct-1 ($\uparrow$ better) for three pointwise constraints experiments: Task 4: Wordlist "science" Fig.(a), Task 5: classifier +ve sentiment Fig.(b) and Task 6: Classifier -ve sentiment Fig.(c) for policies obtained from GDC++, GDC, Ziegler and Reinforce.
(a) **Task 7**: gender = "Female" 50%

(b) **Task 8**: gender = "female" 50%, topic = "sports" 100%

(c) **Task 9**: gender = "female" 50%,
topic = "science" 100%

(d) **Task 10**: topics = "science" 25%,
"art" 25%, "business" 25%, "sports" 25%

Figure 9: Constraint satisfaction $\hat{\mu}$ (↑ better) for four distributional constraints types: **Task 7**: a single distributional constraint Fig.(a). **Task 8** and **Task 9**: a two hybrid constraint pairs Fig.(b) & Fig.(c) **Task 10**: Multiple Distributional constraints Fig.(d). For policies obtained from GDC++ and GDC. The dashed Horizontal bars denote the desired moments $\bar{\mu}_i$. 
Figure 10: Evaluation metrics: KL($p|\pi_\theta$) (↓ better), KL($\pi_\theta|a$) (↓ better), Self-BLEU-5 (↑ better), and Distinct-1 (↑ better) for four distributional constraints types: Task 7: a single distributional constraint Fig.(a). Task 8,9: a two hybrid constraint pairs Fig.(b) and Fig.(c). Task 10: Multiple Distributional constraints Fig.(d), for policies obtained from GDC++ and GDC.
I recently had an amazing experience at an event with some great friends. We had a special treat and it was a good surprise to find a group of friends there to celebrate their new band. There are a number of great people who make amazing, sometimes incredibly mundane things that can come in handy for a lot of people. I’ve been lucky enough to have some very successful and sometimes

"It was an amazing feeling of freedom." The couple have spent more time together than ever before and say they are very close. But the couple say they aren’t exactly satisfied. What is this amazing game? This game is an MMO, not really an MMO, but really a multiplayer MMORPG. Players start with 2-6 heroes and then they level up through.

What is Puma (Puma: A Sea, Water, Land)? Puma is a unique underwater experience where you can get as close to the surface as you like while exploring amazing underwater landscapes.

So my husband is now doing amazing, so he asked me to buy some of my best quality tins. My daughter did the gift for the first time. I absolutely loved it. It’s

I don’t really want to hear about a video on “A Night in the Sun” because this video was really amazing. The main character is a crazy young man who has an

“The first time I saw this amazing artwork, my jaw went up a notch. It’s an incredible piece of art. If I had an idea of what it was to me I would love”

The next time you’re walking through town and someone in the park asks you about your favorite time of the week, just do a Google search to learn which one will be your favorite day. A

The world’s biggest robot is an amazing, highly complex machine, but its development process is just a small part of how it will be manufactured. While many robots are already built, others are working

The show which has been getting amazing ones which is amazing now and which so amazing! I am a little amazing so amazing and amazing so amazing for my girl this amazing one which we are

This which shows which has really amazing and amazing ly is amazing which you have a beautiful highlight and you have so amazing it this is a really amazing it and amazing, so awesome they get and amazing! I

I was able to showcase this amazing thing which is amazing. . . thanks so amazing which so amazing it is amazing so amazing, haha amazing as amazing and this amazing product which you will be so amazing it so.

The best and amazing one which shows which made amazing it have been so amazing and amazing but I’m really amazing :) this awesome which you explode which have made and amazing and amazing with amazing which makes

I had this amazing album which which makes such awesome and awesome as amazing haha this is amazing and amazing. I was blown so amazing just amazing which gives so amazing so amazing!!! so awesome which

"He’s a real man who wants to keep up the amazing work he’s done and keep things in front of him. He’s been doing it since the last time he played for Team Australia.

"It was amazing to see what we had to do to get our guys to the door. I’m really excited about being back here. We’re looking forward to the next season, so let

I had this idea for a story this year. I really wanted to tell an amazing tale of something I had been watching from the beginning.

This is what I had, and I was happy with

I found this amazing recipe on the blog and it was so easy and delicious. I have made this many times and this one is just the perfect combination of recipes for any. Try it and tell

To learn more about these amazing videos and how to watch them in your home, head on over to the video search page. If you have any questions or would like to send us your questions.

---

Table 6: Task 1: Randomly selected samples from the single-word constraint task for the word “amazing” (with overall frequency $10^{-3}$) highlighted in green. Tokens are highlighted with yellow with different intensities to indicate their overall frequencies in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity). $\phi(x) = 1$ indicates the satisfaction of the constraint in the sample and reps the number of repetitions of the very same sample in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity).
Table 7: Task 2: Randomly selected samples from the single-word constraint task for the word “WikiLeaks” (with overall frequency $10^{-4}$) highlighted in green. Tokens are highlighted with yellow with different intensities to indicate their overall frequencies in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity). $\phi(x) = 1$ indicates the satisfaction of the constraint in the sample and reps the number of repetitions of the very same sample in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity).
Table 8: Task 3: Randomly selected samples from the wordlist constraint task for the wordlist "politics". Tokens are highlighted with yellow with different intensities to indicate their overall frequencies in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity). $\phi(x) = 1$ indicates the satisfaction of the constraint in the sample and reps the number of repetitions of the very same sample in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity).
I would love to find a way to use all this energy and energy on my own energy. But we have not yet figured this out. In fact, we seem to not really understand how it can.

The research paper is one of only two to date in recent years, after being published in the American Journal of Psychiatry. “The research team did some basic clinical investigation into the causes.

Fashion is no longer a matter of fashion. In fact, it is no longer a matter of fashion. This is so because it is no longer a matter of fashion. It is no.

I love that this post is about the biology of my gut flora, the microbiome (the living tissue that is used to support and control the gut) and the gut microbiome is basically just a chemical.

I think I did it once. I actually saw him with my brother. That’s how it went, I thought the guy was the same age. I don’t know, you were the same.

A few days ago we reported on the fact that the Obama administration has proposed an executive order that could increase the number of Syrian refugees who have been allowed in the U.S. for over five.

If you are wondering, I am not a scientist. I am just a man who studies human behaviour, as I love the science of nature. My focus is on the evolution of human beings.

The Republican National Convention had come under intense scrutiny for its use of language that used the word “nuclear” in an interview with the Daily Beast on Monday. In a lengthy segment on

In addition to the fact that there is no way to make the changes in the data, there is no way to know what is happening. In fact, all we have know about this project.

I know I am not a scientist. I am a man who studies and researches. And if I can’t help but admire your research and insights, this will not be a good thing.

We review data of primary power of data of data of data of data of question of validity of predictive of data and power of power of of data of data of data of and

In an equity of data of data of log as relationships and then: data of relationships to recall of data of data of relationships of relation. In relation of data of relation.

We, including data of data of data of data of data of in question of about power power of data of influence of relevance data of power of predictive of data.

We, including data of data of data of data of data of predictive of data of data of power of predictive of data of data of power of influence of data of data.

To relation power of data of question of data of: The correlation power of data of cohort of information of data of data of data of data of data of data of relation of of.

As the United States seeks to expand its nuclear energy base, it’s hard to ignore the increasing energy scarce in other countries.

In fact, there’s not much reason to think that the world’s

“People don’t believe you are doing any good in life. They say you’re a bad person who doesn’t control your life. They say you should give up on yourself.”

“A small percentage of our population is women. But that does not mean that all women have to be working. In fact, there are women working, but not all of them are. You

In case you missed it, a number of recent studies have shown that even when people with disabilities have an equal chance of being successful in their career, they are better off working in science.

We understand that it is an experiment which needs to be designed to provide data from the most sensitive and relevant individuals to be available to the most effective and well-funded researchers.

Table 9: Task 4: Randomly selected samples from the wordlist constraint task for the wordlist “science”. Tokens are highlighted with yellow with different intensities to indicate their overall frequencies in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity). \( \phi(x) = 1 \) indicates the satisfaction of the constraint in the sample and reps the number of repetitions of the very same sample in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity).
The American Dream is about more than a dream. It’s about a dream that, if you can’t have it, you can’t have it now. The American dream.

"This is our most expensive movie. You’re not looking to get a lot of good things, but with this one, your best bet is to think about what makes a good movie."

"The most incredible thing I can think of to tell you is that the world has finally found a way to get together. And I can’t tell you where it will go. But you will.

As part of a global effort to build a world where all people have access to affordable food, we are making a huge contribution to helping those at the core of the world to find an environment free.

It is no wonder that such a small and influential body of knowledge is important in the field of astronomy, astrophysics, medicine, and medical research. However, our knowledge of these topics is also

GDC++

"We are proud to announce today that the company has announced our fourth fiscal year. In our most important year, we raised nearly $5.5 billion of our operating revenue from online and mobile.

Election 2016 was the first election that did not involve a massive change in political discourse. But in fact, it was a dramatic change in political discourse in this year’s elections, one

Lemon-filled muffins have become an iconic, but surprisingly expensive option for breakfast, lunch or dinner on

Thank you for

Table 10: Task 5: Randomly selected samples from the classifier-based constraint task for positive sentiments. Tokens are highlighted with yellow with different intensities to indicate their overall frequencies in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity). \( \phi(x) = 1 \) indicates the satisfaction of the constraint in the sample and reps the number of repetitions of the very same sample in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity).
Table 11: 

<table>
<thead>
<tr>
<th>reps</th>
<th>$\phi(x)$</th>
<th>Sample</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>GDC++</strong></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>&quot;We're in a big game. &quot; &quot;It's really bad. It's really bad. &quot; &quot;I'm not going to lie to you. This was a lot of</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>A few years ago, the first issue of my first book had a subtitle with &quot;Slammarized&quot;. I had read it, and I had forgotten how bad it was, but it was</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>One of the things about the media is that it is sometimes too busy to do so much. And that's fine. It's just that the press is busy getting paid for doing so much</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>But that is only to be expected. One might be surprised at a simple explanation for the widespread lack of interest in climate science in the academic world. This is the story of the recent climate denial</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>The new 'Naughty Dog' is already in release. In a leaked release on Steam, the game is set for release in August, making it one of Sony’s most widely</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>GDC</strong></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>The first two tests of the K-12 program are very disappointing. One of the first tests showed a spike in learning rate on the test day and in the third the student reported less information than</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>A &quot;tongue for an ugly nose&quot; message was sent after a woman was told to &quot;dance&quot; after she became so disgusted by her friend’s antics that she sent &quot;a</td>
</tr>
<tr>
<td>0</td>
<td>1 of 16</td>
<td>This could be an old story. It didn’t come close to ending until Sunday night, when we got the first look at the cast on the set of &quot;</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>There are several reasons to think that we may not have a healthy amount of energy if we just eat nothing but pizza. The reason is that we’re not really hungry. So many</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>The word ‘fascism’ isn’t even spelled out in terms of the political spectrum. Some are racist, some are homophobic, and some are bigots. But when you</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Reinforce</strong></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>the evil poor evil annoying evil . the evil annoying the negative doesn’t even sad , the sad bin sad bad sass bin , the sad sad bin sass bin sass bin</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>This needs for long period of disappointing poor , the disappointing negative period of pathetic irrelevant poor annoying awful , even the disgusting poor period bin bin-at-total evil disass disass and that</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>no , is irrelevant , is not annoying , and even disgusting , disass or disass disass disass is disass bin disass disass bin disass disass disass</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>that is a big problem. &quot;thx , even a large non evil is a sad , is a sad , unreasonable , awful sad sad&quot; is evil sad , sad sad awful sad</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>so long , sad s/th0 needs to dav and disass is wrong , the dis ass s/th0 s/th0 predictable s , the dis ass bins</td>
</tr>
</tbody>
</table>

Table 11: **Task 6**: Randomly selected samples from the **classifier-based constraint** task for **negative sentiments**. Tokens are highlighted with **yellow** with different intensities to indicate their overall frequencies in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity). $\phi(x) = 1$ indicates the satisfaction of the constraint in the sample and reps the number of repetitions of the very same sample in the generated corpus of 10k samples (lower intensity indicates higher corpus level diversity).
isabela carolina is an american actress, writer, and former model. she is best known for her role as the teenage neighbor katie staley on the american series ".

0 (born august 3, 1969) is an american politician and lawyer. he is a member of the north dakota house of representatives from the 10th.

0 - born august 1, 1976 in new orleans, louisiana) is a former american football safety in the national football league for the washington redskins.

0 on 26 february 1990, he signed a five-year contract with bayer leverkusen. on 1 october 2000, sheik won the german cup with bayer leverkus.

0 the mcculloughs were an english glam rock band from portsmouth, england. the band formed in 2003, initially as a duo with john mckown, jimmy mc.

aime jacques de sousa is an indonesian television actress. she played a lead role in the 2012 indonesian television series "jayam ".

1 she has played

1 on 11 december 2013, lara klepp-larsen confirmed that she had suffered a heart attack. she was diagnosed with breast cancer at the age of 24.

0 the great olympic gong, born may 6, 1960 in san antonio, texas, was the first and only indy to win the world champion title of the american

0 aaron alexander (born october 27, 1989) is an american professional baseball outfielder for the tampa bay rays of major league baseball.

0 liz carlsson (born 2 june 1990) is a swedish actress and model, most famous for her role as alice in the film ".

0 - " for other people named john c. white, see john white (disambiguation). " john c. white, jr.

0 italo zola (born june 17, 1959) is a former italian footballer. he played as a striker and as a forward for italian clubs pesc.

1 of the year award nominations for 2013, 2014 and 2015. her most recent achievement was a "top 10 debut album" from her debut album, "in the name of the devil", on

1 až klimin (born October 20, 1996) is a latvian artistic gymnast. she is a two-time european junior team

0 brian patrick keane (born may 16, 1970) is an american football defensive end who is currently a free agent. he was drafted by the p.

1 was an english film and television actress. she appeared in many british and american films, and had roles in the tv shows "my big fat greek wedding".

0 - araki (born january 4, 1976 in ivanhoe, lautoka) is a retired brazilian footballer. he played for several clubs

1 . better known by her stage name pepi, is a korean female singer-songwriter. she came to korea after being influenced by kim jin-hoon's

1 - " for other people named john c. white, see john white (disambiguation). " john c. white, jr.

0 - araki (born january 4, 1976 in ivanhoe, lautoka) is a retired brazilian footballer. he played for several clubs

1 . better known by her stage name pepi, is a korean female singer-songwriter. she came to korea after being influenced by kim jin-hoon's

1 (born august 23, 1962) is an american actress. she has appeared in such films as "kojak", "i saw the fire"
<table>
<thead>
<tr>
<th>$\phi_1(x)$</th>
<th>$\phi_2(x)$</th>
<th>$\phi_3(x)$</th>
<th>$\phi_4(x)$</th>
<th>Sample</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>. was a russian politician and journalist.</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>luís alberto herrera carvalho ( born october 6, 1951 ) is a chilean economist, economist, politician and former mayor of mon</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>bernard stanton johnson ( born november 8, 1958 ) is a canadian politician. he was elected to the canadian house of commons in</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>, known as yuichi takashi , is a japanese professional golfer. takashi was born in shizuoka, japan and attended soto japan golf club</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>paul r. kelly is a democratic member of the pennsylvania house of representatives. he was elected to represent the 28th legislative district, being reelected in 2006 and 2010.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>slaw ( born 12 february 1961 ) is a polish politician, politician, sociologist, and member of the european parliament for poland.</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>, ( born in dresden, new jersey ) is a german singer and multi-instrumentalist who has released several solo albums.</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>for the artist, see jean-luc krüger ( painter ). &quot; jean-luc krüger ( j</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>kim ludwin ( born august 11, 1985 ) is a canadian ice hockey player who is currently playing with hc slovan bratislava</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>kazuki shimizu ( born march 30, 1970 in osaka, japan ) is a japanese mixed martial artist who is the current pride lightweight</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>andrew jones ( born 23 december 1970 ) is a former english cricketer. jones was a right-handed batsman who bowled right</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>andré fernández de gómez ( born 20 february 1989 ) is a spanish professional footballer who plays for fc barcelona</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>theodore george hudson ( october 20, 1877 - april 8, 1944 ) was a united states army officer. he served as the 19</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>, he was born in rome, italy on 10 may 1949.</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>linda jane thompson ( born march 10, 1958 ) is an american politician who was the u. s. representative for from 2003 to 2015.</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>kenny hansen ( born april 26, 1982 ) is an american actor best known for his role as the sheriff in the disney channel series &quot; criminal</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>in 2007, he was nominated by the governor of illinois to be the governor of illinois in 2011 for the position of the u. s. representative for illinois’s 22nd congressional</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>the dutch are an influential british reggae music duo, formed in 1982 in dublin. the duo consists of lead vocalist dave schroeder and drummer eric kend</td>
</tr>
</tbody>
</table>

Table 13: Task 8: Randomly selected samples from the experiment with Four distributional constraints: $\hat{\phi}_n(x) = 1$ iff $x$ contains at least one of the words from a corresponding $n$-th wordlist proposed by (Dathathri et al., 2020). The considered wordlists are “science”, “art”, “sports” and “business” and for each $\hat{\mu}_n = 0.25$.
Table 14: Task 9: Randomly selected samples from the experiment with a hybrid distributional constraint where $\phi_1(x) = 1$ iff $x$ contains a mention of a female figure, $\hat{\mu}_1 = 0.5$ and $\phi_2(x) = 1$ iff $x$ contains at least one of the words from the “sports” wordlist proposed by (Dathathri et al., 2020) and $\hat{\mu}_2 = 1$
Table 15: **Task 10**: Randomly selected samples from the experiment with a **hybrid distributional constraint** where $\phi_1(x) = 1$ iff $x$ contains a mention of a female figure, $\hat{\mu}_1 = 0.5$ and $\phi_2(x) = 1$ iff $x$ contains at least one of the words from the "science" wordlist proposed by [Dathathri et al. (2020)] and $\hat{\mu}_2 = 1$.