Proposed a novel data-driven non-heuristic data augmentation method for unsupervised domain adaptation, which requires zero in-domain labeled data. Achieved up to 35% and 40% absolute word error rate reduction in mismatched domains on CHiME-4 and Aurora-4 respectively.

### Robust Automatic Speech Recognition

An ASR system often degrades significantly when testing on a domain mismatched from the training data. Here are a few ways to achieve robustness:
- Use domain-invariant acoustic features.
- Enhance speech (convert out-of-domain data to in-domain data).
- Train an ASR system with as much, and as diverse a dataset as possible.
- Use more data to utilize the full capacity of neural network models.

### Unsupervised Learning of Latent Factors with VAEs

#### Variational Autoencoder (VAEs)

Consider a speech dataset $D = \{x^{(i)}\}_{i=1}^N$ of i.i.d. speech segments. Each $x$ is assumed to be generated by:
1. Draw a latent variable $z$ from $p(z) = N(0, I)$.
2. Draw an observed variable $x$ from $p(x|z) = N(x|f(z), \exp(\phi_{xu}(z)))$.
3. A variational inference model $q_{\theta}(z|x)$ is introduced to approximate the intractable true posterior $p_{\theta}(z|x)$.

Objective Function: Variational Lower Bound

\[
\mathcal{L}(\theta, \phi; x) = \mathbb{E}_{q_{\theta}(z|x)}[\log p_{\phi}(p(z|x)) - D_{KL}(q_{\theta}(z|x)||p_{\theta}(z|x))]
\]

Properties of Latent Variables:
- The prior $p(z)$ is a factorial distribution.
- VAEs are encouraged to encode independent physical attributes (e.g., speaker identity, phonemes) into orthogonal subspaces.

### Latent Attribute Representation

It is suggested and empirically verified in the previous work [Hsu et. al., 2017]:
1. Conditional prior of $z$ and attributes $a$ being $r$ (e.g., phoneme being $a$):
   \[
   p(z|a = r) = N(z|\mu_a, \Sigma_a)
   \]
2. $\mu_a \perp \mu_a$ if $r$ and $\gamma$ are independent attributes that affect the realization of speech.
   \[\mu_a\] is defined as latent attribute representation for $r$.

### Estimating Latent Attribute Representations

\[
\mu_a \approx \sum_{i=1}^{N} f_a(x^{(i)}; \phi_{xu}) \delta(z^{(i)}; \mu_a, \Sigma_a)
\]

#### Nuisance Attributes and Data Augmentation

Nuisance Attributes: factors that affect the surface form of a speech utterance but not the linguistic content, such as speaker identity, channel, background noise.
- Nuisance attributes are independent from linguistic content.

Given a labeled utterance, (1) encode, (2) modify the latent subspace that models these attributes, and (3) decode, to generate augmented labeled data.

Type I: Nuisance Attribute Replacement (Repl.)

Replace the nuisance attribute of one utterance with that of another utterance. Transformation vector:

\[
\Delta u = \mu_{a\text{orig}} - \mu_{a\text{rep}}
\]

Type II: Nuisance Subspace Perturbation (Pert.)

Discover the latent nuisance subspace and perturb that subspace:
1. Determine latent nuisance subspace with principle component analysis (PCA) given a dataset of $D$ utterances, we can compute $[\mu_1, \mu_2,...,\mu_d]$ on which we apply PCA.
   \[\phi_{xu}(z)\] with associated eigenvalues $[\gamma_1, \gamma_2,...,\gamma_d]$.
2. Sample a transformation vector for soft latent nuisance subspace perturbation:

### Experiment Setup

#### Datasets
- CHiME-4: the training set consists of 1660 real noisy utterances and 7138 WSJ0 ST-84 clean utterances.
- Aurora-4: multi-condition speech dataset, 2 microphone types, 6 noise types, 4620 WSJ0-based utterances.

#### VAE Model
- Input: $x$ is a segment of 20 frames, represented as mel-scale filter bank coefficients (Flbank).
- Encoder/Decoder: two-layer LSTM with 512 hidden units. Adam optimizer.
- Training Set: all conditions (clean + noisy).

#### ASR Acoustic Model
- Transcripts available for only clean set. augmentation is based on clean data.
- Three-layer LSTM with 1024 cells and 512-node projection (CHiME-4) / six layer fully-connected with 1024 hidden units (Aurora-4).

### CHiME-4 Results

<table>
<thead>
<tr>
<th>Setting</th>
<th>Avg. Method</th>
<th>Fold</th>
<th>WER (%)</th>
<th>WER (%) by Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Orig.</td>
<td>1</td>
<td>36.22</td>
<td>36.22</td>
</tr>
<tr>
<td>2</td>
<td>Repl. Clean</td>
<td>1</td>
<td>20.03</td>
<td>20.03</td>
</tr>
<tr>
<td>3</td>
<td>Repl. Noisy</td>
<td>1</td>
<td>26.31</td>
<td>26.31</td>
</tr>
<tr>
<td>5</td>
<td>Pert. y = 1</td>
<td>1</td>
<td>20.01</td>
<td>20.01</td>
</tr>
<tr>
<td>2</td>
<td>Uni-Pert. y = 1</td>
<td>1</td>
<td>19.70</td>
<td>19.70</td>
</tr>
<tr>
<td>3</td>
<td>Repl. Pert. y = 1</td>
<td>1</td>
<td>19.75</td>
<td>19.75</td>
</tr>
<tr>
<td>4</td>
<td>Pert. y = 0.5</td>
<td>1</td>
<td>19.55</td>
<td>19.55</td>
</tr>
<tr>
<td>5</td>
<td>Pert. y = 1</td>
<td>1</td>
<td>20.01</td>
<td>20.01</td>
</tr>
<tr>
<td>6</td>
<td>Pert. y = 1</td>
<td>1</td>
<td>19.99</td>
<td>19.99</td>
</tr>
<tr>
<td>7</td>
<td>Orig. + Repl. Noisy 2</td>
<td>1</td>
<td>25.26</td>
<td>25.26</td>
</tr>
<tr>
<td>8</td>
<td>Repl. Pert. y = 1</td>
<td>1</td>
<td>25.26</td>
<td>25.26</td>
</tr>
<tr>
<td>9</td>
<td>Pert. y = 1</td>
<td>1</td>
<td>19.82</td>
<td>19.82</td>
</tr>
</tbody>
</table>

| Table: CHiME-4 development set word error rate of acoustic models trained on different augmented sets. We showed the following results:
- Correctness of soft latent nuisance subspace perturbation
- Effectiveness of both replacement and perturbation, and superiority of the latter.
- Benefit of generating more augmented data.

### Aurora-4 Results

<table>
<thead>
<tr>
<th>Setting</th>
<th>Baselines</th>
<th>Mean</th>
<th>WER (%)</th>
<th>WER (%) by Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Clean DNN/HMM -</td>
<td>16.22</td>
<td>15.50</td>
<td>15.50</td>
</tr>
<tr>
<td>1</td>
<td>DNN-PPL -</td>
<td>15.31</td>
<td>15.10</td>
<td>15.10</td>
</tr>
<tr>
<td>2</td>
<td>Orig.</td>
<td>18.76</td>
<td>18.76</td>
<td>18.76</td>
</tr>
</tbody>
</table>

| Table: Aurora-4 test set word error rate of acoustic models trained on different augmented sets.
- Outperformed a state-of-the-art domain adversarial training-based method (DAA).
- Matched the performance of an enhancement-based method (DNN-PPL), which however requires parallel data.