Deep Encoder-Decoder Models for Unsupervised Learning of Controllable Speech Synthesis

Gustav Eje Henter, *Member, IEEE,* Jaime Lorenzo-Trueba[‡], *Member, IEEE,* Xin Wang, *Student Member, IEEE,* and Junichi Yamagishi, *Senior Member, IEEE*

***Abstract*—Generating versatile and appropriate synthetic speech requires control over the output expression separate from the spoken text. Important non-textual speech variation is seldom annotated, in which case output control must be learned in an unsupervised fashion. In this paper, we perform an in-depth study of methods for unsupervised learning of control in statistical speech synthesis. For example, we show that popular unsupervised training heuristics can be interpreted as variational inference in certain autoencoder models. We additionally connect these models to VQ-VAEs, another, recently-proposed class of deep variational autoencoders, which we show can be derived from a very similar mathematical argument. The implications of these new probabilistic interpretations are discussed. We illustrate the utility of the various approaches with an application to acoustic modelling for emotional speech synthesis, where the unsupervised methods for learning expression control (without access to emotional labels) are found to give results that in many aspects match or surpass the previous best supervised approach.**

***Index Terms*—Controllable speech synthesis, latent variable models, autoencoders, variational inference, VQ-VAE.**

I. Introduction

TEXT to speech (TTS) is the task of turning a given text into an audio waveform of the text message being spoken out loud. While speech waveforms have a very high bitrate (e.g., 705,600 bits per second for CD-quality audio), the spoken text only accounts for a handful of these bits, perhaps 50 or 100 bits per second [1]. A major challenge of text-to-speech synthesis is thus to fill in the additional bits in the audio signal in an appropriate and convincing manner. This is not an easy task, as speech features have complex interdependencies [2]. Furthermore, much of the excess acoustic variation in speech is not completely random and incidental, but conveys additional side-information of relevance to communication. The acoustics may, for instance, reflect characteristics such as speaker identity, speaker condition, speaker mood and emotion, pragmatics (via emphasis and intonation), the acoustic environment, and properties of the communication channel (microphone characteristics, room acoustics). Neither of these are determined by the spoken text.

Ideally, the acoustic cues and variability encountered in natural speech should not only be replicated in the acoustics to make the synthesis more convincing, but also be adjustable to create flexible and expressive synthesisers, and ultimately enhance communication between man and machine. Unfortunately, this is not the case today. Most statistical parametric speech synthesis approaches are based on supervised learning, and only account for the variation that can be directly explained by the annotation provided. Any deviations from the conditional mean as predicted from annotated labels is assumed to be random and largely uncorrelated, regardless of any structure or information it may possess.

At synthesis time, recreating the lost variability by drawing random samples from fitted Gaussian models has been found to be a poor strategy from a perceptual point of view, cf. [3], wherefore the predicted average speech features are used in synthesis instead; in fact, acoustic models must be highly accurate before random sampling outperforms the average speech [2]. Using the model mean for synthesis makes the same utterance sound exactly identical every time it is synthesised (unlike when humans speak), and is still likely to give rise to artefacts, for instance widened formant bandwidths when using spectral or cepstral acoustic feature representations.

In theory, salient variation beyond the text could be annotated in the database, enabling the acoustic effects of the additional labels to be learned during training and controlled during synthesis. However, speech annotation is laborious, difficult, and often subjective. This makes it costly to obtain sufficient amounts of data where non-text variation has been annotated accurately. Instead, synthesis practise has focussed on reducing the amount of (unhandled) acoustic variability by recording TTS databases of single talkers reading text in a consistent neutral tone. The use of such data for building synthesisers may benefit segmental acoustic quality, but likely contributes to the flat and detached delivery that many text-to-speech systems suffer from. Several publications [4]–[7] have meanwhile highlighted the potential benefits of acoustic variation (at least when annotated), for instance [7] presenting multi-speaker synthesisers that are more accurate than could be expected from training on any single speaker in the database alone and additionally allow control over properties of the generated speech, such as the speaker's voice.

Manuscript last revised July 30, 2018.

This research was carried out while all authors were with the Digital Content and Media Sciences Research Division at the National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan.

G. E. Henter is with the Department of Speech, Music and Hearing (TMH) at KTH Royal Institute of Technology, 100 44 Stockholm, Sweden. (e-mail: ghe@kth.se)

J. Lorenzo-Trueba is with Amazon.com in Cambridge, U.K. (e-mail: jaime@nii.ac.jp)

X. Wang is with the Digital Content and Media Sciences Research Division at the National Institute of Informatics, Japan. (e-mail: wangxin@nii.ac.jp)

J. Yamagishi is with the Digital Content and Media Sciences Research Division at the National Institute of Informatics, Japan, as well as with the Centre for Speech Technology Research at the University of Edinburgh, 10 Crichton Street, Edinburgh EH8 9AB, U.K. (e-mail: jyamagis@nii.ac.jp)

[‡] Work performed prior to joining Amazon.

This paper considers a number of alternatives to the standard approach outlined above. The common theme is to investigate and connect methods that attempt to explicitly account for the effects of *unannotated* variation in the data. These methods are able to learn synthesisers with controllable output acoustics (beyond the effects of the input text), albeit without an a-priori labelling of the perceptual effects of the learned control; this can be seen as an important, though not sufficient, step to eventually enable flexible speaking systems that respond appropriately to communicative context. Mathematically, our perspective is that of probabilistic modelling, specifically the theory of latent variables, and a major part of the work is to establish theoretical connections between practical approaches and principles of statistical estimation. Our main scientific contributions can be summarised as follows:

1) We use variational methods to show that several prior methods for learning controllable models from data with unannotated variation – the training heuristic used in [7]–[12], as well as so-called VQ-VAEs from [13] – can be interpreted as approximate maximum-likelihood approaches, and elucidate the approximations involved.
2) We introduce and detail various theoretical connections between the techniques in [7], [10]–[12] and encoder-decoder models, particularly VQ-VAEs.
3) We consider ways in which prior information can be integrated into the heuristic approaches (which lack an explicit prior distribution).
4) We use a large database of emotional speech to perform objective and subjective empirical evaluations of the heuristic approaches (with and without prior information) against comparable VQ-VAEs and a competitive supervised system on the task of acoustic modelling. The unsupervised methods are found to produce equal or better results than the supervised approach.

These contributions all extend preliminary work performed in [14].

The remainder of this article is laid out as follows: Sec. II outlines relevant prior work while Sec. III describes mathematical foundations. Sec. IV then presents novel interpretations of and connections between different encoder-decoder approaches. Sec. V recounts empirical evaluations performed on a database of emotional speech, while Sec. VI concludes.

II. Prior Work

In this section, we introduce controllable speech synthesis (Sec. II-A) and a wide variety of previous work of relevance to our contributions. We especially consider unsupervised learning of control (Sec. II-B) and variational autoencoders (Sec. II-C) and their use in speech generation (Sec. II-D). We also give an introduction to prior work on emotional speech synthesis (Sec. II-E), as this is the control task considered in our experiments.

A. Controllable Speech Synthesis

All text-to-speech systems are in a sense controllable, since the input text influences the output audio. (Voice conversion, similarly, represents a speech synthesiser driven by speech rather than text.) By *controllable speech synthesis*, however, we refer to speech synthesisers that enable additional output control beyond the words alone, such that the same text can be made to be spoken in several, perceptually distinct ways.

Early, rule-based parametric speech synthesisers typically exposed many control knobs ("speech parameters") relating to speech articulation and pronunciation; the text-to-speech aspect was simply a set of rules for how these knobs were to be moved in response to phonemes extracted from text [15], and the resulting parameter trajectories could be manually edited in order to alter pronunciation. Unit selection TTS can achieve control of any properties annotated in the database by including a term in the target cost to preferentially select units with labels similar to the user-selected control input. However, success depends heavily on the database having adequate coverage of the desired control configuration.

With the transition to statistical parametric speech synthesis (SPSS), [16], [17] it became straightforward to learn to control synthesiser output, i.e., to learn a mapping from control inputs to acoustic outputs. This avoids having to design the signal generator to expose the desired speech properties to be controlled or manually tuning weight factors in the target cost, and typically achieves meaningful control from smaller training databases than unit-selection approaches. The decision trees used in early SPSS systems can relatively easily incorporate additional categorical labels as phone- or frame-level inputs. Continuous-valued inputs can be quantised for decision-tree learning, and the quantisation threshold can be learned as well (e.g., through C4.5 [18]). So-called multiple regression HMMs (MR-HMMs) [19] were developed as a more refined method for continuous control of synthesiser output, by endowing each decision-tree node with a linear regression model that maps control inputs to acoustics. MR-HMMs and their extensions have been used for smoothly controlling properties such as speaking style [20], [21] or articulation [22].

B. Learning Control Without Annotation

The approaches covered in Sec. II-A all rely on control either being manually designed, or learned in a supervised manner from annotated data. This paper, in contrast, considers the more difficult situation where salient speech variability has not been annotated, but we nonetheless wish to learn to account for and replicate such variability by adjusting some synthesiser control inputs separate from the input text.

Many approaches to this problem exist. Unlike, e.g., Jauk [23], where the control space is defined by clustering training utterances based on pre-defined acoustic features, we concentrate on approaches that treat the unknown values of the hypothesised control parameters as if they were part of the set of unknown model parameters, and estimate all these unknowns through optimisation over the training data. This will learn a synthesiser that allows the control over the most (mathematically) salient extra-linguistic speech variation, but provides no a-priori indication what perceptual aspects that will be controllable (or how). One example of this approach is so-called cluster-adaptive training (CAT), introduced for automatic speech recognition (ASR) in [24]. It can be seen as

an extension of MR-HMMs to learning and optimising both decision-tree node regression models and their inputs. CAT has for instance been applied to learn expressive TTS with decision trees [25]. However, the method does not include a joint optimisation over the regression tree structure, and the possible uncertainty in the determination of the control input values from the acoustics is ignored.

With modern synthesis techniques based on deep learning there have been multiple independent proposals to improve modelling by using backpropagation to jointly optimise the entire regression model (the unknown weights of one or more neural networks) together with its control inputs. The idea was introduced for speaker adaptation in neural network ASR in [8], [9] under the name "discriminant condition codes" (DCC), and was independently adapted for multi-speaker speech synthesis several times: first by Luong et al. [7] and more recently by Arık et al. [11] (Deep Voice 2) and Taigman et al. [12] (VoiceLoop). In all cases, the result is that training and test speakers all are embedded in a low-dimensional speaker space. Independent of [9], Watts et al. [10] also proposed a mathematically identical setup and applied it to train a TTS acoustic model on a database of expressive speech, specifically children's audiobooks from [26]. (The equivalence between [9] and [10] was first pointed out in [14].) Watts et al. learned a fixed input vector for each utterance in the data, calling the approach "learned sentence-level control vectors". Adjusting the control parameter input when synthesising from the trained system was found to adjust vocal effort (pitch and energy) in a nonlinear and non-uniform manner.

Sawada et al. [27] considered similar data but took a somewhat different approach, wherein a unique "phrase code" was assigned to each phrase in the training data through random draws from a high-dimensional Gaussian distribution; this code was then used as an input to the synthesiser alongside the features extracted from the text. For test sentences, the phrase code of the training-data phrase with the greatest similarity (as computed through by doc2vec [28]) to the text phrase to be spoken was used as the control parameters. (They also assigned "word codes" to each word in a similar manner.) This overall approach is similar to the approaches with learned input codes – especially [10] – in that training-data segments were embedded in a fixed-dimensional space used to control the output, but here the embeddings were random rather than learned, and codes were predicted based on text rather than acoustics. Trained on children's audiobooks the resulting synthesiser achieved notably successful expression control and was one of the best-rated systems in the 2017 Blizzard Challenge [27], [29].

Luong et al. [7] evaluated both random and learned input codes with different dimensionalities for representing speaker variation, and compared them to simple one-hot vector speaker codes. They found no major differences in subjective performance between the methods, though all were better than no adaptation. However, we note that this and other speaker-adaptation evaluations typically involve some degree of supervision, since it generally is pre-specified which utterances that came from each speaker.

In the last year, there have been efforts to learn unsupervised control in the (mostly) end-to-end Tacotron [30] TTS framework. Parallel to this paper being written, these demonstrated the use of encoders and decoders for prosody transfer across speakers (given similar text prompts) [31] and more general style control [32]. This extends and improves on preliminary work presented by the same group in [33], which learned framewise rather than utterance-level control. Among other things, they demonstrate that the style-token approach in [32] is capable of synthesis with high subjective quality even from 95% noisy training data. They also demonstrated the use of a separately-learned speaker verification system as an encoder for controlling and adapting speaker identity [34].

C. Variational Autoencoders

Interestingly, all of the above proposals for unsupervised learning of controllable speech synthesis gloss over the issue that the actual values of any control inputs cannot be determined to exact certainty, since they are neither annotated nor observed. To properly account for the uncertainty regarding the unknown control inputs calls for the use of *latent* (or *hidden*) *variables* associated with each datum. The fundamental idea is simply to model the unknown quantities and their uncertainty as random variables. We can then use the theory of probability and estimation to make inferences about these unobserved variables. In practice, the mathematics are very similar to Bayesian probability, but the prior and posterior distributions pertain to (local) control inputs, not to the (global) model parameters, which may still be treated in a frequentist manner.

Latent-variables are ubiquitous in speech modelling, with two examples being the component in a mixture model and the unobservable state variable in hidden Markov models (HMMs) [35], [36]. Training algorithms for these latent-variable approaches are usually derived from the expectation-maximisation (EM) framework [37]. However, the expressiveness of these classical methods is often quite limited, and new setups generally require careful, manual derivation of update equations, which often is prohibitively difficult for more complex and interesting models.

A recent idea is to harness the power of deep learning to describe and train more flexible latent-variable models. Using techniques similar to [37], Henter et al. [14] showed that, for the special case of EM-like alternate optimisation, the heuristic methods [7]–[12] can be seen as "poor man's latent variables" that can learn a complex mapping from latent to observable variables but ignore any uncertainty in the latent space. A more full-fledged example of deep learning of latent variables is so-called *variational autoencoders* (VAEs) [38], [39]. They use neural networks to parameterise both how observations depend on continuous latent variables (control inputs) along with the act of inferring latent-variable distributions from observations. VAEs are considered autoencoders since the inference process can be seen as encoding an observation into a latent variable value (or distribution) while the generation can be seen as decoding the latent variable back to the observation domain. We elaborate on this connection in Sec. III-C. Furthermore, the two mappings can be learned tractably and jointly through gradient descent [40], in contrast to some mathematically similar models such as Helmholtz machines [41].

A practical issue with VAEs is that they sometimes fail to learn to make proper use of the latent variables to explain the observed variation: in that case, the estimated control inputs do not change appreciably over the training data (their inferred distributions are highly overlapping) and exert little influence over model outputs, cf. [42]. Chen et al. [43], Huszár [44], and Graves et al. [45] provide lucid discussions of this problem. This has been called "posterior collapse" in [13], although it does *not* mean that the posterior collapses to a point – just that the posterior collapses to the same distribution (which is also the prior) regardless of the observation made. A recent proposal to combat this issue is to quantise the encoder output through a vector-quantisation (VQ) step, such that the inferred value of the hidden variable for an observation is taken from a finite codebook. The resulting construction is called *VQ-VAE*, and was introduced in [13]. While the regular VAEs objective function penalises the variational posterior diverging from the prior (which can force "posterior collapse"), this penalty reduces to a constant for the VQ-VAE, and thus does not affect learning. Although the fact that only a single codebook vector is used for each observation means that any uncertainty in the inference step is not represented explicitly, we show in Sec. IV-A that the mathematics still can be derived from the same latent-variable principles that underpin regular VAEs. VQ-VAEs might use discrete latent variables, but these latents are nonetheless embedded in a continuous Euclidean space.

While Gaussian mixture models and HMMs also consider discrete latent variables that are in some sense embedded (through their mean vectors) in a vector space, VQ-VAEs let the latent vectors occupy a space different from that of the observations. The VQ-VAE mapping from latent space to observation space is furthermore strongly nonlinear, which differentiates it from constructions like subspace GMMs [46].

Variational autoencoders also resemble recently-popular generative adversarial networks (GANs) [47], in that the latter also use a random latent variable to explain variation in the observations through a highly-nonlinear mapping parameterised by a neural network. However, VAEs map latent variable values to output distribution parameters, whereas GANs map latent samples directly to observations. Parameter estimation in GANs is also more challenging, since one seeks a Nash equilibrium of a game between two agents, rather than an optimum of a fixed objective function as in VAEs. A taxonomy of different generative models such as VAEs and GANs, along with connections between them, is provided in [48]. In Sec. IV this paper, we bring the widely-used heuristic from Sec. II-B (DCC/sentence-level control vectors) into the fold, by describing its connections to VAEs and latent-variable models.

D. Variational Autoencoders in Synthesis

Variational autoencoders have seen a number applications to speech generation. For example, [42], [49]–[51] all consider applying VAEs to each frame in an acoustic analysis of speech, with the intention of learning to encode something similar to phonetic identity in the absence of transcription. In [49], [50], this was used to identify matching data frames for non-parallel voice conversion. [52], [53] used VAEs to separate and manipulate both speaker and phone identities, though without generating or evaluating speech audio. Very recently [54] used VAEs to identify sentence-level latent variables in the VoiceLoop [12] framework.

VAEs have also been applied to speech waveform modelling, typically based on generalisations of basic VAEs to sequence models such as [55]–[58]. While [56]–[58] all contain applications to speech data, only Chung et al. [56] considered speech signal generation. Unfortunately, the perceptual quality of random waveforms sampled from their model is poor: there is a lot of static, and no intelligible speech is produced, since the models are not conditioned on an input text. Much better segmental quality has been demonstrated by generating signals using WaveNet [5]. In a standard WaveNet the next-step distribution only depends on the previous waveform in the receptive field and possible conditioning information, with no hidden state. Other successful neural networks for waveform generation include SampleRNN [59] and WaveRNN [60], which contain a deterministic (hidden) RNN state. The VQ-VAE paper [13] combines these breakthroughs (specifically WaveNet) with VAEs, using strided convolutions to downsample and encode raw audio into discrete quantisation indices with a WaveNet-like architecture for decoding. This approach was able to reproduce high-quality versions of encoded waveforms, and the quantisation indices were additionally found to be closely related to phones, providing a compelling demonstration of unsupervised acoustic unit discovery.

Wang [61, Ch. 7] investigated VQ-VAEs for F0 modelling on the utterance, mora, and phone levels in Japanese TTS, coupled with a *linguistic linker* to predict VQ-VAE codebook indices from linguistic features. It was found that a combined VQ-VAE approach on the mora and phone levels performed objectively and subjectively on par with a larger deep, autoregressive F0 model [62] without explicit latent variables.

Different from the prior work above, but similar to the heuristics [7]–[12] in Sec. II-B, this paper considers (VQ-)VAE approaches that model and encode utterance-wide, non-phonetic information that complements the known transcription.

The work on speech synthesis with global style tokens (GSTs) in [32] has many similarities to VQ-VAEs and encoder-decoder based synthesis. While the global style tokens are initialised as random vectors (like in, e.g., [27]), only a limited, fixed number of style tokens is used, reminiscent of a vector-quantiser codebook. Unlike VQ-VAEs, however, the style-token approach uses attention to obtain a set of positive interpolation weights between the different tokens. This means that utterances in practice can fall on a continuum in token space, similar to the heuristic approaches in Sec. II-B. Another difference is that the encoders in [31], [32], [34] do not have access to the text features, in contrast to the heuristic and VQ-VAE approaches studied in this paper, which make use of both acoustic and text-derived features in encoding.

E. Emotional Speech Synthesis

The experiments in this paper consider speech synthesis from a large corpus of acted emotional speech, described in [63]. The importance of emotional expression in speech

synthesis can be seen in, e.g., the 2016 Blizzard Challenge [26], where suitably accounting for the expressive nature of the data was a common element of the most successful entries.

There have been successful demonstrations of emotional speech synthesis with speech generation based on unit selection (including hybrid speech synthesis) [64]–[66] as well as through SPSS with decision trees [67]–[71]. Most of these consider a relatively limited number of discrete emotional classes, from binary (e.g., neutral vs. affective as in [66]) to the "big six" (anger, disgust, fear, happiness, sadness and surprise, as considered in [64], [65], [70]); [68], which investigates continuous emotional-intensity control with MR-HMMs, is an exception. Applications of methods based on neural-networks to emotional speech synthesis are less common, though there are a few examples [14], [63] from the last year. This article builds on these two publications and considers the same data in the experiments.

III. Mathematical Background

This section introduces the mathematical preliminaries of speech synthesis as necessary for the novel insights described in Sec. IV. In particular, Sec. III-A outlines controllable speech synthesis through latent variables, while remaining sections describe the fundamental theory of variational inference (Sec. III-B) and variational autoencoders in general (Sec. III-C).

A. Controlling Speech Synthesis Through Latent Variables

Mathematically, statistical parametric speech synthesis is usually formulated as a regression problem. The central statistical modelling task is to map an input sequence $\underline{\boldsymbol{l}}$ of text-based ("linguistic") features to a sequence $\underline{\boldsymbol{x}}$ of acoustic features ("speech parameters") that control a waveform generator (vocoder).[1] Since human speech is stochastic even for a given text and control input (cf. [2]), we typically want to map the input $\underline{\boldsymbol{l}}$ to an entire distribution $\underline{\boldsymbol{X}}(\underline{\boldsymbol{l}})$ of acoustic feature sequences $\underline{\boldsymbol{x}}$. This mapping is learned from a parallel corpus of text and speech using statistical methods. The linguistic features $\underline{\boldsymbol{l}}$ in the mapping are extracted deterministically from input text by a (typically language-dependent) so-called front-end. While the front-end traditionally has been designed rather than learned, this is starting to change, with a number of frameworks [12], [30], [72], [73] learning to predict acoustics directly from sequences of characters or phones. Similarly, the waveform generator is traditionally a fixed, designed component, for example STRAIGHT [74] or WORLD [75], to whose control interface the acoustic feature representation is tied. However, learned (neural) vocoders have recently achieved impressive results, e.g., [76]. Thus, while it is possible to learn both the front-ends and vocoders, only the central linguistic-to-acoustic mapping is consistently learned from speech data.[2]

[1]In this text, bold symbols signify vectors or matrices; the underline denotes a time sequence $\underline{\boldsymbol{l}} = (\boldsymbol{l}_1, \ldots, \boldsymbol{l}_T)$. Capital letters identify random variables, while corresponding lowercase quantities represent specific, non-random outcomes of those variables.

[2]For all the interest in waveform-level speech synthesis, it is worth noting that [76] – the current state of the art in text-to-speech signal quality – still solves a statistical parametric speech synthesis problem. The difference in speech quality comes from matched training of a learned vocoder instead of synthesising waveforms with the Griffin-Lim algorithm as in [30].

Let $\mathcal{D} = \left\{\underline{\boldsymbol{l}}^{(n)}, \underline{\boldsymbol{x}}^{(n)}\right\}_{n=1}^{N}$ be a dataset of N aligned linguistic (input) and acoustic (output) data sequences, which are assumed to be independent and identically distributed draws from a joint distribution of $\underline{\boldsymbol{L}}$ and $\underline{\boldsymbol{X}}$. Let further $f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}}}(\underline{\boldsymbol{x}} \,|\, \underline{\boldsymbol{l}}; \boldsymbol{\theta})$ be a parametric model describing the probability of output $\underline{\boldsymbol{X}}$ given $\underline{\boldsymbol{L}}$. To estimate the unknown model parameters $\boldsymbol{\theta}$ it is standard to use maximum-likelihood estimation

$$\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}(\mathcal{D}) = \operatorname*{argmax}_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta} \,|\, \mathcal{D}) \tag{1}$$

$$\mathcal{L}(\boldsymbol{\theta} \,|\, \mathcal{D}) = \sum_{n=1}^{N} \ln f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}}}\left(\underline{\boldsymbol{x}}^{(n)} \,\middle|\, \underline{\boldsymbol{l}}^{(n)}; \boldsymbol{\theta}\right). \tag{2}$$

To achieve control over how the text message encoded by $\underline{\boldsymbol{l}}$ is spoken, we add a second input representing control parameters, $\boldsymbol{z}$. While one could envision using a sequence $\underline{\boldsymbol{z}} \in \mathbb{R}^D$ of control inputs that may change throughout an utterance, we only develop the mathematics for the case when this input is constant for each data sequence, and thus can be represented by a single vector $\boldsymbol{z}$. If this control signal has been annotated as $\boldsymbol{z}^{(n)}$ for each training data sequence it is straightforward to train a controllable synthesiser by maximising the conditional likelihood

$$\mathcal{L}(\boldsymbol{\theta} \,|\, \mathcal{D}) = \sum_{n=1}^{N} \ln f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}},\boldsymbol{Z}}\left(\underline{\boldsymbol{x}}^{(n)} \,\middle|\, \underline{\boldsymbol{l}}^{(n)}, \boldsymbol{z}^{(n)}; \boldsymbol{\theta}\right). \tag{3}$$

Changing the control signal will then cause the output distribution to be more similar to the examples with similar annotated control-input values, assuming learning was successful.

The situation becomes more interesting if the control parameter is a latent (unobserved) variable. A general and principled approach is to treat the unknown control input as a random variable $\boldsymbol{Z}$ which is jointly distributed with $\underline{\boldsymbol{X}}$ as in

$$f_{\underline{\boldsymbol{X}},\boldsymbol{Z}|\underline{\boldsymbol{L}}}(\underline{\boldsymbol{x}}, \boldsymbol{z} \,|\, \underline{\boldsymbol{l}}; \boldsymbol{\theta}) = f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}},\boldsymbol{Z}}(\underline{\boldsymbol{x}} \,|\, \underline{\boldsymbol{l}}, \boldsymbol{z}; \boldsymbol{\theta})\, f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}}(\boldsymbol{z} \,|\, \underline{\boldsymbol{l}}; \boldsymbol{\theta}), \tag{4}$$

where $f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}}$ is a conditional prior for $\boldsymbol{Z}$. To perform maximum-likelihood parameter estimation in the presence of this latent variation one marginalises out the unknown random variable, and thus maximises

$$\mathcal{L}(\boldsymbol{\theta} \,|\, \mathcal{D}) = \sum_{n=1}^{N} \ln \int f_{\underline{\boldsymbol{X}},\boldsymbol{Z}|\underline{\boldsymbol{L}}}\left(\underline{\boldsymbol{x}}^{(n)}, \boldsymbol{z} \,\middle|\, \underline{\boldsymbol{l}}^{(n)}; \boldsymbol{\theta}\right) \mathrm{d}\boldsymbol{z}; \tag{5}$$

this is termed the *marginal likelihood* or the *model evidence*, but is merely another way of writing $f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}}}$ from Eq. (2).

To generate speech from a latent-variable model like this, there are two conceivable $\underline{\boldsymbol{X}}$-distributions to consider. One could use the same marginalisation principle as in Eq. (5) and generate speech based on $f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}}}$ (i.e., after integrating out $\boldsymbol{Z}$). However, the integral is frequently intractable, as discussed in the next paragraph. Moreover, this does not allow control of the output speech $\underline{\boldsymbol{x}}$. For these reasons we exclusively consider output generation from the $\underline{\boldsymbol{X}}$-distribution conditioned on $\boldsymbol{Z}$, $f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}},\boldsymbol{Z}}$. By adjusting the input $\boldsymbol{z}$-value, the same text may then be spoken in (statistically) distinct ways.

B. Variational Inference

Unfortunately, the integral in Eq. (5) is only tractable to evaluate for quite basic models, which tend to be too simplistic to allow an acceptable description of reality. To fit more advanced statistical models, approximations must be made. Some approximation techniques rely on numerical methods for estimating the value of the integral, e.g., through Monte-Carlo sampling. In this paper, however, we consider analytical approximations based on variational principles, where a parametric and tractable approximation $q(\boldsymbol{z};\,\boldsymbol{\varphi})$ is used in place of the intractable true posterior $f_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}$. Instead of maximising the likelihood $\mathcal{L}$ directly, one then maximises a lower bound $\underline{\mathcal{L}}$ on it, sometimes called the *evidence lower bound* (ELBO). Specifically, one can show [35, Sec. 10.1] that

$$\ln f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}}}\left(\underline{\boldsymbol{x}}\,|\,\underline{\boldsymbol{l}};\,\boldsymbol{\theta}\right) = D_{\mathrm{KL}}\left(q\,\middle\|\,f_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}\right) + \underline{\mathcal{L}}\left(\boldsymbol{\theta},\,\boldsymbol{\varphi}\,|\,\underline{\boldsymbol{x}},\,\underline{\boldsymbol{l}}\right), \tag{6}$$

where

$$D_{\mathrm{KL}}\left(q\,\middle\|\,f_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}\right) = \int q(\boldsymbol{z};\,\boldsymbol{\varphi}) \ln \frac{q(\boldsymbol{z};\,\boldsymbol{\varphi})}{f_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}\left(\boldsymbol{z}\,|\,\underline{\boldsymbol{x}},\,\underline{\boldsymbol{l}};\,\boldsymbol{\theta}\right)}\,\mathrm{d}\boldsymbol{z} \tag{7}$$

is the Kullback-Leibler divergence (or KLD) and

$$\underline{\mathcal{L}}\left(\boldsymbol{\theta},\,\boldsymbol{\varphi}\,|\,\underline{\boldsymbol{x}},\,\underline{\boldsymbol{l}}\right) = \int q\left(\boldsymbol{z};\,\boldsymbol{\varphi}\right) \ln \frac{f_{\underline{\boldsymbol{X}},\,\boldsymbol{Z}|\underline{\boldsymbol{L}}}\left(\underline{\boldsymbol{x}},\,\boldsymbol{z}\,|\,\underline{\boldsymbol{l}};\,\boldsymbol{\theta}\right)}{q\left(\boldsymbol{z};\,\boldsymbol{\varphi}\right)}\,\mathrm{d}\boldsymbol{z} \tag{8}$$

is the evidence lower bound. Since the KLD between two distributions satisfies $D_{\mathrm{KL}}\left(p\,\|\,q\right) \geq 0$, with equality if and only if $p = q$, the desired bound $\mathcal{L} \geq \underline{\mathcal{L}}$ follows. This bound can be applied to every term in Eq. (2) with a separate q-distribution $q(\boldsymbol{z};\,\boldsymbol{\varphi}^{(n)})$ for each datapoint to lower-bound the entire training-data likelihood.

If q is chosen cleverly, the integral in Eq. (8) can sometimes be evaluated analytically. One can then identify a parameter estimate $\widehat{\boldsymbol{\theta}}_{\mathrm{VI}}$ and a set of per-datum q-distribution parameters $\boldsymbol{\varphi}^{\star(n)}$ (producing the variational posteriors $q^{\star(n)}$) that jointly maximise $\underline{\mathcal{L}}$. This framework provides the basis for optimising and using powerful statistical models through the use of an approximate latent posterior. The difference between the optimal lower bound $\underline{\mathcal{L}}$ and the optimal (log-)likelihood $\mathcal{L}$ of the model without the variational approximation is given by $D_{\mathrm{KL}}\left(p\,\|\,q^{\star}\right)$ and is referred to as the *approximation gap* [77].

C. Variational Autoencoders

The main idea of variational autoencoders [38], [39] is to use neural networks to parameterise not only the output-distribution dependence on latent-variable values, but also the act of latent-variable inference, and then learn these two networks simultaneously. Like in variational inference in general, we approximate the true latent posterior $f_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}$ by a variational posterior q, but instead of optimising the set $\left\{\boldsymbol{\varphi}^{(n)}\right\}$ to identify a different posterior distribution $q^{\star(n)}$ for each datapoint, these multiple optimisations are replaced by a single function $q_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}\left(\boldsymbol{z}\,|\,\underline{\boldsymbol{x}},\,\underline{\boldsymbol{l}};\,\boldsymbol{\varphi}\right)$ (here a neural network) that simply maps the values of $\underline{\boldsymbol{x}}$ and $\underline{\boldsymbol{l}}$ to (parameters of) an approximate posterior q.[3] This function $q_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}$, parameterised by the network weights $\boldsymbol{\varphi}$, is sometimes called the *inference network*, the *recognition network*, or the *encoder* and is distinct from the previously-introduced conditional output distribution $f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}},\,\boldsymbol{Z}}$ (sometimes called the *decoder*) that is parameterised by $\boldsymbol{\theta}$.

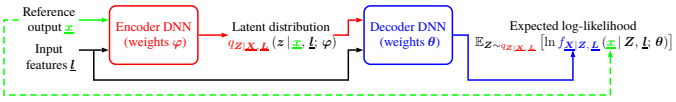


Figure 1. Conditional variational autoencoder training.

Given the parameterised inference $q_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}$ defined above, one can show [38], [40] that

$$\begin{aligned}&\ln f_{\underline{\boldsymbol{X}}}\left(\underline{\boldsymbol{x}};\,\boldsymbol{\theta}\right) - D_{\mathrm{KL}}\left(q_{\boldsymbol{Z}|\underline{\boldsymbol{X}}}\,\middle\|\,f_{\boldsymbol{Z}|\underline{\boldsymbol{X}}}\right)\\ &= \mathbb{E}_{\boldsymbol{Z}\sim q_{\boldsymbol{Z}|\underline{\boldsymbol{X}}}}\left[\ln f_{\underline{\boldsymbol{X}}|\boldsymbol{Z}}\left(\underline{\boldsymbol{x}}\,|\,\boldsymbol{Z};\,\boldsymbol{\theta}\right)\right] - D_{\mathrm{KL}}\left(q_{\boldsymbol{Z}|\underline{\boldsymbol{X}}}\,\middle\|\,f_{\boldsymbol{Z}}\right),\end{aligned} \tag{9}$$

where we for succinctness have suppressed the dependence on $\underline{\boldsymbol{l}}$. (Strictly speaking, our main consideration is *conditional VAEs*, or C-VAEs, where every distribution additionally is conditioned on an input such as $\underline{\boldsymbol{l}}$, but this difference is not of importance to the exposition.) The right-hand side in the equation is a lower bound on the likelihood (since the KLD on the left-hand side cannot be negative) which, it turns out, can be optimised efficiently using stochastic gradient ascent for certain choices of prior $f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}}$ and approximate posterior q. A common choice [38] is to take both distributions to be Gaussian; in this article we will additionally assume that the conditional output distribution $f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}},\,\boldsymbol{Z}}$ is an isotropic Gaussian.

The act of replacing individual optimisations by the regression problem of finding the weights $\boldsymbol{\varphi}$ in VAEs is sometimes called *amortised inference*, since it amortises the computational cost of the separate optimisations (inferring $q^{(n)}$) over the entire training. (See [77], [78] for in-depth explanations.) Since the posterior parameters predicted by the learned q-function may not be optimal for each datapoint, VAEs will in practise usually not reach the same performance as the theoretically optimal $\underline{\mathcal{L}}$ attained using $q^{\star(n)}$. The difference between the ELBO value attained by the VAE and the maximal ELBO possible under the chosen family of approximate posteriors q is known as the *amortisation gap* [77], and is added to the approximation gap due to the use of the approximate variational posterior defined in Sec. III-B.

The "autoencoder" part of "variational autoendcoders" comes from the observation that $q_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}\left(\boldsymbol{z}\,|\,\underline{\boldsymbol{x}},\,\underline{\boldsymbol{l}};\,\boldsymbol{\varphi}\right)$ essentially encodes $\underline{\boldsymbol{x}}$ into a latent variable $\boldsymbol{z}$, such that the original $\underline{\boldsymbol{x}}$ is maximally likely to be recovered from (samples from) $q_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\,\underline{\boldsymbol{L}}}$, as seen in the expectation in Eq. (9). This is illustrated conceptually in Fig. 1. Also note that the two terms on the right-hand side of Eq. (9) pull in different directions during maximisation: the first term is trying to make the approximate

[3] Please note that $\boldsymbol{\varphi}$ now denotes a set of neural network weights that define a mapping from $\underline{\boldsymbol{x}}$ and $\underline{\boldsymbol{l}}$ to distribution parameters, rather than distribution parameters themselves as in Sec. III-B.

posterior $q_{\boldsymbol{Z}|\underline{\boldsymbol{X}},\underline{\boldsymbol{L}}}$ resemble the true posterior as much as possible, while the second instead prioritises q not straying too far from the given prior distribution. If our model class $f_{\underline{\boldsymbol{X}}|\underline{\boldsymbol{L}},\boldsymbol{Z}}$ is sufficiently powerful to describe the observations well without depending on $\boldsymbol{z}$ as an input, the learned latent variables are likely to stay close to the prior and exert minimal influence on the observation distribution [44]. This is a common failure mode of VAEs, and is especially undesirable when learning output control.

To reduce the risk of not learning a useful latent-variable representation ("posterior collapse"), one can introduce a weight between the two terms in Eq. (9), yielding so-called β-VAEs [79], which can also be annealed [80]. This is straightforward to implement, but is not easy to motivate on probabilistic grounds and can not generally be interpreted as a lower bound on the marginal likelihood [81]. Alternatively, one might reduce the capacity/flexibility of the decoder model $f_{\underline{\boldsymbol{X}}|\boldsymbol{z},\underline{\boldsymbol{l}}}$, for instance by modelling speech parameters with a simple Gaussian distribution as in the experiments in Sec. V. VQ-VAEs were conceived as a third option for easily learning meaningful and informative latent representations.

IV. Theoretical Insights

This section presents and discusses the main theoretical developments of this paper. In particular, Sec. IV-A describes a new probabilistic understanding of VQ-VAEs, Sec. IV-B likewise introduces a variational derivation of the heuristic methods from [7]–[12] and connects these to other autoencoder models, while Sec. IV-C discusses how prior information might be incorporated into the heuristic models. To the best of our knowledge, all of these contributions are new.

A. A Variational Interpretation of VQ-VAEs

VQ-VAEs were introduced in [13] as a method of training VAEs when $\boldsymbol{Z}$ is a discrete random variable from a *codebook* $\mathcal{Z} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_M)$, a finite set of vectors in $\mathbb{R}^D$. This replaces the integrals in divergences and expectations with sums. Moreover, the latent prior $f_{\boldsymbol{Z}}$ is taken to be uniform over $\mathcal{Z}$ while the variational posterior q for $\boldsymbol{Z}$ is taken to be a point estimate $\boldsymbol{z}_q \in \mathcal{Z}$. The VQ-VAE encoder is realised as a function $\boldsymbol{z}_e(\underline{\boldsymbol{x}}; \boldsymbol{\varphi})$ taking values on all of $\mathbb{R}^D$, which subsequently is vector quantised using the nearest codebook vector to obtain $\boldsymbol{z}_q$. After adding squared-error regularisation terms to the ELBO to promote codebook vectors and encoded values being close together, the full VQ-VAE objective function for a single datapoint becomes[4]

$$\begin{aligned}\mathcal{L}_{\mathrm{VQ}}(\boldsymbol{\theta}, \boldsymbol{\varphi}, \mathcal{Z} \,|\, \underline{\boldsymbol{x}}) &= \ln f_{\underline{\boldsymbol{X}}|\boldsymbol{Z}}\left(\underline{\boldsymbol{x}} \,\middle|\, \boldsymbol{z}_q(\underline{\boldsymbol{x}}); \boldsymbol{\theta}\right) \\ &\quad - \left\|\mathrm{sg}\left(\boldsymbol{z}_e(\underline{\boldsymbol{x}})\right) - \boldsymbol{z}_q\right\|_2^2 - \beta \left\|\boldsymbol{z}_e(\underline{\boldsymbol{x}}) - \mathrm{sg}\left(\boldsymbol{z}_q\right)\right\|_2^2. \quad (10)\end{aligned}$$

Here $\mathrm{sg}(\cdot)$ is the *stop-gradient operator* implemented in many deep learning frameworks, which essentially means that the argument is to be treated as a constant during differentiation. (For simplicity, we ignore the conditioning on $\underline{\boldsymbol{l}}$ in our treatment of VQ-VAEs.) The straight-through estimator described in [82] is used to backpropagate the gradient through the (non-differentiable) quantisation that turns $\boldsymbol{z}_e(\underline{\boldsymbol{x}})$ into $\boldsymbol{z}_q(\underline{\boldsymbol{x}})$ in the likelihood term. Since this estimator ignores the effect of the VQ codebook, the gradient used to update $\mathcal{Z}$ only depends on the second term in the objective function in Eq. (10) [13].

As originally introduced in [13], the regularisation terms in Eq. (10) (e.g., the "commitment loss") are motivated on geometric, not probabilistic grounds. Together with the quantisation and the stop-gradient operators, this makes it difficult to assign a probabilistic interpretation to the VQ-VAE objective function. However, we will now show that it is possible to interpret the objective function as an actual ELBO maximisation.

Proposition 1: For $\beta = 1$, optimising the VQ-VAE objective in Eq. (10) is equivalent to optimising the combined objective

$$\begin{aligned}&\mathcal{L}_{\mathrm{VQ1}}(\boldsymbol{\theta}, \boldsymbol{\varphi}, \mathcal{Z} \,|\, \underline{\boldsymbol{x}}) \\ &\quad = \ln f_{\underline{\boldsymbol{X}}|\boldsymbol{Z}}\left(\underline{\boldsymbol{x}} \,\middle|\, \boldsymbol{z}_q(\underline{\boldsymbol{x}}); \boldsymbol{\theta}\right) - \left\|\boldsymbol{z}_e(\underline{\boldsymbol{x}}; \boldsymbol{\varphi}) - \boldsymbol{z}_q\right\|_2^2, \quad (11)\end{aligned}$$

which lacks the stop-gradient operators.

This proposition is easily verified by computing and comparing the partial derivatives of $\mathcal{L}_{\mathrm{VQ}}$ and $\mathcal{L}_{\mathrm{VQ1}}$ with respect to $\boldsymbol{\theta}$, $\boldsymbol{\varphi}$, and $\mathcal{Z}$. In practice, the results of learning are said [13] not to depend substantially on the numerical value of the hyperparameter β. Our analysis will henceforth assume $\beta = 1$, although $\beta = 0.25$ is used for the experiments, following [13].

Next we will show how Eq. (11) can be derived in a principled manner from a probabilistic model that includes a statistical model of the effect of quantisation in the latent space. We are not aware of any prior publications that derive VQ-VAEs from probabilistic principles alone.

To begin with, we model the distribution of encoder outputs in the latent space through a Gaussian mixture model (GMM). More concretely, we separate encoding and quantisation through a two-part latent variable $\boldsymbol{Z} = (\boldsymbol{Z}_e, \boldsymbol{Z}_q)$, where $\boldsymbol{z}_e \in \mathbb{R}^D$ represents the encoder output and $\boldsymbol{z}_q \in \mathcal{Z} \subset \mathbb{R}^D$ is the quantised version thereof. Assume that $\underline{\boldsymbol{X}}$ is conditionally independent of $\boldsymbol{Z}_e$ given the codebook vector $\boldsymbol{Z}_q$. (This is the reverse of more conventional uses of mixture models in VAEs [83], [84], where the observation $\underline{\boldsymbol{X}}$ is instead assumed to be conditionally independent of the mixture component identity $\boldsymbol{Z}_q$ given the mixture model sample $\boldsymbol{Z}_e$.) The joint model then factorises as

$$\begin{aligned}&f_{\underline{\boldsymbol{X}},\boldsymbol{Z}_e,\boldsymbol{Z}_q}(\underline{\boldsymbol{x}}, \boldsymbol{z}_e, \boldsymbol{z}_q; \boldsymbol{\theta}) \\ &\quad = f_{\underline{\boldsymbol{X}}|\boldsymbol{Z}_q}\left(\underline{\boldsymbol{x}} \,\middle|\, \boldsymbol{z}_q; \boldsymbol{\theta}\right) f_{\boldsymbol{Z}_e|\boldsymbol{Z}_q}\left(\boldsymbol{z}_e \,\middle|\, \boldsymbol{z}_q\right) f_{\boldsymbol{Z}_q}\left(\boldsymbol{z}_q\right). \quad (12)\end{aligned}$$

We further assume that the latent prior $f_{\boldsymbol{Z}_q}$ over codebook vectors is uniform and that $f_{\boldsymbol{Z}_e|\boldsymbol{Z}_q}$ is an isotropic Gaussian centred on $\boldsymbol{Z}_q$ with fixed covariance matrix $\sigma^2\boldsymbol{I}$. $\boldsymbol{Z}_e$ here provides an explicit representation of the noise introduced by the vector quantiser. Analogous to a regular VAE, the remaining parameters $\boldsymbol{\varphi}$ and (here) $\mathcal{Z}$ define the variational posterior $q_{\boldsymbol{Z}}$. In particular, we choose a posterior of the form

$$\begin{aligned}&q_{\boldsymbol{Z}_e,\boldsymbol{Z}_q|\underline{\boldsymbol{X}}}\left(\boldsymbol{z}_e, \boldsymbol{z}_q \,\middle|\, \underline{\boldsymbol{x}}; \boldsymbol{\varphi}, \boldsymbol{e}\right) \\ &= q_{\boldsymbol{Z}_e|\boldsymbol{Z}_q,\underline{\boldsymbol{X}}}\left(\boldsymbol{z}_e \,\middle|\, \boldsymbol{z}_q, \underline{\boldsymbol{x}}; \boldsymbol{\varphi}\right) q_{\boldsymbol{Z}_q|\underline{\boldsymbol{X}}}\left(\boldsymbol{z}_q \,\middle|\, \underline{\boldsymbol{x}}; \boldsymbol{e}\right) \quad (13) \\ &= f\left(\boldsymbol{z}_e - \boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi})\right) I\left(\boldsymbol{z}_q = \boldsymbol{e}\right), \quad (14)\end{aligned}$$

[4] This formula corrects a sign inconsistency present in Eq. (3) of [13].

Here, $\boldsymbol{e} \in \mathcal{Z}$ (to enforce quantisation), $I(\cdot)$ is the indicator distribution (which equals one if the argument is true and zero otherwise), while $f(\cdot)$ is any fixed, unimodal distribution centred on the origin. To reduce confusion with the latent outcome $\boldsymbol{z}_e$, we have abbreviated the encoder output $\boldsymbol{z}_e(\underline{\boldsymbol{x}}; \boldsymbol{\varphi})$ as $\boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi})$. When $f(\cdot)$ shrinks to a point mass, meaning that we ignore the uncertainty in the latent posterior, we call this model a *GMM-quantised VAE*, or GMMQ-VAE.

Proposition 2: Under the assumptions made in [13], ELBO maximisation over the extended parameter set $\boldsymbol{\psi} = \{\boldsymbol{\theta}, \boldsymbol{\varphi}, \mathcal{Z}, \boldsymbol{e} \in \mathcal{Z}\}$ for the GMMQ-VAE has the same form as parameter estimation with the VQ-VAE objective in Eq. (11).

Proof sketch: From Eq. (8), the GMMQ-VAE ELBO is

$$\underline{\mathcal{L}}_{\text{GMMQ}}(\boldsymbol{\psi} \,|\, \underline{\boldsymbol{x}}) = -h(q_{\boldsymbol{Z}}) + \sum_{\boldsymbol{z}_q} \int q_{\boldsymbol{Z}}(\boldsymbol{z}; \boldsymbol{\varphi}, \boldsymbol{e}) \ln f_{\underline{\boldsymbol{X}}, \boldsymbol{Z}}(\underline{\boldsymbol{x}}, \boldsymbol{z}; \boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{z}_e, \quad (15)$$

where $h(\cdot)$ denotes the differential entropy. Since the entropy of $q_{\boldsymbol{Z}}$ is independent of $\boldsymbol{\psi}$ it has no effect on ELBO maximisation and can be ignored. If we then let $f(\cdot)$ approach a Dirac delta function $\delta(\cdot)$ – thus ignoring any uncertainty in the variational posterior by shrinking it to a point mass – the sum and integral both reduce to simple evaluation, and we obtain

$$\widehat{\boldsymbol{\psi}} = \operatorname*{argmax}_{\boldsymbol{\psi}} \lim_{f \to \delta} \underline{\mathcal{L}}_{\text{GMMQ}}(\boldsymbol{\psi} \,|\, \underline{\boldsymbol{x}}) \quad (16)$$

$$= \operatorname*{argmax}_{\boldsymbol{\psi}} \ln f_{\underline{\boldsymbol{X}}, \boldsymbol{Z}_e, \boldsymbol{Z}_q}(\underline{\boldsymbol{x}}, \boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi}), \boldsymbol{e}; \boldsymbol{\theta}) \quad (17)$$

$$= \operatorname*{argmax}_{\boldsymbol{\psi}} \left(\ln f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}_q}(\underline{\boldsymbol{x}} \,|\, \boldsymbol{e}; \boldsymbol{\theta}) + \ln f_{\boldsymbol{Z}_e | \boldsymbol{Z}_q}(\boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi}) \,|\, \boldsymbol{e}) \right), \quad (18)$$

using Eq. (12) with $f_{\boldsymbol{Z}_q}$ uniform. For the optimisation over $\boldsymbol{e} \in \mathcal{Z}$ in $\boldsymbol{\psi}$, $f_{\boldsymbol{Z}_e | \boldsymbol{Z}_q}$ is unimodal isotropic, and thus maximised by the $\boldsymbol{e}$ closest to $\boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi})$. Also, for good autoencoders (i.e., near the global optimum of $\boldsymbol{\psi} \setminus \boldsymbol{e}$) we expect $f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}_q}(\underline{\boldsymbol{x}} \,|\, \boldsymbol{e}; \boldsymbol{\theta})$ to be greatest for the $\boldsymbol{e} \in \mathcal{Z}$ closest to $\boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi})$. This is essentially a less restrictive version of the VQ-VAE assumption $f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}_q}(\underline{\boldsymbol{x}} \,|\, \boldsymbol{z}; \boldsymbol{\theta}) \approx 0$ whenever $\boldsymbol{z} \neq \boldsymbol{z}_q$ [13]. The optimisation over $\boldsymbol{e}$ can then solved explicitly, with the optimum being

$$\boldsymbol{e}^\star = \boldsymbol{z}_q(\underline{\boldsymbol{x}}; \boldsymbol{\varphi}, \mathcal{Z}) \quad (19)$$

$$= \operatorname*{argmin}_{\boldsymbol{e} \in \mathcal{Z}} \left\| \boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi}) - \boldsymbol{e} \right\|_2^2, \quad (20)$$

the codebook vector closest to the encoder output $\boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi})$, as expected for a vector quantiser. Since $f_{\boldsymbol{Z}_e | \boldsymbol{Z}_q}$ is Gaussian with covariance matrix $\sigma^2 \boldsymbol{I}$, its log-probability reduces to the squared distance between the quantised and unquantised encoder output, plus a constant. We then arrive at

$$\left\{ \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}, \widehat{\mathcal{Z}} \right\} = \operatorname*{argmax}_{\boldsymbol{\theta}, \boldsymbol{\varphi}, \mathcal{Z}} \left(\ln f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}_q}(\underline{\boldsymbol{x}} \,|\, \boldsymbol{z}_q(\underline{\boldsymbol{x}}; \boldsymbol{\varphi}, \mathcal{Z}); \boldsymbol{\theta}) - \frac{1}{2\sigma^2} \left\| \boldsymbol{z}(\underline{\boldsymbol{x}}; \boldsymbol{\varphi}) - \boldsymbol{z}_q(\underline{\boldsymbol{x}}; \boldsymbol{\varphi}, \mathcal{Z}) \right\|_2^2 \right). \quad (21)$$

This expression is of the same form as Eq. (11), as desired. The variance σ^2 of the isotropic Gaussian acts as a weight between the two terms in the objective function, very similar to the hyperparameter β in regular VQ-VAEs.

Proposition 2 shows that the entire VQ-VAE objective function for $\beta = 1$ can be assigned a probabilistic interpretation as a regular VAE with a Gaussian mixture distribution in the latent space, specifically a GMMQ-VAE. The key twist is that $\underline{\boldsymbol{X}}$ depends on the discrete GMM component $\boldsymbol{z}_q$ instead of the continuous-valued, GMM-distributed encoder output $\boldsymbol{z}_e$ like in [83], [84]. This introduces quantisation into the encoder, distinguishing VQ-VAEs from the alternative of a simple, unquantised VAE with a GMM prior on $\boldsymbol{Z}$. We see that different weights on the squared-error term (which is closely related to changing β in Eq. (10)) correspond to different assumptions about the magnitude of the quantisation error.

Our derivation of Proposition 2 suggests a number of natural generalisations of GMMQ/VQ-VAEs, for example by adjusting and potentially learning any combination of the component prior probabilities $f_{\boldsymbol{Z}_q}$ and the component covariance matrices $\boldsymbol{\Sigma}_q$. These extensions are however beyond the scope of the current article, and will not be explored further here. Since the GMMQ-VAEs and VQ-VAEs are so closely related, we will henceforth concentrate on VQ-VAEs for simplicity.

B. A Variational Interpretation of Heuristic Control Learning

In this section, we show how discriminant condition codes [7]–[9], [11], [12] and sentence-level control vectors [10], which we collectively will refer to as the *heuristic approaches* or *poor man's latent variables*, can be connected to variational inference, autoencoders, and VQ-VAEs. We begin by noting that the heuristic approaches are merely different names for the same model-fitting framework, where the likelihood maximisation in Eq. (2) is replaced by a joint log-probability optimisation over both model parameters $\boldsymbol{\theta}$ and the per-sequence latent variables $\left\{ \boldsymbol{z}^{(n)} \right\}$. The resulting estimation problem over the entire training data $\mathcal{D}$ can be written

$$\left\{ \widehat{\boldsymbol{\theta}}_{\text{DCC}}(\mathcal{D}), \widehat{\boldsymbol{z}}^{(n)}_{\text{DCC}}(\mathcal{D}) \right\} = \operatorname*{argmax}_{\{\boldsymbol{\theta}, \boldsymbol{z}^{(n)}\}} \sum_{n=1}^{N} \ln f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}, \underline{\boldsymbol{L}}} \left(\underline{\boldsymbol{x}}^{(n)} \,\middle|\, \boldsymbol{z}^{(n)}, \underline{\boldsymbol{l}}^{(n)}; \boldsymbol{\theta} \right). \quad (22)$$

Proposition 3: The heuristic methods based on joint optimisation of latent inputs and model parameters equivalently be formulated encoder-decoder models, where the encoder for any $\boldsymbol{\theta}$ can be written

$$\widehat{\boldsymbol{z}}^{(n)}_{\text{DCC}}(\mathcal{D}, \boldsymbol{\theta}) = \operatorname*{argmax}_{\boldsymbol{z}} \ln f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}, \underline{\boldsymbol{L}}} \left(\underline{\boldsymbol{x}}^{(n)} \,\middle|\, \boldsymbol{z}, \underline{\boldsymbol{l}}^{(n)}; \boldsymbol{\theta} \right). \quad (23)$$

Proof sketch: Consider

$$\widehat{\boldsymbol{\theta}}_{\text{DCC}}(\mathcal{D}) = \operatorname*{argmax}_{\boldsymbol{\theta}} \max_{\{\boldsymbol{z}^{(n)}\}} \sum_{n=1}^{N} \ln f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}, \underline{\boldsymbol{L}}} \left(\underline{\boldsymbol{x}}^{(n)} \,\middle|\, \boldsymbol{z}^{(n)}, \underline{\boldsymbol{l}}^{(n)}; \boldsymbol{\theta} \right) \quad (24)$$

$$= \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_{n=1}^{N} \max_{\boldsymbol{z}^{(n)}} \ln f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}, \underline{\boldsymbol{L}}} \left(\underline{\boldsymbol{x}}^{(n)} \,\middle|\, \boldsymbol{z}^{(n)}, \underline{\boldsymbol{l}}^{(n)}; \boldsymbol{\theta} \right) \quad (25)$$

$$= \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_{n=1}^{N} \ln f_{\underline{\boldsymbol{X}} | \boldsymbol{Z}, \underline{\boldsymbol{L}}} \left(\underline{\boldsymbol{x}}^{(n)} \,\middle|\, \widehat{\boldsymbol{z}}^{(n)}_{\text{DCC}}(\mathcal{D}, \boldsymbol{\theta}), \underline{\boldsymbol{l}}^{(n)}; \boldsymbol{\theta} \right) \quad (26)$$

where the last line follows from the observation that

$$\max_{\boldsymbol{z}} g(\boldsymbol{x},\, \boldsymbol{z}) = g(\boldsymbol{x},\, \operatorname*{argmax}_{\boldsymbol{z}} g(\boldsymbol{x},\, \boldsymbol{z})), \tag{27}$$

for any function $g(\cdot,\, \cdot)$.

From Proposition 3 we observe that the common heuristics for learning controllable speech synthesis from unannotated data can be seen as encoder-decoder models, where the encoder uses the same network as the decoder. This observation motivates our interest in comparing these heuristics to other encoder-decoder approaches. (The situation is however different from traditional autoencoders with *tied* weights, where the weight matrices in the decoder are transposes of those in the decoder.) Unlike VAEs, where encoding is performed via forward propagation through a second network, encoding here involves solving an optimisation problem through backpropagation. This is likely to be slow, but may give better performance (especially on test data) since each encoded variable solves an independent posterior-probability optimisation problem; there's no amortisation gap, unlike for VAEs [78]. In both VAEs and in the heuristic framework the encoder requires $\underline{\boldsymbol{x}}$ as well as $\underline{\boldsymbol{l}}$ as input, and thus cannot easily be applied in situations where natural speech acoustics are unavailable.

Different from the style-token encoder in [31], [32] and the speaker encoder in [34], the encoder here has access to the text-derived features of the spoken utterance. This is likely to promote encoder output that is more complementary to the text (reduced redundancy), but may or may not be more transferable between different text prompts. Interestingly, while recent Tacotron and VoiceLoop publications [31], [32], [85] have added explicit and distinct encoding networks similar to (VQ-)VAEs, previous work [12], [33] by these groups used backpropagation through the decoder as an implicit encoder, in the same way as the heuristic methods considered here.

Proposition 4: Increasing the heuristic objective function in Eq. (22) increases the evidence lower bound in Eq. (8). The encoder output can be seen as an approximate maximum a-posteriori estimate of the latent variable $\boldsymbol{Z}$ given $\underline{\boldsymbol{X}} = \underline{\boldsymbol{x}}$ and $\underline{\boldsymbol{L}} = \underline{\boldsymbol{l}}$.

Proof sketch: Note that the ELBO in Eq. (8) can be written

$$\begin{aligned}\underline{\mathcal{L}}\,(\boldsymbol{\theta},\, \boldsymbol{\varphi} \,|\, \underline{\boldsymbol{x}},\, \underline{\boldsymbol{l}}) &= \int q\,(\boldsymbol{z};\, \boldsymbol{\varphi}) \ln \frac{f_{\underline{\boldsymbol{X}},\, \boldsymbol{Z}|\underline{\boldsymbol{L}}}\,(\underline{\boldsymbol{x}},\, \boldsymbol{z} \,|\, \underline{\boldsymbol{l}};\, \boldsymbol{\theta})}{q\,(\boldsymbol{z};\, \boldsymbol{\varphi})}\, \mathrm{d}\boldsymbol{z} \\ &= \int q\,(\boldsymbol{z};\, \boldsymbol{\varphi}) \ln f_{\underline{\boldsymbol{X}},\, \boldsymbol{Z}|\underline{\boldsymbol{L}}}\,(\underline{\boldsymbol{x}},\, \boldsymbol{z} \,|\, \underline{\boldsymbol{l}};\, \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{z} - h(q), \end{aligned} \tag{28}$$

where $h(q)$ is the differential entropy of $q\,(\boldsymbol{z};\, \boldsymbol{\varphi})$. Consider choosing the q-distribution from a family which is parameterised by location $\boldsymbol{\mu}$ only, meaning that $\boldsymbol{\varphi} = \boldsymbol{\mu}$ and

$$q\,(\boldsymbol{z};\, \boldsymbol{\mu}) \to q\,(\boldsymbol{z} - \boldsymbol{\mu})\,. \tag{29}$$

This makes $h(q\,(\boldsymbol{z};\, \boldsymbol{\mu}))$ independent of $\boldsymbol{\mu}$, and we get

$$\begin{aligned}&\widehat{\boldsymbol{\mu}}\,(\underline{\boldsymbol{x}},\, \underline{\boldsymbol{l}},\, \boldsymbol{\theta}) \\ &= \operatorname*{argmax}_{\boldsymbol{\mu}} \underline{\mathcal{L}}\,(\boldsymbol{\theta},\, \boldsymbol{\mu} \,|\, \underline{\boldsymbol{x}},\, \underline{\boldsymbol{l}}) \end{aligned} \tag{30}$$

$$= \operatorname*{argmax}_{\boldsymbol{\mu}} \int q\,(\boldsymbol{z};\, \boldsymbol{\mu}) \ln f_{\underline{\boldsymbol{X}},\, \boldsymbol{Z}|\underline{\boldsymbol{L}}}\,(\underline{\boldsymbol{x}},\, \boldsymbol{z} \,|\, \underline{\boldsymbol{l}};\, \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{z}. \tag{31}$$

If the shape of the q-distribution(s) is made increasingly narrow (by making the variance tend to zero) so that it approaches a Dirac delta function $\delta(\cdot)$ we obtain

$$\begin{aligned}&\lim_{q \to \delta} \widehat{\boldsymbol{\mu}}\,(\underline{\boldsymbol{x}},\, \underline{\boldsymbol{l}},\, \boldsymbol{\theta}) \\ &= \operatorname*{argmax}_{\boldsymbol{\mu}} \int \delta\,(\boldsymbol{z} - \boldsymbol{\mu}) \ln f_{\underline{\boldsymbol{X}},\, \boldsymbol{Z}|\underline{\boldsymbol{L}}}\,(\underline{\boldsymbol{x}},\, \boldsymbol{z} \,|\, \underline{\boldsymbol{l}};\, \boldsymbol{\theta})\, \mathrm{d}\boldsymbol{z} \end{aligned} \tag{32}$$

$$= \operatorname*{argmax}_{\boldsymbol{\mu}} \ln f_{\underline{\boldsymbol{X}},\, \boldsymbol{Z}|\underline{\boldsymbol{L}}}\,(\underline{\boldsymbol{x}},\, \boldsymbol{\mu} \,|\, \underline{\boldsymbol{l}};\, \boldsymbol{\theta}) \tag{33}$$

$$= \operatorname*{argmax}_{\boldsymbol{\mu}} \ln \Big(f_{\underline{\boldsymbol{X}}|\boldsymbol{Z},\, \underline{\boldsymbol{L}}}\,(\underline{\boldsymbol{x}} \,|\, \boldsymbol{\mu},\, \underline{\boldsymbol{l}};\, \boldsymbol{\theta}) \cdot f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}}\,(\boldsymbol{\mu} \,|\, \underline{\boldsymbol{l}};\, \boldsymbol{\theta}) \Big) \tag{34}$$

$$= \operatorname*{argmax}_{\boldsymbol{\mu}} \ln f_{\underline{\boldsymbol{X}}|\boldsymbol{Z},\, \underline{\boldsymbol{L}}}\,(\underline{\boldsymbol{x}} \,|\, \boldsymbol{\mu},\, \underline{\boldsymbol{l}};\, \boldsymbol{\theta})\,, \tag{35}$$

where the last line assumes that $f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}}$ is constant. By applying these approximations to each training datapoint independently one obtains Eq. (22).

In summary, we have shown that the heuristic objective in Eq. (22) can be derived from variational principles assuming:

1) That the prior distribution $f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}}$ is flat (constant) across the range of $\boldsymbol{z}$- and $\underline{\boldsymbol{l}}$-values considered.
2) We use a Dirac delta function (a spike) to represent all latent posterior distributions.

Both assumptions are directly analogous to assumptions made in the probabilistic derivation of VQ-VAEs in Proposition 2: VQ-VAEs use a uniform prior over codebook vectors and do not represent any uncertainty in the (encoded) latents. This is another motivation for us to compare the heuristic approach to the largely similar functionality offered by VQ-VAEs. The second assumption explains the nickname "poor man's latent variables", since we see that the heuristic objective does not afford any representation of uncertainty in the latent space.

If the listed assumptions are violated, the variational approximation need not produce a maximum of the true likelihood, though the agreement between the two methods is likely to be greater the more accurate the two assumptions are. Unlike the EM-based derivation in [14], the derivation presented here establishes that any simultaneous modification of that increases Eq. (22) also increases the likelihood lower bound; it is not necessary to perform interleaved optimisation as in the EM-algorithm [37].

While $\underline{\mathcal{L}}$ diverges to minus infinity as $q \to \delta$, and thus does not provide a reasonable numeric lower bound on the likelihood, it is still true that relative differences is $\underline{\mathcal{L}}$-are meaningful and can be mapped to similar changes in the lower bound (consider subtracting one ELBO from another). A similar observation applies to the numerical value of the VQ-VAE objective derived in Proposition 2.

The domain of the optimisation over $\boldsymbol{z}^{(n)}$ in Eq. (22) can also be given a statistical interpretation. Define a binary prior $f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}}$, which is constant and nonzero on feasible $\boldsymbol{z}$-values, but equals zero (so that $\ln f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}} = -\infty$) outside the domain of optimisation. Unconstrained ELBO maximisation with this prior will then only find possible optimal parameters in the feasible set defined by the constraints. Constrained optimisation in the latent space is thus interpretable as normal variational parameter estimation under a particular prior on $\boldsymbol{Z}$.

To summarise, the key similarities between VQ-VAEs and the heuristic approach are:

- Both VQ-VAEs and the heuristic approach can be viewed as autoencoders.
- Both methods are closely related to variational approaches with a flat prior over the permissible z-values.
- Neither approach represents uncertainty in the latent-variable inference (the encoder output value).

The main differences, meanwhile, are:

- The heuristic approach does not quantise latent vectors.
- The heuristic approach uses a single network for both encoding and decoding, with an optimisation operation instead of forward propagation through a separate encoder. In other words, it does not amortise inference.

C. Using Prior Information in Control Learning

It is worth noting that the variational interpretation of the heuristic method requires that a flat, noninformative prior is used. In Bayesian statistics, priors like $f_{\boldsymbol{Z}|\underline{\boldsymbol{L}}}$ can be adjusted by practitioners based on side information about what z-value to expect for any given datapoint. With a fixed prior, this opportunity goes away.

There are, however, other methods for potentially biasing learning based on side information. In particular, since speech synthesisers are trained by local refinements of a previous parameter estimate and the parameter set includes explicit estimates of the latent encodings, the system can be initialised based on an informed guess about appropriate latent-variable values. We compare this strategy against random initalisation in the experiments in Sec. V-D. A finding that these two schemes do not differ in behaviour would indicate that learning is robust to initialisation. The opposite finding would suggest a more brittle learning process, but also one with room to straightforwardly inject prior information into the learning.

V. Experiments

Following the theoretical developments in the previous section, we now investigate the practical performance of different methods for unsupervised learning of control in an example application to acoustic modelling of emotional speech, using a corpus described in Sec. V-A. The systems and baselines considered are introduced in Sec. V-B, and their training presented in Sec. V-C. The results of training and the associated learned latent representations are evaluated objectively in Sec. V-D. Sec. V-E then details the subjective listening test performed, along with its analysis and resulting findings. Wherever possible, the experiments have been designed to be as similar as possible to the experiments with supervised speech-synthesis control in [63], which used the same data.

A. Data and Preprocessing

For the experiments in this paper, we decided to use the large database of studio-recorded, high-quality acted emotional speech from [63]. (An earlier subset of this database was used for the research in [14].) The database contains recordings of isolated utterances in Japanese, read aloud by a female voice talent who is a native speaker of Japanese. Each prompt text was chosen to not harbour any inherent emotion, but was spoken in one or more of seven different emotional styles: emotionally-neutral speech as well as the three pairs happy vs. sad, calm vs. insecure, and excited vs. angry. This means that the database contains speech variation of communicative importance that cannot be predicted from the text alone. 1200 utterances (133–158 min) were recorded for each emotion, for a total of 8400 utterances and nearly 17 hours of audio (beginning and ending silences included), all recorded at 48 kHz. The talker was instructed to keep their expression of each emotion constant throughout the recordings.

Each audio recording in the data is annotated with the text prompt (in kanji and kana) as well as the prompted emotion. Lorenzo-Trueba et al. [63] considered a number of different methods for encoding this emotional information for speech synthesiser control, while also leveraging information on listener perception of the different emotions. They found the best-performing encoding of emotional categories to be based on listener responses to emotional speech (confusion-matrix columns) rather than one-hot categorical vectors. Relabelling the data based on listener perception of individual utterances did not improve performance. In contrast to this previous work, we will treat the emotional content as a latent source of variation, to be discovered and described by the different unsupervised methods we are investigating.

To simplify comparison, we used the same partitioning, preprocessing, and forced alignment of the database as Lorentzo-Trueba et al. [63]. In particular 10% of the data were used for validation and 10% for testing, with these held-out sets only incorporating sentences where annotators' perceived emotional categories agreed with the prompted emotion. We also used the exact same linguistic and acoustic features as those extracted in [63]. In particular, Open JTalk [86] was used to extract 389 linguistic features while WORLD [75], [87] was used for acoustic analysis and signal synthesis. The analysis produced a total of 259 acoustic features at 5 ms intervals. The features comprised linearly interpolated log pitch estimated using SWIPE [88], 60 mel-cepstrum features (MCEPs, with frequency warping 0.77 to approximate the Bark scale), and 25 band-aperiodicity coefficients (BAPs) based on critical bands. Each of these had static, delta, and delta-delta coefficients. These continuous-valued features were all normalised to zero mean and unit variance, and subsequently complemented with a binary voiced/unvoiced flag.

Linguistic and acoustic features were forced-aligned with five-state left-to-right no-skip HMMs trained with HTS [89], given access to the prompted emotion as an additional decision-tree feature. These HMMs were also used for duration prediction during synthesis, which was identical for all models; only different approaches to acoustic modelling (trained with or without emotional labels) were compared in the experiments. At synthesis time, predicted static and dynamic features were reconciled through most likely parameter generation (MLPG) [90] and enhanced using the postfilter described in [91] with coefficient 0.2.

B. Systems

To investigate how supervised and unsupervised approaches for learning acoustic-model control behave on data with im-

portant non-textual variation (specifically emotion), we considered eight different sources of speech stimuli, or *systems*, of three different kinds: stimuli based on natural speech (functioning as toplines), systems with only supervised learning (functioning as baselines for comparisons), and systems capable of learning output control from unannotated variation. In brief, the eight systems were defined as follows:

- **NAT:** Natural speech from the held-out test-set.
- **VOC:** Natural speech from the held-out test-set, subjected to analysis synthesis as described in Sec. V-A.
- **SUP:** A supervised approach to controllable speech synthesis, trained and evaluated with labels derived from the ground-truth prompted emotion as input. Specifically, this system is equivalent to the best setup with emotional strength from [63], since the approaches based on unannotated data presumably can learn to moderate emotional strength as well. The only difference from [63] is that the system was optimised using Adam [92] rather than stochastic gradient descent.
- **BOT:** A bottom-line system, same as SUP but with no control input, only linguistic features $\underline{\boldsymbol{l}}$. This system cannot accommodate the differences between the different emotions in the database and provides a bottom line in terms of prediction performance.
- **VQS:** A VQ-VAE with the same ('S') number of hidden nodes and layer order in the encoder as in the decoder.
- **VQR:** A VQ-VAE with the same number of hidden nodes and but reverse ('R') layer order in the encoder compared to the decoder.
- **HZI:** Poor man's latent variables with latent-space control vectors initialised with all zeros ('ZI').
- **HSI:** Poor man's latent variables with supervised initialisation ('SI') of latent-space control vectors. This gives an idea of the impact of using prior information in initialisation, as discussed in Sec. IV-C.

All synthesisers used the same duration model and duration predictions as the experiments in [63]; only the acoustic models differed. They also used exact same decoder structure, identical to the one used in [14], [63], [93] (among others). Based on the proposal in [94], it contains two 256-unit feedforward layers with logistic sigmoid nonlinearities, followed by two 128-unit BLSTM layers and a linear output layer. The neural networks were implemented in CURRENNT [95].

Based on our observation in Prop. 3 in Sec. IV-B – that the heuristic methods can be interpreted as encoder-decoder models that use the same network for both encoding and decoding – we made the VQ-VAE encoders in the experiments have the same internal structure (hidden layers and unit counts) as the decoder. There is, however, some ambiguity as for how to order the hidden layers in the encoder: the encoder is a function $\boldsymbol{z}_q\left(\underline{\boldsymbol{x}}, \underline{\boldsymbol{l}}\right)$ while the decoder is a function $\underline{\boldsymbol{x}}\left(\boldsymbol{z}_q, \underline{\boldsymbol{l}}\right)$. An argument based on $\boldsymbol{z}_q$ or $\underline{\boldsymbol{x}}$ suggests that the order of the feedforward and recurrent layers be swapped in the encoder compared to the decoder, placing the recurrent layers closer to the input side of the encoder (as in system VQR), while a reference to $\underline{\boldsymbol{l}}$ suggests that the layer order should not be altered between encoder and decoder (as in system VQS).

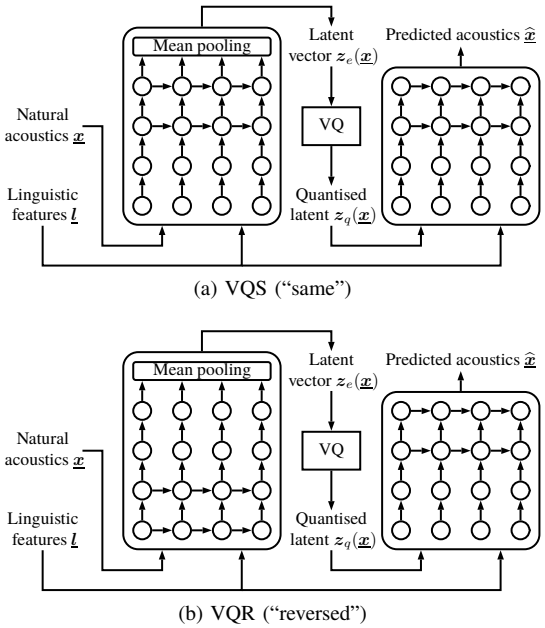
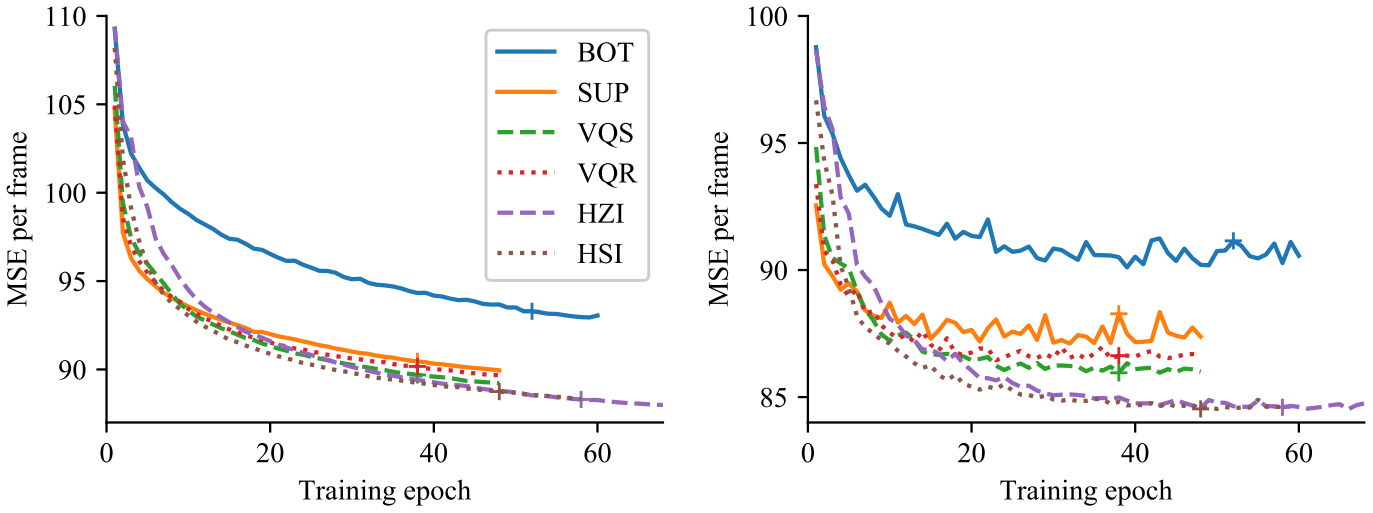


Figure 2. VQ-VAE schematics showing the two different encoder structures.

The situation is illustrated in Fig. 2. For completeness, both topologies were considered in the experiments. In either case, the final per-sentence encoding vector $\boldsymbol{z}_e$ was extracted from a mean-pooling layer across all timesteps, similar to how the backpropagated gradients for the latent control vectors sum across frames in the heuristic approach.

Prior to training, all networks were initialised with small random weights based on Glorot & Bengio [96]. The autoencoder-based approaches in this study also require that the latent representations (the per-sentence control vectors or the codebook) be initialised as well. We set the control-vector dimensionality D to 8 throughout the experiments, the same value as in [63] (based on 7 emotions plus a scalar emotional strength). The latent control vector elements for HZI and HSI were then initialised deterministically (either all zeros, or with the same values as for as SUP, also on the validation and test sets). For the VQ-VAEs the codebook size was set to 1344 and the codebook vectors were initialised with small random values as part of neural network initialisation. The size of the codebook was chosen to be the same as the maximum number of distinct emotional-category encodings used by SUP on the training set [63], computed as 192 35-utterance mini-batches with 7 emotions in each. It is good practice to use a larger VQ codebook than might be necessary, since some codebook vectors are likely to end up in regions that the encoder does not visit, yielding "dead" vectors that are neither trained or used; with too few vectors, the presence of local optima means that not all control modes or nuances may be learned.

In purely objective terms, we may expect the unsupervised approaches to achieve a better fit to the training data than the supervised method, since the former can tailor their output to each individual utterance in the corpus. The heuristic methods are furthermore likely to give better objective prediction accuracy than VQ-VAEs, due to the amortisation gap and the VQ-VAE restriction to a discrete set of latent-space values.



(a) Training set (b) Test set

Figure 3. Training curves for different systems. Note the different scales on the y-axes. Plus signs indicate the best epoch on the validation set.

Table I
OBJECTIVE RESULTS OF SYSTEM TRAINING.

			MSE per frame		
System	#NN weights	Best epoch	Train	Val.	Test
BOT	1.58M	52	93.3	105.1	91.1
SUP	1.58M	38	90.5	101.3	88.3
VQS	3.24M	38	89.7	100.2	86.0
VQR	3.18M	38	90.2	100.7	86.6
HZI	1.58M	58	88.3	98.9	84.6
HSI	1.58M	48	88.8	98.9	84.5

Subjectively, however, SUP will be hard to beat, since it is trained using supervised knowledge to explicitly control the perceptually most relevant variation in the data.

C. Training

All mathematical approaches considered in this work are probabilistic methods that operate on the principle of likelihood maximisation. For this experiment, we assume that the conditional output distribution $\underline{\boldsymbol{X}}(\underline{\boldsymbol{l}}, \boldsymbol{z})$ (or $\underline{\boldsymbol{X}}(\underline{\boldsymbol{l}})$ for BOT) is an isotropic Gaussian with fixed variance. Log-likelihood maximisation is then mathematically equivalent to (mean) squared-error (MSE) minimisation. The MSE is a common loss function in synthesiser training, used for instance in Tacotron 1 and 2 [30], [76]. In our case each extracted acoustic feature is normalised to unit variance prior to neural network training (see [63]), so our setup altogether corresponds to an assumption that the speech-feature outputs are Gaussian, uncorrelated, and that each feature-vector element has a standard deviation proportional to the global standard deviation of that feature on the training set; the network outputs, in turn, can also be interpreted probabilistically as estimated conditional Gaussian means. It was seen in [97] that the use of such a globally-constant covariance matrix did not significantly affect synthesis quality compared to the alternative of letting the variance depend on linguistic context.

Encoder and decoder parameters (including the VQ codebook) were trained to minimise per-frame MSE using Adam [92] with default hyperparameter values. However, since each per-utterance control-vector input for the heuristic systems HZI and HSI only is updated once per epoch, these $\boldsymbol{z}$-vectors may not be a good fit for the per-parameter moment estimates that Adam maintains. The control vectors were therefore instead updated using stochastic gradient descent (SGD) with a fixed learning rate $2 \cdot 10^{-4}$, the same rate as used for the latent vectors in [14].[5] The HZI and HSI control-vector inputs for validation and test utterances were updated similarly using the corresponding synthesis network from each epoch, but without modifying the network weights on these utterances (cf. [10]). In an encoder-decoder view, this maximisation performed by SGD on training, validation, and test data is an instantiation of the encoder in Eq. (23).

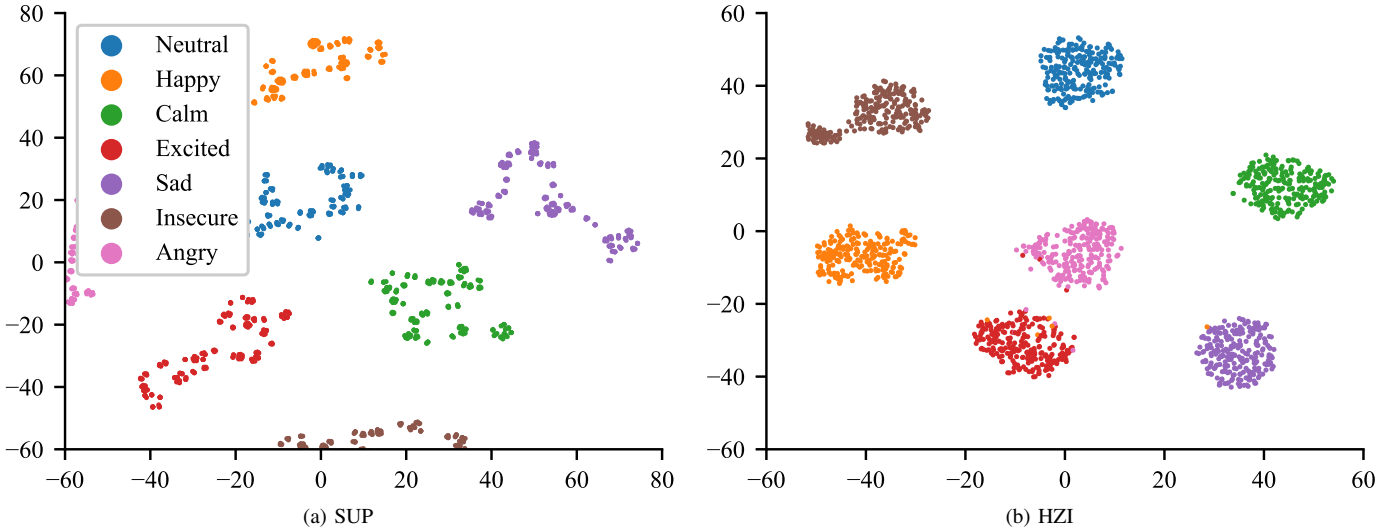
Training was run until the validation-set MSE failed to improve for ten consecutive epochs (or eight in the case of BOT), whereafter the network with the lowest validation-set error was returned. In the present experiment, this scheme required at most 68 epochs for termination.

D. Objective Evaluation

1) Evaluation of Training: Fig. 3 presents learning curves from the synthetic systems in Sec. V-B, chronicling the evolution of per-frame mean-squared error on training and test-set data for each epoch of optimisation. The number of iterations until termination and final performance numbers on all three data partitions are listed in Table I, along with the number of neural network weights used by CURRENT for each system.

Looking at Table I, a handful of general trends become evident. To begin with, validation set numbers are consistently inferior to both training and test set numbers; this appears to be a consequence of the data partitioning in [63], and recurs in other systems trained on this data split. The most notable difference between the methods is that all schemes with control achieved better MSE performance than the emotionally-

[5] Paper [14] contains a typo where $0.2 \cdot 10^{-3}$ is incorrectly listed as $2 \cdot 10^{-3}$.



(a) SUP (b) HZI

Figure 4. 2D t-SNE embeddings of latent control vectors $\boldsymbol{z}$, coloured by the prompted emotion. Scale and rotation are arbitrary.

unaware bottom line BOT by at least 3.0 on all data partitions. This is entirely expected, since only BOT is unable to adjust its output based on the emotional content of the speech. The fact that methods with learned control inputs slightly outdo SUP is not surprising either, since they had access to the natural ground-truth acoustics for each test-set utterance as a decoder input. These numbers do not imply that the resulting systems achieve subjectively better quality or emotional control.

The heuristic systems required more epochs than most other systems to terminate training, but also achieved lower per-frame MSE than VQS and VQR by at least 1.4 on the test set. This difference is likely due to the amortisation gap [77], since the VQ-VAEs use learned inference while the heuristic systems use direct per-utterance optimisation. The use of SGD rather than Adam for updating the latent-variable values of each utterance might explain the slower convergence rate and longer training seen in Fig. 3 for the heuristic systems.

As a side note, an earlier version of our VQ-VAE encoder extracted the final state on the LSTM (in each direction) and mapped these to the latent space through a linear output layer; such a design is perhaps more traditional in encoder-decoder models, and resembles the one used in [32]. However, VQ-VAEs with this encoder design did not perform much differently from BOT. It seems that relevant information from mid-utterance acoustics did not propagate well to the end states, resulting in encoder output of little predictive value. Without emotional information (from label or acoustics), the resulting network is then essentially a version of BOT. Once the choice to extract the end state of the LSTM was replaced by a mean pooling operation, performance improved to the levels seen in Table I.[6]

[6]As an alternative, the work in [31] chose used the final state of a unidirectional RNN as the encoder output, but since their encoder contained several strided convolutions, the training sequences were effectively downsampled such that the RNN had to run over less than ten timesteps. Similar to our mean pooling, this allowed the encoder to better incorporate information from the entire utterance, but their setup is more likely to retain some order information of relevance to the intonation patterns they studied.

2) Evaluation of Learned Latent Vectors: While the low MSE achieved by the encoder-decoder models in Table I are encouraging, it does not follow that the trained systems must have learned to represent and control emotion specifically. To investigate this, we performed objective analyses on the learned latent representations. For the heuristic systems, we used t-distributed stochastic neighbour embedding (t-SNE) [98] to reduce dimensionality and visualise the latent-space vectors in two dimensions. The results for HZI can be seen in Fig. 4b, and can be compared against a similar embedding of the SUP control vectors in Fig. 4a. It is clear that the different emotions are grouped into well-defined clusters with minimal overlap. The degree of separation can be quantified by looking at how frequently the nearest neighbour of an utterance vector in the latent space is from a different prompted emotion. Across the 1680 latent vectors in the test set, this happened 18 times for HZI and 7 times for HSI. If we measure how many times at least one of the five nearest neighbours is from a different emotion, the numbers rise to 41 for HZI and 21 for HSI. (For SUP, the corresponding number is 0.) All in all, this indicates that the heuristic approach has been highly successful at identifying the different base emotions in the database and then separating them in the latent space.

While exhibiting faster convergence, supervised initialisation (HSI) did not seem to confer any lasting benefit over the purely unsupervised approach HZI initialised with all zeros. This suggests that latent vectors learned through standard heuristics are robust against differences in initialisation.

For the systems based on VQ-VAE we performed a clustering analysis on the 1680 quantised latent vectors $\boldsymbol{z}_q$ from the test set. The results are provided in Table II. We see that most vectors in the codebooks were not used at all (at most 61 vectors out of 1344 were used), so a parsimonious discrete representation was learned despite starting from a very large codebook. Of the vectors that did see use on the test set, each emotion only used a subset of these (first group of numbers in the table). Standard measures of clustering quality like

Table II
ANALYSIS OF QUANTISED LATENT VECTORS IN VQ-VAE SYSTEMS.

System	VQ indices used min / mean / max	Emotion entropy min / mean / max	Total indices	Purity (frac)	NMI (bits)
VQS	2 / 11.7 / 33	0.19 / 2.03 / 3.98	61	0.96	0.17
VQR	1 / 5.7 / 13	0 / 1.24 / 2.71	29	0.98	0.10

Table III
MEAN OPINION SCORES FOR QUALITY AND EMOTIONAL STRENGTH.

System	Quality		Emotional strength	
	Per utt.	Per emo.	Per utt.	Per emo.
NAT	4.01	-	3.38	-
VOC	2.94	-	3.18	-
SUP	3.41	-	2.94	-
VQS	3.42	3.51	2.92	2.99
VQR	3.41	3.50	2.89	2.97
HZI	3.43	3.53	2.89	2.99
HSI	3.44	3.54	2.86	2.98

purity and normalised mutual information (NMI) [99, Ch. 16] indicate that the prompted emotions were very well separated by the VQ-VAE. Beyond the emotion, there is relatively little information in the encoded latent vectors, as shown by the low per-emotion entropies (second set of numbers in the table). This suggests that the talker's emotional expression might be quite consistent across the database, precisely as intended during recording, and does not leave much room for the encoded vectors $\boldsymbol{z}_q$ to pick up additional nuances in emotional expression. While VQR seems to yield smaller and more well-defined clusters than VQS, the differences are marginal and unlikely to have substantial impact on the synthesis.

In summary, we find that the unsupervised methods very successfully identified the emotional classes in held-out speech data on our task, despite not having access to explicit emotional annotation. This confirms that these methods are capable of identifying and representing salient, unannotated variation in the data, just like the unsupervised style tokens in [32].

E. Subjective Evaluation

Reduced objective error does not necessarily imply a perceptually better system. In fact, the true minimiser of the MSE objective we use is the conditional mean of $\underline{\boldsymbol{X}}$. This mean was estimated directly from repeated speech in [2] and found to be perceptually inferior to random sampling in highly accurate models. In order not to be led astray by the objective performance, we complemented our observations above with a crowdsourced subjective listening test similar to those in [63].

1) Listening Test Design: For the listening test, the BOT system was excluded, as it is incapable of control. Each of the four unsupervised systems, however, was represented twice: once synthesising from control vectors derived from encoding the ground-truth held-out test sentences (the normal autoencoder approach), and once with the latent input to the encoder always set equal to the mean latent vector $\overline{\boldsymbol{z}}$ for the relevant emotion across the entire training set. While the former control scheme varies the control input $\boldsymbol{z}$ from utterance to utterance, the latter holds $\boldsymbol{z}$ constant for each emotion, wherefore we refer to these schemes as *per-utterance* and *per-emotion* control, respectively.

Our per-utterance control may in principle be able to reproduce nuances in the emotional expression of each test utterance, but requires access to the held-out test-set acoustics to do so. Per-emotion control is derived from emotional labels on the training data (instead of using test-set acoustics), but any systematic variation in perceived emotional strength across utterances must then be attributed to the text input alone. Together, the two control schemes can be used to assess the systems' abilities to replicate nuances in emotional expression on the test set. Many other control schemes are also possible, but studying them is left as future work.

A system paired with a control scheme will be termed a *condition*, of which we investigated a total of 11: NAT, VOC, SUP, and two each (for the two control schemes) for the unsupervised systems VQS, VQR, HZI, and HSI. Each of the 1680 utterances in the test set (240 per emotion) can then be realised in any condition, producing a *stimulus* waveform.

Our subjective evaluation recruited native Japanese listeners through CrowdWorks$^{\text{LTD}}$ to evaluate sets of 22 randomly-selected stimuli through a web-based interface. The sets were constrained such that all stimuli were unique and each condition appeared exactly twice in each set. No listener was permitted to evaluate more than 10 sets.

Evaluators processed the stimuli in the set in sequence. For each stimulus, they were asked to supply three pieces of information: i) perceived speech quality (traditional MOS scale of integers "1 – bad" through "5 – excellent"); ii) perceived emotional category (response options being the seven emotions in the database plus "other"); and iii) perceived emotional strength (integer scale "1 – almost no emotion" through "5 – very emotional", or 6 for "no emotion"). Evaluators could listen to each stimulus as many times as desired before responding. In total, 700 response triplets were gathered for each emotion, from a total of 50 different listeners.

2) Evaluation of Synthesis Quality: The first set of columns in Table III shows the mean opinion scores (MOS) for speech quality for the different systems and control strategies investigated. To check if the differences were significant we applied two-sided Mann-Whitney U tests comparing all condition pairs, with Holm-Bonferroni correction [100] used to keep the familywise error rate below 5%. These tests found NAT and VOC to be significantly different from all other systems, as well as from each other. No other differences in quality were found to be statistically significant. t-tests (also with Holm-Bonferroni correction) gave the same conclusions. We thus observe that SPSS, while not achieving the same performance as natural speech, can achieve good output quality both through supervised as well as unsupervised control in this application. The difference between the best and the worst (SUP) synthesiser MOS is a mere 0.13 points on the five-point MOS scale. While there was evidence of a minor amortisation gap between VQ-VAEs and heuristic systems in terms of objective performance (i.e., MSE), this gap does not appear to have affected speech quality. Given that VQ-VAEs have advantages of being easier to train and allow straightforward latent-variable inference through amortisation, this makes them an appealing practical choice.

Table IV
FROBENIUS DISTANCES BETWEEN EMOTIONAL CONFUSION MATRICES. THE BEST UNSUPERVISED PERFORMANCE IN EACH COLUMN IS BOLDED.

	Per-utterance control			Per-emotion control		
System	vs. ID	vs. ref	vs. NAT	vs. ID	vs. ref	vs. NAT
NAT	0.50	1.04	0.00	-	-	-
VOC	0.68	1.26	0.37	-	-	-
SUP	0.71	1.51	0.69	-	-	-
VQS	0.63	1.39	**0.46**	**0.48**	**1.27**	**0.53**
VQR	**0.58**	**1.35**	0.51	0.65	1.44	0.70
HZI	0.60	1.39	0.53	0.59	1.37	0.55
HSI	0.64	1.42	0.52	0.62	1.42	0.63

3) Evaluation of Output Control: Our primary interest in this work is not synthesis quality but controllability. We therefore assessed the synthesisers' ability to reproduce the emotions in the database by studying the emotional classifications assigned by the listeners in the listening test. These classifications can be summarised through a confusion matrix, tabulating the distribution of listener classifications conditioned on the different prompted emotions. In the ideal case when all emotions are perceived as intended, this matrix should be the identity matrix. For completely natural speech there are nonetheless some confusions between emotions (as discussed in [63]), leading to some off-diagonal matrix structure.

Following the same methodology as in [63, Sec. 8.1.1], we computed emotional classification confusion matrices for each and every condition in the listening test (700 classifications per condition). These matrices were then compared against three different reference matrices: the ideal (identity matrix, 'ID') as well as two confusion matrices from natural speech, namely the one tabulated in [63, Table 5] ('ref') as well as the one computed from listener classifications of natural speech in the present listening test ('NAT'). Specifically, we computed the Frobenius norm of the difference between every confusion matrix and every reference matrix. Table IV presents the results of this comparison. A system that well separates and reproduces the different emotions should have low distance to the three references in the table.

While identifying statistically significant differences between confusion matrices is not a solved problem (see, e.g., [101]), we note that (with one single exception) NAT is better than all other conditions in all metrics; this agrees with our expectation that the recorded natural speech should perform at least as well as SPSS control schemes learned from the same data. On the other end of the spectrum, SUP is found to have greater distance to the reference matrices than all other conditions (again with a single exception). All other conditions exhibit broadly comparable numbers for each reference. Taken together, these patterns suggest that unsupervised approaches are at least as good (or better) than supervised learning of control in the present application, but that there is little difference between VQ-VAEs and the heuristic methods (and between different control schemes) in how reliably they reproduce the base emotions in the corpus.

As the controllable speech synthesisers considered in this work are capable of control inputs that differentiate more than just the seven base emotions, there is the possibility that they may learn to control other aspects of speech variability such as emotional nuance (cf. [14]), assuming such variation is present in the training data. This might be reflected in the emotional strength ratings, whose means are tabulated in the last two columns of Table III. (For this analysis, a response of "no emotion" was mapped to an emotional strength of zero.) Holm-Bonferroni corrected Mann-Whitney U tests between conditions (the same methodology used to analyse synthesis quality earlier) show that NAT and VOC perform similarly, and better than other conditions, which otherwise exhibit no significant differences. Thus the unsupervised approaches are again competitive with the supervised system.

No differences are evident between per-utterance and per-emotion control in this evaluation. This might not be too surprising, given the lack of diversity (only one or two bits of entropy) observed in Table II among control inputs in the same emotion class. Such a finding is consistent with expectations that the range of nuances within each emotion is quite limited in our speech corpus. It is possible that exaggerating the differences between utterance control inputs, as done in [14], would give more noticeable differences in expression within each emotion class.

To summarise, we have found that the unsupervised approaches under consideration are comparable to the supervised system also in terms of perceived speech quality, emotion recognition, and perceived emotional strength. Moreover, the different unsupervised systems and control schemes appear essentially perceptually equivalent in our evaluation.

VI. CONCLUSION

This paper has studied the theory and practice of unsupervised learning of output control in statistical text to speech. On the theory side, we have established novel connections between traditional unsupervised heuristics from speech-technology, like DCC and sentence-level control vectors, and variational latent-variable inference in autoencoder models. We have likewise connected the heuristics to VQ-VAEs, which we have shown have a similar interpretation as variational inference neglecting uncertainty in a Gaussian mixture model.

In terms of empirical insights, we have compared supervised and unsupervised methods for learning controllable acoustic models on a large corpus of emotional speech. The objective and subjective results show that the unsupervised methods successfully learn and reproduce the emotional classes in the speech data and often outperform a competitive supervised baseline. This bodes well for unsupervised learning for enabling output control in speech synthesis at large. Methods incorporating amortised inference stand out as particularly appealing for future applications, since they achieve similar performance as the established heuristics but enable easier training and latent-variable inference.

REFERENCES

[1] S. Van Kuyk, W. B. Kleijn, and R. C. Hendriks, "On the information rate of speech communication," in *Proc. ICASSP*, 2017, pp. 5625–5629.
[2] G. E. Henter, T. Merritt, M. Shannon, C. Mayo, and S. King, "Measuring the perceptual effects of modelling assumptions in speech synthesis using stimuli constructed from repeated natural speech," in *Proc. Interspeech*, 2014, pp. 1504–1508.

[3] B. Uria, I. Murray, S. Renals, C. Valentini-Botinhao, and J. Bridle, "Modelling acoustic feature dependencies with artificial neural networks: Trajectory-RNADE," in *Proc. ICASSP*, 2015, pp. 4465–4469.

[4] Y. Fan, Y. Qian, F. K. Soong, and L. He, "Multi-speaker modeling and speaker adaptation for DNN-based TTS synthesis," in *Proc. ICASSP*, 2015, pp. 4475–4479.

[5] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," *arXiv preprint 1609.03499*, 2016.

[6] B. Li and H. Zen, "Multi-language multi-speaker acoustic modeling for LSTM-RNN based statistical parametric speech synthesis," in *Proc. Interspeech*, 2016, pp. 2468–2472.

[7] H.-T. Luong, S. Takaki, G. E. Henter, and J. Yamagishi, "Adapting and controlling DNN-based speech synthesis using input codes," in *Proc. ICASSP*, 2017, pp. 4905–4909.

[8] O. Abdel-Hamid and H. Jiang, "Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code," in *Proc. ICASSP*, 2013, pp. 7942–7946.

[9] S. Xue, O. Abdel-Hamid, H. Jiang, L.-R. Dai, and Q. Liu, "Fast adaptation of deep neural network based on discriminant codes for speech recognition," *IEEE/ACM T. Audio Speech*, vol. 22, no. 12, pp. 1713–1725, 2014.

[10] O. Watts, Z. Wu, and S. King, "Sentence-level control vectors for deep neural network speech synthesis," in *Proc. Interspeech*, 2015, pp. 2217–2221.

[11] S. Ö. Arık, G. Diamos, A. Gibiansky, J. Miller, K. Peng, W. Ping, J. Raiman, and Y. Zhou, "Deep Voice 2: Multi-speaker neural text-to-speech," in *Proc. NIPS*, 2017, pp. 2962–2970.

[12] Y. Taigman, L. Wolf, A. Polyak, and E. Nachmani, "VoiceLoop: Voice fitting and synthesis via a phonological loop," in *Proc. ICLR*, 2018.

[13] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Proc. NIPS*, 2017, pp. 6309–6318.

[14] G. E. Henter, J. Lorenzo-Trueba, X. Wang, and J. Yamagishi, "Principles for learning controllable TTS from annotated and latent variation," in *Proc. Interspeech*, 2017, pp. 3956–3960.

[15] D. H. Klatt, "Review of text-to-speech conversion for English," *The Journal of the Acoustical Society of America*, vol. 82, no. 3, pp. 737–793, 1987.

[16] H. Zen, K. Tokuda, and A. W. Black, "Statistical parametric speech synthesis," *Speech Commun.*, vol. 51, no. 11, pp. 1039–1064, 2009.

[17] S. King, "An introduction to statistical parametric speech synthesis," *Sadhana*, vol. 36, no. 5, pp. 837–852, 2011.

[18] J. R. Quinlan, "Improved use of continuous attributes in C4.5," *J. Artif. Intel. Res.*, vol. 4, pp. 77–90, 1996.

[19] K. Fujinaga, M. Nakai, H. Shimodaira, and S. Sagayama, "Multiple-regression hidden Markov model," in *Proc. ICASSP*, 2001, pp. 513–516.

[20] T. Masuko, T. Kobayashi, and K. Miyanaga, "A style control technique for HMM-based speech synthesis," in *Proc. Interspeech*, 2004, pp. 1437–1439.

[21] T. Nose, Y. Kato, and T. Kobayashi, "Style estimation of speech based on multiple regression hidden semi-Markov model," in *Proc. Interspeech*, 2007, pp. 2285–2288.

[22] Z.-H. Ling, K. Richmond, and J. Yamagishi, "Articulatory control of HMM-based parametric speech synthesis using feature-space-switched multiple regression," *IEEE T. Audio Speech*, vol. 21, no. 1, pp. 207–219, 2013.

[23] I. Jauk, "Unsupervised learning for expressive speech synthesis," Ph.D. dissertation, Polytechnic University of Catalonia, Barcelona, Spain, Jun 2017.

[24] M. J. F. Gales, "Cluster adaptive training of hidden Markov models," *IEEE T. Speech Audi. P.*, vol. 8, no. 4, pp. 417–428, 2000.

[25] L. Chen, M. J. F. Gales, V. Wan, J. Latorre, and M. Akamine, "Exploring rich expressive information from audiobook data using cluster adaptive training," in *Proc. Interspeech*, 2012, pp. 959–962.

[26] S. King and V. Karaiskos, "The Blizzard Challenge 2016," in *Proc. Blizzard Challenge Workshop*, 2016.

[27] K. Sawada, K. Hashimoto, K. Oura, and K. Tokuda, "The NITech text-to-speech system for the Blizzard Challenge 2017," in *Proc. Blizzard Challenge Workshop*, 2017.

[28] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. ICML*, 2014, pp. 1188–1196.

[29] S. King, L. Wihlborg, and W. Guo, "The Blizzard Challenge 2017," in *Proc. Blizzard Challenge Workshop*, 2017.

[30] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: A fully end-to-end text-to-speech synthesis model," in *Proc. Interspeech*, 2017, pp. 4006–4010.

[31] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. J. Weiss, R. Clark, and R. A. Saurous, "Towards end-to-end prosody transfer for expressive speech synthesis with Tacotron," *arXiv preprint arXiv:1803.09047*, 2018.

[32] Y. Wang, D. Stanton, Y. Zhang, R. Skerry-Ryan, E. Battenberg, J. Shor, Y. Xiao, F. Ren, Y. Jia, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," *arXiv preprint arXiv:1803.09017*, 2018.

[33] Y. Wang, R. Skerry-Ryan, Y. Xiao, D. Stanton, J. Shor, E. Battenberg, R. Clark, and R. A. Saurous, "Uncovering latent style factors for expressive speech synthesis," in *NIPS ML4Audio Workshop*, 2017.

[34] Y. Jia, Y. Zhang, R. J. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. L. Moreno, and W. Yonghui, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *arXiv preprint arXiv:1806.04558*, 2018.

[35] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed. New York, NY: Springer, 2006.

[36] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[37] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc. B*, vol. 39, no. 1, pp. 1–38, 1977.

[38] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. ICLR*, 2014.

[39] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proc. ICML*, vol. 32, no. 2, 2014, pp. 1278–1286.

[40] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[41] P. Dayan, G. E. Hinton, R. M. Neal, and R. S. Zemel, "The Helmholtz machine," *Neural Comput.*, vol. 7, no. 5, pp. 889–904, 1995.

[42] M. Blaauw and J. Bonada, "Modeling and transforming speech using variational autoencoders," in *Proc. Interspeech*, 2016, pp. 1770–1774.

[43] 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.

[44] F. Huszár. (2017) Is maximum likelihood useful for representation learning? [Online]. Available: http://www.inference.vc/maximum-likelihood-for-representation-learning-2/

[45] A. Graves, J. Menick, and A. van den Oord, "Associative compression networks for representation learning," *arXiv preprint arXiv:1804.02476*, 2018.

[46] D. Povey, L. Burget, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. K. Goel, M. Karafiát, A. Rastrow, R. C. Rose, P. Schwarz, and S. Thomas, "Subspace Gaussian mixture models for speech recognition," in *Proc. ICASSP*, 2010, pp. 4330–4333.

[47] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. NIPS*, 2014, pp. 2672–2680.

[48] I. Goodfellow, "NIPS 2016 tutorial: Generative adversarial networks," *arXiv preprint arXiv:1701.00160*, 2016.

[49] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," in *Proc. APSIPA*, 2016, pp. 1–6.

[50] ——, "Voice conversion from unaligned corpora using variational autoencoding Wasserstein generative adversarial networks," in *Proc. Interspeech*, 2017, pp. 3364–3368.

[51] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "ACVAE-VC: Non-parallel many-to-many voice conversion with auxiliary classifier variational autoencoder," *arXiv preprint arXiv:1808.05092*, 2018.

[52] W.-N. Hsu, Y. Zhang, and J. Glass, "Learning latent representations for speech generation and transformation," in *Proc. Interspeech*, 2017, p. 12731277.

[53] ——, "Unsupervised learning of disentangled and interpretable representations from sequential data," in *Proc. NIPS*, 2017, pp. 1878–1889.

[54] K. Akuzawa, Y. Iwasawa, and Y. Matsuo, "Expressive speech synthesis via modeling expressions with variational autoencoder," in *Proc. Interspeech*, 2018, to appear.

[55] O. Fabius and J. R. van Amersfoort, "Variational recurrent auto-encoders," *Proc. ICLR Workshop Track*, 2014.

[56] J. Chung, K. Kastner, L. Dinh, K. Goel, A. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Proc. NIPS*, 2015, pp. 2980–2988.

[57] M. Fraccaro, S. K. Sønderby, U. Paquet, and O. Winther, "Sequential neural models with stochastic layers," in *Proc. NIPS*, 2016, pp. 2199–2207.

[58] J. Marino, M. Cvitkovic, and Y. Yue, "A general framework for amortizing variational filtering," in *ICML 2018 Workshop Theor. Found. Appl. Deep Gener. Model.*, 2018.

[59] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. Courville, and Y. Bengio, "SampleRNN: An unconditional end-to-end neural audio generation model," in *Proc. ICLR*, 2017.

[60] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. van den Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *Proc. ICML*, 2018, pp. 2410–2419.

[61] X. Wang, "Fundamental frequency modeling for neural-network-based statistical parametric speech synthesis," Ph.D. dissertation, SOKENDAI (The Graduate University for Advanced Studies), Tokyo, Japan, Sep 2018.

[62] X. Wang, S. Takaki, and J. Yamagishi, "Autoregressive neural F0 model for statistical parametric speech synthesis," *IEEE/ACM T. Audio Speech*, vol. 26, no. 8, pp. 1406–1419, 2018.

[63] J. Lorenzo-Trueba, G. E. Henter, S. Takaki, J. Yamagishi, Y. Morino, and Y. Ochiai, "Investigating different representations for modeling and controlling multiple emotions in dnn-based speech," *Speech Commun.*, 2018.

[64] R. Barra-Chicote, J. Yamagishi, S. King, J. M. Montero, and J. Macias-Guarasa, "Analysis of statistical parametric and unit selection speech synthesis systems applied to emotional speech," *Speech Commun.*, vol. 52, no. 5, pp. 394–404, 2010.

[65] D. Erro, E. Navas, I. Herndez, and I. Saratxaga, "Emotion conversion based on prosodic unit selection," *IEEE T. Audio Speech*, vol. 18, no. 5, pp. 974–983, 2010.

[66] P. Tsiakoulis, S. Raptis, S. Karabetsos, and A. Chalamandaris, "Affective word ratings for concatenative text-to-speech synthesis," in *Proc. PCI*, 2016.

[67] J. Yamagishi, K. Onishi, T. Masuko, and T. Kobayashi, "Acoustic modeling of speaking styles and emotional expressions in HMM-based speech synthesis," *IEICE T. Inf. Syst.*, vol. 88, no. 3, pp. 502–509, 2005.

[68] T. Nose and T. Kobayashi, "An intuitive style control technique in HMM-based expressive speech synthesis using subjective style intensity and multiple-regression global variance model," *Speech Commun.*, vol. 55, no. 2, pp. 347–357, 2013.

[69] J. Lorenzo-Trueba, R. Barra-Chicote, R. San-Segundo, J. Ferreiros, J. Yamagishi, and J. M. Montero, "Emotion transplantation through adaptation in HMM-based speech synthesis," *Comput. Speech Lang.*, 2015.

[70] J. P. Cabral, C. Saam, E. Vanmassenhove, S. Bradley, and F. Haider, "The ADAPT entry to the Blizzard Challenge 2016," in *Proc. Blizzard Challenge Workshop*, 2016.

[71] Q. T. Do, T. Toda, G. Neubig, S. Sakti, and S. Nakamura, "A hybrid system for continuous word-level emphasis modeling based on HMM state clustering and adaptive training," in *Proc. Interspeech*, 2016, pp. 3196–3200.

[72] J. Sotelo, S. Mehri, K. Kumar, J. a. F. Santos, K. Kastner, A. Courville, and Y. Bengio, "Char2Wav: End-to-end speech synthesis," in *Proc. ICLR Workshop Track*, 2017.

[73] W. Ping, K. Peng, A. Gibiansky, S. Ö. Arık, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep Voice 3: Scaling text-to-speech with convolutional sequence learning," in *Proc. ICLR*, 2018.

[74] H. Kawahara, "STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds," *Acoust. Sci. Technol.*, vol. 27, no. 6, pp. 349–353, 2006.

[75] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE T. Inf. Syst.*, vol. 99, no. 7, pp. 1877–1884, 2016.

[76] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *Proc. ICASSP*, 2018, pp. 4799–4783.

[77] C. Cremer, X. Li, and D. Duvenaud, "Inference suboptimality in variational autoencoders," in *Proc. ICLR Workshop Track*, 2018.

[78] R. Shu, H. H. Bui, S. Zhao, M. J. Kochenderfer, and S. Ermon, "Amortized inference regularization," *arXiv preprint arXiv:1805.08913*, 2018.

[79] 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," in *Proc. ICLR*, 2016.

[80] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," in *Proc. CoNLL*, 2016, pp. 10–21.

[81] M. D. Hoffman, C. Riquelme, and M. J. Johnson, "The β-vaes implicit prior," in *Proc. NIPS 2017 Workshop Bayesian Deep Learn.*, vol. 2, 2017.

[82] Y. Bengio, N. Léonard, and A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," *arXiv preprint arXiv:1308.3432*, 2013.

[83] E. T. Nalisnick, L. Hertel, and P. Smyth, "Approximate inference for deep latent Gaussian mixtures," in *Proc. NIPS 2016 Workshop Bayesian Deep Learn.*, vol. 1, 2016.

[84] J. M. Tomczak and M. Welling, "VAE with a VampPrior," *arXiv preprint arXiv:1705.07120*, 2017.

[85] E. Nachmani, A. Polyak, Y. Taigman, and L. Wolf, "Fitting new speakers based on a short untranscribed sample," in *Proc. ICML*, 2018, pp. 3683–3691.

[86] K. Oura, S. Sako, and K. Tokuda, "Japanese text-to-speech synthesis system: Open JTalk," in *Proc. ASJ Spring*, 2010, pp. 343–344.

[87] M. Morise, "Cheaptrick, a spectral envelope estimator for high-quality speech synthesis," *Speech Commun.*, vol. 67, pp. 1–7, 2015.

[88] A. Camacho and J. G. Harris, "A sawtooth waveform inspired pitch estimator for speech and music," *The Journal of the Acoustical Society of America*, vol. 124, no. 3, pp. 1638–1652, 2008.

[89] H. Zen, T. Nose, J. Yamagishi, S. Sako, T. Masuko, A. W. Black, and K. Tokuda, "The HMM-based speech synthesis system (HTS) version 2.0," in *Proc. SSW*, 2007, pp. 294–299.

[90] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, 2000, pp. 1315–1318.

[91] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Incorporating a mixed excitation model and postfilter into HMM-based text-to-speech synthesis," *Syst. Comput. Jpn.*, vol. 36, no. 12, pp. 43–50, 2005.

[92] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

[93] X. Wang, S. Takaki, and J. Yamagishi, "An autoregressive recurrent mixture density network for parametric speech synthesis," in *Proc. ICASSP*, 2017, pp. 4895–4899.

[94] Y. Fan, Y. Qian, F.-L. Xie, and F. K. Soong, "TTS synthesis with bidirectional LSTM based recurrent neural networks," in *Proc. Interspeech*, 2014, pp. 1964–1968.

[95] F. Weninger, J. Bergmann, and B. W. Schuller, "Introducing CURRENNT: The Munich open-source CUDA recurrent neural network toolkit," *J. Mach. Learn. Res.*, vol. 16, no. 3, pp. 547–551, 2015.

[96] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. AISTATS*, 2010, pp. 249–256.

[97] O. Watts, G. E. Henter, T. Merritt, Z. Wu, and S. King, "From HMMs to DNNs: where do the improvements come from?" in *Proc. ICASSP*, 2016, pp. 5505–5509.

[98] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[99] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[100] S. Holm, "A simple sequentially rejective multiple test procedure," *Scand. J. Stat.*, vol. 6, no. 2, pp. 65–70, 1979.

[101] A. Leijon, G. E. Henter, and M. Dahlquist, "Bayesian analysis of phoneme confusion matrices," *IEEE/ACM T. Audio Speech*, vol. 24, no. 3, pp. 469–482, March 2016.