Thank you for the valuable comments. We first address several general concerns, and then reply to each reviewer.

G1: Computational cost. We discussed it in Supplementary 2.3 and will do a more thorough analysis in the revision. On a single 1080 Ti, our model took 6 hours and VAE 1.5 hours, and thus being 4 times slower, a similar factor observed on other image datasets. For text models with an autoregressive generator, our model does not have an obvious disadvantage compared to SOTA VAEs in terms of total training time (despite its longer per-iteration time) because of better posterior samples from short run MCMC than amortized inference and the overhead of the techniques that VAEs take to address posterior collapse (a common phenomenon for text VAEs). For the Yahoo dataset trained with the largest autoregressive generator, on a single 1080 Ti, our model took 14 hours, SA-VAE (combining amortized inference and MCMC) took 48 hours, FB-VAE took 5 hours (autoencoder pretraining) + 7 hours, and ARAE took 15 hours. Our approach trades feasible computational cost for expressive prior and simple and accurate inference.

G2: Scalability. To test our method's scalability, we trained a large generator with 49 million parameters on CelebA (128×128) on 4 V100 GPUs. It converged in 12 hours and produced faithful samples (see Figure 1 in Supplementary).

G3: RAE (Ghosh et al. ICLR 2020). We apologize for the citation typo. RAE in Table 1 refers to the regularized autoencoder, a SOTA VAE model, proposed by Ghosh et al. 2020.

R1. Thank you for the insightful comments. We will cite and discuss Hoffman 2017. We shall also discuss computational efficiency in the main text as you suggested.
Q1: Relations to MLE/EM/VAE. We shall tune down the claim to "our method is a practical modification of MLE." In Section 3.5 on theoretical understanding, eqn (13), the first D_{KL} is related to VAE and EM (line 181; see also Supplementary line 133). We will add more explanations as you suggested.
Q2: Log-likelihood. Yes, it is possible to estimate the log-likelihood by AIS. We shall include it in the revision.

R2. We deeply appreciate your positive feedback and insightful summary of our work.
Q1: Computational cost. Please see **G1** and **G2** above, which hopefully address your concern.

R4. We appreciate your positive comments on theoretical section and supplementary content. We shall do our best to further strengthen experiments as you suggested. We wish you could reconsider your rating.
Q1: Up-to-date literature (GAN and VAE). (1) We wish to clarify that the VAE models we compared against are up-to-date models. 2sVAE and RAE are considered the SOTA VAEs for image modeling. FB-VAE is the SOTA model for latent-variable-based language modeling. (2) We shall compare with GAN models in the revision. GANs underperform basic language models on text modeling due to the non-differentiability of text generation (Caccia et al. ICLR 2020). Moreover, GAN by itself is not equipped with an inference mechanism for inferring latent variables.
Q2: Advantage of learnable EBM prior. Compared to standard Gaussian, the learnable EBM prior leads to more accurate modeling of data distribution, illustrated by the learned model producing faithful (image and text) samples and being able to detect anomaly samples. Also see Table 4 and line 171 in Supplementary for a direct comparison to models with standard Gaussian prior.
Q3: Semi-supervised learning. Thanks for the suggestion! We are working on semi-supervised learning with our model and obtained promising results, which will be included in the revision.

R5. We appreciate your in-depth questions. We will discuss relevant work, DVAEs and VQ-VAE, and correct the citation typo for RAE (see **G3**). Also please see **G1** and **G2** above for our responses to computational concerns.
Q1: Persistent chain (PC). We will include the comparison to PC in the revision. In our experiments, short run chains (SRC) in latent space mixes quickly (see Figure 2 in the main text for SRC and Figure 2 in Supplementary for long run chains) so that $D_{KL}(\tilde{p}_\alpha(z) \| p_\alpha(z))$ can be small. This is a big advantage of EBM in latent space as compared to EBM in data space. Advantages of SRC over PC are: 1) theoretical underpinning of the learning method with SRC is cleaner; 2) in both training and testing, the same short run MCMC is used.
Q2: Amortized inference. We shall explore amortized inference in revision as you suggested. We did not use amortized inference for the following reasons. (1) We need posterior samples of z to learn the EBM prior $p_\alpha(z)$, and short run MCMC can be more reliable for generating posterior samples than amortized inference. (2) VAE is not conveniently applicable because $\log p_\theta(x, z)$ involves intractable $\log Z_\alpha$ term. (3) Short run MCMC in latent space is simple and reasonably computationally efficient.
Q3: Stability. In our experience and as observed by Grathwohl et al. ICLR 2020, training EBMs with PC as done by Du & Mordatch is less stable than using short run MCMC. Du & Mordatch stabilizes training with spectral normalization and L_2 regularization which are not needed in our experiments. Since our EBM is defined in low-dimensional latent space, short run MCMC is stable and scalable.
Q4: Baseline implementations. Prior art results are copied from the published papers if available. For instance, the best RAE FID scores for CIFAR-10 and CelebA are taken from Table 1 in Ghosh et al. 2020. FID scores for 2sVAE are also taken from Ghosh et al. given the comparability of their generator architecture to ours. We shall include the numbers reported in the 2sVAE paper in the revision.