COLD Decoding: Energy-based Constrained Text Generation with Langevin Dynamics

Lianhui Qin · Sean Welleck · Daniel Khashabi · Yejin Choi

Text generation requires producing text that is not only fluent, but also satisfies different constraints that control the semantics or style of the generated text.

The dominant approach: fine-tune a pretrained LM with task-specific data
- prohibitively expensive
- can hardly scale to the infinite possible combinations of constraints

This work: constrained generation as sampling from an energy-based model (EBM):
- specify an energy function by plugging in any desired constraint functions
- then sample from the induced energy-based distribution
- No training/fine-tuning — control on the fly!

Key challenge of sampling from the text EBM:
- the normalizing factor $Z$ is intractable
- the common discrete MCMC methods (e.g., Gibbs sampling) is too inefficient!

Solution: Use gradient-based MCMC, Langevin dynamics, for efficient sampling!

Algorithm of COLD

Algorithm 1 Constrained Decoding w/ Langevin Dynamics.
input Constraints $\{f_i\}$, length $T$, iterations $N$.
output Sample sequence $y$.

\[ \tilde{y}^{(0)} = \text{init(); for all position } t \text{ // init soft-tokens} \]
for $n \in \{1, \ldots, N\}$ do
\[ E^{(n)} \leftarrow E(y^{(n)}; \{f_i\}) \text{// compute energy (§3.2)} \]
\[ \tilde{y}^{(n+1)} \leftarrow \tilde{y}^{(n)} - \eta \nabla y E(y^{(n)}) + \epsilon^{(n)} \text{// update soft tokens (Eq.2)} \]
end for
\[ y = \arg\max_{y'} \text{topk-filter } \tilde{y}^{(N)}(v) \text{ for all } t \text{// discretize (Eq.6)} \]
return: $y = (y_1, \ldots, y_T)$

Illustrations of example differentiable constraints

1. Fluency constraint
2. Future contextualization constraint
3. n-gram similarity constraint

Experiment Results

Lexically Constrained Generation
- Abductive Reasoning
- Counterfactual Reasoning

Check out COLD decoding paper!!

- continuous relaxation of discrete text: each token $y_i$ is modeled with its logit vector $\tilde{y}_i$
- Langevin dynamics: $\tilde{y}^{(n+1)} \leftarrow \tilde{y}^{(n)} - \eta \nabla y E(\tilde{y}^{(n)}) + \epsilon^{(n)}$
- Discretize the sampled continuous text vector with top-$k$ filtering (see paper for more details)