Supplementary Materials

The supplementary file includes additional content related to the following:

1. Synthetic Experiments (Section 5.1 in main paper): We explain why the synthetic data loosely align with Definition 1, describe the network architectures of the CycleBase and CycleCVAE models, and include additional generation results.
2. WebNLG Experiments (Section 5.2 in main paper): We describe the experimental setup for WebNLG, including the task description, cycle-consistency model design, and all baseline and implementation details. We also include an ablation study varying $\dim[\boldsymbol{z}]$.
3. T5 Extension (Section 5.3 in main paper): We provide details of the CycleCVAE+T5 extension and include additional generated samples showing textual diversity.
4. Proof of Proposition 2.
5. Proof of Corollary 3.

7 Synthetic Dataset Experimental Details and Additional Results

7.1 Dataset Description and Relation to Definition 1

To motivate how the synthetic data used in Section 5.1 from the main paper at least partially align with Definition 1, we let $\boldsymbol{c}$ and $\boldsymbol{e}$ be zero vectors and $\boldsymbol{A} \in \mathbb{R}^{d \times 10}$ be a $d \times 10$ transformation matrix from images to digits, where d is the total number of pixels in each image $\boldsymbol{x}$. In other words, each column $i \in \{0, 1, \dots, 9\}$ of $\boldsymbol{A}$ is a linearized pixel sequence of the 2D image of digit i from top left to bottom right. Based on $\boldsymbol{A}$, we construct an example inverse matrix $\boldsymbol{D}$ so that $\boldsymbol{DA} = \boldsymbol{I}$. Specifically, $\boldsymbol{D}$ can be a $10 \times d$ matrix where each row $i \in \{0, 1, \dots, 9\}$ is a linearized pixel sequence of a masked version of the image of the digit i, and this image can have, for example, only one non-zero pixel that is sufficient to distinguish the digit i from all other nine possibilities. We also construct $\boldsymbol{B}$, a $d \times 9$ transformation matrix from the image to the border position, which surrounds one out of the nine tiles in each image. Each column $i \in \{0, 1, \dots, 8\}$ of $\boldsymbol{B}$ is a linearized pixel sequence of the 2D image of the border surrounding the i-th tile. Since the patterns of the digit and border do not share any non-zero pixels, we should have that $\boldsymbol{DB} = \boldsymbol{0}$. Moreover, each digit's image is distinct and cannot be produced by combining other digit images, so $\text{rank}[\boldsymbol{A}] = r_y$ and also $r_y \leq r_x$ because border patterns are orthogonal to digit patterns. Hence, we also have $\text{rank}[\boldsymbol{B}] \leq r_x - r_y$. Note however that the synthetic data do not guarantee that $\boldsymbol{Wy} + \boldsymbol{Vu}$ is equivalent to ρ^y_{gt} iff $\boldsymbol{W} = \boldsymbol{I}$ and $\boldsymbol{V} = \boldsymbol{0}$.

7.2 Network Architectures

We train two models on this dataset, a base model CycleBase using standard cycle training, and our CycleCVAE that incorporates the proposed CVAE into a baseline cycle model.

CycleBase The base model uses multilayer perceptrons (MLPs) for both the image($\boldsymbol{x}$)-to-digit($\boldsymbol{y}$) mapping $h^+_\theta(\boldsymbol{x})$ (shared with CycleCVAE), and the digit($\boldsymbol{y}$)-to-image($\boldsymbol{x}$) mapping denoted $h^{\text{Base}}_\theta(\boldsymbol{y})$. Each MLP hidden layer (two total) has 50 units with the tanh activation function. The last layer of $h^+_\theta(\boldsymbol{x})$ uses a softmax function to output a vector of probabilities $\boldsymbol{\alpha}$ over the ten digits, and therefore we can apply $p_\theta(\boldsymbol{y}|\boldsymbol{x}) = \text{Cat}(\boldsymbol{y}|\boldsymbol{\alpha})$, a categorical distribution conditioned on $\boldsymbol{\alpha}$, for training purposes. The last layer of digit-to-image $h^{\text{Base}}_\theta(\boldsymbol{y})$ adopts a per-pixel sigmoid function (since the value of each pixel is between 0 and 1), and we assume $p_\theta(\boldsymbol{x}|\boldsymbol{y})$ is based on the binary cross entropy loss.

CycleCVAE Our CycleCVAE uses the same function $h_\theta^+(\boldsymbol{x})$ as the base model. However, for the digit-to-image generation direction, CycleCVAE includes a 1-dimensional latent variable $\boldsymbol{z}$ sampled from $\mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$, where $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ are both learned by 50-dimensional, 3-layer MLPs (including output layer) with input $\boldsymbol{x}$. Then $h_\theta(\boldsymbol{y}, \boldsymbol{z})$ takes the digit $\boldsymbol{y}$ and latent variable $\boldsymbol{z}$ as inputs to another 3-layer MLP with 50 hidden units and the same activation function as the base model.

7.3 Generation Results

In addition to Figure 1 in the main paper, we list more example images generated by our model in the figure below. As we can see, the base model fails to learn the diverse border which should randomly surround only one of the nine tiles. However, CycleCVAE learns the border in its latent variable $\boldsymbol{z}$ and by random sampling, CycleCVAE can generate an arbitrary border around one of the nine digits as expected.

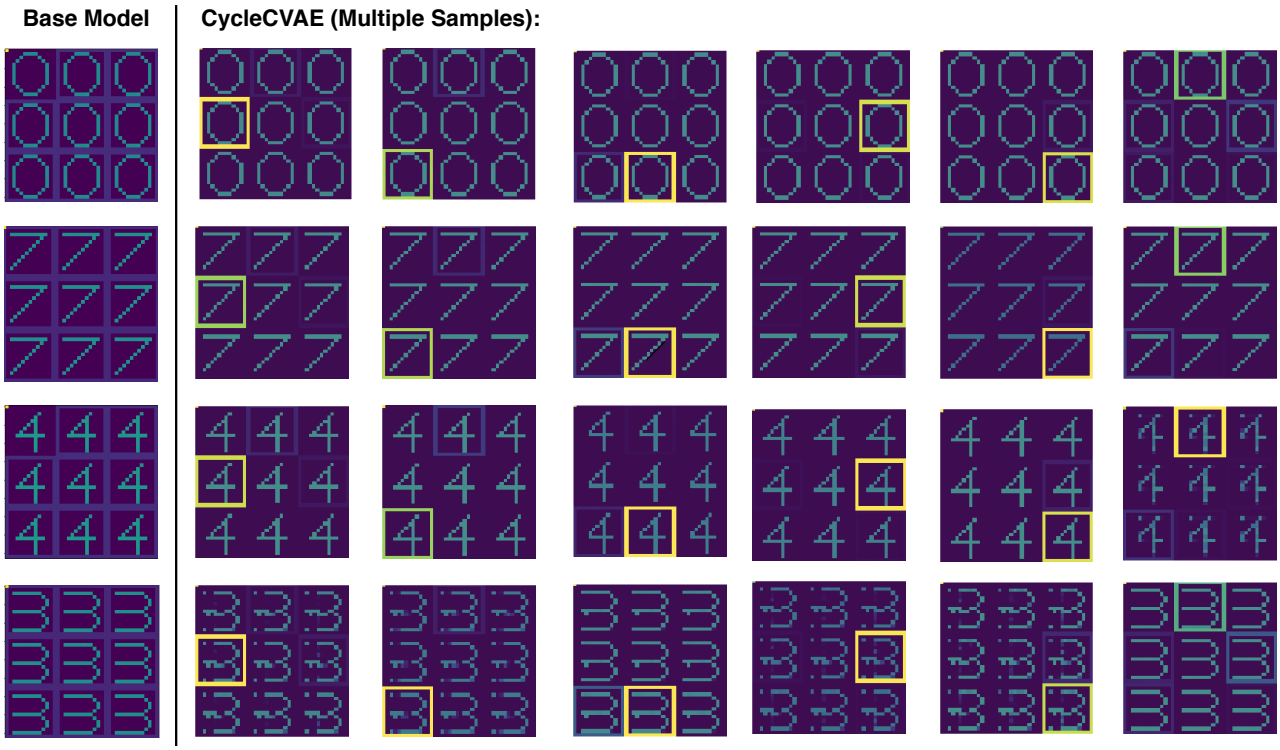

Figure 4: Example images generated by CycleCVAE.

8 WebNLG Experimental Setup and Ablation Study

The WebNLG dataset[6] is widely used for conversions between graph and text. Note that WebNLG is the most appropriate dataset for our purposes because in other candidates (e.g., relation extraction datasets (Walker et al., 2006)) the graphs only contain a very small subset of the information in the text.

8.1 Task Description

The WebNLG experiment includes two directions: text-to-graph (T2G) and graph-to-text (G2T) generation. The G2T task aims to produce descriptive text that verbalizes the graphical data. For example, the knowledge graph triplets "*(Allen Forest, genre, hip hop), (Allen Forest, birth year, 1981)*" can be verbalized as "*Allen Forest, a hip hop musician, was born in 1981.*" This has wide real-world applications, for instance, when a digital assistant needs to translate some structured information (e.g., the properties of the restaurant) to the human user. The other task, T2G is also important, as it extracts structures in the form of knowledge graphs from the text, so that

[6]It can be downloaded from `https://webnlg-challenge.loria.fr/challenge_2017/`.

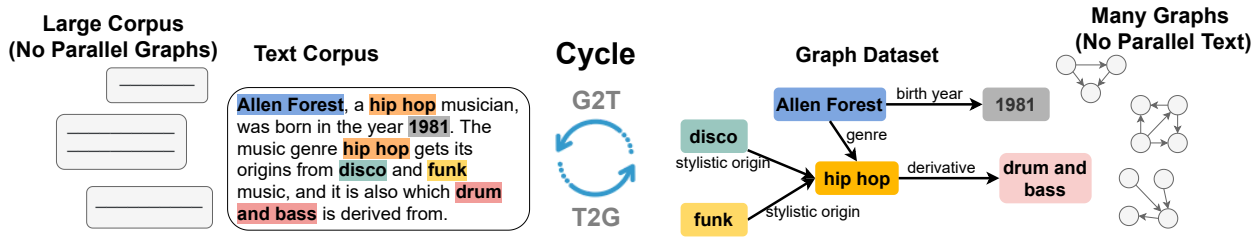

Figure 5: The graph-to-text generation task aims to verbalize a knowledge graph, while the text-to-graph task extracts the information of text into the form of a knowledge graph.

all entities become nodes, and the relationships among entities form edges. It can help many downstream tasks, such as information retrieval and reasoning. The two tasks can be seen as a dual problem, as shown in Figure 5.

Specifically, for unsupervised graph-to-text and text-to-graph generation, we have two non-parallel datasets:

- A text corpus $\boldsymbol{X} = \{\boldsymbol{x}_i\}_{i=1}^N$ consisting of N text sequences, and
- A graph dataset $\boldsymbol{Y} = \{\boldsymbol{y}_j\}_{j=1}^M$ consisting of M graphs.

The constraint is that the graphs and text contain the same distribution of latent content, but are different forms of surface realizations, i.e., there is no alignment providing matched pairs. Our goal is to train two models in an unsupervised manner: h_θ that generates text based on the graph, and h_θ^+ that produces a graph based on text.

8.2 Cycle Training Models

CycleBase Similar to the synthetic experiments mentioned above, we first propose the base cycle training model CycleBase that jointly learns graph-to-text and text-to-graph generation. To be consistent with our main paper, we denote text as $\boldsymbol{x}$ and graphs as $\boldsymbol{y}$, and the graph-to-text generation is a one-to-many mapping. The graph cycle, $\boldsymbol{y} \to \hat{\boldsymbol{x}} \to \hat{\boldsymbol{y}}$ is as follows: Given a graph $\boldsymbol{y}$, the cycle-consistent training first generates synthetic text $\hat{\boldsymbol{x}} = h_\theta^{\text{Base}}(\boldsymbol{y})$, and then uses it to reconstruct the original graph $\hat{\boldsymbol{y}} = h_\theta^+(\hat{\boldsymbol{x}})$. The loss function is imposed to align the generated graph $\hat{\boldsymbol{y}}$ with the original graph $\boldsymbol{y}$. Similarly, the text cycle, $\boldsymbol{x} \to \hat{\boldsymbol{y}} \to \hat{\boldsymbol{x}}$, is to align $\boldsymbol{x}$ and the generated $\hat{\boldsymbol{x}}$. Both loss functions adopt the cross entropy loss.

Specifically, we instantiate the graph-to-text module $h_\theta^{\text{Base}}(\boldsymbol{y})$ with the GAT-LSTM model proposed by (Koncel-Kedziorski et al., 2019), and the text-to-graph module $h_\theta^+(\boldsymbol{x})$ with a simple BiLSTM model we implemented. The GAT-LSTM module has two layers of graph attention networks (GATs) with 512 hidden units, and two layers of a LSTM text decoder with multi-head attention over the graph node embeddings produced by GAT. This attention mechanism uses four attention heads, each with 128 dimensions for self-attention and 128 dimension for cross-attention between the decoder and node features. The BiLSTM for text-to-graph construction uses 2-layer bidirectional LSTMs with 512 hidden units.

CycleCVAE Our CycleCVAE uses the same $h_\theta^+(\boldsymbol{x})$ as the base model. As for $h_\theta(\boldsymbol{y}, \boldsymbol{z})$ (the CycleCVAE extension of CycleBase), we first generate a 10-dimensional latent variable $\boldsymbol{z}$ sampled from $q_\phi(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$, where $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ are both learned by bidirectional LSTMs plus a fully connected feedforward layer. We form $p(\boldsymbol{z}|\boldsymbol{y})$ as a Gaussian distribution whose mean and variance are learned from a fully connected feedforward layer which takes in the feature of the root node of the GAT to represent the graph. Note that applying this $p(\boldsymbol{z}|\boldsymbol{y})$ as the CycleCVAE prior is functionally equivalent to using a more complicated encoder, as mentioned in the main paper.

Implementation Details For both cycle models, we adopt the Adam optimizer with a learning rate of 5e−5 for the text-to-graph modules, and learning rate of 2e−4 for graph-to-text modules. For the graph-to-text module, we re-implement the GAT-LSTM model (Koncel-Kedziorski et al., 2019) using the DGL library (Wang et al., 2019b). Our code is available `https://github.com/QipengGuo/CycleGT`.

8.3 Details of Competing Methods

Unsupervised Baselines As cycle training models are unsupervised learning methods, we first compare with unsupervised baselines. *RuleBased* is a heuristic baseline proposed by (Schmitt et al., 2020) which simply iterates through the graph and concatenates the text of each triplet. For example, the triplet "(AlanShepard, occupation, TestPilot)" will be verbalized as "Alan Shepard occupation test pilot." If there are multiple triplets, their text expressions will be concatenated by "and." The other baseline, *UMT* (Schmitt et al., 2020), formulates the graph and text conversion as a sequence-to-sequence task and applies a standard unsupervised machine translation (UMT) approach. It serializes each triplet of the graph in the same way as RuleBased, and concatenates the serialization of all triplets in a random order, using special symbols as separators.

Supervised Baselines We also compare with *supervised* systems using the original supervised training data. Since there is no existing work that jointly learns graph-to-text and text-to-graph in a supervised way, we can only use models that address one of the two tasks. For graph-to-text generation, we list the performance of state-of-the-art supervised models including (1) *Melbourne*, the best supervised system submitted to the WebNLG challenge 2017 (Gardent et al., 2017), which uses an encoder-decoder architecture with attention, (2) *StrongNeural* (Moryossef et al., 2019) which improves the common encoder-decoder model, (3) *BestPlan* (Moryossef et al., 2019) which uses a special entity ordering algorithm before neural text generation, (4) *G2T* (Koncel-Kedziorski et al., 2019) which is the same as the GAT-LSTM architecture adopted in our cycle training models, and (5) *Seg&Align* (Shen et al., 2020), which segments the text into small units, and learns the alignment between data and target text segments. The generation process uses the attention mechanism over the corresponding data piece to generate the corresponding text. For text-to-graph generation, we compare with state-of-the-art models including *OnePass* (Wang et al., 2019a), a BERT-based relation extraction model, and *T2G*, the BiLSTM model that we adopt as the text-to-graph component in the cycle training of CycleBase and CycleCVAE.

8.4 Ablation Study

We conduct an ablation study using the 50%:50% unsupervised data of WebNLG. Note that our models do not use an adversarial term, so we only tune the CVAE latent dimension to test robustness to this factor. The hyperparameter tuning of the size of the latent dimension is shown in Table 5, where we observe that our CycleCVAE is robust against different $\boldsymbol{z}$ dimensions. Note that because $\boldsymbol{z}$ is continuous while generated text is discrete, just a single dimension turns out to be adequate for good performance for these experiments. Even so, the encoder variance can be turned up to avoid 'overusing' any continuous latent dimension to roughly maintain a bijection.

	Text (BLEU)	Diversity (# Variations)
Latent Dimension		
$\boldsymbol{z} = 1$	46.3	4.62
$\boldsymbol{z} = 10$	46.5	4.67
$\boldsymbol{z} = 50$	46.2	4.65

Table 5: Text quality (by BLEU scores) and diversity (by the number of variations) under different dimensions of $\boldsymbol{z}$.

9 T5 Model Details and More Generated Samples

9.1 CycleCVAE+T5 Implementational Details

We adopted the pretrained T5 model (Raffel et al., 2020) to replace the GAT-LSTM architecture that we previously used for the graph-to-text module within the cycle training. T5 is a sequence-to-sequence model that takes as input a serialized graph (see the serialization practice in Schmitt et al., 2020; Ribeiro et al., 2020; Kale, 2020) and generates a text sequence accordingly. We finetune the T5 during training with the Adam optimizer using a learning rate of 5e−5.

9.2 Additional Text Diversity Examples

We list the text diversity examples generated by CycleCVAE+T5 in Table 6.

No.	Variations
1	– Batagor, a variation of Siomay and Shumai, can be found in Indonesia, where the leader is Joko Widodo and Peanut sauce is an ingredient. – Batagor is a dish from Indonesia, where the leader is Joko Widodo and the main ingredient is Peanut sauce. It can also be served as a variation of Shumai and Siomay.
2	– The AMC Matador, also known as "American Motors Matador", is a Mid-size car with an AMC V8 engine and is assembled in Thames, New Zealand. – AMC Matador, also known as "American Motors Matador", is a Mid-size car. It is made in Thames, New Zealand and has an AMC V8 engine.
3	– Aleksandr Chumakov was born in Moscow and died in Russia. The leader of Moscow is Sergey Sobyanin. – Aleksandr Chumakov, who was born in Moscow, was a leader in Moscow where Sergey Sobyanin is a leader. He died in Russia.
4	– A Wizard of Mars is written in English language spoken in Great Britain. It was published in the United States, where Barack Obama is the president. – A Wizard of Mars comes from the United States where Barack Obama is the leader and English language spoken in Great Britain.
5	– The Addiction (journal), abbreviated to "Addiction", has the ISSN number "1360-0443" and is part of the academic discipline of Addiction. – Addiction (journal), abbreviated to "Addiction", has the ISSN number "1360-0443".
6	– Atlantic City, New Jersey is part of Atlantic County, New Jersey Atlantic County, New Jersey, in the United States. – Atlantic City, New Jersey is part of Atlantic County, New Jersey, United States.
7	– Albuquerque, New Mexico, United States, is lead by the New Mexico Senate, led by John Sanchez and Asian Americans. – Albuquerque, New Mexico, in the United States, is lead by the New Mexico Senate, where John Sanchez is a leader and Asian Americans are an ethnic group.
8	– Aaron Turner plays the Electric guitar and plays Black metal, Death metal and Black metal. He also plays in the Twilight (band) and Old Man Gloom. – Aaron Turner plays the Electric guitar and plays Black metal. He is associated with the Twilight (band) and Old Man Gloom. He also plays Death metal.

Table 6: Examples of diverse text generated by CycleCVAE based on the same input knowledge graph.

10 Proof of Proposition 2

The high-level proof proceeds in several steps. First we consider optimization of $\ell_x(\theta, \phi)$ over ϕ to show that no suboptimal local minima need be encountered. We then separately consider optimizing $\ell_x(\theta, \phi)$ and $\ell_y(\theta)$ over the subset of θ unique to each respective loss. Next we consider jointly optimizing the remaining parameters residing between both terms. After assimilating the results, we arrive at the stated result of Proposition 2. Note that with some abuse of notation, we reuse several loss function names to simplify the exposition; however, the meaning should be clear from context.

10.1 Optimization over encoder parameters ϕ in $\ell_x(\theta, \phi)$

The energy term from the $\boldsymbol{x} \to \hat{\boldsymbol{y}} \to \hat{\boldsymbol{x}}$ cycle can be modified as

$$\begin{aligned}
\ell_x(\theta,\phi) &= \int \left\{ \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \left[\tfrac{1}{\gamma} \left\| \boldsymbol{x} - \boldsymbol{\mu}_x \right\|_2^2 \right] + d \log \gamma + \sum_{k=1}^{r_z} \left(s_k^2 - \log s_k^2 \right) + \left\| \boldsymbol{\mu}_z \right\|_2^2 \right\} \rho_{gt}^x(d\boldsymbol{x}) \\
&= \int \left\{ \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \left[\tfrac{1}{\gamma} \left\| \left(\boldsymbol{I} - \boldsymbol{W}_x \boldsymbol{W}_y \right) \boldsymbol{x} - \boldsymbol{V}_x \boldsymbol{z} - \boldsymbol{W}_x \boldsymbol{b}_y - \boldsymbol{b}_x \right\|_2^2 \right] \right. \\
&\quad \left. + d \log \gamma + \sum_{k=1}^{r_z} \left(s_k^2 - \log s_k^2 \right) + \left\| \boldsymbol{W}_z \boldsymbol{x} + \boldsymbol{b}_z \right\|_2^2 \right\} \rho_{gt}^x(d\boldsymbol{x}) \qquad (13) \\
&= \int \left\{ \tfrac{1}{\gamma} \left\| \left(\boldsymbol{I} - \boldsymbol{W}_x \boldsymbol{W}_y \right) \boldsymbol{x} - \boldsymbol{V}_x \left(\boldsymbol{W}_z \boldsymbol{x} + \boldsymbol{b}_z \right) - \boldsymbol{W}_x \boldsymbol{b}_y - \boldsymbol{b}_x \right\|_2^2 \right. \\
&\quad \left. + d \log \gamma + \sum_{k=1}^{\kappa} \left(s_k^2 - \log s_k^2 + \tfrac{1}{\gamma} s_k^2 \| \boldsymbol{v}_{x,k} \|_2^2 \right) + \left\| \boldsymbol{W}_z \boldsymbol{x} + \boldsymbol{b}_z \right\|_2^2 \right\} \rho_{gt}^x(d\boldsymbol{x}),
\end{aligned}$$

where $\boldsymbol{v}_{x,k}$ denotes the k-th column of $\boldsymbol{V}_x$. Although this expression is non-convex in each s_k^2, by taking derivatives and setting them equal to zero, it is easily shown that there is a single stationary point that operates as the unique minimum. Achieving the optimum requires only that $s_k^2 = \left[\frac{1}{\gamma} \| \boldsymbol{v}_{x,k} \|_2^2 + 1 \right]^{-1}$ for all k. Plugging this value into (13) then leads to the revised objective

$$\begin{aligned}
\ell_x(\theta,\phi) \equiv \int \left\{ \right. & \tfrac{1}{\gamma} \left\| \left(\boldsymbol{I} - \boldsymbol{W}_x \boldsymbol{W}_y \right) \boldsymbol{x} - \boldsymbol{V}_x \left(\boldsymbol{W}_z \boldsymbol{x} + \boldsymbol{b}_z \right) - \boldsymbol{W}_x \boldsymbol{b}_y - \boldsymbol{b}_x \right\|_2^2 \qquad (14) \\
& \left. + \sum_{k=1}^{\kappa} \log \left(\tfrac{1}{\gamma} \| \boldsymbol{v}_{x,k} \|_2^2 + 1 \right) + d \log \gamma + \left\| \boldsymbol{W}_z \boldsymbol{x} + \boldsymbol{b}_z \right\|_2^2 \right\} \rho_{gt}^x(d\boldsymbol{x})
\end{aligned}$$

ignoring constant terms. Similarly we can optimize over $\boldsymbol{\mu}_z = \boldsymbol{W}_z \boldsymbol{x} + \boldsymbol{b}_z$ in terms of the other variables. This is just a convex, ridge regression problem, with the optimum uniquely satisfying

$$\boldsymbol{W}_z \boldsymbol{x} + \boldsymbol{b}_z = \boldsymbol{V}_x^\top \left(\gamma \boldsymbol{I} + \boldsymbol{V}_x \boldsymbol{V}_x^\top \right)^{-1} \left[\left(\boldsymbol{I} - \boldsymbol{W}_x \boldsymbol{W}_y \right) \boldsymbol{x} - \boldsymbol{W}_x \boldsymbol{b}_y - \boldsymbol{b}_x \right], \qquad (15)$$

which is naturally an affine function of $\boldsymbol{x}$ as required. After plugging (15) into (14), defining $\boldsymbol{\epsilon}_x \triangleq \left(\boldsymbol{I} - \boldsymbol{W}_x \boldsymbol{W}_y \right) \boldsymbol{x} - \boldsymbol{W}_x \boldsymbol{b}_y - \boldsymbol{b}_x$, and applying some linear algebra manipulations, we arrive at

$$\begin{aligned}
\bar{\ell}_x(\theta) &\triangleq \min_\phi \ \ell_x(\theta,\phi) \qquad (16) \\
&= \int \left\{ \boldsymbol{\epsilon}_x^\top \left(\boldsymbol{V}_x \boldsymbol{V}_x^\top + \gamma \boldsymbol{I} \right)^{-1} \boldsymbol{\epsilon}_x \right\} \rho_{gt}^x(d\boldsymbol{x}) + \sum_{k=1}^{\kappa} \log \left(\| \boldsymbol{v}_{x,k} \|_2^2 + \gamma \right) + (d - \kappa) \log \gamma,
\end{aligned}$$

noting that this minimization was accomplished without encountering any suboptimal local minima.

10.2 Optimization over parameters θ that are unique to $\bar{\ell}_x(\theta)$

The optimal $\boldsymbol{b}_x$ is just the convex maximum likelihood estimator given by the mean

$$\boldsymbol{b}_x = \int \left(\boldsymbol{I} - \boldsymbol{W}_x \boldsymbol{W}_y \right) \boldsymbol{x} \rho_{gt}^x(d\boldsymbol{x}) - \boldsymbol{W}_x \boldsymbol{b}_y = \left(\boldsymbol{I} - \boldsymbol{W}_x \boldsymbol{W}_y \right) \boldsymbol{c} - \boldsymbol{W}_x \boldsymbol{b}_y, \qquad (17)$$

where the second equality follows from Definition 1 in the main text. Plugging this value into (16) and applying a standard trace identity, we arrive at

$$\bar{\ell}_x(\theta) \equiv \operatorname{tr} \left[\boldsymbol{S}_{\epsilon_x} \left(\boldsymbol{V}_x \boldsymbol{V}_x^\top + \gamma \boldsymbol{I} \right)^{-1} \right] + \sum_{k=1}^{\kappa} \log \left(\| \boldsymbol{v}_{x,k} \|_2^2 + \gamma \right) + (d - \kappa) \log \gamma, \qquad (18)$$

where

$$\boldsymbol{S}_{\epsilon_x} \triangleq \text{Cov}_{\rho^x_{gt}}[\boldsymbol{\epsilon}_x] = (\boldsymbol{I} - \boldsymbol{W}_x\boldsymbol{W}_y)\,\text{Cov}_{\rho^x_{gt}}[\boldsymbol{x}]\,(\boldsymbol{I} - \boldsymbol{W}_x\boldsymbol{W}_y)^\top. \tag{19}$$

The remaining parameters $\{\boldsymbol{W}_x, \boldsymbol{W}_y, \boldsymbol{V}_x\}$ are all shared with the $\boldsymbol{y} \to \hat{\boldsymbol{x}} \to \hat{\boldsymbol{y}}$ cycle loss $\ell_y(\theta)$, so ostensibly we must include the full loss $\bar{\ell}_x(\theta) + \ell_y(\theta)$ when investigating local minima with respect to these parameters. However, there is one subtle exception that warrants further attention here. More specifically, the loss $\ell_y(\theta)$ depends on $\boldsymbol{V}_x$ only via the outer product $\boldsymbol{V}_x\boldsymbol{V}_x^\top$. Consequently, if $\boldsymbol{V}_x = \bar{\boldsymbol{U}}\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{V}}^\top$ denotes the singular value decomposition of $\boldsymbol{V}_x$, then $\ell_y(\theta)$ is independent of $\bar{\boldsymbol{V}}$ since $\boldsymbol{V}_x\boldsymbol{V}_x^\top = \bar{\boldsymbol{U}}\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\Lambda}}^\top\bar{\boldsymbol{U}}^\top$, noting that $\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\Lambda}}^\top$ is just a square matrix with squared singular values along the diagonal. It then follows that we can optimize $\bar{\ell}_x(\theta)$ over $\bar{\boldsymbol{V}}$ without influencing $\ell_y(\theta)$.

To this end we have the following:

Lemma 1 *At any minimizer (local or global) of $\bar{\ell}_x(\theta)$ with respect to $\bar{\boldsymbol{V}}$, it follows that $\bar{\boldsymbol{V}} = \boldsymbol{P}$ for some permutation matrix $\boldsymbol{P}$ and the corresponding loss satisfies*

$$\bar{\ell}_x(\theta) = \text{tr}\left[\boldsymbol{S}_{\epsilon_x}\boldsymbol{\Sigma}_{\epsilon_x}^{-1}\right] + \log|\boldsymbol{\Sigma}_{\epsilon_x}|, \quad \textit{where } \boldsymbol{\Sigma}_{\epsilon_x} \triangleq \boldsymbol{V}_x\boldsymbol{V}_x^\top + \gamma\boldsymbol{I}. \tag{20}$$

This result follows (with minor modification) from (Dai et al., 2019)[Corollary 3]. A related result also appears in (Lucas et al., 2019).

10.3 Optimization over parameters θ that are unique to $\ell_y(\theta)$

Since $\boldsymbol{y}$ has zero mean per Definition 1, the optimal $\boldsymbol{b}_y$ is the convex maximum likelihood estimator satisfying $\boldsymbol{b}_y = -\boldsymbol{W}_y\boldsymbol{b}_x$ (this assumes that $\boldsymbol{W}_y\boldsymbol{b}_x$ has not been absorbed into $\boldsymbol{y}$ as mentioned in the main text for notational simplicity). This leads to

$$\ell_y(\theta) \equiv \text{tr}\left[\boldsymbol{S}_{\epsilon_y}\boldsymbol{\Sigma}_{\epsilon_y}^{-1}\right] + \log|\boldsymbol{\Sigma}_{\epsilon_y}|, \quad \text{where } \boldsymbol{S}_{\epsilon_y} \triangleq (\boldsymbol{I} - \boldsymbol{W}_y\boldsymbol{W}_x)\,(\boldsymbol{I} - \boldsymbol{W}_y\boldsymbol{W}_x)^\top \tag{21}$$

and $\boldsymbol{\Sigma}_{\epsilon_y}$ is defined in the main text.

10.4 Optimizing the combined loss $\bar{\ell}_{cycle}(\theta)$

The above results imply that we may now consider jointly optimizing the combined loss

$$\bar{\ell}_{cycle}(\theta) \triangleq \bar{\ell}_x(\theta) + \ell_y(\theta) \tag{22}$$

over $\{\boldsymbol{W}_x, \boldsymbol{W}_y, \boldsymbol{V}_x\boldsymbol{V}_x^\top\}$; all other terms have already been optimized out of the model without encountering any suboptimal local minima. To proceed, consider the distribution $\rho^{\hat{y}}_{gt}$ of

$$\hat{\boldsymbol{y}} = \boldsymbol{W}_y\boldsymbol{x} + \boldsymbol{b}_y = \boldsymbol{W}_y\boldsymbol{A}\boldsymbol{y} + \boldsymbol{W}_y\boldsymbol{B}\boldsymbol{u} + \boldsymbol{W}_y\boldsymbol{c} + \boldsymbol{b}_y. \tag{23}$$

To satisfy the constraint the stipulated constraint $\rho^{\hat{y}}_{gt} = \rho^y_{gt}$ subject to the conditions of Definition 1, it must be that $\boldsymbol{W}_y\boldsymbol{A} = \boldsymbol{I}$ and $\boldsymbol{B} \in \text{null}[\boldsymbol{W}_y]$ (it will also be the case that $\boldsymbol{b}_y = -\boldsymbol{W}_y\boldsymbol{c}$ to ensure that $\hat{\boldsymbol{y}}$ has zero mean). From this we may conclude that

$$\begin{aligned}
\boldsymbol{S}_{\epsilon_x} &= (\boldsymbol{I} - \boldsymbol{W}_x\boldsymbol{W}_y)\,\text{Cov}_{\rho^x_{gt}}[\boldsymbol{x}]\,(\boldsymbol{I} - \boldsymbol{W}_x\boldsymbol{W}_y)^\top \\
&= (\boldsymbol{I} - \boldsymbol{W}_x\boldsymbol{W}_y)\left[\boldsymbol{A}\boldsymbol{A}^\top + \boldsymbol{B}\boldsymbol{B}^\top\right](\boldsymbol{I} - \boldsymbol{W}_x\boldsymbol{W}_y)^\top \\
&= (\boldsymbol{A} - \boldsymbol{W}_x)\,(\boldsymbol{A} - \boldsymbol{W}_x)^\top + \boldsymbol{B}\boldsymbol{B}^\top,
\end{aligned} \tag{24}$$

where the middle equality follows because $\boldsymbol{y}$ and $\boldsymbol{u}$ are uncorrelated with identity covariance. Furthermore, let $\widetilde{\boldsymbol{D}} \in \mathbb{R}^{r_y \times r_x}$ denote any matrix such that $\widetilde{\boldsymbol{D}}\boldsymbol{A} = \boldsymbol{I}$ and $\boldsymbol{B} \in \text{null}[\widetilde{\boldsymbol{D}}]$. It then follows that $\boldsymbol{W}_y$ must equal some such $\widetilde{\boldsymbol{D}}$ and optimization of (22) over $\boldsymbol{W}_x$ will involve simply minimizing

$$\bar{\ell}_{cycle}(\theta) \equiv \text{tr}\left[(\boldsymbol{A} - \boldsymbol{W}_x)\,(\boldsymbol{A} - \boldsymbol{W}_x)^\top\boldsymbol{\Sigma}_{\epsilon_x}^{-1}\right] + \text{tr}\left[\left(\boldsymbol{I} - \widetilde{\boldsymbol{D}}\boldsymbol{W}_x\right)\left(\boldsymbol{I} - \widetilde{\boldsymbol{D}}\boldsymbol{W}_x\right)^\top\boldsymbol{\Sigma}_{\epsilon_y}^{-1}\right] + C \tag{25}$$

over $\boldsymbol{W}_x$, where C denotes all terms that are independent of $\boldsymbol{W}_x$. This is a convex problem with unique minimum at $\boldsymbol{W}_x = \boldsymbol{A}$. Note that this choice sets the respective $\boldsymbol{W}_x$-dependent terms to zero, the minimum possible value. Plugging $\boldsymbol{W}_x = \boldsymbol{A}$ into (25) and expanding the terms in C, we then arrive at the updated loss

$$\begin{aligned}\bar{\ell}_{cycle}(\theta) &\equiv \operatorname{tr}\left[\boldsymbol{B}\boldsymbol{B}^\top \boldsymbol{\Sigma}_{\epsilon_x}^{-1}\right] + \log|\boldsymbol{\Sigma}_{\epsilon_x}| + \log\left|\boldsymbol{\Sigma}_{\epsilon_y}\right| \\ &= \operatorname{tr}\left[\boldsymbol{B}\boldsymbol{B}^\top\left(\boldsymbol{V}_x\boldsymbol{V}_x^\top + \gamma\boldsymbol{I}\right)^{-1}\right] + \log\left|\boldsymbol{V}_x\boldsymbol{V}_x^\top + \gamma\boldsymbol{I}\right| + \log\left|\widetilde{\boldsymbol{D}}\boldsymbol{V}_x\boldsymbol{V}_x^\top\widetilde{\boldsymbol{D}}^\top + \gamma\boldsymbol{I}\right|.\end{aligned} \tag{26}$$

Minimization of this expression over $\boldsymbol{V}_x$ as γ becomes arbitrarily small can be handled as follows. If any $\boldsymbol{V}_x$ and γ are a local minima of (26), then $\{\alpha = 1, \beta = 0\}$ must also be a local minimum of

$$\begin{aligned}&\bar{\ell}_{cycle}(\alpha,\beta) \triangleq \\ &\quad \operatorname{tr}\left[\boldsymbol{B}\boldsymbol{B}^\top\left(\alpha\boldsymbol{\Sigma}_{\epsilon_x} + \beta\boldsymbol{B}\boldsymbol{B}^\top\right)^{-1}\right] + \log\left|\alpha\boldsymbol{\Sigma}_{\epsilon_x} + \beta\boldsymbol{B}\boldsymbol{B}^\top\right| + \log\left|\alpha\boldsymbol{\Sigma}_{\epsilon_y} + \beta\widetilde{\boldsymbol{D}}\boldsymbol{B}\boldsymbol{B}^\top\widetilde{\boldsymbol{D}}^\top\right| \\ &= \operatorname{tr}\left[\boldsymbol{B}\boldsymbol{B}^\top\left(\alpha\boldsymbol{\Sigma}_{\epsilon_x} + \beta\boldsymbol{B}\boldsymbol{B}^\top\right)^{-1}\right] + \log\left|\alpha\boldsymbol{\Sigma}_{\epsilon_x} + \beta\boldsymbol{B}\boldsymbol{B}^\top\right| + \log\left|\alpha\boldsymbol{\Sigma}_{\epsilon_y}\right|.\end{aligned} \tag{27}$$

If we exclude the second log-det term, then it has been shown in (Wipf and Nagarajan, 2007) that loss functions in the form of (27) have a monotonically decreasing path to a unique minimum as $\beta \to 1$ and $\alpha \to 0$. However, given that the second log-det term is a monotonically decreasing function of α, it follows that the entire loss from (27) has a unique minimum as $\beta \to 1$ and $\alpha \to 0$. Consequently, it must be that at any local minimum of (26) $\boldsymbol{V}_x\boldsymbol{V}_x^\top = \boldsymbol{B}\boldsymbol{B}^\top$ in the limit as $\gamma \to 0$. Moreover, the feasibility of this limiting equality is guaranteed by our assumption that $r_z \geq r_c - r_y$ (i.e., if $r_z < r_c - r_y$, then $\boldsymbol{V}_x$ would not have sufficient dimensionality to allow $\boldsymbol{V}_x\boldsymbol{V}_x^\top = \boldsymbol{B}\boldsymbol{B}^\top$).

10.5 Final Pieces

We have already established that at any local minimizer $\{\theta^*, \phi^*\}$ it must be the case that $\boldsymbol{W}_x^* = \boldsymbol{A}$ and $\boldsymbol{W}_y^* = \widetilde{\boldsymbol{D}}$. Moreover, we also can infer from (17) and Section 10.3 that at any local minimum we have

$$\boldsymbol{b}_x^* = \left(\boldsymbol{I} - \boldsymbol{W}_x^*\boldsymbol{W}_y^*\right)\boldsymbol{c} - \boldsymbol{W}_x^*\boldsymbol{b}_y^* = \left(\boldsymbol{I} - \boldsymbol{W}_x^*\boldsymbol{W}_y^*\right)\boldsymbol{c} + \boldsymbol{W}_x^*\boldsymbol{W}_y^*\boldsymbol{b}_x^* = \left(\boldsymbol{I} - \boldsymbol{A}\widetilde{\boldsymbol{D}}\right)\boldsymbol{c} + \boldsymbol{A}\widetilde{\boldsymbol{D}}\boldsymbol{b}_x^* \tag{28}$$

from which it follows that $\left(\boldsymbol{I} - \boldsymbol{A}\widetilde{\boldsymbol{D}}\right)\boldsymbol{c} = \left(\boldsymbol{I} - \boldsymbol{A}\widetilde{\boldsymbol{D}}\right)\boldsymbol{b}_x^*$. This along is not sufficient to guarantee that $\boldsymbol{b}_x^* = \boldsymbol{c}$ is the unique solution; however, once we include the additional constraint $\rho_{gt}^y = \rho_\theta^{\hat{y}}$ per the Proposition 2 statement, then $\boldsymbol{b}_x^* = \boldsymbol{c}$ is uniquely determined (otherwise it would imply that $\hat{\boldsymbol{y}}$ has a nonzero mean). It then follows that $\boldsymbol{b}_y^* = -\boldsymbol{W}_y^*\boldsymbol{b}_x^* = -\widetilde{\boldsymbol{D}}\boldsymbol{c}$.

And finally, regarding $\boldsymbol{V}_x^*$, from Section 10.4 we have that $\boldsymbol{V}_x^*\left(\boldsymbol{V}_x^*\right)^\top = \boldsymbol{B}\boldsymbol{B}^\top$. Although this does *not* ensure that $\boldsymbol{V}_x^* = \boldsymbol{B}$, we can conclude that $\operatorname{span}[\bar{\boldsymbol{U}}] = \operatorname{span}[\boldsymbol{B}]$. Furthermore, we know from Lemma 1 and the attendant singular value decomposition that $\boldsymbol{V}_x^* = \bar{\boldsymbol{U}}\bar{\boldsymbol{\Lambda}}\boldsymbol{P}^\top$ and $\left(\boldsymbol{V}_x^*\right)^\top\boldsymbol{V}_x^* = \boldsymbol{P}^\top\bar{\boldsymbol{\Lambda}}^\top\bar{\boldsymbol{\Lambda}}\boldsymbol{P}$. Therefore, up to an arbitrary permutation, each column of $\boldsymbol{V}_x^*$ satisfies

$$\|\boldsymbol{v}_{x,k}^*\|_2^2 = \begin{cases}\bar{\lambda}_k^2, & \forall\ k = 1,\ldots,\operatorname{rank}[\boldsymbol{B}] \\ 0, & \forall\ k = \operatorname{rank}[\boldsymbol{B}]+1,\ldots,r_z\end{cases} \tag{29}$$

where $\bar{\lambda}_k$ is an eigenvalue of $\bar{\boldsymbol{\Lambda}}$. Collectively then, these results imply that $\boldsymbol{V}_x^* = \left[\widetilde{\boldsymbol{B}}, \boldsymbol{0}\right]\boldsymbol{P}^\top$, where $\widetilde{\boldsymbol{B}} \in \mathbb{R}^{r_x \times \operatorname{rank}[\boldsymbol{B}]}$ satisfies $\operatorname{span}[\widetilde{\boldsymbol{B}}] = \operatorname{span}[\boldsymbol{U}] = \operatorname{span}[\boldsymbol{B}]$.

11 Proof of Corollary 3

From (15) in the proof of Proposition 2 and the derivations above, we have that at any optimal encoder solution $\phi^* = \{\boldsymbol{W}_z^*, \boldsymbol{b}_z^*\}$, both $\boldsymbol{W}_z^*$ and $\boldsymbol{b}_z^*$ are formed by left multiplication by $\left(\boldsymbol{V}_x^*\right)^\top$. Then based on Proposition 2 and

the stated structure of $\boldsymbol{V}_x^*$, it follows that $\boldsymbol{W}_z^* = \boldsymbol{P}\begin{bmatrix} \widetilde{\boldsymbol{W}}_z^* \\ \boldsymbol{0} \end{bmatrix}$ and $\boldsymbol{b}_z^* = \boldsymbol{P}\begin{bmatrix} \widetilde{\boldsymbol{b}}_z^* \\ \boldsymbol{0} \end{bmatrix}$, where $\widetilde{\boldsymbol{W}}_z^*$ has rank$[\boldsymbol{B}]$ rows and $\widetilde{\boldsymbol{b}}_z^* \in \mathbb{R}^{\text{rank}[\boldsymbol{B}]}$. Finally, there exists a bijection between $\boldsymbol{x}$ and $\{\boldsymbol{y}, \widetilde{\boldsymbol{\mu}}_z\}$ given that

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{W}_y^* \boldsymbol{x} + \boldsymbol{b}_y^* \text{ and } \widetilde{\boldsymbol{\mu}}_z = \widetilde{\boldsymbol{W}}_z^* \boldsymbol{x} + \widetilde{\boldsymbol{b}}_z^* \quad (\text{for } \boldsymbol{x} \rightarrow \{\boldsymbol{y}, \widetilde{\boldsymbol{\mu}}_z\} \text{ direction}) \\ \boldsymbol{x} &= \boldsymbol{W}_x^* \boldsymbol{y} + \boldsymbol{V}_x^* \boldsymbol{P}\begin{bmatrix} \widetilde{\boldsymbol{\mu}}_z \\ \boldsymbol{0} \end{bmatrix} + \boldsymbol{c} \quad (\text{for } \{\boldsymbol{y}, \widetilde{\boldsymbol{\mu}}_z\} \rightarrow \boldsymbol{x} \text{ direction}), \end{aligned} \tag{30}$$

completing the proof.