
On Controlled De Entanglement

Sai Krishna Rallabandi

Language Technologies Institute, Carnegie Mellon University
srallaba@cs.cmu.edu

Abstract

This is the writeup accompanying my Sphinx lunch talk in Fall 2019¹. In this draft, I present an argument for De-Entanglement: a property that has potential to isolate the factors of variation in the data distribution. I posit that explicitly isolating relevant factors using such an approach is helpful with respect to the downstream tasks. I introduce the concept using analogies and then present mathematical analysis from information theory highlighting different ways to accomplish ‘De-Entanglement’. I point out how this concept can be interpreted as different popular deep learning approaches. I conclude by arguing that this serves as a neat framework to build systems.

1 Introduction

Imagine a property referred to as De-Entanglement. It is (currently) defined as the ability of a model of world to isolate the factors of variation which perhaps were involved in the design of the world itself. It is easy to see that such a property is extremely useful. Consider kids playing with Lego toys as opposed to a static toy like a TeddyBear or a Barbie. The freedom to dismantle the structure apart and re-compose variants of it has been shown to improve creativity[Gauntlett, 2014]. The implications become even more apparent when we consider a real life application such as speech processing. It is extremely difficult to reason about speech in the time domain by inspecting the individual samples. However, transforming the same utterance into frequency domain - a process that isolates the contribution from individual frequencies(factors of variation) - makes reasoning easier, to the point of even identification of the individual linguistic units within the utterance. This can be generalized to other types of data: (visible) light in terms of its spectral decomposition enabling estimation of cosmic evolution of celestial bodies[Keller et al., 2014]. In this writeup, I argue that designing systems with an explicit consideration for ‘De-Entanglement’ not only leads to better systems but also has additional benefits in terms of scalability, controllability and interpretability.

This writeup is organized as follows: In this section, I present the basis for De-Entanglement using analogies. In the subsequent section 2, I present a more mathematical version followed by a brief perspective with respect to speech processing in section 3. I conclude this writeup with a summary in section 9.

1.1 Intuitions / Analogies

Let us first try to derive the concept of ‘De-Entanglement’ using analogies.

1.1.1 Information Propagation

Consider a noise less communication between Alice and Bob. Perhaps it is about the quality of this write-up itself. For the sake of simplicity, consider that there are only two possible messages Alice wants to communicate to Bob and are the following:

¹Previous Writeups: <http://www.cs.cmu.edu/~srallaba/ProjectPeregrine/>

Message
This writeup sucks!
This is a reasonable writeup

One of the most efficient ways to communicate this message would be to use a signal that has two states: Head and Tail or spin up and spin down of a spin- $\frac{1}{2}$ particle like electron.

Signal 1	Signal 2	Message
Head	Spin Up	This writeup sucks!
Tail	Spin Down	This is a reasonable writeup

The probability of decoding the right message depends on the ability of Bob to associate the signal to its intended message. Let us extend this to a more practical scenario. Alice is still using an electron but Bob is using a Hydrogen atom to decode the message. Now the problem starts becoming interesting. For Bob to decode the right message, two things need to happen:

- Identify which two of the ten stable states of the Hydrogen atom correspond to the signal states. For now let us assume that there is a one to one correspondence between the states.
- Attribute the identified states to the right signals.

Due to the large number of possibilities a Hydrogen atom can encode (high model capacity), the error probability of the communication increases [Schumacher, 1995]. To compensate, Bob needs to learn a projection function (ϕ) that projects the 10 stable states into 2 plausible states that Alice perhaps used to encode the message. I argue that our current deep learning approaches fall under this scenario: The information capacity of these models is very high where as the tasks, due to the way they are formulated, do not require such high capacity. When these powerful models are employed, the model capacity is utilized simply to capture the surface level associations [Goyal et al., 2017] and biases [Agrawal et al., 2017] in the data distribution. In other words, the most popular paradigm of training - supervised training - by its formulation reduces many interesting tasks (like Visual Question Answering) to equivalent of model fitting on the observed data, which models with high capacity can accomplish by memorizing the observed data distribution. The problem becomes even more pronounced in the context of tasks that lack strict objective measures (such as image captioning). On the other hand, weakening the models (by going shallow) makes it harder for the models to sufficiently model the stochasticity and capture all the degrees of freedom required to handle tasks involving humans (such as Visual Dialog). I posit that a better strategy is to have the models at full capacity but to explicitly regulate the amount of information (similar to learning ϕ) thus forming an information bottleneck for designing more efficient systems.

1.1.2 Information Bottleneck and Factors of Variation in the Data distribution

But what has information bottleneck got to do with the term ‘De-Entanglement’? Let us perform a small thought experiment: Imagine that a blob of ink was dropped (say, at $t = t_0$) in a large bucket of still water. We know that this process likely creates waves in the bucket which then spread from the point of contact to the horizon. For the sake of simplicity, let us assume that no water molecules jumped off the bucket leading to a decrease in the total mass of water. Also assume that the whole process is adiabatic, and so no energy is lost from the system in the form of heat. It is interesting to think about the information in this system. The information most likely is about the properties of the ink blob and is contained solely in the water molecules - manifested in the form of waves within the system. Consider one such property: diameter of the ink blob. Sampling the characteristics of the generated waves should enable us to describe the properties of ink blob. It should be apparent that this formulation is very similar to supervised training. We drop more ink blobs, record the characteristics of the waves and hopefully a sufficiently sophisticated model can be fit to predict the diameter of the blob. Let us now turn our attention to a slightly more interesting version of the task: We are observing the system at a later point in time (say, $t = t_1$) and the waves are dying down, solely due to the reasons related to viscosity and not due to loss of heat to an external medium. In order to now estimate the properties of the blob, we need to sample at a much finer resolution compared to the previous scenario. In this scenario, we are constrained only by the technology requirements in terms of the ability to sample at those finer resolution; In principle we can still recover diameter of

the blob. Finally, let us consider a much more interesting variant of the task: The waves have now completely died down. Where is the information about the blob now? Before proceeding further, we need to establish the relationship between entropy and information of this hypothetical system. From second law of thermodynamics, we know that the system always strives to reach highest entropy. Therefore, the system at $t = t_2$ which coincides with the initial state right before $t = t_0$ has highest entropy. Returning to our question, of course the information about the blob is still implicit within our system but it seems implausible to extract it from the current state of the system. However, we made an interesting observation in this process: We can explicitly disturb the system using another blob of ink and make new measurements: thermodynamics ensures that the system returns to t_0 . We can do this multiple times over and under the mentioned constraints, generalize the findings to not just ink blobs but also to pebbles and other objects. It can be seen that this scenario is equivalent to the paradigm of unsupervised learning. Specifically, if we build a model capable of reducing the total entropy of the system, we can make inferences about the blob. However, it has to be noted that the observation about waves plays a significant role in our success in this endeavour. Attempting to decipher information about the object without the observation about waves is likely futile: Consider an approach of trying to decode information about the blob using the motion of water molecules instead of considering the wave motion. Since the number and organization of molecules in seemingly uniform and homogeneous water in the bucket is astronomical (extremely high degrees of freedom), the capacity of even a sophisticated model aimed at deciphering information about the blob falls short. Since there are multiple ways the model can attempt to reduce the overall entropy, the optimization of the model becomes intractable. Even in scenarios where a sufficiently sophisticated model does recover the diameter, it is apparent that employing the observation about wave nature simplifies the model complexity. The saved model capacity can instead be used to make better inferences about nuances and finer details. I argue that information about the factors of variation in the system (waves) provides a reliable signal assisting us to design an efficient model that can decode the relevant information (diameter). I refer to this mechanism: of explicitly controlling the process of isolating the relevant factors of variation in the data distribution as **‘De-Entanglement’**.

1.1.3 Disentanglement vs De-Entanglement

An interesting possibility to consider as factors of variation is the motion of molecules themselves. I argue that while this is a causal factor of variation, it is not necessarily a relevant factor of variation with respect to the downstream task at hand: measurement of blob diameter. This argument can be generalized as well: Not all causal factors of variation are relevant to the task. I specifically argue for those factors of variation that are most relevant to the task we are interested in. While the process of isolating causal factors of variation in the data is typically referred to as ‘Disentanglement’ and follows an unsupervised approach, I am more interested in isolating the relevant factors using an explicit control. In this interpretation, the relevant factors of variation are a subset of the causal factors of variation. Continuing the analogy, we are not particularly interested in the factors of variation in the system when it is in state t_0 but between t_0 and t_2 . In addition, the term ‘disentanglement’ is typically used to imply a representation where each dimension of the representation corresponds to one particular factor of variation. I argue that such fine grained isolation is not always possible in the distribution of real world data. Consider the prosodic aspects of a speech signal. There are several confounding factors related to Fundamental Frequency (F_0), duration of the lexical item and the intensity/power/magnitude that contribute to the prosody and are perhaps not orthonormal to each other. It is not clear what a disentangled representation in such scenarios denotes. On the other hand, ‘De-Entanglement’ allows us to specify our observations (waves in the analogy, one of the three factors for prosody) and obtain a decomposition of the signal constrained to the observations.

Mechanism	Disentanglement	De-Entanglement
Approach	Unsupervised	No constraint
Representation Type	Degenerate/Orthonormal	No constraint

2 De-Entanglement

I posit that designing learning paradigms such that we explicitly control de-entanglement of relevant factors of variation while marginalizing the nuisance factors of variation leads to massive improvements. Such an approach, I claim, leads to further advantages in the context of both generative

processes: in terms of generation of novel content and discriminative processes: in terms of robustness of such models to noise and attacks. Let us consider a typical deep learning architecture such as AlexNet[Krizhevsky et al., 2012]. It is characterized by a series of convolutional layers (feature extraction module) followed by a pooling layer and a SoftMax layer(classification module). Note that while I mention AlexNet as an example, this abstraction can be extended to most sequence to sequence architectures with encoder as feature extraction module and decoder as the classification module[Rousseau and Tsafaris, 2019] across modalities and tasks. It can be shown that the pooling layer acts as information bottleneck[Tishby et al., 2000] module in such architectures. I point out[Rallabandi and Black, 2019] that in case of conventional Seq2Seq architectures deployed today, attention plays the role of information bottleneck module regulating the amount of information being utilized by the decoder. In [Rallabandi and Black, 2019, Vyas et al., 2019] I show that this module controls optimization in encoder decoder models leading to (1) Disentanglement of Causal Factors of variation in the data distribution (2) Marginalization of nuisance factors of variation from the input distribution. In case of models that employ stochasticity, two more effects can be observed : (a) Posterior collapse or Degeneration due to powerful decoders and (b) Loss of output fidelity due to finite capacity decoders. In current architectures, marginalization and disentanglement are realized implicitly and often lead to (a) and (b) when deployed in practise.

The most popular approach to obtain isolation of factors of variation in neural models is by employing stochastic random variables. This approach provides flexibility to jointly train the latent representations as well as the downstream network. It has been observed that the latent representations resemble disentangled representations under certain conditions [Chen et al., 2018, Burgess et al., 2018b, Esmaili et al., 2018, Ansari and Soh, 2018]. Note that although obtaining such degenerate representations is considered typical, it is not the only manifestation: it also manifests as continuous representations[Ravanelli and Bengio, 2018] and other abstract phenomena(e.g. grounding). I argue that explicitly controlling what and how much gets de-entangled [Burgess et al., 2018b] is better than implicit disentanglement as is followed today[Locatello et al., 2018]. I identify four ways to computationally control de-entanglement in encoder decoder models

- (1) By employing suitable priors about task or data distribution
- (2) By incorporating additional adversarial or multi task objectives within the model
- (3) By utilizing a different divergence objective
- (4) By employing an alternative formulation of probability density estimation

2.1 Case for Controlled De-Entanglement

I believe that complete disentanglement of input data into its independent causal factors of variation is not fully useful. A more attractive option is to control what and how much gets de-entangled in a task dependent manner. It has to be noted that given a particular downstream task, some causal factors of variation might not be relevant, in which case modeling them would be unnecessary. Let us consider a data distribution X which consists of class examples $\{x_1, x_2, \dots, x_n\}$, where each x_i is described by attribute-set (a, b, c) . The prior distribution of X can be represented by a parametric function g such that g maximizes the likelihood of X over the set of its attributes:

$$P_\omega(X) = g_\omega(a, b, c) \quad (1)$$

Note that the attribute-set can either contain individual entities or the relationships between them or both. To illustrate this, let us consider a toy-example where we build a binary classifier to predict if a given integer triplet is a Pythagorean triplet. Pythagorean triplets are a triplet of numbers that follow Pythagoras Theorem such as $\{3, 4, 5\}$ and $\{5, 12, 13\}$. In this task, the attribute-set consists of the relationship between the first two-elements of the triplet. If the model is able to discover this attribute, it can generalize for any given numbers. However, if we have a more complicated task like building a classifier for MNIST digits, then the attribute-set has multiple first and second order relations like brush strokes, shape of the digits etc. The success of modelling $P_\omega(X)$, and ultimately the success on the downstream task, relies on how well can the model isolate these individual attributes from the observed data X_t . This isolation ability becomes even more important in case we want to regenerate

Table 1: Implicit Realization of information bottleneck in popular deep learning mechanisms

Architecture	Manifestation of Information Bottleneck	Type of Bottleneck
AlexNet	Pooling	Spatial
Attention	Activation	Temporal
VAE	Priors	Spatio temporal
Neural Module Networks	Softmax over Modules	Spatial
LISA	Linguistic Priors	Temporal
BERT	Masking	Spatio temporal
XLNet	Permutation	Spatio temporal

the digits using a generative model. Mathematically, let us consider the posterior probability of a training instance x_1 expressed as

$$P_{\theta}(x_1) = f_{\theta}(x_1) \quad (2)$$

where f denotes arbitrary function and θ denotes the parametric family used to model the distribution X . It can be seen that compositionality over an unseen training instant x_{new} would be possible if f is related to g . In other words, f needs to intuitively have some information about the latent causal factors of variation that generated X in the first place. In such scenarios, the test instance can be appropriately expressed as

$$P_{\theta}(x_{new}) = h(a, k(b, c)) \quad (3)$$

where h and k can be a novel combination of functions that embed these attributes in the manifold of original distribution of X . Not tracking the relevant factors of variation typically leads to model memorizing only the surface level associations leading to mode collapse and lack of diversity in the generated outputs. On the other hand, explicitly caring about the factors of variation can be seen as a way of incorporating inductive bias into the model and has the potential to avoid such pitfalls.

2.2 Implicit De-Entanglement in Seq2Seq Models: Deterministic Attention vs Stochastic Attention

I will illustrate this sub section with a typical generative model of speech: Text to Speech. Consider that we are interested in building a code mixed version: a model that can accomodate two languages in a single utterance. Let us also consider a speech corpus X consisting of languages $\{l_1, \dots, l_n\}$, where each l_i might comprise of multiple speakers. Let $y_1, \dots, y_n$ denote acoustic frames in the target sequence y while $x_1, \dots, x_n$ denote the encoded text sequence x from one of the languages. A typical attention based encoder decoder network such as Tacotron[Wang et al., 2017] factorizes the joint probability of acoustic frames as product of conditional probabilities. Mathematically, this can be shown as below:

$$P_{\theta}(y|x) = \prod_{t=1}^n P(y_t|x_1 \dots x_m, s_t) \quad (4)$$

where s_t is a decoder state summarizing $y_1, \dots, y_{t-1}$. Parameters θ of the model are set by maximizing either the log likelihood of training examples or the divergence between predicted and true target distributions. At each time step t in these models, an attention variable a_t is used to denote which encoded state of $x_1 \dots x_m$ aligns with y_t . The most common form of attention used is soft attention, a convex combination from encoded representation of input text. It has to be noted that soft attention in such scenarios is essentially a latent deterministic variable that computes an expectation over the alignment between input and output sequences. Empirically, soft attention provides surprisingly good alignment often correlating with human intuitions. Having said that, to synthesize speech from different languages at test time, the generative process needs to disentangle appropriate individual language attributes from observed data X_{obs} and also compose them to form a coherent utterance in

the voice of desired speaker. However, presence of deterministic alignment method limits the ability of models to generalize to such scenario.

On the other hand, variational attention[Deng et al., 2018] provides a mechanism to factorize this alignment and mediate the generative process of y through a stochastic variable z . In addition, both soft and hard attention mechanisms can be shown as special cases of ELBO[Deng et al., 2018]. Therefore, incorporating latent stochastic variables allows us to directly optimize ELBO. In this context, model parameters are set by maximizing the log marginal likelihood of the training samples. But direct maximization of this marginal in the presence of latent variable is often difficult due to expectation involved. To address this, a recognition network q is employed to approximate the posterior probability using reparameterization. It is interesting to note that the encoder in a deterministic Seq2Seq network functions as the recognition network in latent stochastic variable models and is incentivized to search over variational distributions to improve ELBO. Intuitively, the lower bound is tight when the inferred variational distribution is closer to the true posterior of the data. This has a sense of grounding in our understanding of the task as well. Perhaps there are a set of universal phonemes, around 120, which should be enable us to speak in any language subject to the phonotactic constraints of the language. Having such prior information greatly reduces the model size as opposed to naively using a combination of all phones from all the languages to build a polyglot model.

2.3 Analysis of role of priors in Latent Stochastic Models

The choice of priors plays a significant role in optimization within latent stochastic models. In this subsection, we present an analysis to show that priors control the disentanglement of causal factors of variation in such models. Let us consider the ELBO being optimized in a VAE:

$$E_{q_\phi(z|x,c)}[\log p_\theta(x|c,z)] - |D_{KL}(q_\phi(z|x,c)||p_\theta(z|c))| \quad (5)$$

where the first term is the reconstruction error while the second is the divergence between approximate and true posteriors. Here are the four phenomenon that are manifested due to choices of priors:

(1) *Disentanglement or Factorization of causal factors of variation*

The KL divergence forces the posterior distribution output by encoder to follow an appropriate prior about the data generation process. Typically, prior space is assumed to be continuous distribution and a unit Gaussian. The global optimum value for the divergence in such cases is 0 and is reached only when both the distributions exactly match each other. Since the prior information about the data generation process typically involves some causal factors of variation of the data, this naturally is assumed to translate to a constraint on the encoder to track such factors. Thus, such models have potential to disentangle or factorize the causal factors of variation in the distribution.

(2) *Marginalization of Nuisance Factors of Variation*

It has to be noted that during training optimization is performed in expectation over minibatches. Therefore, the expectation of KL divergence can be rewritten as related to the amount of mutual information between the latent representation and the data distribution [Makhzani and Frey, 2017]. As this divergence decreases, the amount of information the encoder can place in the latent space also decreases. As a result, encoder is forced to discard some nuisance factors that may not have contributed to the generation of data. Thus, KL divergence also forces the model to marginalize the nuisance variables.

(3) *Posterior Collapse due to simple priors*

Consider the scenario where the prior is too simplistic, such as the aforementioned unit normal distribution. In such cases, the model is incentivized to force the posterior distribution to closely follow the Gaussian distribution [Chen et al., 2016]. Typically the decoders in variational models are implemented using universal approximators such as RNNs. In the context of a TTS systems, decoder segment of the acoustic model along with the neural vocoder act as the decoders. Since such decoders are very powerful, they are able to learn or ignore the priors about data distribution themselves and hence marginalize out the latent representation input from the encoder. In other words, the prediction of next sample is based solely on the marginal distribution at the current timestep which can be implemented by learning a dictionary per time step. Therefore, the encoder is no longer forced to track the causal factors of variation in the data. This is referred to as posterior collapse or mode collapse.

(4) *Loss of output fidelity due to complex priors*

A reasonable and intuitive solution to posterior collapse is making the prior space more complex thereby pressurizing the posterior distribution to track the prior space more closely. For instance, [Burgess et al., 2018a] attempt to accomplish this by adding a hyperparameter β to promote disentanglement and gradually increasing channel capacity, something that increases loss. However, it has to be noted that simply making the prior distribution arbitrarily complex also perhaps leads to unreasonable constraints on the decoder. For instance, in scenarios that have categorical distribution as their output (tasks such as language modeling, machine translation, image captioning among others) it is unintuitive to assume that the true prior that generates latent distribution is a Gaussian when the likelihood is based on discrete sequential data in such tasks. Having such strong priors directly affects the reconstruction ability in these models.

Therefore, priors in latent stochastic models play a significant role in the optimization and facilitate disentanglement of causal factors of variation on the one hand, as well as help the ability of the model to reconstruct the data distribution on the other. Having this knowledge enables us to engineer various components to tune the model behavior as per our requirements.

3 Speech Processing

Consider a speech corpus such as WSJ[Paul and Baker, 1992] or LibriSpeech[Panayotov et al., 2015]. Such corpora are an amalgamation from different sources and are characterized by the presence of non trivial variations. For instance:

- Variations in terms of content (music and other noises)
- Variations in terms of speaker(s)
- Variations in terms of channel (microphone vs ..)
- Variations in terms of speaking style (slow - speech vs clear - slurred)
- Variations in terms of prosodic variations (high valence vs low valence, etc)

Consider a speech chain architecture built on such corpora. From the perspective of a Text to Speech system, such as system provides a rich source of natural variations in terms of, for example, filler sounds. However, current models are not typically designed to handle such variations: while speech recognition system aim to normalize these differences, speech generation systems expect precise labeling to ensure appropriate fidelity of non speech events. Therefore, Found Data presents us with a number of challenges before rendering it useful for building good voices. I argue that de-entanglement provides a mechanism to incorporate such interesting phenomena into a speech synthesis system without having to significantly modify the speech recognition components.

Speech signal is a rich encoded form of distinct information - ranging from language and content to speaker, age, emotions, etc. There are works which have shown that it is possible to decipher even the behavioral patterns and diagnose certain medical conditions from speech. Visualizing mathematically, a speech utterance (U) can be seen as a non-linear combination (because linear is a subset of nonlinear) of speaker(s), content, emotion and other personality/channel traits as follows:

$$U = f(s, c, e, \theta) \quad (6)$$

Here θ is a variable we are choosing to account for all the extra information present in the utterance. It might be tempting to attempt to clean such data and assume that we can use a preprocessing pipeline to our existing unimodal or multimodal algorithms that expect balanced datasets without bias . However, it is important to note that even in controlled setting, professional speakers also tend to produce variations in these factors (leading to different pronunciations or different durations for same pronunciation, etc). This is also apparent if we take a look at the distribution of durations used by speakers in speech related tasks. For instance, see figure 1 which depicts the different durations for the same amount of recorded speech. This database was part of voice conversion challenge 2018.

It has to be noted that Found data will comprise many more variations in addition to the ones mentioned. In order to build a robust model that can be appropriately controlled, it is desirable to isolate the individual factors of variation in the data. It is also possible that more than a single feature

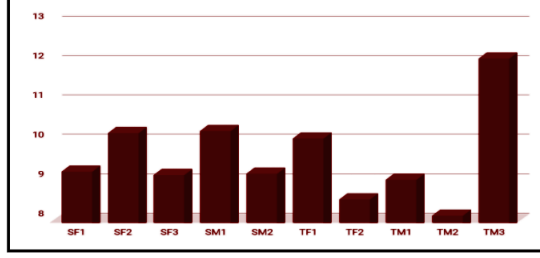


Figure 1: Bar chart depicting the duration of speech per speaker in minutes. X axis has different speakers. Y axis has the length of the spoken content in minutes. This is part of the Voice Conversion Challenge 2018. Note that the content of speech is same across the speakers. The variation in duration points to the differences in pronunciation, style or accentuation corresponding to the speaker set.

is involved in accurately accounting for some of those different characteristics. Given a chunk of speech signal (lets call this a frame and denote it by F), it is reasonable to assume that at least 3 features play a role in deciphering the aforementioned traits. Mathematically,

$$F = g(\text{spectrum}, \text{fundamental frequency}, \text{voicing}) \quad (7)$$

where spectrum and voicing correspond to the content spoken - Vowels are voiced and consonants are unvoiced. Fundamental frequency and spectrum correspond to the speaker - formants are speaker specific and pitch is speaker specific. However, this is just a rough approximation as the content is also present in the spectrum (harmonics for voiced and random for unvoiced).

Traditionally we have been doing feature extraction by signal processing to isolate the latent causal factors of variation. However, noise affects these algorithms. Therefore, it might be more beneficial to learn this isolation / feature extraction automatically from data using approaches employing observations about the distribution of the data [Ebberts et al., 2017].

Attempting to tackle such tasks using vanilla encoder decoder models that just learn surface level associations between the modalities leads to learning biases present in the data [Goyal et al., 2017, Agrawal et al., 2018]. Such models usually fail to recognize previously unseen compositions of the existing objects [Agrawal et al., 2017] since while they are universal function approximators, they lack the hierarchy to learn robust representations in their formulation. Deep latent stochastic models [Chung et al., 2015, Kingma et al., 2016] on the other hand, provide a flexible framework that promises to address such concerns [Chen et al., 2016]. These models provide a mechanism to jointly train both the latent representations as well as the generator network. Therefore, such models are expected to both discover and isolate relevant factors of variation present in the distribution of original data.

In this section, we first present our analysis of the optimization that happens in WaveNet, followed by latent stochastic models. We then present a case for controlling the disentanglement.

3.1 Analysis of optimization in Stochastic Generative Models for Speech - VQ-VAE

Lets take WaveNet. WaveNet [Van Den Oord et al., 2016] is an autoregressive neural model with a stack of 1D convolutional layers that is capable of directly generating audio signal. It has been shown to produce generated speech that rivals natural speech when conditioned on predicted mel spectrum [Shen et al., 2017]. The input to WaveNet is subjected to corresponding gated activations while passing through each dilated convolutional layer and is classified by the final softmax layer into a μ law encoding. The concrete form of the residual gated activation function is given by following equation:

$$r_d(x) = \tanh(W_f * x) \odot \sigma(W_g * x) \quad (8)$$

where x and $r_d(x)$ are the input and output with dilation d , respectively. The symbol $*$ is a convolution operator with dilation d and the symbol $\odot$ is an element-wise product operator. W represents a

convolution weight. The subscripts f and g represent a filter and a gate, respectively. The joint probability of a waveform $\mathbf{X}$ can be written as:

$$P(X|\theta) = \prod_{t=1}^T P(x_t|x_1, x_2 \dots x_{t-1}, \theta) \quad (9)$$

given model parameters θ . During implementation of WaveNet, the autoregressive process is realized by a stack of dilated convolutions. The final output y_t at time step t can be expressed mathematically as:

$$\hat{y}_t \sim \sum_{d=0}^D h_d * r_d(x) \quad (10)$$

where x, y represent input and output vectors; D is the number of different dilation used and d is the dilation factor; h_d is the convolution weights. This stack of convolutions is repeated multiple times in the original WaveNet. Optimization in WaveNet is performed based on the error between predicted sample and the ground truth sample conditioned on previous samples in the receptive field alongside the local conditioning. Expressing the loss function being optimized mathematically the error at sample t is:

$$l_t = Div(\hat{y}_t || y_t) \quad (11)$$

Here, we define the divergence similar to the [Salimans et al., 2017], To optimize this loss, the contribution from the individual convolution layers towards this global error function must be nullified. Now let us consider the expression for intermediate output for a single filter in Eqn 10:

$$x_{out}(t) = \sum_{\tau=0}^t h(\tau)x(t-\tau) \quad (12)$$

where τ is the receptive field covered by the model and $h(\tau)$ represents the discrete state representation at time t . Without loss of generality and dropping the term τ for brevity, the spectral representation generated by the model can be expressed as:

$$Y(z) = H(z)X(z) \quad (13)$$

Considering the discrete nature of input from Eqn 11, an interpretation of Eqn 13 is that the neural autoregressive model acts as the transfer function and is discretized by convolving with the samples from original signal. It has to be noted that this is similar to the formulation of source filter model of speech, specifically the periodic components aka voiced sounds. Voiced sounds typically represented as impulse train are convolved with the transfer function to generate spectral envelope. As a corollary, from Eqn 11 and 13, we posit that the optimization in WaveNet model is performed by minimizing the divergence between true and approximate spectral envelope. Note that latent stochastic models such as VAEs are aimed to minimize the divergence between true and approximate posterior distributions of input data. The advantage with such models is the presence of stochastic random variables that capture the causal factors of variation in input based on some prior information about the distributional characteristics of data. Techniques aimed at this [Higgins et al., 2016] have shown that it is possible to effectively disentangle the factors of variation using stochastic variables. Hence, we postulate that it should be possible to augment WaveNet decoder with a suitable encoder and an appropriate prior distribution to disentangle the acoustic phonetic units from a given utterance.

Table 2: Overview of Systems with variation in Grapheme to Phoneme Mapping.

Level	Config	Description
Text	P2P_Mono	Phone to Phone in Monolingual System
Text	W2P_Mono	Word to Phone in Monolingual System
Text	Translit_Mono	Word to transliteration in Monolingual System
Spectral	Separate	Combined phoneset with language tag
Spectral	Shared	Combined phoneset without language tag
Spectral	Word	Language of word as question
Spectral	WC	Tri context for word is used as question
Prosodic	BL	Baseline statelevel Duration Prediction
Prosodic	Ratio	Ratio used for modification
Prosodic	Gauss	Gaussian used for modification
Prosodic	OLA	Only Look ahead features
Prosodic	OPF	Only phonetic features

4 De-Entanglement by Priors

4.1 Code Mixing

CodeMixing is a phenomenon where linguistic units such as phrases, words and morphemes of one language are embedded into an utterance of another language Gella et al. [2014]. This is a common phenomenon in multilingual societies such as in India where English has transitioned from the status of a foreign language to that of a second language. Moreover, due to the diversity in terms of the regionality and the proficiency, the patterns of codemixing found are rather different from one another, often leading to confusion. Muysken [2000] states that there are, in general, three types of mixing:

- *insertion* or *embedding* of content (lexical items or entire constituents) from English into a regional language.
- *alternation* between structures from both the languages.
- *congruent lexicalization* of material from different lexical inventories into a shared grammatical structure.

Muysken [2000] also identifies that the lexical items that can be inserted during mixing are adverbial phrases, single nouns and determiner-nouns. We performed an informal analysis on a Hindi recipe blog on the web and found that while the content was all in ASCII, around 15 percent of the words were English words (though often misspelt), and almost all of them were nouns or adverbs, in line with the observation in Muysken [2000]. A similar analysis of a Telugu blog showed that around 20-30 percent of the text is in English (ASCII), and again, most of them being nouns. Such mixed text data poses a variety of challenges to the speech synthesis system due to their innate characteristics such as contractions, non-standard pronunciation, and non-standard sentence constructions, etc.

Current approaches handling this scenario fall into three categories: phone mapping, multilingual or polyglot. In phone mapping, the phones of the foreign language are substituted with the closest sounding phones of the primary language, often resulting in a strong accented speech. In a multilingual setting, each text portion in a different language is synthesised by a corresponding monolingual TTS system. This typically means that the different languages will have different voices unless each of the voices is trained on the voice of same multilingual speaker. The polyglot solution refers to the case where a single system is trained using data from a multilingual speaker. Similar approaches to dealing with codemixing have been focused on assimilation at the linguistic level, and advocate applying a foreign linguistic model to a monolingual TTS system. The linguistic model might include text analysis and normalisation, a G2P module and a mapping between the phone set of the foreign language and the primary language of the TTS system Tomokiyo et al. [2005], Campbell [2001], Badino et al. [2004]. Other approaches utilise cross-language voice conversion techniques Mashimo

et al. [2001] and adaptation on a combination of data from multiple languages Latorre et al. [2005]. Assimilation at the linguistic level is fairly successful for phonetically similar languages Badino et al. [2004], and the resulting foreign synthesized speech was found to be more intelligible compared to an unmodified non-native monolingual system but still retains a degree of accent of the primary language. This might in part be attributed to the non-exact correspondence between individual phone sets.

4.2 Approaches for Grapheme to Phoneme Conversion

Grapheme-to-phoneme conversion (G2P) is one of the first tasks in speech synthesis and essentially is a conversion from a word in orthography to its spoken form or pronunciation. This can be seen, in an oversimplification, as maximizing $Prob(P/G)$ where P is the phoneme sequence and G is the grapheme sequence of a single language. However, in case of ‘embedding’, the G contains graphemes from both native language as well as English. In this case, a phone to phone mapping is employed to map the phones from English to the native language.

However, in practice, this method results in a strong foreign accent while synthesizing the english words. Elluru et al. [2013], Rallabandi et al. proposed a method to use a word to phone mapping instead, where an english word is statistically mapped to Indian language phones. This can be seen as maximizing the expression:

$$\prod_{i,j \in S,W} Prob(s_i|w_j) \quad (14)$$

where $w \in W^d$ is a word in source language with a vocabulary(W) of size d. The intuition in this can be seen as

$$\prod_{i,j,k \in S,M,W} Prob(m_k|w_j) * Prob(s_i|m_k) \quad (15)$$

where $m \in M$ was referred to as the mental mapping of the native speaker. In this paper, we take a more direct approach and investigate the use of transliterations as the phoneme internal representation. We hypothesize that there is a single model which has generated both the transliteration and the phoneme itself. This serves as a more concrete mapping problem and can be seen as maximizing the expression

$$\prod_{i,j,k \in S,T,W} Prob(t_k|w_j) * Prob(s_i|t_k) \quad (16)$$

where $concat(t) \in T$ is the transliterated form. We have used this approach previously, Sitaram [2015], but this the first time we are systematically comparing the three possible G2P approaches in such scenarios.

4.3 Approaches for Acoustic Modeling

There are two dimensions in which we can vary the input features for synthesis. First at the phone level itself, choosing to explicitly separate the phones by original language (we add a language id suffix to the phone name), or taking the union of the phones across the languages (e.g. the data for English /t/ and Hindi /t/ are treated as one class). Secondly we provide contextual features to identify the language that the phone actually appears in (e.g. is it in an English or a Hindi word – and also the language id of the surrounding words). In the second case of language contextual tagging, these features may allow pronunciation distinctions between longer phrases in a particular language and isolated words in a codemixed utterance.

4.3.1 Systems built

The systems we built at this level are mentioned in table 2. The system ‘Separate’ uses a combined phoneset, where the phones of English are explicitly separated from the phones of Indic by adding

a language tag `_E` or `_I` denoting English and Indic respectively. The system ‘Shared’, as the name indicates, uses a combined phoneset obtained by the union of phones from English and Indic phonesets - if a phone is common in both the sets it is retained as is, and the disjoint phones are added separately. We then build systems ‘Word’ and ‘WC’ incorporating contextual features to identify the language. These systems differ in the amount of context used. The system ‘WC’ uses a tri word context while the system ‘Word’ uses a single word context.

4.4 Approaches for Modeling Prosody

In bilingual sentences, it was shown that the context of source language will influence the embedded target language words, which will change the original prosody of the target language Zhang and Tao [2008]. In order to establish a mixed-language speech synthesis system based on separate corpora, it is therefore important to consider such influences to generate natural mixed-lingual prosody. From the corpus, we did not observe much differences in pitch between the monolingual and mixed lingual synthesis systems, as in Zhang and Tao [2008]. We suspect that this might be due to the proficiency level of bilingual speech, which is used on a daily basis in India. However, we did observe marked differences in the duration of the words, specifically at the point of switch from source language to the target language. In this subsection, we present our approaches to build combined prosodic models using separate monolingual speech and text data. We specifically look at two different ways of achieving this: (1) By manipulating the durations predicted by the baseline model and (b) Incorporating extra features while building the model, thereby modifying the prediction model itself.

- **Ratio based Manipulation System** Zhang and Tao [2008] In this system we first obtain the predicted durations from the baseline prosodic model and then transform the durations of English segments to account for the contextual effect that might be caused by the source language. The multiplication factor λ was obtained using the mean durations of the segments during the training stage.

$$dur_{new} = \lambda * dur_{pred} \quad (17)$$

- **Gaussian based mapping system**

We have observed that there seem to be two separate gaussian distributions followed by the phones from arctic and the indic recording sets. Therefore, we built system which modifies the durations of individual phonemes based on the following:

$$dur_{new} = \lambda * \frac{\sigma_{indic}}{\sigma_{arctic}} * (dur_{pred} - \mu_{arctic}) + \mu_{indic} \quad (18)$$

where μ and σ indicate the mean and standard deviation of the individual phonemes respectively.

- **System with only look ahead models** - These are the models that do not take the previous context into account while predicting the duration of the current state.
- **System with only phonetic feature based models** - These are models trained without taking the names of phones into account and considering only the context in which they occur.

4.4.1 Front End

The evaluation results for the front end systems are presented in table ?? . The word to phone mapping based systems seem to outperform the rest of the systems across all the languages consistently. The transliteration based systems seem to be performing better than the basic phone mapping based systems in the languages they were deployed in, but seem to be lagging behind the word to phone mapping systems. An informal preference test showed that the transliterated system does reduce the accented nature of phone mapping procedure for some words, but the reduction itself was not found to be as much as that obtained by the word to phone based modeling approach. Therefore it appears that using a separate orthographic system might not necessarily result in the best quality synthesis. One reason for this might be that the errors in the transliteration process itself act as barriers hindering the system from reaching its full potential in terms of voice quality.

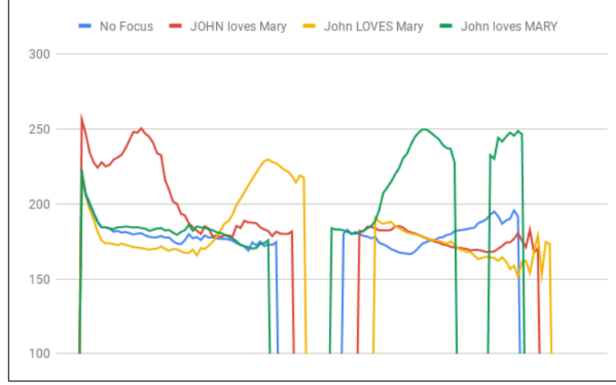


Figure 2: Plot of Fundamental Frequency(F_0) trajectories obtained from generated waves using proposed approach. We have employed our approach to render the sentence ‘John loves Mary’ with emphasis on individual words. The emphasized word is mentioned in capitals. The blue trajectory corresponds to F_0 when no emphasis was applied to any word. The plot highlights that the proposed approach allows explicit local control at the desired level in the generated speech. These and other samples are attached in the supplementary material for reference.

We analysed the sentences that were not preferred by either of the systems in this case and found atleast two interesting characteristics common across the languages: (1) The English parts of the sentences seems to be a bit dull and flattened out compared to the native language counterparts. We hypothesize that this might be due to the manner in which the models were trained: using separate corpora as opposed to a multilingual corpus which has codemixed sentences, leading to a train test mismatch no matter how shared the phonesets are. The model when predicting the English segments might therefore be tending towards the mean of the training observations due to lack of proper context. It might be interesting to see if this artifact can be corrected/addressed by either using a codemixed database or by using some form of adaptation. (2) There seems to be an uncharacteristic gap between the English words which have two parts, ex: shortcut as short PAU cut , download as down PAU load, etc. We have anticipated this behaviour at the point of switch between the languages and therefore explicitly tried to model the context, but we didnt expect this in case of English words. In addition, we have observed that the stress pattern in these instances became a bit wierd.

4.4.2 Prosodic Modeling

In this case, none of the systems that we have tried were successful in completely eliminating the seemingly disjoint(and therefore sudden variations in) speeds in the English and the Indic parts of test sentences. This is also clear from the evaluation results for the front end systems are presented in table ???. Surprisingly, none of the systems were able to outperform the baseline system, indicating that incorporating modifications artificially in the duration of the sentences is easily noticeable as unnatural by human evaluators.

5 De-Entanglement by Loss

5.1 Emphasis by Disentangling Tonal Heuristics (EDITH)

We are interested in incorporating inductive biases into the architecture of Seq2Seq TTS enabling explicit control over prosody. Ideally, we desire the priors to push model towards learning and exploiting latent representations that can generate different prosodic events. In addition, we also desire the learnt representations to be generalized so that the approach can be extended to different speakers as well as languages or language combinations such as code mixing. Given an utterance, a typical decomposition in the context of speech synthesis historically involves three components:

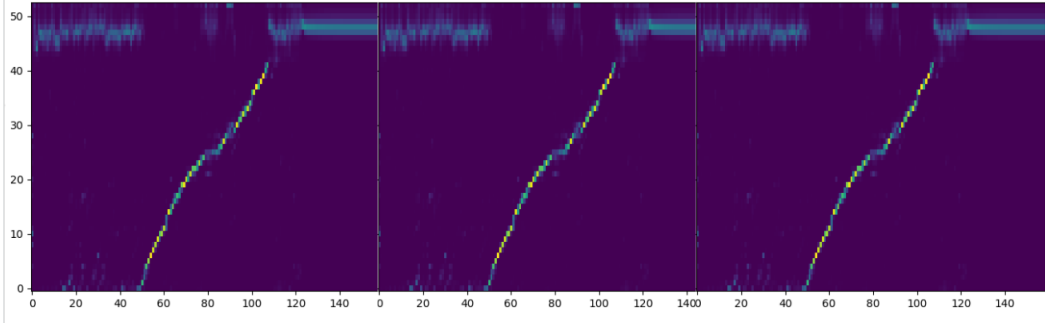


Figure 3: Attention alignment plots from different systems for the sentence 'John loves Mary'. To investigate the reason for sub optimal performance of subsentence systems, we have built three variants that differ in hyperparameters in terms of batch size and number of output frames per prediction. The first three subplots correspond to these variants built with $bsz=32, r=5$; $bsz=64, r=3$ and $bsz=64, r=5$ respectively. The final subplot on the extreme right is the one corresponding to the system built with access to full sentences.

spectrum, fundamental frequency and aperiodicity/voicing. Mathematically, this decomposition can be represented as

$$U = g(\text{spectrum}, F_0, \text{voicing}) \quad (19)$$

where g is a (non linear) function approximating the composition of the individual components. A general consensus states that spectrum typically is associated with the content spoken while the fundamental frequency is associated with the prosody - how the content is spoken. While there are many other parameters associated with prosody, F_0 can be considered one of the most important attributes. Based on this, we have chosen to incorporate priors in terms of F_0 . To accomplish this, we have extracted fundamental frequency using SPTK toolkit Group et al. [2009] and then quantized it into multiple bins with each bin spanning 25 Hz. In other words, continuous F_0 values were mapped to their discrete bin indices, resulting in ordinal F_0 values for each utterance. We then incorporated these values into the model architecture. We have investigated two ways of incorporating F_0 priors: (1) Explicit labeling by directly adding the ordinal F_0 as additional input to the model and (2) Implicit incorporation by adding prediction of ordinal F_0 as a secondary task in the model. The predicted F_0 is used as conditioning information to the decoder along with the input from soft attention. This approach can be interpreted as VQVAE?. It can also be seen as GSTWang et al. [2018] based encoding but our approach has two differences: (1) We do not use a different encoder for spectral information and (2) We explicitly constrain the latent classes to correspond to the quantized F_0 s.

5.2 Voice Building

As preprocessing, we perform the following steps:

- *Alignment and Segmentation:* Since our approach requires segment level information, we have performed alignment of the utterances using EHMM to obtain segment level duration.
- *Tokenization and G2P:* We consider any text entry separated by white space as a token. Once tokens are obtained, we have used US phoneset to perform mapping from graphemes to phonemes based on CMU pronunciation dictionary.
- *Acoustic Feature Extraction:* We perform acoustic feature vector extraction over a 50ms frames obtained by applying a hamming window with a frame shift of 12.5 msec. The speech files have been high pass filtered with cut off frequency of 60 Hz prior to feature extraction. We obtain 1025 dimensional linear and 80 dimensional mel frequencies.
- *Acoustic Model and Vocoder:* Our acoustic model and vocoder are extensions to the ones used in our baseline approach.
- *Refinement of Segment Boundaries:* For local control, it is important that we obtain accurate segment boundaries. Misalignment at this stage might lead to undesired consequences such

as vowels getting assigned zero F_0 . Therefore, we have employed a refinement step to improve the segment boundaries obtained from initial labeling. we utilize a different speech representation, MCEPs in this process. For each of the states obtained from segmentation, we extract acoustic feature vectors over a 5ms frames obtained by applying a hamming window. Spectral representation that we use is MCEPs and were extracted using the SPTK toolkit. The order of MCEP was chosen to be 24 with a frequency warping factor of 0.42 and a small value (1.0E-08) was added to the periodogram. For F_0 , we interpolate between unvoiced section ensuring breaks during silences and then apply a post smoothing using a 25 ms window. We examine each segment boundary and consider moving it forward or backward (by one frame) and investigate whether this decreases the distance between original and predicted frame. This process is performed over all the labels and then the models are rebuilt. The distance is measured in terms of unnormalized MCD including the energy coefficient but not the deltas. We have performed 10 iterations over the entire database as the improvement in MCD stopped at that point. The results of this procedure have been outlined in the table 3.

Table 3: Refinement of segment durations

Pass	Number of Moves	Mel Cepstral Distortion	F0 Error	Duration Error
1	58941	7.065	32.808	0.964
2	55467	7.049	32.712	0.964
3	51224	7.047	32.7	0.963
4	48930	7.041	32.4	0.959
5	47844	7.036	32.35	0.948
6	47342	7.022	32.23	0.946
7	46541	7.021	31.91	0.942
8	43451	7.018	31.82	0.942
9	42872	7.020	31.86	0.947

5.2.1 Sub Sentence Training

We posit that it is possible to improve the association learnt by acoustic model by using sub sentence training: selecting aligned segments of text and acoustics within a sentence. In addition, selecting segments within a sentence might lead to the model utilizing the available data more efficiently. We note that such an approach is already used for vocoding: Typical vocoders the authors are aware of are trained using aligned chunks of acoustic vectors and corresponding speech samples as opposed to full utterances. While this is due to GPU memory constraints in the context of vocoders, we investigate applying similar strategy to acoustic modeling. We believe that this facilitates local attention mechanism to better learn the mapping from phonetic space to the acoustic space.

6 Experiments

6.1 Data and Model Specifications

We have used data from LJSpeech dataset² to build our systems. We have used all of the 13100 sentences. The text was normalized manually to convert non standard forms (for ex. 1993) to written forms (nineteen ninety three). We have utilized 1300 sentences as validation data to tune our hyperparameters. We have quantized F_0 into bins based on absolute values at frame level. 25 Hz was chosen as the size of bin. This effectively resulted in a total of 14 bins.

6.2 Systems

We have built multiple variants to compare the performance of the proposed approach. Extending our baseline, we have built systems that employ quantized F_0 in their architectures. Additionally, we have also built models trained with access to only part of the full sentence. We refer to these systems

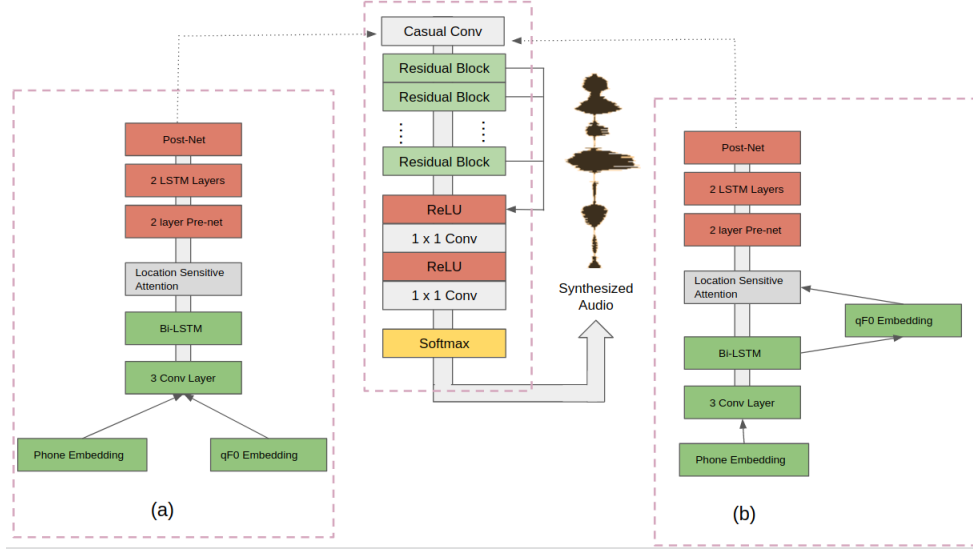


Figure 4: Architecture of our proposed approaches. (a) denotes the systems FSQ_i and SSQ_i and (b) denotes the systems FSQ_l and SSQ_l . The vocoder is same for both the systems.

as Full Sentence Quantized F0 (FSQ) and Sub Sentence Quantized F0 (SSQ) respectively. For each of the above systems, we have incorporated the prior information about F0 either at the input or as an additional loss term after the encoder. We refer to these systems with subscripts. For instance, the system employing full sentences for training with quantized F0 as inputs in addition to phoneme embeddings is referred to as FSQ_i . Similarly, the system employing sub sentence training but with additional loss term is referred to as SSQ_l . In addition to the proposed approaches, we have also built systems that employ AuToBI? labels. We have not built a sub sentence variant in this approach and refer to the full sentence variant as AUTOBI.

6.3 Evaluation

Evaluation was performed in the form of listening tests using Parlikar [2012]. We have conducted two types of listening tests: (1) Rating the naturalness in terms of Mean Opinion Score (MOS) on a scale of 1 (least natural) to 5 (highly natural) and (2) ABX Preference test on Emphasis where the users need to mention their preference towards either of the systems or state that they prefer neither. We have manually curated 50 sentences where the meaning was implied based on prosody. Participants were shown the entire sentence and its implication in parenthesis. An example sentence from our testset is ‘It looks like a starfish (but it really is not).’ Every system was used to generate this test set (in the mentioned example, the systems generated just the part ‘It looks like a starfish’ and not the part in parenthesis.) The participants are to mention their preference to the system that faithfully generates prosody in line with the information in parenthesis. We had 25 listeners and each participant rated 20 random sentences giving us a total of 500 ratings per pair of systems.

6.4 Systems

We have built multiple variants to compare the performance of the proposed approach. Extending our baseline, we have built systems that employ quantized F0 in their architectures. Additionally, we have also built models trained with access to only part of the full sentence. We refer to these systems as Full Sentence Quantized F0 (FSQ) and Sub Sentence Quantized F0 (SSQ) respectively. For each of the above systems, we have incorporated the prior information about F0 either at the input or as an additional loss term after the encoder. We refer to these systems with subscripts. For instance, the system employing full sentences for training with quantized F0 as inputs in addition to phoneme embeddings is referred to as FSQ_i . Similarly, the system employing sub sentence training but with additional loss term is referred to as SSQ_l . In addition to the proposed approaches, we have also

built systems that employ AuToBI? labels. We have not built a sub sentence variant in this approach and refer to the full sentence variant as AUTOBI.

Table 4: Results from Preference Test for Emphasis generation. The entries indicate preference values obtained by the systems in the first column against every other system in the subsequent columns. Each pair of systems received a total of 500 ratings. We have excluded the *No Preference* values from this table for brevity. However, they can be estimated based on the values in the table.

Config	FSQ_i	FSQ_l	SSQ_i	SSQ_l	AUTOBI
FSQ_i	-	112	396	363	441
FSQ_l	125	-	424	378	477
SSQ_i	91	86	-	235	278
SSQ_l	64	86	243	-	227
AUTOBI	47	19	219	256	-

6.5 Discussion

6.5.1 Proposed Approaches vs AuToBI based system

The preference evaluation results for the proposed approaches are presented in table 4. The full sentence based systems seem to outperform sub sentence as well as AuToBI based systems consistently. Within the full sentence systems, there does not seem to be a clear preference (27 out of the 50 participants chose no preference to either of the systems). This indicates that the full sentence systems seem identical in performance while clearly outperforming their other counterparts. An informal listening test in the scenarios where full sentence models were not preferred revealed an interesting finding: All these scenarios were when the emphasized word was the first in the sentence. Some listeners have expressed that they felt multiple words were focused in the sentence but nevertheless the intended word more or less stood out from the rest except when it was the first word in sentence. We hypothesize that this might be due to the data distribution and plan to investigate this further. The sub sentence based approaches seem to match the performance of AUTOBI systems while clearly under performing their full sentence counterparts. To investigate this, we have looked at the alignment distributions for various sub sentence systems. We have presented the alignment information of three sub sentence systems in figure 3.

7 De-Entanglement by Divergence

7.1 Visual Question Answering

Visual Question Answering (VQA) involves answering a natural language query about an image. Questions can be arbitrary and they encompass many sub-problems in computer vision: (1) Object recognition (2) Object detection (3) Attribute classification (4) Scene classification (5) Counting. VQA is characterized by wide ranging applications from helping visually impaired people through human machine interaction. It has the potential to serve as an effective media content retrieval framework. A primary form of implementing a VQA system would be to use a bucketing approach: by learning image and text features and fusing them to get an answer. In recent years, there have been several extensions to the trivial approach mentioned above ?; ?; ?; ? claim to learn good representations of abstract concepts needed to answer questions. However, it has been shown Agrawal et al. [2017] that most of the approaches capture surface level correlations and fail to handle unseen novel combinations during test time.

In this work, we investigate approaches to improve compositionality in VQA, where we explicitly focus on learning compositionality between concepts and objects. Language and vision are inherently composite in nature. For example different questions share substructure viz *Where is the dog?* and *Where is the cat?* Similarly images share abstract concepts and attributes viz *green pillow* and *green light*. Hence it is vital not only to focus on understanding the information present across both these modalities, but also to model the abstract relationships so as to capture the unseen compositions of

seen concepts at test time. Achieving this would then allow the model to generalize better by learning an inference procedure, resulting in true success on this task.

In this work, we propose **JUPITER** - **J**ustification via **P**ointwise combination of **I**mage and **T**ext based on **E**xpected **R**ewards, is built on top of the Neural Module Networks ?. This is motivated from our hypothesis that generating captions can provide additional information to improve VQA. Additionally, JUPITER uses Reward Augmented Maximum Likelihood Norouzi et al. [2016], which improves caption generation.

7.2 Related Work

Visual Question Answering: ? provided a strong baseline for VQA using a simple CNN-LSTM architecture, and achieved 64.6% on the VQA 1.0 Openended QA challenge. This further proved that the dataset is biased. ? introduced grounding to prevent the model from memorizing this bias. Similarly, ? used a zero-shot training approach to improve the generalizability of the model, and prevent the model to learn the bias. However, recently ? showed that most models degrade in performance when tested on unseen samples. In this work, we aim to tackle this lack of generalizability.

Neural Module Networks: To the best of our knowledge, the work by authors in ? and ? is the only work so far that explicitly uses a divide and conquer approach for compositionality. Natural language questions are best answered when broken down into their subparts. The authors use a similar intuition and propose a modular architecture. This approach first parses the natural language question into linguistic components. Second, each component is assigned to a sub-module that solves a single task. Lastly, these modules are then composed into an appropriate layout that predicts an answer for each training example. Such a dynamic network not only helps learning object-object relationships well via compositionally, but also improves the reasoning abilities of the model.

Multitask Learning: There have been number of works that explore multitask learning as an approach to joint learning of vision and language tasks. In one such work ?, authors learn related regions of the image by simultaneously training three different semantic tasks - scene graph generation, object detection, and image captioning. A multi-task learning architecture was also proposed by ? for image captioning where they enable sharing of a CNN encoder and an LSTM decoder between object classification task and the syntax generation tasks. ?? show multitask learning reduces overfitting in limited-resource settings, and can learn representations to improve downstream (part-of-speech tagging and name-entity recognition) tasks. Our purpose of joint training in multitask learning is to provide regularization on the learned features for VQA, with an added benefit of achieving better performance on the auxiliary task (of generating captions).

Incorporating additional knowledge: In ? authors show that incorporating captions helps resolve some ambiguities in visual question answering. In ? authors first obtain captions and then use them for improving VQA via the framework of predicate logic. In ? authors learn attributes from an image using an image labeling and then query using an external knowledge base.

7.3 JUPITER - Justification by Pointwise combination of Image and Text based on Expected Rewards

The key motivation of this approach [depicted in Figure: 5] was to manipulating the loss function to account for captions. We hypothesize that explicitly accounting for captions in the loss function will affect the downstream VQA predictions. Figure 5 shows the framework architecture and functioning.

7.3.1 Model Description

Our model uses Neural Module Networks (NMN), along with multiple proposed extensions. More specifically, we use the following extensions:

- *Multitask Learning:* We modify the decoder to perform multiple tasks namely, caption generation and VQA. We use the attention grid generated by ‘Find’ module in the NMN, the encoded question layout, and the input image to generate captions in an auto regressive.

Our hypothesis is that using this conditioning, we can force the model to generate attention grid that is suitable to both downstream tasks, in turn improving VQA performance.

- *Conditional Generation*: As opposed to multitask learning approach, in this extension we explicitly provide the generated captions as input to VQA decoder. More specifically, we train the model to first generate a relevant image caption using previously defined setup. Next we condition the answer decoder on the generated caption. The intuition is that providing the model with information more explicitly will help to predict answers based on this information.
- *Re-weighting*: In this extension, we re-weight the answer hypothesis using the generated caption. We hypothesize that this will help the model to disambiguate between answer logits that have maximum entropy.
- *M-Hybrid and C-Hybrid*: In order to harvest complimentary benefits from our primary extensions, we also implemented two hybrid systems. M-Hybrid extension combined multi-task learning and re-weighting approach, and the C-Hybrid extension combined conditional generation and re-weighting approach.
- *Reinforcement Learning*: This extension uses Reward Augmented Maximum Likelihood (RAML) as opposed to Maximum Likelihood (MLE) for generating captions. The intuition for this extension was to enable the agent to generate captions that will help the model to answer the given question. More specifically, the agent at each caption generation step can perform one of the two tasks: (1) Generate next word for the captions or (2) Answer the question based on caption generated so far. The agent is rewarded based on VQA accuracy. Since training with REINFORCE is known to be unstable, we use a baseline wherein we generate answers based on the final hidden state of a decoder trained using MLE.

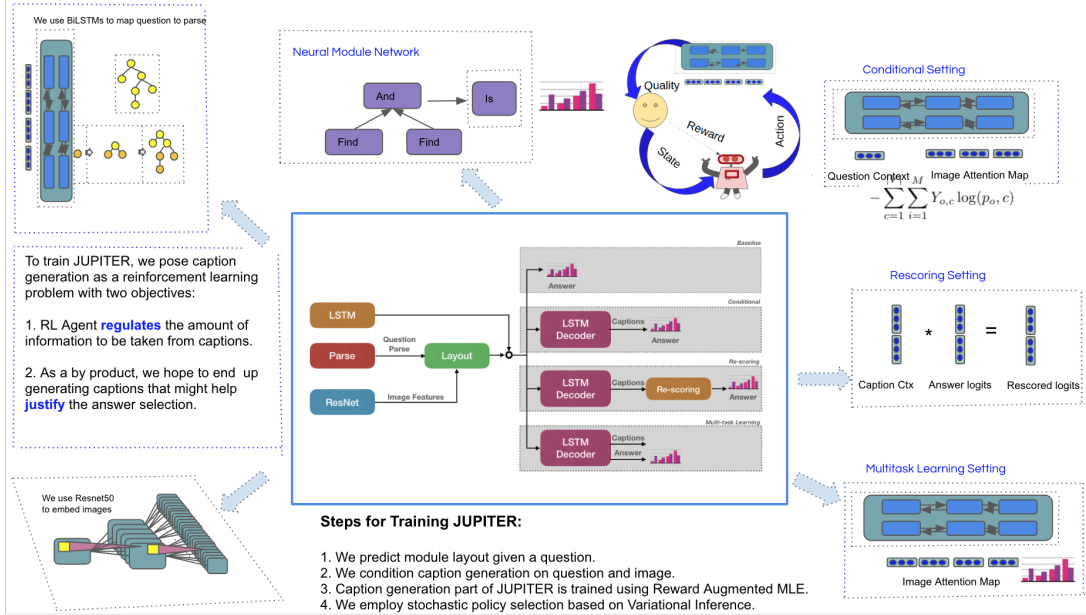


Figure 5: Justification by Pointwise combination of Image and Text based on Expected Rewards

7.3.2 Learning

We denote input question as Q and input image as I . L^* denotes the gold layout for Q and C^* is gold caption for I . We denote L as the layout generated by NMN for Q . C is caption generated from JUPITER. We denote answer classes by y and the correct answer class by y^* . T is the training data samples of type (I, Q, y^*) . Next, we describe the objective function for each extension in detail.

- *Multitask Learning*: We use a two-part objective function for multitask learning. The first part is generating captions from the input and the second is generating answer logits from

the input and the generated NMN layout.

$$L(\theta) = \sum_{(I, Q, y^*) \in T} \log P_\theta(y|I, Q, L) + \log P_\theta(C|I, Q) \quad (20)$$

- *Conditional Generation*: This extension uses a similar objective function. However, we generate answer logits from the input, generated NMN layout as well as the generated captions.

$$L(\theta) = \sum_{(I, Q, y^*) \in T} \log P_\theta(y|I, Q, L, C) + \log P_\theta(C|I, Q) \quad (21)$$

- *Re-weighting*: This extension uses a similar objective as conditioned generation. Further, for re-weighting we define new answer logits y' .

$$y' = C_T y \quad (22)$$

where, C_T is the final hidden state of generated caption, and y is the previous answer logits. The updated objective function is:

$$L(\theta) = \sum_{(I, Q, y^*) \in T} \log P_\theta(y'|I, Q, L, C) + \log P_\theta(C|I, Q) \quad (23)$$

- *Reinforcement Learning*: The agent transitions between generating next word in the caption and generating final answer. The agent receives minibatch VQA accuracy as its reward. The Baseline we use to stabilize the training and the expected reward of our agent respectively are expressed as

$$L_{baseline}(\theta) = \sum_{(I, Q, y^*) \in T} \log P_\theta(y|I, Q, L, C) \quad (24)$$

We use cross-entropy loss to train the model. We jointly train our captions module in JUPITER alongside NMN, which learns a question layout L .

7.4 Dataset and Input Modalities

VQA dataset by ? has 265016 images, 614163 questions. The dataset consists of 82,783 training, 40,504 validation, and 40,775 test images. Each image has 3 questions on average and 10 ground truth answers. Questions as well as answers are open ended, accounting for a more real-world scenario. The questions are rich in a way, as they require the model to have complex reasoning and understanding abilities.

8 Results and Discussion

In this section, we discuss the results from our proposed approaches viz. JUPITER, VENUS and MARS, and compare them against our baselines. Table 5 consolidates the results of our experiments. To better understand the performance of these models, we report the performance across different answer categories namely, Number, Yes/No and Other. The overall best baseline model for VQA is NMN by ?.

8.1 Results: Baseline Models

The input to our baseline models is the image and the question. We do not use any external knowledge. Our results show that the baseline models have highest accuracy on the Yes/No questions. However, the Number type questions often require deeper understanding of the image, and so our baselines have lowest performance on them. Humans tend to have low agreement for Yes/No questions. We attribute this to question ambiguity or missing information in the image. It has to be noted that our implementation of the NMN baseline achieves better scores compared to the open source original implementation. This can be attributed to the presence of additional modules in our implementation, specifically OR, COUNT, FILTER, and EXIST modules.

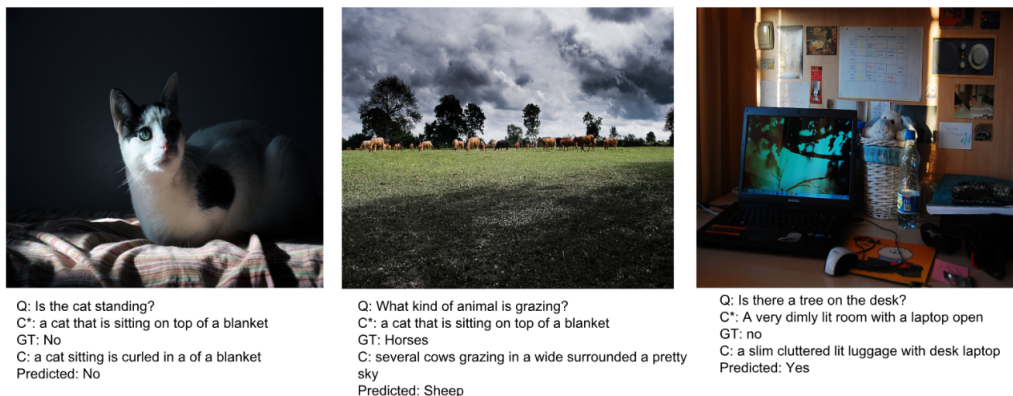


Figure 6: Qualitative Analysis from JUPITER: left image depicts a scenario where generating caption helped the model in selection of the right answer. Image in the center depicts a scenario where captions end up confusing the model. Image in the right most highlights an interesting scenario where the generated caption seems irrelevant.

Model	System	Input	Number	Yes/No	Other
Human	Best	Image + Question	83.39	95.77	72.67
Human	Worst	Image + Question	65.28	46.52	78.02
NMN	Baseline (Replicated)	Image + Question	23.31	63.93	26.65
NMN	Baseline (Our implementation)	Image + Question	26.35	64.49	31.55
RNN	Baseline	Image + Question	19.34	57.82	17.77
VED	Baseline	Image + Question	17.76	58.00	10.43
RNN	MARS	Image + Question + Caption*	23.09	57.88	18.22
RNN	MARS	Question + Caption*	21.72	57.95	21.59
VED	VENUS	Image + Question + Caption*	19.25	57.83	10.10
VED	VENUS	Question + Caption*	18.13	58.10	10.33
NMN	M-Hybrid	Image + Question + Caption	26.31	64.27	30.43
NMN	C-Hybrid	Image + Question + Caption	27.48	65.8	32.2
NMN	JUPITER	Image + Question + Caption	32.82	67.95	33.15

Table 5: Results from human, baselines and proposed approaches. * denotes systems that employ Gold captions

8.2 Results: Proposed Models

Looking at the objective evaluation results from table 5, it is clear that incorporating captions leads to improvements across the approaches. This result empirically validates our hypothesis related to captions: Captions help VQA. To understand the extent of this, we have also performed ablation analysis wherein we have used just captions to answer the question ignoring the input image. Surprisingly, systems built in this fashion seem to perform better than our baselines. This leads to an interesting observation: *Captions seem to contain supplementary and in some cases complementary information to the images themselves*. However, we acknowledge that proving such hypothesis would require additional experimentation. For instance, it would be interesting to perform similar ablation analyses employing computationally more powerful frameworks such as attention as baselines or adding more visual information such as ground truth bounding boxes. It is also interesting to note that the proposed approaches achieve better scores compared against the *worst* human performance in Yes/No category.

Our approach JUPITER outperforms all other approaches across all the categories. In addition, within the models employing module networks, the system employing reinforcement learning outperforms other approaches. This is in line with our hypothesis related to Reward Augmented Maximum Likelihood and raises interesting questions related to *comparison between supervised approaches such as Maximum Likelihood and their reward based reinforcement counterparts*. It would be interesting to perform a much larger scale evaluation comprehensively comparing the effectiveness of these approaches in the context of downstream tasks. In figure 6, we present some scenarios that highlight the way captions get utilized for answering question about the corresponding images.

9 Conclusion

In this writeup, I present an argument for De-Entanglement: a property that has potential to isolate the factors of variation in the data distribution. I posit that explicitly isolating relevant factors using such an approach is helpful with respect to the downstream tasks. I introduce the concept using analogies and then present mathematical analysis from information theory highlighting different ways to accomplish ‘De-Entanglement’. I point out how this concept can be interpreted as different popular deep learning approaches.

References

- Aishwarya Agrawal, Aniruddha Kembhavi, Dhruv Batra, and Devi Parikh. C-vqa: A compositional split of the visual question answering (vqa) v1. 0 dataset. *arXiv preprint arXiv:1704.08243*, 2017.
- Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don’t just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4971–4980, 2018.
- Abdul Fatir Ansari and Harold Soh. Hyperprior induced unsupervised disentanglement of latent representations. *arXiv preprint arXiv:1809.04497*, 2018.
- Leonardo Badino, Claudia Barolo, and Silvia Quazza. Language independent phoneme mapping for foreign tts. In *Fifth ISCA Workshop on Speech Synthesis*, 2004.
- Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in beta-vae. *arXiv preprint arXiv:1804.03599*, 2018a.
- Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in Beta VAE. *arXiv preprint arXiv:1804.03599*, 2018b.
- Nick Campbell. Talking foreign. In *Proc. Eurospeech*, pages 337–340, 2001.
- Tian Qi Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems*, pages 2615–2625, 2018.
- Xi Chen, Diederik P Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. Variational lossy autoencoder. *arXiv preprint arXiv:1611.02731*, 2016.
- Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. In *Advances in neural information processing systems*, pages 2980–2988, 2015.
- Yuntian Deng, Yoon Kim, Justin Chiu, Demi Guo, and Alexander Rush. Latent alignment and variational attention. In *Advances in Neural Information Processing Systems*, 2018.
- Janek Ebbers, Jahn Heymann, Lukas Drude, Thomas Glarner, Reinhold Haeb-Umbach, and Bhiksha Raj. Hidden markov model variational autoencoder for acoustic unit discovery. In *INTERSPEECH*, pages 488–492, 2017.

- Naresh Kumar Elluru, Anandaswarup Vadapalli, Raghavendra Elluru, Hema Murthy, and Kishore Prahallad. Is word-to-phone mapping better than phone-phone mapping for handling english words? In *ACL (2)*, pages 196–200, 2013.
- Babak Esmaeili, Hao Wu, Sarthak Jain, Alican Bozkurt, N Siddharth, Brooks Paige, Dana H Brooks, Jennifer Dy, and Jan-Willem van de Meent. Structured disentangled representations. *stat*, 1050:12, 2018.
- David Gauntlett. The lego system as a tool for thinking, creativity, and changing the world. *LEGO Studies: Examining the Building Blocks of a Transmedial Phenomenon*, pages 1–16, 2014.
- Spandana Gella, Kalika Bali, and Monojit Choudhury. “ye word kis lang ka hai bhai?” testing the limits of word level language identification, 2014.
- Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913, 2017.
- SPTK Working Group et al. Speech signal processing toolkit (sptk). *http://sp-tk.sourceforge.net*, 2009.
- Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.
- SC Keller, Michael Stanley Bessell, Anna Frebel, AR Casey, Martin Asplund, HR Jacobson, Karin Lind, John Edward Norris, David Yong, Alexander Heger, et al. A single low-energy, iron-poor supernova as the source of metals in the star smss j031300. 36- 670839.3. *Nature*, 506(7489):463, 2014.
- Diederik P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*, pages 4743–4751, 2016.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 2012.
- Javier Latorre, Koji Iwano, and Sadaoki Furui. Polyglot synthesis using a mixture of monolingual corpora. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP’05). IEEE International Conference on*, volume 1, pages I–1. IEEE, 2005.
- Francesco Locatello, Stefan Bauer, Mario Lucic, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. *arXiv preprint arXiv:1811.12359*, 2018.
- Alireza Makhzani and Brendan J Frey. Pixelgan autoencoders. In *Advances in Neural Information Processing Systems*, pages 1975–1985, 2017.
- Mikiko Mashimo, Tomoki Toda, Kiyohiro Shikano, and Nick Campbell. Evaluation of cross-language voice conversion based on gmm and straight. 2001.
- Pieter Muysken. *Bilingual speech: A typology of code-mixing*, volume 11. Cambridge University Press, 2000.
- Mohammad Norouzi, Samy Bengio, Navdeep Jaitly, Mike Schuster, Yonghui Wu, Dale Schuurmans, et al. Reward augmented maximum likelihood for neural structured prediction. In *Advances In Neural Information Processing Systems*, 2016.
- Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

- Alok Parlikar. TestVox: web-based framework for subjective evaluation of speech synthesis. *Open-source Software*, 2012.
- Douglas B Paul and Janet M Baker. The design for the wall street journal-based csr corpus. In *Proceedings of the workshop on Speech and Natural Language*, pages 357–362. Association for Computational Linguistics, 1992.
- Sai Krishna Rallabandi and Alan Black. Variational Attention using Articulatory priors for generating code mixed speech using monolingual corpora. In *proceedings of Interspeech*, 2019.
- Sai Krishna Rallabandi, Anandaswarup Vadapalli, Sivanand Achanta, and S Gangashetty. Iiit hyderabad’s submission to the blizzard challenge. In *Proc. of Blizzard Challenge 2015*.
- Mirco Ravanelli and Yoshua Bengio. Interpretable convolutional filters with sincnet. *arXiv preprint arXiv:1811.09725*, 2018.
- David Rousseau and Sotirios Tsaftaris. Data augmentation techniques for deep learning. In *Tutorial Session, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.
- Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017.
- Benjamin Schumacher. Quantum coding. *Physical Review A*, 51(4):2738, 1995.
- Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, et al. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. *arXiv preprint arXiv:1712.05884*, 2017.
- Sunayana et. al Sitaram. Experiments with cross-lingual systems for synthesis of code-mixed text. In *9th ISCA Speech Synthesis Workshop*, pages 76–81, 2015.
- Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.
- Laura Mayfield Tomokiyo, Alan W Black, and Kevin A Lenzo. Foreign accents in synthetic speech: development and evaluation. In *Interspeech*, pages 1469–1472, 2005.
- Aaron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.
- Nidhi Vyas, SaiKrishna Rallabandi, Lalitesh Morishetti, Eduard Hovy, and Alan W Black. Learning disentangled representation in latent stochastic models: A case study with image captioning. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019.
- Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, et al. Tacotron: Towards end-to-end speech synthesis. *arXiv preprint arXiv:1703.10135*, 2017.
- Yuxuan Wang et al. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. *arXiv preprint arXiv:1803.09017*, 2018.
- Yi Zhang and Jianhua Tao. Prosody modification on mixed-language speech synthesis. In *Chinese Spoken Language Processing, 2008. ISCSLP’08. 6th International Symposium on*, pages 1–4. IEEE, 2008.