# Plug and Play Language Models: a Simple Approach to Controlled Text Generation

Authors: Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu

This folder contains the original code used to run the Plug and Play Language Model (PPLM).

Paper link: https://arxiv.org/abs/1912.02164

Blog link: https://eng.uber.com/pplm

Please check out the repo under uber-research for more information: https://github.com/uber-research/PPLM

# Note

⚠ This project should be run with pytorch-lightning==1.0.4 which has a potential security vulnerability

## Setup

```
git clone https://github.com/huggingface/transformers && cd transformers
pip install .
pip install nltk torchtext # additional requirements.
cd examples/research_projects/pplm
```

## PPLM-BoW

### Example command for bag-of-words control

```
python run_pplm.py -B military --cond_text "The potato" --length 50 --gamma 1.5 --
num_iterations 3 --num_samples 10 --stepsize 0.03 --window_length 5 --kl_scale 0.01
--gm_scale 0.99 --colorama --sample
```

### Tuning hyperparameters for bag-of-words control

1. Increase `--stepsize` to intensify topic control, and decrease its value to soften the control. `--stepsize 0` recovers the original uncontrolled GPT-2 model.

2. If the language being generated is repetitive (For e.g. "science science experiment experiment"), there are several options to consider:
   a) Reduce the `--stepsize`
   b) Increase `--kl_scale` (the KL-loss coefficient) or decrease `--gm_scale` (the gm-scaling term)
   c) Add `--grad-length xx` where xx is an (integer <= length, e.g. `--grad-length 30` ).

## PPLM-Discrim

### Example command for discriminator based sentiment control

```
python run_pplm.py -D sentiment --class_label 2 --cond_text "My dog died" --length
50 --gamma 1.0 --num_iterations 10 --num_samples 10 --stepsize 0.04 --kl_scale 0.01
--gm_scale 0.95 --sample
```

### Tuning hyperparameters for discriminator control

1. Increase `--stepsize` to intensify topic control, and decrease its value to soften the control. `--stepsize 0` recovers the original uncontrolled GPT-2 model.

2. Use `--class_label 3` for negative, and `--class_label 2` for positive