A Hierarchical Taxonomy For Deep State Space Models

Shiqin Tang
Department of Data Science
City University of Hong Kong
Hong Kong, Country
t.sq@my.cityu.edu.hk

Pengxing Feng
Department of Electrical Engineering
City University of Hong Kong
Hong Kong
pengxfeng2-c@my.cityu.edu.hk

Shujian Yu
Department of Artificial Intelligence
Vrije Universiteit Amsterdam
Amsterdam, Netherlands
s.yu3@vu.nl

Yining Dong
Department of Data Science
City University of Hong Kong
Hong Kong
yining.dong@cityu.edu.hk

S. Joe Qin
Department of Computing and Decision Science
Lingnan University
Hong Kong
joeqin@lin.edu.hk

***Abstract*—Modeling nonlinear dynamical systems is a challenging task in fields such as speech processing, music generation, and video prediction. This paper introduces a hierarchical framework for Deep State Space Models (DSSMs), categorizing them by their conditional independence properties and Markov assumptions and positioning existing models within this framework, including the Stochastic Recurrent Neural Network (SRNN), Variational Recurrent Neural Network (VRNN), and Recurrent State Space Model (RSSM). We discuss different options for the inference networks and demonstrate how integrating normalizing flows can enhance model flexibility by capturing complex distributions. Our work not only clarifies the relationships among existing models but also paves the way for the development of new, more effective approaches for modeling nonlinear dynamics. In particular, we propose the Autoregressive State Space Model (ArSSM) and evaluate its effectiveness in speech and polyphonic music modeling tasks.**

***Index Terms*—Hierarchical Taxonomy, Dynamical Variational Autoencoders, Deep State Space Models, Normalizing Flows**

I. Introduction

Modeling nonlinear dynamical systems is a crucial challenge in fields such as speech processing, music generation, and video prediction. Dynamic Variational Autoencoders (DVAEs) are a class of methods that use dynamic latent states to capture the inherent randomness in sequential data. Despite the existence of surveys and reviews in this field (e.g. [1]), they often describe models in isolation, making it difficult to compare them and understand their interrelationships. This paper presents a hierarchical framework for Deep State Space Models (DSSMs), a subset of DVAEs where latent states evolve over time, categorizing them based on their conditional independence properties and Markov assumptions. By positioning existing models within this framework, we also identify new models with distinct sets of properties.

Recent advancements in Dynamic Variational Autoencoders have significantly enhanced the modeling of sequential data by incorporating temporal dependencies into the generative process. These advancements can be seen across several tracks. The Stochastic Video Generation (SVG) model [2] is among the earliest approaches to apply DVAEs in the task of video prediction, enabling the generation of plausible future frames given a handful of conditional frames as a starting point. Following SVG, the Hierarchical Variational RNN (VRNN) [3] improves video prediction quality by using ConvLSTM [4] as a backbone for generation and inference and introducing hierarchical latent variables, allowing the model to capture both high-level and fine-grained temporal dependencies. More recently, the Greedy Hierarchical Variational Autoencoders (GHVAE) [5] have provided an optimization scheme tailored for deep hierarchical DVAEs, making them suitable for large-scale video prediction tasks. DVAEs also have a significant impact on speech and audio processing. First proposed by the Stochastic Recurrent Network (STORN) [6], the idea of using RNN hidden states to store information from previous latent and observed variables has inspired the invention of various DVAE variants such as the Variational RNN (VRNN) [7], Stochastic RNN (SRNN) [8], and Recurrent VAE (RVAE) [9–11], which have successful applications in speech synthesis, and audio enhancement.

Incorporating stochastic latent states offers several advantages over purely deterministic approaches like RNNs or LSTMs. The inclusion of latent states facilitates a disentangled representation of observed data, thereby enhancing the model's generalization ability and robustness to changes [12, 13]. Models with latent variables can generate more diverse samples, which is essential for applications that require variability, such as music and speech generation. Furthermore, latent variable models are better equipped to handle uncertainty, as demonstrated by the VRNN's ability to generate consistent handwriting samples [7] and the SVG-LP's capability to predict uncertain events in dynamic environments [2].

The motivation behind proposing this framework of deep state-space models is to highlight that no single model outperforms the rest across all scenarios, and selecting the appropriate model depends on its alignment with the true underlying assumptions of the data and the objective of the task. As shown in Fig. 1, the hierarchical framework we propose classifies models into four groups based on the presence or absence of specific connections. With the Hidden Markov Model (HMM) as the base case, we consider the addition of an autoregressive connection among the observations, a feedforward connection from previous observations to the current latent state, or both; these modifications result in the base models referred to as Autoregressive HMM (AR-HMM), Feedforward HMM (F-HMM), and Augmented Deep Markov Model (DMM-Aug) [14]. Models characterized by AR-HMM are well-suited when the observations exhibit strong temporal dependencies but do not causally influence the state transitions. For instance, AR-HMMs are effectively used in financial time series analysis to uncover shifts in economic conditions based on observations such as asset prices or treasury bill rates [15] (specifically, the model proposed in [15] is a linear Gaussian version of the model in Fig. 1 (d) with finite autoregressive orders). Conversely, models with feedforward connections capture the causal influence of past observations on the current latent state, making them suitable for tasks where leveraging these dependencies can improve performance [14, 16].

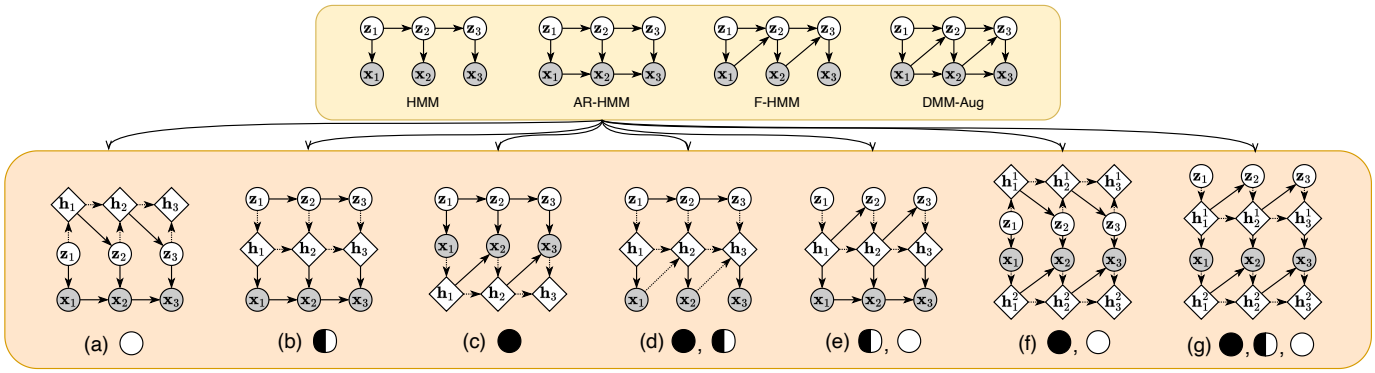


Fig. 1: A hierarchical framework for DSSMs. The models are first categorized into four groups, each defined by a base model, as shown in the first row, according to their conditional independence properties. Each base model then has multiple variations based on different relaxations of its Markov properties. This figure highlights the group of DSSMs characterized by the AR-HMM. The shaded nodes represent observations, while the unshaded nodes represent latent variables. The diamond-shaped nodes denote the RNN hidden states. The solid arrows denote stochastic mappings, while the dotted arrows denote deterministic transformations. Each individual caption includes markers reflecting the respective Markov assumptions. Sub-figure (g) describes the proposed Autoregressive State-Space Model (ArSMM).

This paper is organized as follows: Section II-A details the structures of the DSSMs in the generative process. In Section II-B, we discuss different options for the networks used in approximate inference. Section II-C demonstrates how normalizing flows can be integrated into our framework to add expressiveness to the posterior approximations and improve the tightness of the variational bound. Experimental results are presented in Section III.

The key contributions of this paper include: 1) the introduction of a hierarchical taxonomy for DSSMs, which facilitates the understanding of their interrelationships, 2) a discussion on various types of inference networks, 3) the seamless integration of normalizing flows, and 4) the proposal of the Autoregressive State-Space Model (ArSSM) that performs favorably compared to existing methods.

II. Deep State Space Models

Before diving into the DSSMs, we introduce some notations. Let $\mathbf{z}_t$ and $\mathbf{x}_t$ denote the latent and observed state at time step t, for $t = 1, 2, \ldots, T$. Let $\mathbf{h}_t$ and $\mathbf{g}_t$ denote the deterministic RNN hidden states in the generative and inference processes respectively.

A. Generative Process

As shown in Fig. 1, we divide the deep state-space models (DSSMs) into four groups, each characterized by a base model with distinct sets of conditional independence properties. Viewing HMM as a baseline, AR-HMM applies an autoregressive connection between $\mathbf{x}_{t-1}$ and $\mathbf{x}_t$, F-HMM adds a feedforward connection between $\mathbf{x}_{t-1}$ and $\mathbf{z}_t$, and DMM-Aug utilizes both connections.

The models within each group have different Markov assumptions. Given all previous observations $\mathbf{x}_{1:t-1}$, we consider four types of interactions among the latent and observed variables: $\mathbf{z}_t$ to $\mathbf{z}_{1:t-1}$, $\mathbf{x}_t$ to $\mathbf{x}_{1:t-1}$, $\mathbf{x}_t$ to $\mathbf{z}_{1:t}$, and $\mathbf{z}_t$ to $\mathbf{z}_{1:t-1}$. We introduce the markers ○, ●, ◐, and ◑ to denote respectively the conditions where the corresponding first order Markov assumption is being relaxed, i.e.

- ○: $p(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{c}) \rightarrow p(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{c})$
- ●: $p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{c}) \rightarrow p(\mathbf{x}_t|\mathbf{x}_{1:t-1}, \mathbf{c})$
- ◐: $p(\mathbf{x}_t|\mathbf{z}_t, \mathbf{c}) \rightarrow p(\mathbf{x}_t|\mathbf{z}_{1:t}, \mathbf{c})$
- ◑: $p(\mathbf{z}_t|\mathbf{x}_{t-1}, \mathbf{c}) \rightarrow p(\mathbf{z}_t|\mathbf{x}_{1:t-1}, \mathbf{c})$

where $\mathbf{c}$ denotes the previous states the distribution is conditioned on. Note that each model is not confined to have just one Markov property being relaxed. Inspired by [6, 17], each combination of relaxed Markov properties can be implemented with the help of RNN hidden states.

For demonstration, we examine several models corresponding to different base models. Recurrent State-Space Models (RSSMs) can be regarded as the representative of the group featured by HMM as it relaxes both types of first-order Markov assumptions that exist in HMM, i.e. ○ and ◐, and it has various applications in robotics and control [18]. Following the same methology, we propose the Feedforward SSMs (FSSMs) as the representative of the F-HMM group. It is worth noting that the model equipped with both types of connections and in which all four Markov assumptions are relaxed is known as the Variational RNNs (VRNNs) [7], which is the representative of the DMM-Aug group. The Stochastic RNNs (SRNNs), although belonging to the same group as VRNN, do not model the long term temporal dependencies among the latent states. Due to space limitations, we move the graphical representations of the remaining DSSMs to Appendix A, and the models related to our framework are discussed in Appendix E.

The Autoregressive State-Space Model (ArSSM) as shown in Fig. 1 (g) can be viewed as a representative of the group characterized by AR-HMM because it has all three types of Markov assumptions relaxed. The state transition and emission rules of ArSSM are given by

$$p(\mathbf{z}_t, \mathbf{x}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:t-1}) = p(\mathbf{z}_t|\mathbf{z}_{1:t-1})p(\mathbf{x}_t|\mathbf{z}_{1:t}). \tag{1}$$

ArSSM requires two RNN chains in its generative structure to implement the assumption that its current latent state can depend on the previous latent states but not the previous observations. Given the previous RNN hidden states, $\mathbf{h}^1_{t-1}$ and $\mathbf{h}^2_{t-1}$, the current states are updated as:

$$\mathbf{z}_t \sim p(\mathbf{z}_t|d_z(\mathbf{h}^1_{t-1})), \qquad \mathbf{h}^1_t = d_{h^1}(\mathbf{z}_t, \mathbf{h}^1_{t-1}), \tag{2}$$

$$\mathbf{x}_t \sim p(\mathbf{x}_t|d_x(\mathbf{h}^1_t, \mathbf{h}^2_{t-1})), \qquad \mathbf{h}^2_t = d_{h^2}(\mathbf{x}_t, \mathbf{h}^2_{t-1}), \tag{3}$$

where d_z and d_x are nonlinear functions, and d_{h^1} and d_{h^2} are generic RNN cells. The outputs of d_z and d_x parameterize the distribution of $\mathbf{z}_t$ and $\mathbf{x}_t$ respectively. If $\mathbf{z}_t$ and $\mathbf{x}_t$ are assumed to satisfy spherical Gaussian distributions, it is a common practice to make d_z and d_x dual-head multilayer perceptrons (MLPs) to allow parameter sharing. One can verify by recursively applying (2) and (3) that we are indeed modeling the relationship specified in (1).

B. Posterior Approximation & Training

Since the exact posterior distribution of the latent states given the observations is intractable to find, we consider two types of posterior approximation for each DSSM:

1) **Partial Alignment**: Based on the exact posterior factorization of the latent states, if $\mathbf{z}_t \perp\!\!\!\perp \mathbf{x}_{1:t-1} | \mathbf{z}_{t-1}$, we approximate the posterior distribution as

$$q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_{t:T}). \tag{4}$$

Otherwise, if $\mathbf{z}_t \not\perp\!\!\!\perp \mathbf{x}_{1:t-1} | \mathbf{z}_{t-1}$, we instead model

$$q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_{1:T}). \tag{5}$$

Inspired by [14, 19], the former case requires a backward RNN on $\mathbf{x}_{1:T}$, while the latter requires a bidirectional RNN.

2) **Full Alignment**: The partial alignment approach, albeit easy to implement, fails to consider the temporal dependency among the latent states. The full alignment approach aims to mimic the exact posterior factorization. Inspired by [7], the dependence of $\mathbf{z}_t$ on $\mathbf{z}_{1:t-1}$ is achieved by reusing the RNN structure in the generative process. We also implement new RNNs to store historical information of the latent states sampled in the inference process, but such models show no apparent improvement, as shown in Appendix D.

The exact posterior factorization for ArSSM is given by

$$q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T}). \tag{6}$$

As a result, the posterior approximations in partial and full alignment styles are given, respectively, by

$$q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1}, \overrightarrow{\mathbf{g}}_t^x, \overleftarrow{\mathbf{g}}_t^x), \tag{7}$$

$$q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{h}_{t-1}^1, \mathbf{h}_t^2, \overleftarrow{\mathbf{g}}_t^x), \tag{8}$$

where $\overrightarrow{\mathbf{g}}_t^x$ and $\overleftarrow{\mathbf{g}}_t^x$ denote the t-th hidden states of the forward and backward RNN on $\mathbf{x}_{1:T}$ respectively. For conditional sampling of $\mathbf{z}_t$, we use a structure similar to the combiner function in [14].

The training objective for the ArSSM, with the posterior approximation specified in (8), is to maximize the evidence lower bound (ELBO):

$$\begin{aligned}
&\mathbb{E}_{q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T})}\Big[\log \frac{p(\mathbf{z}_{1:T}, \mathbf{x}_{1:T})}{q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T})}\Big] \\
&= \sum_{t=1}^{T} \mathbb{E}_{q(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T})}[\log p(\mathbf{x}_t|\mathbf{z}_{1:t}, \mathbf{x}_{1:t-1})] \\
&\quad - \sum_{t=1}^{T} \mathbb{E}_{q(\mathbf{z}_{t-1}|\mathbf{z}_{1:t-2}, \mathbf{x}_{1:T})}[D_{\mathrm{KL}}(q(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T}) \| p(\mathbf{z}_t|\mathbf{z}_{1:t-1}))]
\end{aligned} \tag{9}$$

with respect to the parameters used in both the generative and inference processes. By making the approximate posterior more consistent in form with the exact posterior factorization, the full alignment approach should result in a tighter lower bound during training [1]. Since the parameters of the density functions involved in the expectation in (9) also need to be optimized, we apply the reparameterization trick when sampling from the approximate posterior [20].

C. Normalizing Flows

Initially proposed for density estimation, normalizing flows have gradually gained popularity in modeling distributions within latent variable models [21]. For instance, inverse autoregressive flows (IAFs) are used as approximate posterior distributions in VAEs due to their improved expressiveness and fast sampling capabilities [22]. Later, [23] points out that using autoregressive flows (AFs) for modeling the prior distribution is equivalent to using IAFs to model the corresponding posterior distribution. Beyond static settings, normalizing flows have also been applied in sequential modeling. For example, normalizing flows are used to augment the emission model of the Kalman filter [24], and autoregressive flows are employed to model the prior or the generative process in VRNNs [25].

In the previous section, the approximate posterior distributions, albeit parameterized by deep neural nets, are confined to spherical Gaussian forms, which limits their expressiveness to some extent. In this paper, we propose to use IAFs to enhance the inference process using the posterior approximations in Section II-B as base distributions.

We use ArSSM in full alignment approach as a demonstration (the generative and inference models are given in (1) and (8)). Let $\mathbf{z}_t$ be of dimensions d. Let $f_\phi : \mathbb{R}^d \to \mathbb{R}^d$ be an IAF parameterized by ϕ, and let g_ϕ be its inverse operation. At time step t, we sample from base distribution $\mathbf{u}_t \sim q(\mathbf{u}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T})$, and use it as input for the flow f_ϕ to compute $\mathbf{z}_t$, as shown below:

$$\mathbf{u}_t \sim q(\mathbf{u}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T}) \xrightarrow{f_\phi} \mathbf{z}_t \sim q_\phi(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T}). \tag{10}$$

The approximate posterior factorization with IAFs is given by

$$q_\phi(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q_\phi(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T}), \tag{11}$$

where

$$q_\phi(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T}) = q(g_\phi(\mathbf{z}_t)|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:T}) \cdot |\det J(g_\phi)(\mathbf{z}_t)|, \tag{12}$$

where $J(g_\phi)(\mathbf{z}_t) = \mathrm{d}g_\phi(\mathbf{z}_t)/\mathrm{d}\mathbf{z}_t$ denotes the Jacobian. The parameters of the IAF can be jointly optimized along with the parameters from the generative and inference networks by maiximizing the updated ELBO:

$$\mathbb{E}_{q_\phi(\mathbf{z}_{1:T}|\mathbf{x}_{1:T})}\Big[\log \frac{p(\mathbf{x}_{1:T}, \mathbf{z}_{1:T})}{q_\phi(\mathbf{z}_{1:T}|\mathbf{x}_{1:T})}\Big]. \tag{13}$$

Every DSSM can augment its approximate posterior following this manner with minute adjustments.

III. Experiments & Results

In this section, we test the proposed ArSSM against existing methods like VRNN, SRNN, and RSSM, in the speech and polyphonic music modeling tasks. Specifically, we conduct an analysis-resynthesis task in teacher-forcing mode [1], where latent states are sampled from the inference network and the quality of the reconstructed data is assessed. This task is commonly used in speech modeling to evaluate a model's capacity to capture the underlying data distribution and accurately reconstruct the input.

For speech modeling we use a subset of LibriSpeech [26], which consists of audiobooks read aloud from the LibriVox project. From the collected data, 90% is used for training, while the remaining 10% is reserved for testing. The Polyphonic Music Generation dataset is a widely used benchmark [8, 14], consisting of four sets of piano music represented as 88-dimensional binary time series: folk tunes

	SI-SDR (dB) ↑	PESQ ↑	ESTOI ↑
Without IAFs			
ArSSM	**5.42**/5.132	**1.69**/**1.675**	**0.785**/**0.795**
SRNN	4.99/**5.146**	1.60/1.667	0.783/0.788
FSSM	0.367/1.46	1.17/1.23	0.634/0.691
VRNN	3.32/3.23	1.25/1.19	0.687/0.664
RSSM	0.633/0.787	1.18/1.19	0.654/0.663
Using IAFs			
ArSSM	5.24/**5.56**	1.65/**1.70**	0.791/**0.802**
SRNN	**5.73**/4.31	**1.70**/1.42	**0.795**/0.770
FSSM	1.44/1.55	1.23/1.25	0.688/0.706
VRNN	2.09/2.80	1.13/1.16	0.601/0.623
RSSM	1.30/1.47	1.20/1.20	0.665/0.670

TABLE I: Test SI-SDR, PESQ, and ESTOI scores for different models with and without IAFs. Each entry consists scores for the model in partial/full alignment (in that order). Bolded numbers highlight the best scores in each category under partial and full alignment.

(Nottingham), orchestral music (Musedata), classical piano music (Piano-midi), and Bach chorales (JSB). The models are trained using the Adam optimizer [27] with a norm clipping of 10 and minibatches of size 100. Additionally, we apply the KL annealing technique during training to improve convergence; specifically, we set the initial weight of the regularization factor (the second term of (9)) to a small positive value and gradually increase it to one over the course of 100 epochs. We implement the above models using Pyro [28], a probabilistic programming language. We direct our readers to Appendix D for additional details in implementation and experiments. The code for replicating the experiments in this section can be found at https://github.com/marcusstang/DSSMs.

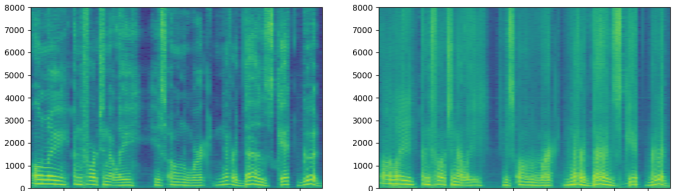
Speech Modeling. We evaluate the quality of the resynthesized speech using three metrics across all methods: the scale-invariant signal-to-distortion ratio (SI-SDR) in dB, the perceptual evaluation of speech quality (PESQ) score, and the extended short-time objective intelligibility (ESTOI) score. Typically, SI-SDR values are above 0 dB. PESQ ranges from -0.5 to 4.5, and ESTOI ranges from 0 to 1. In all cases, higher SI-SDR, PESQ, and ESTOI scores indicate better resynthesized speech quality.

Table I summarizes the performance of the tested models across different configurations. First, consider the models without IAFs. As seen, ArSSM performs favorably compared to its peers in nearly every metric, except for a slight underperformance in SI-SDR under the full alignment configuration, where SRNN achieves a higher score. While switching to full alignment improves the performance of FSSM, this improvement is not always guaranteed for all models. One possible reason is that a posterior approximation more consistent in form with the exact posterior factorization may lead to a tighter variational bound but doesn't necessarily translate to better performance, as discussed in [1, 14]. FSSM and RSSM show lower overall performance, due to the absence of autoregressive connections among the observed variables, limiting their ability to capture intricate temporal dependencies in speech data.

Next, we examine the models with the IAF posterior, where three affine autoregressive layers of dimension 20 are employed. Incorporating IAFs significantly boosts the performance of SRNN, which now surpasses other models in SI-SDR under partial alignment. The integration of IAFs also greatly enhances the performance of ArSSM in the full alignment setting, making it the top performer across all three metrics.

Models	**Nottingham** ↓	**Musedata** ↓	**Piano-midi** ↓	**JSB** ↓
ArSSM PA	(3.07)	(6.44)	(7.79)	(7.46)
ArSSM FA	(2.87)	(6.34)	(7.76)	(7.17)
SRNN	(2.94)	(6.28)	(8.20)	(4.74)
DMM	2.77 (2.96)	6.83 (6.99)	7.84 (7.98)	6.39 (6.93)
DMM-Aug	2.68 (2.86)	6.36 (6.48)	7.59 (7.72)	6.29 (6.77)
HMSBN	5.24	9.74	9.56	8.05
STORN	2.85	6.16	7.13	6.91
RNN	4.46	8.13	8.37	8.71
TSBN	(3.67)	(6.81)	(7.98)	(7.48)

TABLE II: Test negative log likelihood (NLL) and its upper bound (Negative ELBO) on Polyphonic Music Generation dataset. Each entry is in the format "NLL (ELBO)". "PA" and "FA" stand for partial and full alignment respectively. The results for models other than ArSSM are from [8, 14].



(a) Original Spectrogram (b) ArSSM in Full Alignment

Fig. 2: Mel spectrograms for an audio sample and its reconstructed forms using ArSSM in full alignment.

The mel spectrograms generated by the proposed ArSSM are shown in Fig. 2. Comparing the power spectra of the original speech with the reconstructed one, the reconstructed speech looks like a smoothed or blurred version of the original speech. The proposed models can predict the position of the speech at high frequency. The details of the speech at high frequency are partially lost. Nevertheless, the proposed methods recover the power spectrum of speech signals well in the mid-low frequency band, which is the main frequency band for speech.

Polyphonic Music Modeling. We report a conservative upper bound on the negative log likelihood (NLL) as the evaluation metric for ArSSM, since log likelihood estimates obtained via importance sampling may not be reliable, as noted in [8]. As shown in Table II, ArSSM performs comparably to other models across the four datasets. In the JSB dataset, ArSSM slightly underperforms its peers, which we attribute to the scarcity, short length, and low complexity of the training data. These factors reduce the need for modeling long-term temporal dependencies in the latent states, an area where ArSSM typically excels.

In all cases, switching to full alignment improves the performance of ArSSM. This is because the inference network in the full alignment approach is more consistent with the true posterior factorization, leading to tighter variational bounds. However, this may not hold for the speech modeling task, where the training objective differs from the evaluation metrics.

IV. Discussion

This work presented a hierarchical taxonomy for deep state space models (DSSMs), categorizing models based on their properties. Our experiments in speech modeling show that the proposed ArSSM consistently outperforms existing comparable methods. Given that multiple graphical models can share the same conditional independence structure, future work could examine the impact of these variations on model efficiency, as well as extend applications to areas like object tracking and video generation.

REFERENCES

[1] L. Girin, S. Leglaive, X. Bie, J. Diard, T. Hueber, and X. Alameda-Pineda, "Dynamical variational autoencoders: A comprehensive review," *arXiv preprint arXiv:2008.12595*, 2020.

[2] E. Denton and R. Fergus, "Stochastic video generation with a learned prior," in *International conference on machine learning*. PMLR, 2018, pp. 1174–1183.

[3] L. Castrejon, N. Ballas, and A. Courville, "Improved conditional vrnns for video prediction," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 7608–7617.

[4] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in neural information processing systems*, vol. 28, 2015.

[5] B. Wu, S. Nair, R. Martin-Martin, L. Fei-Fei, and C. Finn, "Greedy hierarchical variational autoencoders for large-scale video prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2318–2328.

[6] J. Bayer and C. Osendorfer, "Learning stochastic recurrent networks," *arXiv preprint arXiv:1411.7610*, 2014.

[7] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," *Advances in neural information processing systems*, vol. 28, 2015.

[8] M. Fraccaro, S. K. Sønderby, U. Paquet, and O. Winther, "Sequential neural models with stochastic layers," *Advances in neural information processing systems*, vol. 29, 2016.

[9] S. Leglaive, X. Alameda-Pineda, L. Girin, and R. Horaud, "A recurrent variational autoencoder for speech enhancement," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 371–375.

[10] M. Sadeghi and R. Serizel, "Fast and efficient speech enhancement with variational autoencoders," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[11] P. Wang and X. Li, "Rvae-em: Generative speech dereverberation based on recurrent variational auto-encoder and convolutive transfer function," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 496–500.

[12] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[13] A. Dittadi, "On the generalization of learned structured representations," *arXiv preprint arXiv:2304.13001*, 2023.

[14] R. Krishnan, U. Shalit, and D. Sontag, "Structured inference networks for nonlinear state space models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[15] J. D. Hamilton, "Analysis of time series subject to changes in regime," *Journal of econometrics*, vol. 45, no. 1-2, pp. 39–70, 1990.

[16] M. C. Ngô, M. P. Heide-Jørgensen, and S. Ditlevsen, "Understanding narwhal diving behaviour using hidden markov models with dependent state distributions and long range dependence," *PLoS computational biology*, vol. 15, no. 3, p. e1006425, 2019.

[17] K. P. Murphy, *Probabilistic machine learning: Advanced topics*. MIT press, 2023.

[18] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," in *International conference on machine learning*. PMLR, 2019, pp. 2555–2565.

[19] A. G. ALIAS PARTH GOYAL, A. Sordoni, M.-A. Côté, N. R. Ke, and Y. Bengio, "Z-forcing: Training stochastic recurrent networks," *Advances in neural information processing systems*, vol. 30, 2017.

[20] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[21] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *International conference on machine learning*. PMLR, 2015, pp. 1530–1538.

[22] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," *Advances in neural information processing systems*, vol. 29, 2016.

[23] X. Chen, D. P. Kingma, T. Salimans, Y. Duan, P. Dhariwal, J. Schulman, I. Sutskever, and P. Abbeel, "Variational lossy autoencoder," *arXiv preprint arXiv:1611.02731*, 2016.

[24] E. de Bézenac, S. S. Rangapuram, K. Benidis, M. Bohlke-Schneider, R. Kurle, L. Stella, H. Hasson, P. Gallinari, and T. Januschowski, "Normalizing kalman filters for multivariate time series analysis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 2995–3007, 2020.

[25] J. Marino, L. Chen, J. He, and S. Mandt, "Improving sequential latent variable models with autoregressive flows," in *Symposium on advances in approximate bayesian inference*. PMLR, 2020, pp. 1–16.

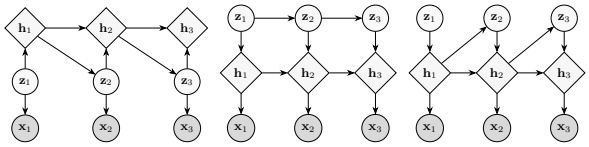
[26] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[28] E. Bingham, J. P. Chen, M. Jankowiak, F. Obermeyer, N. Pradhan, T. Karaletsos, R. Singh, P. Szerlip, P. Horsfall, and N. D. Goodman, "Pyro: Deep universal probabilistic programming," *Journal of machine learning research*, vol. 20, no. 28, pp. 1–6, 2019.

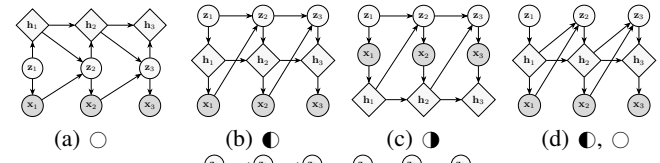
APPENDIX A
SUPPLEMENTARY GRAPHICAL MODELS

The deep state-space models corresponding to base models HMM, F-HMM, and HMM-Aug are shown in Fig. 3, 4, and 5 respectively.



(a) ○ (b) ◐ (c) ○, ◐

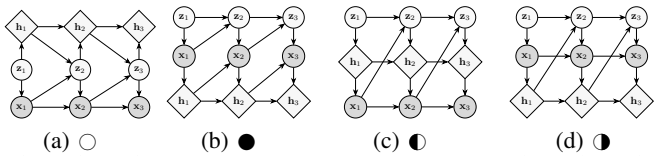
Fig. 3: DSSMs characterized by HMM.



(a) ○ (b) ◐ (c) ◑ (d) ◐, ○

(e) ○, ◑ (f) ◑, ◐ (g) ○, ◑, ◐

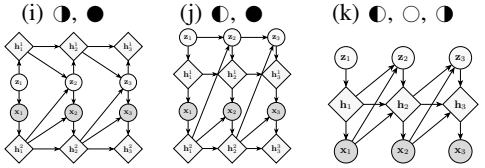
Fig. 4: DSSMs characterized by Feedforward HMM.



(a) ○ (b) ● (c) ◐ (d) ◑

(e) ○, ● (f) ◑, ◐ (g) ○, ◐ (h) ○, ◑

(i) ◑, ● (j) ◐, ● (k) ◐, ○, ◑ (l) ●, ◐, ○



(m) ○, ◑, ● (n) ◑, ●, ◐ (o) ○, ◑, ●, ◐

Fig. 5: DSSMs characterized by DMM-Aug. Figure (i) and (o) describe the generative models of SRNN and VRNN respectively.

APPENDIX B
DETAILED GENERATIVE AND INFERENCE MODELS

In this section, we describe the generative and inference processes of the DSSMs used for comparison in Section III. Feedforward SSM (FSSM) as shown in Fig 4 (g) has the following update rules:

$$\begin{aligned}\mathbf{z}_t &\sim p(\mathbf{z}_t|d_z(\mathbf{h}^1_{t-1},\mathbf{h}^2_{t-1}))\\ \mathbf{h}^1_t &= d_{h^1}(\mathbf{z}_t,\mathbf{h}^1_{t-1})\\ \mathbf{x}_t &\sim p(\mathbf{x}_t|d_x(\mathbf{h}^1_t))\\ \mathbf{h}^2_t &= d_{h^2}(\mathbf{x}_t,\mathbf{h}^2_{t-1})\end{aligned} \tag{14}$$

The inference models of FSSM are exactly the same as those of ArSSM as the exact posterior factorization of the models have the same form. The generative process of VRNN is given by:

$$\begin{aligned}\mathbf{z}_t &\sim p(\mathbf{z}_t|d_z(\mathbf{x}_{t-1},\mathbf{h}_{t-1}))\\ \mathbf{h}_t &= d_h(\mathbf{x}_{t-1},\mathbf{z}_t,\mathbf{h}_{t-1})\\ \mathbf{x}_t &\sim p(\mathbf{x}_t|d_x(\mathbf{h}_t))\end{aligned} \tag{15}$$

We consider two posterior approximations for VRNN:

$$\begin{aligned}q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) &= \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1},\overrightarrow{\mathbf{g}}^x_t,\overleftarrow{\mathbf{g}}^x_t)\\ q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) &= \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{h}_{t-1},\overleftarrow{\mathbf{g}}^x_{t-1})\end{aligned} \tag{16}$$

The generative process of SRNN is given by

$$\begin{aligned}\mathbf{z}_t &\sim p(\mathbf{z}_t|d_z(\mathbf{h}_{t-1},\mathbf{z}_{t-1}))\\ \mathbf{x}_t &\sim p(\mathbf{x}_t|d_x(\mathbf{h}_{t-1},\mathbf{z}_t))\\ \mathbf{h}_t &= d_h(\mathbf{x}_t,\mathbf{h}_{t-1})\end{aligned} \tag{17}$$

We apply the following posterior approximations to SRNN:

$$\begin{aligned}q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) &= \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1},\overrightarrow{\mathbf{g}}^x_t,\overleftarrow{\mathbf{g}}^x_t)\\ q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) &= \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1},\mathbf{h}_t,\overleftarrow{\mathbf{g}}^x_t)\end{aligned} \tag{18}$$

The generative process for RSSM is given by

$$\begin{aligned}\mathbf{z}_t &\sim p(\mathbf{z}_t|d_z(\mathbf{h}_{t-1}))\\ \mathbf{h}_t &= d_h(\mathbf{z}_t,\mathbf{h}_{t-1})\\ \mathbf{x}_t &\sim p(\mathbf{x}_t|d_x(\mathbf{h}_t))\end{aligned} \tag{19}$$

Similarly, we implement two posterior approximations:

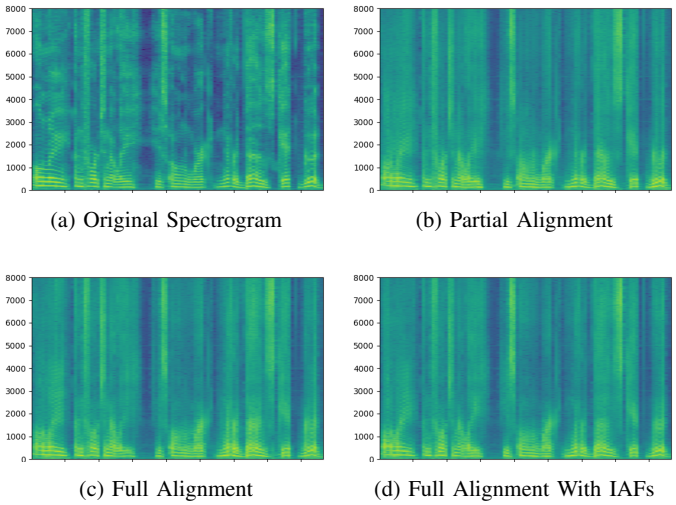
$$\begin{aligned}q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) &= \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1},\overleftarrow{\mathbf{g}}^x_t)\\ q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) &= \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{h}_{t-1},\overleftarrow{\mathbf{g}}^x_t)\end{aligned} \tag{20}$$

Note that each pair of approximated posterior distributions follow the order of partial alignment and full alignment.

APPENDIX C
EXPERIMENT SPECIFICS

The raw speech is sampled at 16 kHz. The analysis-resynthesis task is performed on a power spectrogram. The original time-domain speech is processed with the short-time Fourier transform (STFT), using 64-ms sine window with 25%-overlap to obtain 513-dimensional discrete Fourier spectra. Then, the power spectrograms are computed by squaring the spectra. In our experiment, the sequence length is chosen as 50, which means speech utterances pf 0.8 s are extracted from the original speech data and preprocessed with the STFT as mentioned above. The learning rate we set is 0.001. The training phase involves 1000 epochs without early stopping.

The corresponding phase spectrogram is directly combined with the output magnitude spectrogram of the DSSMs to reconstruct the output speech signal using inverse STFT with overlap-add. The Mel spectrograms for the original data and reconstructed form using ArSSM and its variants are shown in Fig. 6.



(a) Original Spectrogram (b) Partial Alignment

(c) Full Alignment (d) Full Alignment With IAFs

Fig. 6: Mel spectrograms for an audio sample and its reconstructed forms using three variations of ArSSM.

Appendix D
Extended Experiments

Apart from the two styles of approximate inference, we also test three variations, which we denote as approach A, B, and C as defined below:

- A: $q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{H}_{t-1}^1, \mathbf{h}_t^2, \overleftarrow{\mathbf{g}}_t^x)$
- B: $q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{H}_{t-1}^1, \overrightarrow{\mathbf{g}}_t^x, \overleftarrow{\mathbf{g}}_t^x)$
- C: $q(\mathbf{z}_{1:T}|\mathbf{x}_{1:T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{h}_{t-1}^1, \overrightarrow{\mathbf{g}}_t^x, \overleftarrow{\mathbf{g}}_t^x)$

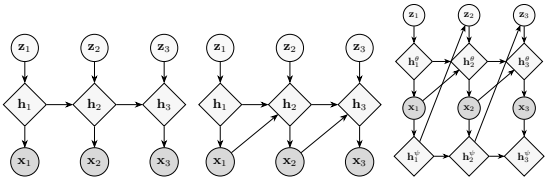
where $\mathbf{H}_{t-1}^1$ is the hidden state of an RNN in the inference process that carries information from the previous latent states. As shown in Table III, the above variations shown no apparent improvements.

	SI-SDR (dB)	PESQ	ESTOI
Partial Alignment	**6.050/5.416**	**1.783/1.690**	0.804/0.786
Full Alignment	5.791/5.132	1.751/1.675	0.806/0.795
A	6.000/5.128	1.767/1.665	**0.812/0.795**
B	5.249/4.639	1.682/1.640	0.781/0.774
C	5.280/4.892	1.651/1.570	0.791/0.776

TABLE III: SI-SDR, PESQ, and ESTOI results for ArSSM with different inference models. Values are reported as training/test. Bold values indicate the best performance for each metric.

Appendix E
Related Models

We briefly discuss a few models that are not mentioned in the main paper. The graphical models for Recurrent VAE (RVAE) [9], Stochastic Recurrent Network (STORN) [6], and Stochastic Video Generation with learned priors (SVG-LP) [2] are shown in Fig. 7. Apart from SVG-LP, [2] also introduced a model called SVG with fixed priors (SVG-FP), which shares the same graphical structure as STORN. These models are not included in the main body because our focus is on models where the latent states evolve over time. Notably, these models lack the autoregressive connections among latent states that are central to our framework.



(a) ◐ (b) ◑, ● (c) ●, ◐, ◑

Fig. 7: Graphical models for RVAE (a), STORN (b), and SVG-LP (c). The relaxation of the first-order Markov properties are denoted by the markers in the individual captions.

Appendix F
Acknowledgments

This research work is supported by a Math and Application Project (2021YFA1003504) under the National Key R&D Program, a Collaborative Research Fund by RGC of Hong Kong (Project No. C1143-20G), a grant from the Natural Science Foundation of China (U20A20189), a grant from ITF - Guangdong-Hong Kong Technology Cooperation Funding Scheme (Project Ref. No. GHP/145/20), and an InnoHK initiative of The Government of the HKSAR for the Laboratory for AI-Powered Financial Technologies.