DisCoV: Disentangling Time Series Representations via Contrastive based l-Variational Inference

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Learning disentangled representations is crucial for Time Series, offering benefits like feature derivation and improved interpretability, thereby enhancing task performance. We focus on disentangled representation learning for home appliance electricity usage, enabling users to understand and optimize their consumption for a reduced carbon footprint. Our approach frames the problem as disentangling each attribute's role in total consumption (e.g., dishwashers, fridges, . . .). Unlike existing methods assuming attribute independence, we acknowledge real-world time series attribute correlations, like the operating of dishwashers and washing machines during the winter season. To tackle this, we employ weakly supervised contrastive disentanglement, facilitating representation generalization across diverse correlated scenarios and new households. Our method utilizes innovative l-variational inference layers with self-attention, effectively addressing temporal dependencies across bottom-up and top-down networks. We find that **DisCoV** (**Dis**entangling via **Co**ntrastive l-**V**ariational) can enhance the task of reconstructing electricity consumption for individual appliances. We introduce TDS *(Time Disentangling Score)* to gauge disentanglement quality. TDS reliably reflects disentanglement performance, making it a valuable metric for evaluating time series representations. Code available at `https://anonymous.4open.science/r/DisCo`

1 Introduction

Disentangled representation learning is crucial in various fields like computer vision, speech processing, and natural language processing [2]. It aims to improve model performance by learning latent disentangled representations and enhancing generalizability, robustness, and explainability. These representations have latent units that respond to single attribute changes while remaining invariant to others. Existing approaches assume independent attributes, but in real-world time series data, latent attributes are often causally related. This necessitates a new framework for causal disentanglement. For instance, in Fig 1, the consumption profile of "Dishwasher" and "Profile 2" cause variations in "Washing machine" and "Profile 1," showing the inadequacy of existing methods in capturing these non-independent attributes [31, 29]. One of the most common frameworks for disentangled

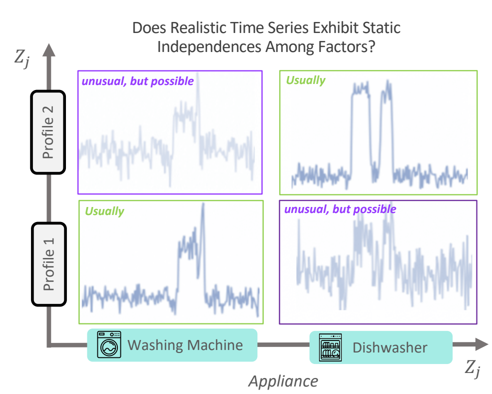

Figure 1: Illustrative of Real-world data often showcases attributes exhibit strong positive correlation: seasonal changes.

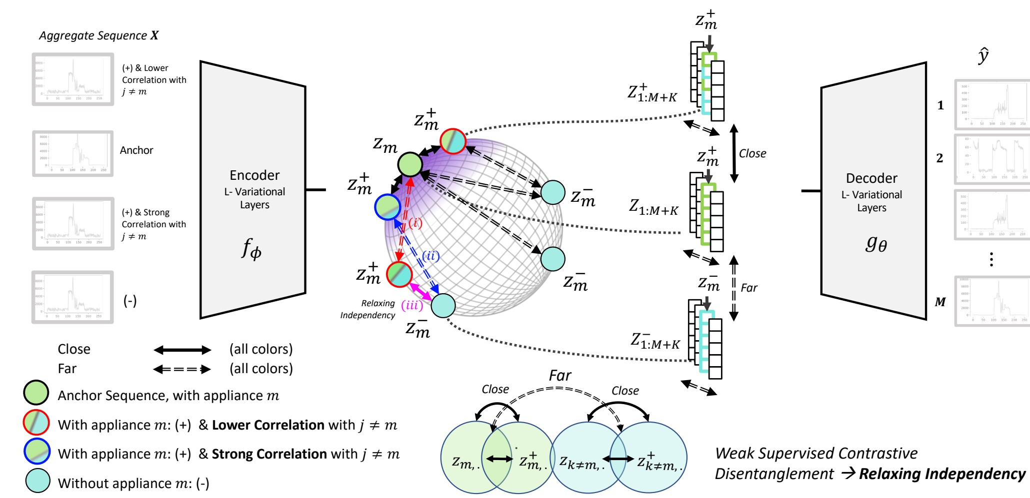

Figure 2: Latent attributes are causally [25] correlated, allows positive pairs $\mathbf{x}, \mathcal{T}(\mathbf{x}^+)$ to decrease their distance, while negative pairs increase it, and allows cases where unlikely combinations occur ((i) and (ii) lead to the existence of (iii)), although forcing statically independence does not prohibit these cases. Our framework is based on *contrastive disentanglement* to relax z to have a support factorization (allowing for some dependency).

representation learning is Variational Autoencoders (VAE) [11], a deep generative model trained to disentangle the underlying explanatory attributes. Disentanglement via VAE can be achieved by a regularization term of the Kullback-Leibler divergence between the posterior of the latent attributes and a standard Multivariate Gaussian prior [11], which enforces the learned latent attribute to be as independent as possible. It is expected to recover the latent variables if the observation in the real world is generated by countable independent attributes. To further enhance the independence, various extensions of VAE consider minimizing the mutual information among latent attributes [15]. [15] further encourage independence by reducing the total correlation among attributes. Our focus in this work is a more general case, where the data does not have specificity like domain frequency, or amplitude to analysis. Household energy consumption disaggregation, also known as Non-Intrusive Load Monitoring (NILM), is a key application. Given only the main consumption of a household, the energy disaggregation algorithm identifies which appliances are operating. Such a capability is extremely vital given the growing interest in reducing carbon footprints through user energy behavior, which poses a challenge to conventional algorithms. Many households rely on past bills to adjust future energy use, underscoring the importance of energy disaggregation algorithms. Recent work [3, 32, 23] hold promising results, yet persistent challenges in generalisability and robustness stem from the correlations occurring within time series a challenge that spans beyond the domain of time series in general. In this work, we tackle the energy disaggregation problem from the perspective of disentanglement.

Our work is distinguished by instead of assuming independent factors we will only assume that the support of the distribution factorizes. We explore how to design an efficient and disentangling representation under correlated attributes using weak supervised contrastive learning. An ablation investigation to understand the impact of considering statical independence versus the case where we avoid it by giving the latent space a support factorization through weakly supervised contrastive learning. This addresses latent space misalignment between attributes, maintains generalizability, and preserves disentanglement through the *Pairwise similarity* over $\mathbf{z}$ setting it apart from methods relying on *independence*. More clearly, we break the concept of independence, allowing any combination of individual attributes, to be possible, even if some combinations are unlikely, our experiments on three datasets and increasingly difficult correlation settings, show that DisCoV improves robustness to attribute correlation and improves disentanglement (as measured by SAP, DCI, RMIG,TDS) by up to $+21.7\%$ over state of the art (c.f. §5.3). Furthermore, we introduce an in-depth l-variational-based self-attention for extracting high semantic representations from time series. An ablation study shows that l-VAE learns complex representations; added attention improves further (in-depth model

$l = 4, 8, 16, 32$ c.f. fig. 6). This approach retains dimension reduction while avoiding temporal locality. Additionally, our proposed Time Disentanglement Metric (TDS) aligns more effectively with decoder output compared to existing metrics. These findings establish it as a strongly recommended for time series representations.

2 Related Work

Recent work [3, 23] has produced promising results. However, they are confronted with problems of interpretability, generalization, and robustness. Various approaches have been proposed to solve these problems. For instance, [3] introduced Convolutional Neural Networks (CNNs) for feature extraction from power consumption data, showing promise on the UK-DALE dataset [13]. Generalization concerns persist despite leveraging Gated Recurrent Units (GRUs) and attention mechanisms. Other works attempt meaningful representation of time series, but disentangling remains challenging [29, 27, 22]. Recurrent VAE (RVAE) [7] for sequential data, D3VAE [20] improves prediction using a diffusion model after decoding the latent space. In representation learning, [34] employs contrastive learning, but in correlated data scenarios it is not explored. [21] based on specific propriety of time series like frequency and amplitude to disentangling Time series, but disentangling the latent space through data-driven methods poses a challenge. Nevertheless, recent approaches like Support Factorization as described in the works of [35, 24] show promise in addressing this challenge and have yielded encouraging results.

3 Formulation

We consider a c-variate time series observed at times $t = 1, \ldots, \tau$. We denote by $\mathbf{x} \in \mathbb{R}^{c\times\tau}$ the $c \times \tau$ resulting matrix with rows denoted by $x_1, \ldots, x_c$. Each row can be seen as a univariate time series. In the electric load application, we have $c = 3$, and x_1 is the sampled active power, x_2 the sampled reactive power and x_3 the sampled apparent power. The goal of non-intrusive load monitoring (NILM) is to use $\mathbf{x}$ in order to express x_1 as

$$x_1 = \sum_{m=1}^{M} y_m + \xi \, , \tag{1}$$

where, for each $m = 1, \ldots, M$, $y_m \in \mathbb{R}^\tau$ represents the contribution of the m-th electric device among the M ones identified in the household, and $\xi \in \mathbb{R}^\tau$ denotes a residual noise. We further denote by $\mathbf{y}$ the $M \times \tau$ matrix with row-wise stacked devices' contributions.

The NILM mapping $\mathbf{x} \mapsto \{\mathbf{y}_1 \ldots \mathbf{y}_k\}$, where $\mathbf{x} = \sum_i \mathbf{y}$ is generally learnt from a training data set $\mathcal{S} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^N$. VAEs rely on two main ingredients: 1) a generative model (p_θ) based on a latent variable, and a decoder g_θ; 2) a variational family (q_ϕ), which approximates the conditional density of the latent variable given the observed variable based on an encoder f_ϕ.

In a VAE, both (unknown) parameters θ and ϕ are learnt from the training data set $\mathcal{S} = \{\mathbf{x}_n\}_{n=1}^N$. A key idea for defining the goodness of fit part of the learning criterion is to rely the **E**vidence **L**ower **B**ound (ELBO), which provides a lower bound on (and a proxy of) the log-likelihood

$$\log p_\theta(\mathbf{x}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z})) \, , \tag{2}$$

where we denoted the latent variable by $\mathbf{z}$, defined as a $(M + K) \times d_z$ matrix and p denotes its distribution. The use of ELBO goes back to traditional variational Bayes inference. An additional feature of VAE's is to define q_ϕ and p_θ through an encoder/decoder pair of neural networks (f_ϕ, g_θ). A standard choice in a VAE is to rely on Gaussian distributions and, for instance, to set $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \mu(\mathbf{x}, \phi), \sigma^2(\mathbf{x}, \phi))$, where $\mu(\mathbf{x}, \phi)$ and $\sigma^2(\mathbf{x}, \phi)$ are the outputs of the encoder f_ϕ.

As mentioned in Section 1, various additional features such as β/TC/Factor/DIP-VAE have been proposed, where a specific distribution $p(\mathbf{z})$ is learned. The objective is to disentangle the latent variable $\mathbf{z}$, and align it with the corresponding attribute. However, they assume statistical independence among attributes, leading to the assumption: $p(\mathbf{z}) = p(\mathbf{z}_1) \ldots p(\mathbf{z}_{\mathbf{M+K}})$. As we explained in the introduction, appliances are not used independently. In [24], correlated attributes have been taken into account by replacing the factorization constraint with support factorization via Hausdorff Factorized

Support (HFS). In order to meet this criterion, they penalize the Hausdorff pairwise estimate Eq.3, based solely on the distance without any alignment on the input.

$$\hat{d}_H(\mathbf{z}) = \sum_{i=1}^{(M+K)-1} \sum_{j=i+1}^{(M+K)} \max_{z \in \{\mathbf{z}_{:,i}\} \times \{\mathbf{z}_{:,j}\}} \left[\min_{z' \in \{\mathbf{z}_{:,(i,j)}\}} \|z - z'\| \right] . \tag{3}$$

We are investigating an alternate way to achieve both alignment and disentanglement leading to a generalizable representation. To that end, we draw on support factorization, and we replace $\hat{d}_H(\mathbf{z})$ by a *Pairwise Similarity* penalty. In the next section, we develop our proposed method based on weakly contrastive learning to have factorized support, and it provides an advantage in terms of computation and latent representation.

4 Proposed Methods

Our objective is to disentangle latent space by relaxing the independence, for this, we now define a concrete training criterion that encourages factorized support. Let us consider deterministic representations obtained by the encoder $\mathbf{z} = f_\phi(\mathbf{x})$. We enforce the factorial support criterion on the aggregate distribution $\bar{q}_\phi(\mathbf{z}) = \mathbb{E}_\mathbf{x}[f_\phi(\mathbf{x})]$, where $\bar{q}_\phi(\mathbf{z})$ is conceptually similar to the aggregate posterior $q_\phi(z)$ in, e.g., TCVAE, though we consider points produced by a deterministic mapping f_ϕ rather than a stochastic one. To match our factorized support assumption on the ground truth, we want to encourage the support of $\bar{q}_\phi(z)$ to factorize, i.e., that $Supp(\bar{q}_\phi(z))$ and the Cartesian product of each dimension support, $Supp^\times(\bar{q}_\phi(z))$, are equal. In practical scenarios, we often deal with a finite sample of observations $\{\mathbf{x}_i\}_{i=1}^N$ and can only estimate support on a finite set of representations $\{f_\phi(\mathbf{x}_i)\}_{i=1}^N$. To encourage such a pairwise factorized support, we can minimize sliced/pairwise contrastive with the additional benefit of keeping computation tractable when k is large. Specifically, we approximate the support as $Supp \approx \mathbf{z}$ and the Cartesian product of each dimension's support as $Supp^\times \approx z_{:,1} \times z_{:,2} \times \ldots \times z_{:,k} = \{(z_1, \ldots, z_k) \mid z_1 \in z_{:,1}, \ldots, z_k \in z_{:,k}\}$.

4.1 Support factorization via Weakly supervised Constrastive

Let us first formalize the contrastive learning setup. Each training triplet comprises a reference sample $\mathbf{x}$ along with a positive (similar) sample $\mathbf{x}^+$ and negative (dissimilar) samples $\mathbf{x}_1^-, \ldots, \mathbf{x}_N^-$ against which it is to be contrasted. As introduced in the previous section, we assume that these samples generate corresponding latent: $\mathbf{z}, \mathbf{z}^+, \mathbf{z}_1^-, \ldots, \mathbf{z}_N^-$. The positive sample, denoted as $\mathbf{z}^+$, is generated from a closely related dataset in which appliance m is activated. In contrast, the negative samples, $\mathbf{z}_1^-, \ldots, \mathbf{z}_N^-$, are drawn from a dataset where appliance m remains inactive. This formalization of contrastive learning ensures that positive samples are semantically similar and negatives are dissimilar. Self-supervised contrastive learning is widely used in computer vision, in [14], the loss is defined as:

$$\mathcal{L}_{\text{self}} = -\sum_{i \in I} \log \frac{e^{(z_i \cdot z_{j(i)}/\tau)}}{\sum_{a \in \mathcal{A}(i)} e^{(z_i \cdot z_a/\tau)}}. \tag{4}$$

where, $z_i \in Z, \textit{where, } Z = f_\phi(\mathbf{x})$, the $\cdot$ symbol denotes the inner (dot) product, $\tau \in \mathbb{R}^+$ is a scalar temperature parameter, and $\mathcal{A}(i) \equiv I \setminus \{i\}$. The index i is called the anchor z_i, index $j(i)$ is refer to the positive z_i^+, and the other $2(N-1)$ indices ($\{k \in \mathcal{A}(i) \setminus \{j(i)\}\}$) are called the negatives $z_{k \neq i}^-$. We note that for each anchor i, there is 1 positive pair and $2N-2$ negative pairs. The denominator has a total of $2N-1$ terms (the positives and negatives). In a multiclass scenario, disentangling and aligning data encounters challenges when several samples belong to the same class, as we aim to match certain pairs of data points (e.g., $z_{i,j}$ to $z_{i,j}^+$) and drive others away (i.e. $z_{i,j}$ from $z_{k \neq i,j}$ or $z_{k \neq i,j}^+$). We link the learned latent representation to ground-truth attributes using a limited number of pair labels. This connection is facilitated by employing positive and negative samples, as demonstrated in [34]. We adapt this, by firstly, the loss should not rely on statically independent attributes, mirroring realistic data scenarios; secondly, it should prioritize attribute alignment to maintain sufficient information [35]. To achieve this, the proposed disentanglement loss combines two terms. The first term enforces axis alignment based on the correlation between $z_{:,m}$ and $z_{:,m}^+$ (positive augmentation of $z_{:,m}$). This ensures that only one latent variable learns this alignment for fixed attributes (invariant). The second term minimises information redundancy by measuring the correlation between $z_{:,m}$ and $z^{+}_{:,p\neq m}$ or ($z_{:,m}$ and $z_{:,p\neq m}$), which are almost equivalent in a contrasting sense.

$$\mathcal{L}_{DIS} = \sum_{m=1}^{r} \left(1 - d(z_{:,m}, z^{+}_{:,m})\right)^2 + \sum_{m=1}^{r} \sum_{:,p\neq m}^{r-1} d(z_{:,m}, z^{+}_{:,p})^2 \tag{5}$$

$r = M + K$, we define $d(z_{:,m}, z^{+}_{:,m})$ as the cosine similarity between vectors $z_{:,m}$ and $z^{+}_{:,m}$ in a mini-batch. Furthermore, this helps to support factorizing the latent space as it measures the correlation between $z_{:,m}$ and $z^{+}_{:,p\neq m}$ or $z_{:,p\neq m}$, and performs better than estimating Eq.3. Augmentation affects only one attribute, with others remaining fixed. We assume sufficient augmentation for each factor across the batch. Our results indicate both terms equally contribute to improved disentangling without weighted hyperparameters (c.f. ablation §6.1).

4.2 Attentive l-Variational auto-encoders and Objective function

To avoid time locality during dimension reduction, and keep long-range capability we refer to an in-depth Temporal Attention with l-Variational layers. NVAE [26, 1] proposed an in-depth autoencoder for which the latent space $\mathbf{z}$ is level-structured and attended locally [1], this shows an effective results for image reconstruction. In this work, we enable the model to establish strong couplings, as depicted. Our core idea aims to address construct $\hat{T}^l$ (Time context) that effectively captures the most informative features from a given sequence $T^{<l} = \{T^i\}_{i=1}^{l}$ across bottom-up and top-down, where $T^{<l}$ is the output of the residual network. Both $\hat{T}^l$ and T^l are features with the same dimensionality: $\hat{T}^l \in \mathbb{R}^{T\times C}$ and $T^i \in \mathbb{R}^{T\times C}$. In our model, we employ Temporal Self-attention [28] to construct either the prior or posterior beliefs of variational layers, which enables us to handle long context sequences with large dimensions τ effectively. The construction of $\hat{T}^l$ relies on a query feature $\mathbf{Q}^l \in \mathbb{R}^{T\times Q}$ of dimensionality Q with $Q \ll C$, and the corresponding context T^l is represented by a key feature $\mathbf{K}^l \in \mathbb{R}^{T\times Q}$. Importantly, $\hat{T}^l(t)$ of time step i in sequence τ depends solely on the time instances in $T^{<l}$. For more consistency, using Multihead-attention [28] allows the model to focus on different aspects of the input sequence simultaneously, which can be useful for capturing various relationships and patterns. which allows the model to jointly attend to information from different representation subspaces at different scales. Instead of computing a single attention function, this method first projects $\mathbf{Q}^l$, $\mathbf{K}^{<l}$), $\mathbf{T}^{<l}$) into h different vectors, respectively. Attention is applied individually to these h projections. The output is a linear transformation of the concatenation of all attention outputs. An in-depth description of this mechanism is given in Appendix 8.2. For the remainder of this paper, we presume that **DisCoV** employs self-attention.

We adopt the Gaussian residual parametrization between the prior and the posterior. The prior is given by $p(\mathbf{z}_l|\mathbf{z}_{<l}) = \mathcal{N}(\mu(T^l_p, \theta), \sigma(T^l_p, \theta)$. The posterior is then given by $q(\mathbf{z}_l|\mathbf{x}, \mathbf{z}_{<l}) = \mathcal{N}(\mu(T^l_p, \theta) + \boldsymbol{\Delta}\mu(\hat{T}^l_q, \phi), \sigma(T^l_p, \theta) \cdot \boldsymbol{\Delta}\sigma(\hat{T}^l_q, \phi))$ where the sum $(+)$ and product $(\cdot)$ are pointwise, and T^l_q is defined in Eq 14. $\mu(\cdot)$, $\sigma(\cdot)$, $\boldsymbol{\Delta}\mu(\cdot)$, and $\boldsymbol{\Delta}\sigma(\cdot)$ are transformations implemented as convolutions layers. Based on this, For $\mathcal{L}_{KL}$ in Eq 2, the last term is approximated by: $0.5 \times \left(\frac{\boldsymbol{\Delta}\mu_1^2}{\sigma_l^2} + \boldsymbol{\Delta}\sigma_l^2 - \log \boldsymbol{\Delta}\sigma_l^2 - 1\right)$. Our DisCoV objective function combines the VAE loss (Eq.2), consisting of a reconstruction term $\mathcal{L}_{rec}$ (focused on minimizing Mean Squared Error), with the contrastive term on $\mathbf{z}$ (Eq.5). We introduce balancing factors β and λ (discussed in §6.2) to control their impact.

$$\mathcal{L}_{DisCo} = \underbrace{\mathcal{L}_{rec} + \beta\mathcal{L}_{KL}}_{\beta\text{-VAE}} + \lambda\mathcal{L}_{DIS} \tag{6}$$

4.3 How to evaluate disentanglement for Time Series?

Evaluating disentanglement in series representation is more challenging than established computer vision metrics. Existing time series methods rely on qualitative observations and predictive performance, while metrics like Mutual Information Gap (MIG) [20] have limitations with continuous labels. To address this, we adapted RMIG [4] for continuous labels and used DCI metrics from [8].

Additionally, we employed SAP [17] to measure prediction error differences in the most informative latent dimensions for ground truth attributes. Our evaluation, including β-VAE and FactorVAE scores, can be found in Appendix 8.1. These metrics face challenges with sequential data and do not provide measures of attribute alignment.

To overcome this limitation, we introduce the Time Disentanglement Score (TDS) from an information-gain perspective. TDS assesses how well the latent representation $\mathbf{z} = f_\phi(\mathbf{x})$ maintains the invariance of an attribute m in $\mathbf{x}$ when this attribute changes. TDS relies on the correlation matrix between $\mathbf{z}$ and $\mathbf{z}^+$, where $\mathbf{z} = f_\phi(\mathbf{x})$ and $\mathbf{z}^+ = f_\phi(\mathcal{T}(\mathbf{x}))$, with $\mathcal{T}$ denoting an augmentation function. This correlation matrix quantifies the consistency of attribute components. Additionally, TDS evaluates how well $\mathbf{z}$ contributes to the reconstruction of $\mathbf{y}$ and how $\mathbf{z}^+$ contributes to the reconstruction of $\mathcal{T}(\mathbf{y})$. Specifically, it assesses whether each z_m (or z_m^+) can effectively reconstruct the corresponding y_m (or y_m^+). TDS aligns with qualitative observations of disentanglement (c.f. fig. 5).

Metric	Align-axis	Unbiased	General
β-VAE [12]	No	No	No
FactorVAE [15]	Yes	No	No
RMIG [4]	Yes	No	Yes
SAP [18]	Yes	No	Yes
DCI [8]	Yes	Yes	No
TDS (Ours)	Yes	Yes	Yes

Table 1: In comparison to prior metrics, our proposed TDS detects axis alignment, is unbiased for all hyperparameter settings and can be generally applied to any latent distributions provided efficient estimation exists.

$$TDS = \frac{1}{2}\left[(1 - \sum_i Corr^{(I)}(\mathbf{z}, \mathbf{z}^+)_{ii})^2 + (1 - \sum_i Corr^{(I)}(\mathbf{y} - \hat{\mathbf{y}}, \mathbf{y}^+ - \hat{\mathbf{y}}^+)_{ii})\right]^2 \tag{7}$$

where $Corr_{ij}^{(I)} = \sum_b \mathbf{z}_{b,i}\mathbf{z}_{b,j}^+$ divided by $\sqrt{\sum_b(\mathbf{z}_{b,i})^2}\sqrt{\sum_b(\mathbf{z}_{b,j}^+)^2}$, b indexes batch samples and i, j index the vector dimension of the networks' f_ϕ outputs for $Corr^{(I)}$ (resp. dimension of the networks' outputs of g_θ for $Corr^{(II)}$). $Corr$ is a square matrix with the size of the dimensionality of the network's output and with values comprised between -1 (i.e. perfect anti-correlation) and 1 (i.e. perfect correlation). In practice, the augmentation function $\mathcal{T}$ is effectively a sampling of appliance activation (i.e. from different sources, houses/datasets) for the positive case and sequences where the device is not activated for the negative case. We note that high TDS informativeness signifies strong disentanglement, while a significant distance implies reduced disentanglement and higher attribute correlation, aligning with [9]. More in-depth explanation can be found in the appendix 8.1.4.

5 Experiments

5.1 Experimental Setup

Datasets. We conducted experiments on two publicly available datasets, namely UK-DALE [13] and REDD [16]. The dataset UK-DALE [13] consists of 5 dwellings with a varying number of sub-metered devices and includes aggregate and individual aggregate and individual equipment-level power measurements, sampled equipment, sampled at 1/6 Hz.

Evaluation Metrics. We adopt RMSE to evaluate the accuracy of all compared methods. Details of these three metrics can be found in Appendix 11.1.1

Baseline. We compare DisCoV with down task models in energy, Bert4NILM [33] and S2P [30], S2P [5], for those model we keep the same configuration as the original implémentation. We provide also a varieté de β-TC/Factor/-VAE implemented for time series, compared to D3VA [20] and NVAE [27], and RVAE [7]

Experimental Platform. We conduct 5 rounds of experiments, reporting the averaged results and standard deviation. The experiments are performed on four NVIDIA A40 GPUs and 40 Intel(R) 281 Xeon(R) Silver 4210 CPU @ 2.20GHz. The models are implemented in PyTorch. Detailed hyperparameter settings are available in Appendix 8.3.

5.2 Architecture Settings

Our model uses a bi-directional encoder, which processes the input data in a hierarchical manner to produce a low-resolution latent code that is refined latent code that is refined by a series of oversampling layers. This code is then refined by a series of oversampling layers in *Residual Decoders* blocks, which progressively increases the resolution.

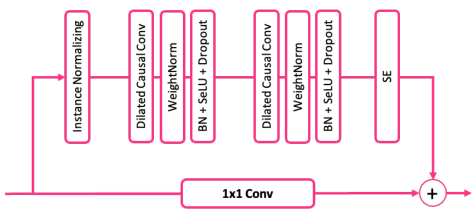

Figure 3: Residual Cell for Encodeur (Inference Model q_ϕ)

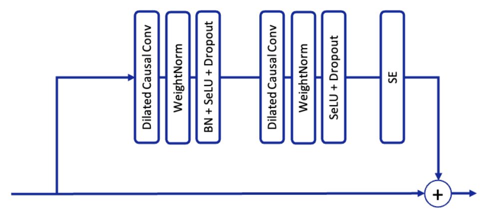

Figure 4: Residual Cell for DisCoV Decodeur (Generative Model p_θ).

Residual Blocs. Activation functions are pivotal for enabling models to learn nonlinear representations, but vanishing and exploding gradients can hinder learning. The Temporal Convolutional Network (TCN) [19] tackles these issues using Rectified Linear Unit (ReLU), weight normalization, and dropout layers. In our Residual model, we simplify the residual block by replacing these components with the Sigmoid Linear Units, which offers advantages and immunity to gradient problems. It reduces training time, efficiently learns robust features, and outperforms weight normalization. SiLU [10] is defined as $\text{SiLU}(x) = x \times \sigma(x)$ where $\sigma(x)$ is the logistic sigmoid.

Squeeze-and-Excitation on Spatial and Temporal. SE block enhances our neural networks by selectively emphasizing important features and suppressing less relevant ones. It does this through global information gathering (squeezing) and feature recalibration (excitation). We find that extending SE for time series data improves the capture of significant temporal patterns in sequence. Our Residual encoders (Inference Model q_ϕ) in Fig 3 and Decodeur (Generative Model p_θ) in Fig 4.

DisCoV ($L = 8$)	KL ↓	RMSE ↓	Time (s) ↓
ReLU	0.734	0.734	28800
SiLU	0.671	0.671	**21600**
ReLU+SE	0.721	0.721	32760
SiLU+SE	**0.582**	**0.582**	23040

Table 2: RMSE Scores for Different DisCoV Variants activation function and SE, as L Increases. (↓ the lower values are better).

5.3 Performance and Informativity of Contrastive

Finding: *DisCoV retains its robustness in correlated scenarios and achieves comparable performance to baseline models.*

In evaluating the robustness of DisCoV regarding correlations in appliance signatures or consumption, we consider several pairs of appliances. Firstly, there's the **No Correlation** scenario, where we examine the correlation between the refrigerator's signature and the dishwasher's signature. These appliances are typically active at different times, resulting in less correlated signatures. Moving on to specific pairs, **Pair 1** involves analyzing the correlation between the washing machine's signature and the dryer's signature. Given that these appliances are often used sequentially, their signatures might exhibit some level of correlation. In **Pair 2**, the focus is on evaluating the correlation between the microwave's signature and the oven's signature. These appliances have distinct power profiles and usage patterns, potentially leading to lower correlation. **Pair 3** explores the correlation between the lighting's power consumption and the television's power consumption. Since these appliances are often used independently, their signatures may exhibit a lower level of correlation. Lastly, the **Random Pair** approach involves selecting two random appliances from a dataset.

Table 3: Disentanglement by Contrastive on **UK-DALE**, **Uk-Dale** across various correlated appliances (columns) and correlation increasing from left (no correlation) to right (every appliance correlated to one confounder). Scores denote DCI metric computed on uncorrelated test data. Bold denotes the best performance per correlation. $[x, y]$ indicate 25/75th percentiles.

Method	**No Corr**	**Pairs: 1**	**Pairs: 2**	**Pairs: 3**	**Random Pair.**	σ
REDD [16]						
β-VAE	72.4 [68.1, 76.9]	70.3 [62.8, 73.5]	54.5 [49.3, 59.1]	39.8 [34.2, 42.7]	40.6 [37.8, 41.9]	3.10
HFS	79.8 [76.5, 84.6]	78.6 [75.2, 80.1]	57.8 [52.0, 59.7]	48.7 [43.4, 50.5]	47.1 [41.9, 48.7]	1.10
β-VAE + HFS	93.1 [78.2, 101.3]	81.9 [77.2, 82.4]	69.4 [64.3, 71.7]	49.2 [45.2, 52.2]	65.1 [62.5, 67.5]	2.12
β-TCVAE	78.0 [77.5, 79.2]	71.9 [67.1, 73.3]	64.7 [61.0, 66.0]	49.0 [38.3, 52.5]	51.6 [47.5, 57.6]	1.01
β-TCVAE + HFS	87.2 [84.0, 98.8]	76.5 [64.4, 77.9]	69.9 [62.6, 73.4]	52.1 [48.2, 53.3]	62.1 [54.4, 64.8]	1.01
FactorVAE	68.4 [53.5, 71.4]	73.2 [72.9, 73.6]	59.7 [58.4, 64.5]	48.4 [42.4, 50.6]	33.0 [29.3, 36.5]	3.12
DisCoV	**63.5 [62.0, 64.5]**	**58.5 [50.8, 60.3]**	**32.9 [28.2, 35.4]**	**34.9 [32.3, 39.3]**	**24.3 [21.4, 27.2]**	1.35
Uk-dale [13]						
β-VAE	34.2 [27.3, 39.9]	11.5 [9.9, 12.3]	9.5 [8.7, 10.3]	N/A	13.4 [11.9, 15.9]	0.48
HFS	37.9 [30.4, 39.0]	15.6 [9.6, 18.7]	13.9 [11.7, 15.8]	N/A	17.2 [13.1, 18.0]	1.38
β-VAE + HFS	52.1 [32.2, 52.6]	21.9 [19.2, 23.3]	19.5 [8.2, 21.8]	N/A	17.9 [14.3, 18.8]	0.22
β-TCVAE	32.1 [30.1, 36.4]	25.2 [24.8, 25.6]	12.4 [8.6, 14.6]	N/A	21.9 [18.5, 24.6]	0.13
β-TCVAE + HFS	55.4 [44.1, 55.5]	27.9 [26.6, 28.6]	29.2 [17.5, 33.0]	N/A	26.2 [25.2, 27.7]	0.11
FactorVAE	29.7 [24.9, 34.9]	19.1 [15.9, 20.3]	17.4 [16.4, 19.0]	N/A	18.7 [17.5, 19.3]	0.23
DisCoV	**42.4 [41.7, 43.0]**	**16.8 [16.3, 17.9]**	**10.5 [8.9, 12.3]**	N/A	**16.3 [16.1, 16.5]**	0.42

Table 4: RMSE in $Watt^2$ on **UK-DALE** and **REDD** data.

Machine	**Dataset Test**	**S2P**	**S2S**	**Bert4NILM**	**RVAE**	β**-TCVAE**	**FactorVAE**	**NVAE**	**D3VAE**	**DisCoV (Ours)**
Fridge	UK-DALE	25.70	25.68	25.69	25.74	27.36	26.70	27.36	28.36	**19.55**
	REDD	25.49	25.47	25.48	26.56	30.68	26.56	30.68	21.18	**19.48**
Washing Machine	UK-DALE	25.78	25.76	25.77	25.63	28.92	24.72	28.92	21.12	**18.33**
	REDD	25.59	25.57	25.58	25.34	28.40	24.78	28.40	23.22	**18.31**
Oven	UK-DALE	25.61	25.59	25.60	25.46	25.28	23.98	25.28	22.18	**19.30**
	REDD	25.45	25.43	25.44	25.42	25.04	23.94	25.04	20.78	**19.82**

6 Ablation Studies

In this section, we conduct ablation experiments to assess DisCo's effectiveness and robustness in comparison to traditional variant VAEs. Our experiments utilize the Uk-Dale, REDD, and REFIT datasets with a fixed random seed. We include additional ablation results in Appendix **??**.

6.1 In-depth self-attention l-VAEs learn an effective representation.

Finding: *DisCoV with increasing depth, the representation becomes over 20% more separable (40% in terms of TDS), downtasking improves performance by 50%, and attention mechanisms contribute to a 10% enhancement in results.*

Table 6, we observe notable differences in performance as the depth (L) of the model architecture varies including Root Mean Square Error (RMSE), Relative Mutual Information Gain (RMIG), and Task Discriminative Score (TDS) for various methods, with a particular emphasis on DisCoV variants with and without attention as the depth (L) increases. Regarding RMSE, which measures the accuracy of the models, we find that the baseline methods VAE, β-TCVAE, and DIP-VAE exhibit consistently higher RMSE values compared to the DisCoV variants. Furthermore, introducing the 'DIS' significantly improves RMSE values across all methods, indicating the effectiveness of the DisCoV loss in enhancing model performance. Additionally, as depth (L) increases from 4 to 16, we observe that the DisCoV variants consistently outperform the baseline methods in terms of RMSE. Notably, when L reaches 16, both DisCoV and DisCoV attention achieve the lowest RMSE value of 0.48, showcasing the superior performance of DisCo-based models with higher depth. It is also worth mentioning that RMIG and TDS metrics follow a similar trend, with DisCoV variants demonstrating superior performance, especially as L increases. These findings suggest that increasing the depth of the model architecture and incorporating DisCoV loss play pivotal roles in improving model accuracy and task discriminative capabilities, highlighting the significance of attention mechanisms in enhancing performance.

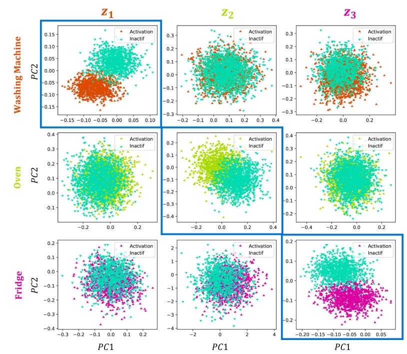

Figure 5: PCA visualization for $M = 3$, $K = 1$: Rows represent latent representations of activated appliances (Washing Machine, Oven, Fridge from top to bottom), columns correspond to $\mathbf{z}_m$ components of structured latent variable $\mathbf{z}$.

Method	Depth (L)	RMSE ↓	RMIG ↓	TDS ↓
VAE (baseline)	-	0.928	0.921	0.935
VAE (baseline)+DIS	-	0.929	0.924	0.931
FactorVAE	-	0.942	0.931	0.973
β-TCVAE	-	0.931	0.918	0.937
β-TCVAE+DIS	-	0.930	0.922	0.933
DIP-VAE	-	0.932	0.915	0.939
DIP-VAE+DIS	-	0.928	0.926	0.930
DisCoV	8	0.50	0.73	0.71
DisCoV w/o Attention	8	0.54	**0.71**	0.72
DisCoV	16	0.49	0.74	0.70
DisCoV w/o Attention	16	0.52	0.72	0.73
DisCoV	32	**0.48**	0.75	**0.69**

Figure 6: RMSE, RMIG, and TDS Scores for Variants DisCoV w/,w/o Attention, as L Increases. (↓ lower values are better).

6.2 Robustness, Disentanglement, and Strong Generalization

Finding: *DisCoV demonstrates robust disentanglement performance across varying dimensions, while FactorVAE exhibits degradation as dimensionality increases M ↑.*

We report the disentanglement performance of DisCoV and FactorVAE on the Uk-dale dataset as M is increased. FactorVAE [11] is the closest TC-based method: it uses a single monolithic discriminator and the density-ratio trick to explicitly approximate $TC(\mathbf{z})$. Computing $TC(\mathbf{z})$ is challenging to compute as M increases. The results for $M = 10$ (scalable $\approx \times 3$) are included for comparison. The average disentanglement scores for DisCoV $M = 7$ and $M = 10$ are very close, indicating that its performance is robust in M. This is not the case for FactorVAE it performs worse on all metrics when m increases. Interestingly, FactorVAE $M = 10$ seems to recover its performance on most metrics with higher β than is beneficial for FactorVAE $M = 10$. Despite this, the difference suggests that FactorVAE is not robust to changes in M.

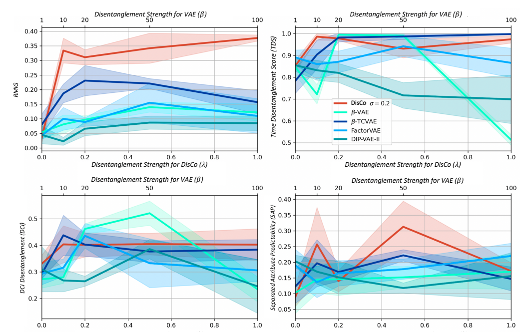

Figure 7: Disentanglement metric comparison of **DisCoV** with **VAE** baselines on UKDALE. **DisCoV** λ is plotted on the lower axis, and VAE-based method regularization strength β is plotted on the upper axis. Dark lines average scores. Shaded areas one standard deviation.

7 Conclusion

To address the limitation of assuming independence in traditional disentangling methods, which doesn't align with real-world correlated data, we explore an approach focused on recovering correlated data. This method achieves untangling by enabling the model to encode diverse combinations of generative attributes in the latent space. Using DisCo, we demonstrate that promoting pairwise factorized support is adequate for traditional untangling techniques. Additionally, we find that DisCoV performs competitively with downstream tasks (i.e. NILM methods) and delivers significant relative improvements of over +60% on common benchmarks across various correlation shifts in datasets.

References

[1] I. Apostolopoulou, I. Char, E. Rosenfeld, and A. Dubrawski. DEEP ATTENTIVE VARIATIONAL INFERENCE. 2022.

[2] Y. Bengio, A. Courville, and P. Vincent. Representation Learning: A Review and New Perspectives, Apr. 2014. arXiv:1206.5538 [cs].

[3] G. Bucci, E. Fiorucci, S. Mari, and A. Fioravanti. A New Convolutional Neural Network-Based System for NILM Applications. *IEEE Transactions on Instrumentation and Measurement*, 2021.

[4] M.-A. Carbonneau, J. Zaidi, J. Boilard, and G. Gagnon. Measuring Disentanglement: A Review of Metrics, May 2022. arXiv:2012.09276 [cs].

[5] K. Chen, Q. Wang, Z. He, K. Chen, J. Hu, and J. He. Convolutional sequence to sequence non-intrusive load monitoring. *the Journal of Engineering*, 2018(17):1860–1864, 2018. Publisher: Wiley Online Library.

[6] R. T. Q. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud. Isolating Sources of Disentanglement in Variational Autoencoders. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[7] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio. A Recurrent Latent Variable Model for Sequential Data. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[8] K. Do and T. Tran. Theory and Evaluation Metrics for Learning Disentangled Representations, Mar. 2021. arXiv:1908.09961 [cs, stat].

[9] C. Eastwood and C. K. I. Williams. A FRAMEWORK FOR THE QUANTITATIVE EVALUATION OF DISENTANGLED REPRESENTATIONS. 2018.

[10] S. Elfwing, E. Uchibe, and K. Doya. Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning, Nov. 2017. arXiv:1702.03118 [cs].

[11] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. Nov. 2016.

[12] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. -VAE: LEARNING BASIC VISUAL CONCEPTS WITH A CONSTRAINED VARIATIONAL FRAMEWORK. 2017.

[13] J. Kelly and W. Knottenbelt. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Scientific data*, 2, 2015. Publisher: Nature Publishing Group.

[14] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan. Supervised Contrastive Learning, Mar. 2021. arXiv:2004.11362 [cs, stat].

[15] H. Kim and A. Mnih. Disentangling by Factorising, July 2019. arXiv:1802.05983 [cs, stat].

[16] J. Z. Kolter and M. J. Johnson. REDD: A public data set for energy disaggregation research. In *Workshop on data mining applications in sustainability (SIGKDD), San Diego, CA*, volume 25, 2011. Issue: Citeseer.

[17] A. Kumar, P. Sattigeri, and A. Balakrishnan. VARIATIONAL INFERENCE OF DISENTANGLED LATENT CONCEPTS FROM UNLABELED OBSERVATIONS. 2018.

[18] A. Kumar, P. Sattigeri, and A. Balakrishnan. Variational Inference of Disentangled Latent Concepts from Unlabeled Observations, Dec. 2018. arXiv:1711.00848 [cs, stat].

[19] C. Lea, R. Vidal, A. Reiter, and G. D. Hager. Temporal Convolutional Networks: A Unified Approach to Action Segmentation, Aug. 2016. arXiv:1608.08242 [cs].

[20] Y. Li, X. Lu, Y. Wang, and D. Dou. Generative Time Series Forecasting with Diffusion, Denoise, and Disentanglement, Jan. 2023. arXiv:2301.03028 [cs].

[21] S. Liu, X. Li, G. Cong, Y. Chen, and Y. Jiang. MULTIVARIATE TIME-SERIES IMPUTATION WITH DIS- ENTANGLED TEMPORAL REPRESENTATIONS. 2023.

[22] L. Maaløe, M. Fraccaro, V. Liévin, and O. Winther. BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling, Nov. 2019. arXiv:1902.02102 [cs, stat].

[23] C. Nalmpantis and D. Vrakas. On time series representations for multi-label NILM. *Neural Computing and Applications*, 32(23), 2020.

[24] K. Roth, M. Ibrahim, Z. Akata, P. Vincent, and D. Bouchacourt. Disentanglement of Correlated Factors via Hausdorff Factorized Support, Feb. 2023. arXiv:2210.07347 [cs, stat].

[25] R. Shanmugam. Elements of causal inference: foundations and learning algorithms. *Journal of Statistical Computation and Simulation*, 88(16):3248–3248, Nov. 2018.

[26] A. Vahdat and J. Kautz. NVAE: A Deep Hierarchical Variational Autoencoder. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, 2020.

[27] A. Vahdat and J. Kautz. NVAE: A Deep Hierarchical Variational Autoencoder, Jan. 2021. arXiv:2007.03898 [cs, stat].

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Kaiser, and I. Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[29] G. Woo, C. Liu, D. Sahoo, A. Kumar, and S. Hoi. COST: CONTRASTIVE LEARNING OF DISENTANGLED SEASONAL-TREND REPRESENTATIONS FOR TIME SERIES FORECASTING. 2022.

[30] M. Yang, X. Li, and Y. Liu. Sequence to Point Learning Based on an Attention Neural Network for Nonintrusive Load Decomposition. *Electronics*, 2021.

[31] M. Yang, F. Liu, Z. Chen, X. Shen, J. Hao, and J. Wang. CausalVAE: Disentangled Representation Learning via Neural Structural Causal Models. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9588–9597, Nashville, TN, USA, June 2021. IEEE.

[32] Y. Yang, J. Zhong, W. Li, T. A. Gulliver, and S. Li. Semisupervised Multilabel Deep Learning Based Nonintrusive Load Monitoring in Smart Grids. *IEEE Transactions on Industrial Informatics*, 16(11):6892–6902, Nov. 2020.

[33] Z. Yue, C. R. Witzig, D. Jorde, and H.-A. Jacobsen. BERT4NILM: A Bidirectional Transformer Model for Non-Intrusive Load Monitoring. In *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, NILM'20, pages 89–93, New York, NY, USA, Nov. 2020. Association for Computing Machinery.

[34] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny. Barlow Twins: Self-Supervised Learning via Redundancy Reduction, June 2021. arXiv:2103.03230 [cs, q-bio].

[35] R. S. Zimmermann, Y. Sharma, S. Schneider, M. Bethge, and W. Brendel. Contrastive Learning Inverts the Data Generating Process, Apr. 2022. arXiv:2102.08850 [cs].

8 Extension and Implementation Details

8.1 Implementation of Metrics

All our metrics consider the expected representation of training samples (except total correlation for which we also consider the sampled representation as described bellow).

8.1.1 BetaVAE Metric

[12] suggest fixing a random factor of variation in the underlying generative model and sampling two mini-batches of observations x. Disentanglement is then measured as the accuracy of a linear classifier that predicts the index of the fixed factor based on the coordinate-wise sum of absolute differences between the representation vectors in the two mini-batches. We sample two batches of 64 points with a random factor fixed to a randomly sampled value across the two batches, and the others varying randomly. We compute the mean representations for these points and take the absolute difference between pairs from the two batches. We then average these 64 values to form the features of a training (or testing) point. We train a Scikit-learn logistic regression with default parameters on 10,000 points and test on 5,000 points.

8.1.2 FactorVAE Metric

[15] address several issues with this metric by using a majority vote classifier that predicts the index of the fixed ground-truth factor based on the index of the representation vector with the least variance. First, we estimate the variance of each latent dimension by embedding 10,000 random samples from the data set, excluding collapsed dimensions with variance smaller than 0.05. Second, we generate the votes for the majority vote classifier by sampling a batch of 64 points, all with a factor fixed to the same random value. Third, we compute the variance of each dimension of their latent representation and divide it by the variance of that dimension computed on the data without interventions. The training point for the majority vote classifier consists of the index of the dimension with the smallest normalized variance. We train on 10,000 points and evaluate on 5,000 points.

8.1.3 Mutual Information Gap Metric

[6] argue that the BetaVAE metric and the FactorVAE metric are neither general nor unbiased as they depend on some hyperparameters. They compute the mutual information between each ground truth factor and each dimension in the computed representation $r(x)$. For each ground-truth factor zk, they then consider the two dimensions in $r(x)$ that have the highest and second highest mutual information with zk. The Mutual Information Gap (MIG) is then defined as the average, normalized difference between the highest and second highest mutual information of each factor with the dimensions of the representation. The original metric was proposed evaluating the sampled representation. Instead, we consider the mean representation, in order to be consistent with the other metrics. We estimate the discrete mutual information by binning each dimension of the representations obtained from 10,000 points into 20 bins. Then, the score is computed as follows:

$$\frac{1}{K}\sum_{k=1}^{K}[I(v_{jk}, zk) - \max I(v_j, zk)]$$

Where zk is a factor of variation, vj is a dimension of the latent representation, and $jk = \arg\max_j I(vj, zk)$.

8.1.4 Foundation of Time Disentanglement Score (TDS)

Time series data often exhibit variations that may not always align with conventional metrics, especially when considering the presence or absence of underlying attributes. To address this challenge, we introduce the Time Disentanglement Score (TDS), a metric designed to assess the disentanglement of attributes in time series data. The foundation of TDS lies in an Information Gain perspective, which measures the reduction in entropy when an attribute is present compared to when it's absent.

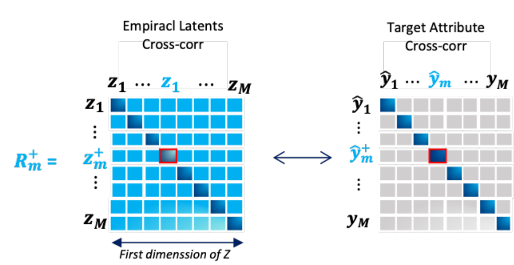

Figure 8: For disentangled presentation, the perturbation of factor m in $X_{t:t+\tau}$ affects Z_m and consequently the time domaine prediction y_m.

In the context of TDS, we augment factor m in a time series window $X_{t:t+\tau}$ with a specific objective: to maintain stable entropy when the factor is present and reduce entropy when it's absent. This augmentation aims to capture the essence of attribute-related information within the data.

TDS relies on the correlation matrix between $\mathbf{z}$ and $\mathbf{z}^+$, where $\mathbf{z} = f_\phi(\mathbf{x})$ and $\mathbf{z}^+ = f_\phi(\mathcal{T}(\mathbf{x}))$, with $\mathcal{T}$ denoting an augmentation function. This correlation matrix quantifies the consistency of attribute components. Additionally, TDS evaluates how well $\mathbf{z}$ contributes to the reconstruction of $\mathbf{y}$ and how $\mathbf{z}^+$ contributes to the reconstruction of $\mathcal{T}(\mathbf{y})$. Specifically, it assesses whether each z_m (or z_m^+) can effectively reconstruct the corresponding y_m (or y_m^+). TDS aligns with qualitative observations of disentanglement.

$$TDS = \frac{1}{2}\left[(1 - \sum_i Corr^{(I)}(\mathbf{z}, \mathbf{z}^+)_{ii})^2 + (1 - \sum_i Corr^{(I)}(\mathbf{y} - \hat{\mathbf{y}}, \mathbf{y}^+ - \hat{\mathbf{y}}^+)_{ii})\right]^2 \tag{8}$$

where $Corr_{ij}^{(I)} = \sum_b \mathbf{z}_{b,i}\mathbf{z}_{b,j}^+$ divided by $\sqrt{\sum_b(\mathbf{z}_{b,i})^2}\sqrt{\sum_b(\mathbf{z}_{b,j}^+)^2}$, b indexes batch samples and i, j index the vector dimension of the networks' f_ϕ outputs for $Corr^{(I)}$ (resp. dimension of the networks' outputs of g_θ for $Corr^{(II)}$). $Corr$ is a square matrix with the size of the dimensionality of the network's output and with values comprised between -1 (i.e. perfect anti-correlation) and 1 (i.e. perfect correlation). High TDS informativeness signifies strong disentanglement, while a significant distance implies reduced disentanglement and higher attribute correlation, aligning with [9].

8.2 Inference and Generative Procedure

To avoid time locality during dimension reduction, and keep long-range capability we refer to an in-depth Temporal Attention with l-Variational layers. Unlike NVAE [26] for which the latent space Z is level-structured locally, in this work, we enable the model to establish strong couplings, as depicted. The core problem we aim to address is to construct a feature $\hat{T}^l$ (Time context) that effectively captures the most informative features from a given sequence $T^{<l} = \{T^i\}_{i=1}^l$. Both $\hat{T}^l$ and T^l are features with the same dimensionality: $\hat{T}^l \in \mathbb{R}^{T \times C}$ and $T^i \in \mathbb{R}^{T \times C}$. In our model, we employ Temporal Attention to construct either the prior or posterior beliefs of variational layers, which enables us to handle long context sequences with large dimensions τ effectively. The construction of $\hat{T}^l$ relies on a query feature $\mathbf{Q}^l \in \mathbb{R}^{T \times Q}$ of dimensionality Q with $Q \ll C$, and the corresponding context T^l is represented by a key feature $\mathbf{K}^l \in \mathbb{R}^{T \times Q}$. Importantly, $\hat{T}^l(t)$ of time step i in sequence τ depends solely on the time instances in $T^{<l}$.

$$\hat{T}^l(t) = \sum_{i<l} \alpha_{i\to l}(t) \cdot T^l(t),\ \alpha_{i\to l}(t) = \frac{\exp(Q_l^\intercal(t) \cdot \mathbf{K}^l(t))}{\sum_{i<l}\exp(Q_l^\intercal(t) \cdot \mathbf{K}^l(t))} \tag{9}$$

In words, feature $\mathbf{Q}^l(t) \in \mathbb{R}^Q$ queries the Temporal significance of feature $T^l(t) \in \mathbb{R}^C$, represented by $\mathbf{K}^l(t) \in \mathbb{R}^Q$, to form $\hat{T}^l(t) \in \mathbb{R}^C$. $\alpha_{i\to l}(t) \in \mathbb{R}$ is the resulting relevance metric of the i-th term, with $i < l$, at time step t. The overall procedure is denoted as $\hat{T} = \mathbf{A}(T^{<l}, \mathbf{Q}^l, \mathbf{K}^{<l})$.

A powerful extension to the above single attention mechanism is the multi-head attention introduced in [**?**], which allows the model to jointly attend to information from different representation subspaces at different scales. Instead of computing a single attention function, this method first projects Q, K, V onto h different vectors, respectively. An attention function $\mathbf{A}(\cdot)$ is applied individually to these h projections. The output is a linear transformation of the concatenation of all attention outputs:

$$\text{Multi-}\mathbf{A}(Q, K, V) = \oplus\{\mathbf{A}(QW_{qi}, KW_{ki}, VW_{vi})\}_{i=1}^h W_o, \tag{10}$$

Where W_o, W_{qi}, W_{ki}, W_{vi} are learnable parameters of some linear layers. $QW_{qi} \in \mathbb{R}^{n_q \times d_{hq}}$, $KW_{ki} \in \mathbb{R}^{n_v \times d_{hk}}$, $VW_{vi} \in \mathbb{R}^{n_v \times d_{hv}}$ are vectors projected from Q, K, V respectively. $d_{hq} = \frac{d_q}{h}$ and $d_{hv} = \frac{d_v}{h}$. Following the architecture of the transformer [**?**], we define the following multi-head attention block:

$$Q_0 = \text{LayerNorm}(\oplus\{QW_{q1}\}_{i=1}^h + \text{MultiAtt}(Q, K, V)), \tag{11}$$

$$\text{MultiBloc-}\mathbf{A}(Q,K,V) = \text{LayerNorm}(Q_0 + Q_0 W_{q0}), \tag{12}$$

where $W_{q0} \in \mathbb{R}^{d_q \times d_q}$ is a learnable linear layer.

Decodeur (Generative Model p_θ). The conditioning factor of the prior distribution at variational layer l is represented by context feature $T_p^l \in \mathbb{R}^{T\times C}$. A convolution is applied on T_p^l to obtain parameters θ defining the prior. $\mathbf{Res}_p^l$ is a non-linear transformation of the immediately previous latent information Z_l and prior context T_p^l containing latent information from distant layers $\mathbf{z}_{<l}^l$, such that $T_p^l = \mathbf{Res}_p^l(Z_l \oplus T_p^l)$. $\mathbf{Res}_p^l(\cdot)$ is a transformation operation, typically implemented as a cascade of residual cells and corresponds to the blue residual module in Fig 3. Z_l and T_p^l are passed in from the previous layer. Because of the architecture's locality, the influence of Z_l could potentially overshadow the signal coming from T_p^l. To prevent this, we adopt direct connections between each pair of stochastic layers. That is, variational layer l has direct access to the prior temporal context of all previous layers $T_p^{<l}$ accompanied by keys $\mathbf{K}_p^{<l}$. This means each variational layer can actively determine the most important latent contexts when evaluating its prior. During training, the temporal context T_p, $\mathbf{Q}_p$, and $\mathbf{K}_p$ are jointly learned:

$$[T_p^l, \mathbf{Q}_p^l, \mathbf{K}_p^l] \leftarrow \mathbf{Res}_p^l(Z_l \oplus (T_p^l + \eta_p^l \mathbf{A}(T_p^{<l}, \mathbf{Q}_p^l, \mathbf{K}_p^{<l}))) \text{ for } l = L, L-1, ..., 1. \tag{13}$$

Where $\eta_p^l \in \mathbb{R}$ is a learnable scalar parameter initialized by zero, $T_p^{<l} = \{T_p^i\}_{i=1}^l$ with $T_p^i \in \mathbb{R}^{T\times C}$, $\mathbf{Q}_p^l \in \mathbb{R}^{T\times Q}$, $\mathbf{K}_p^{<l} = \{\mathbf{K}_p^i\}_{i=1}^l$ with $\mathbf{K}_p^i \in \mathbb{R}^{\times Q}$, and $Q \ll C$. We initially let variational layer l rely on nearby dependencies captured by T_p^l. During training, the prior is progressively updated with the holistic context $\hat{T}_p^l$ via a residual connection.

Encodeur (Infrence Model q_ϕ) As shown in Fig 2, the conditioning context T_q^l of the posterior distribution results from combining deterministic factor h^l and stochastic factor T_p^l provided by the decoder: $T_q^l = h^l \oplus T_p^l$. To improve inference, we let layer l's encoder use both its own h^l and all subsequent hidden representations $h^{\geq l}$, as shown in Fig 2. As in the generative model, the bottom-up path is extended to emit low-dimensional key features $\mathbf{K}_q^l$, which represent hidden features h^l:

$$[h^l, \mathbf{K}_q^l] \leftarrow \mathbf{T}_q^l(h_{l+1} \oplus \mathbf{K}_q^{l+1}) \text{ for } l = L, L-1, ..., 1.$$

Prior works [26] have sought to mitigate against exploding Kullback-Leibler divergence (KL) in Eq 2 by using parametric coordination between the prior and posterior distributions. Motivated by this insight, we seek to establish further communication between them. We accomplish this by allowing the generative model to choose the most explanatory features in $h^{\geq l}$ by generating the query feature $\mathbf{Q}_q^l$. Finally, the holistic conditioning factor for the posterior is:

$$\hat{T}_q^l \leftarrow \mathbf{A}(h^{\geq l}, \mathbf{Q}_q^l, \mathbf{K}_q^{\geq l}) \text{ for } l = L, L-1, ..., 1. \tag{14}$$

We adopt the Gaussian residual parametrization between the prior and the posterior. The prior is given by $p(\mathbf{z}_l|\mathbf{z}_{<l}) = \mathcal{N}(\mu(T_p^l, \theta), \sigma(T_p^l, \theta)$. The posterior is then given by $q(\mathbf{z}_l|\mathbf{x}, \mathbf{z}_{<l}) = \mathcal{N}(\mu(T_p^l, \theta) + \boldsymbol{\Delta}\mu(\hat{T}_q^l, \phi), \sigma(T_p^l, \theta) \cdot \boldsymbol{\Delta}\sigma(\hat{T}_q^l, \phi))$ where the sum $(+)$ and product $(\cdot)$ are pointwise, and T_q^l is defined in Eq 14. $\mu(\cdot)$, $\sigma(\cdot)$, $\boldsymbol{\Delta}\mu(\cdot)$, and $\boldsymbol{\Delta}\sigma(\cdot)$ are transformations implemented as convolutions layers. Based on this, For $\mathcal{L}_{KL}$ in Eq 2, the last term is approximated by: $0.5 \times \left(\frac{\boldsymbol{\Delta}\mu_1^2}{\sigma_l^2} + \boldsymbol{\Delta}\sigma_l^2 - \log \boldsymbol{\Delta}\sigma_l^2 - 1\right)$.

8.3 Hyperparameter and Training

Table 5 presents a comparison of the computational requirements for training different VAE models, including NVAE (Normal VAE), and DisCoV on the Uk-dale dataset.

The table shows the batch size per GPU, the number of GPUs utilized for training, and the corresponding training time in hours for each model. The batch size for all models is set to 128, and four GPUs are used in parallel for training in each case.

As observed from the table, the DisCoV model exhibits longer training times compared to NVAE. This indicates that the additional computational cost associated with computing attention scores in

Table 5: We compare the computational requirements for training DisCoV and NVAE models on the Uk-dale dataset. The training is performed using Nvidia A100 GPUs, each equipped with 80GB of memory.

Model	Batch/GPU	# GPUs	Time (hour)
NVAE	128	4	68
DisCoV	128	4	84

the DisCoV model is offset by the benefits of having a smaller number of stochastic layers in the hierarchical architecture without compromising the generative capacity of the models.

This information provides valuable insights into the computational efficiency and trade-offs among these state-of-the-art VAE models when applied to the Uk-dale dataset.

8.4 Impact of window parameter τ

To perform Non-Intrusive Load Monitoring (NILM) effectively, it is crucial to select an appropriate window time series. This involves determining a time interval for analyzing energy consumption data that allows for the detection and classification of individual appliance activities. The chosen window should strike a balance between being long enough to capture complete appliance activity cycles and short enough to avoid overlaps with other activities or periods of inactivity. The optimal window size depends on factors such as the energy meter's sampling rate, the number and types of appliances being monitored, and the specific NILM algorithm employed. Experimentation and optimization may be necessary to identify the ideal window size for a specific NILM application. In our study, we tried to detect the consumption of the washing machine, which averages 3 to 4 hours of use per cycle. Therefore, we chose a window of 4h30, equivalent to 256-time steps of 60 seconds. In addition, we've noticed that a window of 128 and 300 steps doesn't detect the washing machine.

8.5 Optimization

In all of our experiments, we used the Adam optimizer with an initial learning rate of 10^{-3} and a cosine decay of the learning rate. We also reduced the learning rate to 7×10^{-4} to increase the stability of the training and applied an early stop after 5 iterations. We set $\alpha = 0.5$ and $\beta = 2.5$ after a grid search on the best convergence of the model on the validation data.

9 Extended Ablation Studies

9.1 Empirical Evidence of Enhanced Latent using self-attentive l-VAE

Depth (L)	bits/dim ↓	$\Delta()\%$
4	3.12	-8.7
8	2.96	-8.1
16	3.81	-10.1
32	5.12	-13.7

Table 6: Negative log-likelihood per dimension (bits/dim) for varying depth L for the attentive DisCO.

10 Explicability underlying latent space structuring

An interpretable representation of learning is obtained when the latent space is factorized and the multidimensional components are statistically independent, which is a complex task in the context of information theory for generative models. A variety of methods have been proposed to solve this problem, such as β-TCVAE [**?**]. The most commonly used method is derived from the information theory known as *Total Correlation*, which introduces the TC penalty that is defined by the divergence $\mathrm{KL}(p_\phi(Z)|| \prod_j p_\phi(z_j))$. Nevertheless, estimating this divergence is both expensive and difficult to perform.

Estimation of TC. To avoid costly *TC* estimation and guarantee time-series robustness, we try to apply this penalty using a discriminator across Z. It has been previously used as a disentangling metric for image generation [**?**]. In our case, we use it as a loss function. For its training, the latent variables of half the batch are randomly permuted, creating positive z_{perm} *(i.e all components are independent)*, and the other half is left untouched, corresponding to negative case *(i.e components are correlated)*. A D_ψ discriminator is used to replace the penalty, denoted TC in the following, by optimizing the performance of a discriminator between the distribution of the latent variable and a permuted of it. The D_ψ discriminator and the model are trained simultaneously.

$$\mathcal{L}_{TC} = \mathbb{E}[\log(D_\psi(Z_{\text{permuted}}))] + \mathbb{E}[\log(1 - D_\psi(Z))] \tag{15}$$

The overarching training objective for the sequence-to-sequence model, incorporating residual KL in each layer $l = L, L-1, \ldots 1$ as discussed in our proposed method above (Section 6), can be summarized as follows:

$$\mathcal{L}(\gamma, \beta, \delta; \theta, \phi, \psi) = \mathcal{L}_{rec} + \beta\, \mathcal{L}_{\text{KL}} + \gamma\, \mathcal{L}_{TC} \tag{16}$$

Here, we have a hyperparameter β_{KL} to balance the reconstruction loss and KL losses and γ to balance the disentangling effect of TC.

11 More Quantitative Comparison

11.1 Case where $M = 7$ and $K = 3$

11.1.1 Elaboration on Metrics for the Downstream Task (Reconstruction of Appliance Powers)

Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Squared Error (MSE), and Mean Absolute Percentage Error (MAPE) are adopted to evaluate the imputation accuracy of all compared methods. These four metrics are defined as:

$$\text{RMSE} = \sqrt{\frac{\sum_{ij\in\Omega}(X_{ij} - \hat{X}_{ij})^2}{|\Omega|}}, \tag{17}$$

$$\text{MAE} = \frac{\sum_{ij\in\Omega} |X_{ij} - \hat{X}_{ij}|}{|\Omega|}, \tag{18}$$

$$\text{MSE} = \frac{\sum_{ij\in\Omega}(X_{ij} - \hat{X}_{ij})^2}{|\Omega|}, \tag{19}$$

$$\text{MAPE} = \frac{\sum_{ij\in\Omega} |X_{ij} - \hat{X}_{ij}|}{|\Omega| \cdot |X_{ij}|}, \tag{20}$$

where X_{ij} denotes the ground-truth values, $\hat{X}_{ij}$ is the imputed values, and Ω is the index set of missing entries to be evaluated.