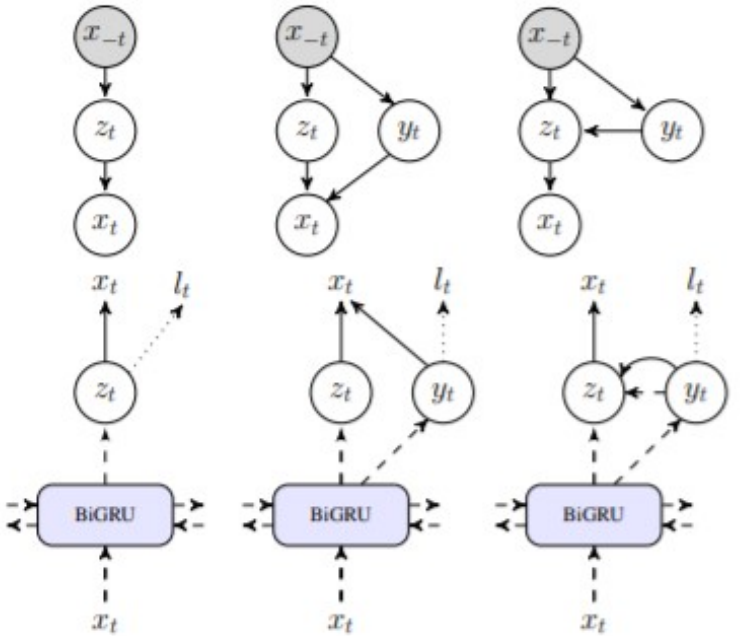
Variational Sequential Labelers for Semi-Supervised Learning

Mingda Chen **Qingming Tang** **Karen Livescu** **Kevin Gimpel**
Toyota Technological Institute at Chicago, Chicago, IL, 60637, USA
{mchen,qmtang,klivescu,kgimpel}@ttic.edu

Limited annotated data

- One approach for both **regularization** and **semi-supervised training** is to design latent-variable generative models and then develop neural variational method for learning and inference.

Proposed Methods



(a) VSL-G (b) VSL-GG-Flat (c) VSL-GG-Hier

Figure 1: Variational sequential labelers. The first row shows the original graphical models of each variant where shaded circles are observed variables. The second row shows how we perform inference and learning, showing inference models (in dashed lines), generative models (in solid lines), and classifier (in dotted lines). All models are trained to maximize $p_\theta(x_t|x_{-t})$ and predict the label l_t.

Proposed Methods

- We denote the input word sequence by $x_{1:T}$, the corresponding label sequence by $l_{1:T}$, the input words other than the word at position t by x_{-t}, the generative model by $p_\theta(\cdot)$, and the posterior inference model by $q_\varphi(\cdot)$.

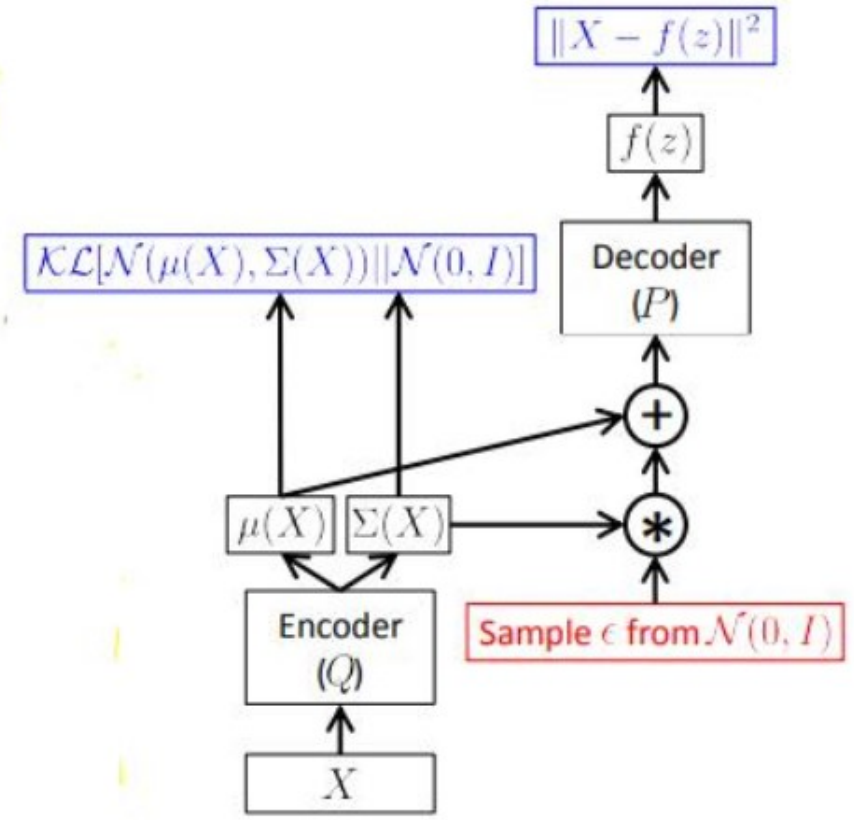
1. Single Latent Variable
2. Flat Latent Variables
3. Hierarchical Latent Variables

Background: Variational Autoencoders

$$\log\ p_\theta(x_{1:T}) \geq$$

$$\mathop{\mathbb{E}}_{z\sim q_\phi(\cdot|x_{1:T})}\left[\log p_\theta(x_{1:T}|z) - \log \frac{q_\phi(z|\ x_{1:T})}{p_\theta(z)}\right] =$$

$$\underbrace{\mathop{\mathbb{E}}_{z\sim q_\phi(\cdot|x_{1:T})}[\log p_\theta(x_{1:T}|z)]}_{\text{Reconstruction Loss}} - \underbrace{KL(q_\phi(z|x_{1:T})\|p_\theta(z))}_{\text{KL divergence}} \tag{1}$$



Single Latent Variable(VSL-G)

- VSL maximizes the conditional probability of $p_\theta(x_t \mid x_{-t})$ and minimizes a classification loss using the latent variables as the input to the classifier

We begin by defining a basic VSL and corresponding parametrization, which will also be used in other variants. This first model (which we call VSL-G and show in Figure 1a) has a Gaussian latent variable at each time step. VSL-G uses two training objectives; the first is similar to the lower bound on log-likelihood used by VAEs:

$$\log\ p_\theta(x_t|x_{-t}) \geq \mathop{\mathbb{E}}_{z_t \sim q_\phi(\cdot|\ x_{1:T}, t)} [\log p_\theta(x_t|\ z_t) - \log \frac{q_\phi(z_t|\ x_{1:T}, t)}{p_\theta(z_t|\ x_{-t})}] = \mathop{\mathbb{E}}_{z_t \sim q_\phi(\cdot|\ x_{1:T}, t)} [\log p_\theta(x_t|\ z_t)] - KL(q_\phi(z_t|\ x_{1:T}, t) \| \ p_\theta(z_t|\ x_{-t})) = U_0(x_{1:T}, t) \quad (2)$$

VSL-G additionally uses a classifier f on the latent variable z_t which is trained with the following objective:

$$C_0(x_{1:T}, l_t) = \mathop{\mathbb{E}}_{z_t \sim q_\phi(\cdot|x_{1:T}, t)} [-\log f(l_t|z_t)] \quad (3)$$

The final loss is

$$L(x_{1:T}, l_{1:T}) = \sum_{t=1}^{T} [C_0(x_{1:T}, l_t) - \alpha U_0(x_{1:T}, t)]$$

Flat Latent Variables

We next consider ways of factorizing the functionality of the latent variable into label-specific and other word-specific information. We introduce VSL-GG-Flat (shown in Figure 1b), which has two conditionally independent Gaussian latent variables at each time step, denote z_t and y_t for time step t.

$$
\begin{aligned}
&\log\ p_\theta(x_t|x_{-t}) \geq \\
&\quad \mathop{\mathbb{E}}_{z_t,y_t\sim q_\phi(\cdot|x_{1:T},t)}[\log p_\theta(x_t|\ z_t, y_t) \\
&\quad - \log \frac{q_\phi(z_t|x_{1:T},t)}{p_\theta(z_t|x_{-t})} - \log \frac{q_\phi(y_t|x_{1:T},t)}{p_\theta(y_t|x_{-t})}] \\
&\quad = \mathop{\mathbb{E}}_{z_t,y_t\sim q_\phi(\cdot|x_{1:T},t)}[\log p_\theta(x_t|z_t,y_t)] \\
&\quad - KL(q_\phi(z_t|x_{1:T},t)\|p_\theta(z_t|x_{-t})) \\
&\quad - KL(q_\phi(y_t|x_{1:T},t)\|p_\theta(y_t|x_{-t})) \\
&\quad = U_1(x_{1:T},t)
\end{aligned}
\tag{4}
$$

The classifier f is on the latent variable y_t and its loss is

$$
C_1(x_{1:T},l_t) = \mathop{\mathbb{E}}_{y_t\sim q_\phi(\cdot|x_{1:T},t)}[-\log f(l_t|y_t)] \tag{5}
$$

The final loss for the model is

$$
L(x_{1:T},l_{1:T}) = \sum_{t=1}^{T}[C_1(x_{1:T},l_t) - \alpha U_1(x_{1:T},t)] \tag{6}
$$

Where α is a trade-off hyperparameter.

Hierarchical Latent Variables

- This model encodes the intuition that the word-specific latent information zt may differ depending on the class-specific information of yt

For this model, the derivations are similar to Equations (4) and (5). The first is:

$$
\begin{aligned}
&\log\ p_\theta(x_t|x_{-t}) \geq \\
&\quad \mathop{\mathbb{E}}_{z_t,y_t\sim q_\phi(\cdot|x_{1:T},t)}[\log p_\theta(x_t|z_t) \\
&\quad - \log \frac{q_\phi(z_t|y_t,x_{1:T},t)}{p_\theta(z_t|y_t,x_{-t})} - \log \frac{q_\phi(y_t|x_{1:T},t)}{p_\theta(y_t|x_{-t})}] \\
&\quad = \mathop{\mathbb{E}}_{z_t,y_t\sim q_\phi(\cdot|x_{1:T},t)}[\log p_\theta(x_t|z_t)] \\
&\quad - KL(q_\phi(z_t|y_t,x_{1:T},t)\|p_\theta(z_t|y_t,x_{-t})) \\
&\quad - KL(q_\phi(y_t|x_{1:T},t)\|p_\theta(y_t|x_{-t})) \\
&\quad = U_2(x_{1:T},t)
\end{aligned}
\tag{7}
$$

The classifier f uses y_t as input and is trained with the following loss:

$$
C_2(x_{1:T},l_t) = \mathop{\mathbb{E}}_{y_t\sim q_\phi(\cdot|x_{1:T},t)}[-\log f(l_t|y_t)] \tag{8}
$$

Note that C_1 and C_2 have the same form. The final loss is

$$
L(x_{1:T},l_{1:T}) = \sum_{t=1}^{T}[C_2(x_{1:T},l_t) - \alpha\, U_2(x_{1:T},t)] \tag{9}
$$

Where α is a trade-off hyperparameter.

Experiments

Twitter POS Dataset. The Twitter dataset has 25 tags. We use OCT27TRAIN and OCT27DEV as the training set, OCT27TEST as the development set, and DAILY547 as the test set. We randomly sample {1k, 2k, 3k, 4k, 5k, 10k, 20k, 30k, 60k} tweets from 56 million English tweets as our unlabeled data and tune the amount of unlabeled data based on development set accuracy.

UD POS Datasets. The UD datasets have 17 tags. We use French, German, Spanish, Russian, Indonesian and Croatian. We follow the same setup as Zhang et al. (2017), randomly sampling 20% of the original training set as our labeled data and 50% as unlabeled data. There is no overlap between the labeled and unlabeled data. See Zhang et al. (2017) for more details about the setup.

NER Dataset. We use the BIOES labeling scheme and report micro-averaged F_1. We preprocessed the text by replacing all digits with 0. We randomly sample 10% of the original training set as our labeled data and 50% as unlabeled data. We also ensure there is no overlap between the labeled and unlabeled data.

Experiments

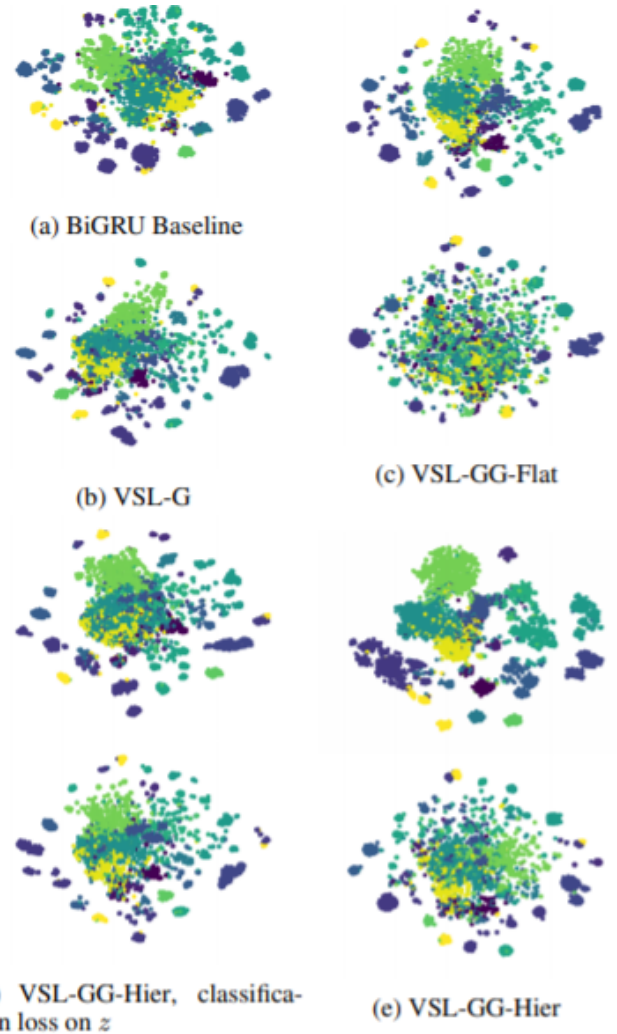
	dev.		test	
	acc.	ULΔ	acc.	ULΔ
BiGRU baseline	90.8	-	90.6	-
VSL-G	91.1	+0.1	-	-
VSL-GG-Flat	91.4	+0.1	-	-
VSL-GG-Hier	***91.6***	***+0.3***	***91.6***	*+0.3*

(a) Twitter tagging accuracies (%)

	dev.		test	
	F_1	ULΔ	F_1	ULΔ
BiGRU baseline	87.6	-	83.7	-
VSL-G	87.8	+0.1	-	-
VSL-GG-Flat	88.0	+0.1	-	-
VSL-GG-Hier	***88.4***	***+0.2***	***84.7***	*+0.0*

(b) NER F_1 score (%)

Table 1: For dev and test, we show results when only using labeled data and the change in performances ("ULΔ") when adding unlabeled data. Bold is highest in each column. Italic is the best model including unlabeled data. We only show test results for the baseline and our best-performing model, which achieves 91.9% accuracy on the Twitter test set and 84.7% F_1 on the NER test set when using unlabeled data.



(a) BiGRU Baseline

(b) VSL-G

(c) VSL-GG-Flat

(d) VSL-GG-Hier, classification loss on z

(e) VSL-GG-Hier

Figure 3: t-SNE visualization of Gaussian latent variables and baseline hidden states for Twitter development set. In plot 3c, 3d, and 3e, the upper subplot is latent variable y and the lower is z. Each point in the plot is a token and the color represents the true tag of the token.

Experiments

	French		German		Indonesian		Spanish		Russian		Croatian	
	acc.	ULΔ	acc.	ULΔ	acc.	ULΔ	acc.	ULΔ	acc.	ULΔ	acc.	ULΔ
NCRF	93.4	-	90.4	-	88.4	-	91.2	-	86.6	-	86.1	-
NCRF-AE	93.7	+0.2	90.8	+0.2	89.1	+0.3	91.7	+0.5	87.8	+1.1	87.9	+1.2
BiGRU baseline	95.9	-	92.6	-	92.2	-	94.7	-	95.2	-	95.6	-
VSL-G	96.1	+0.0	92.8	+0.0	92.3	+0.0	94.8	+0.1	95.3	+0.0	95.6	+0.1
VSL-GG-Flat	96.1	+0.0	93.0	**+0.1**	92.4	**+0.1**	95.0	+0.1	95.5	**+0.1**	95.8	+0.1
VSL-GG-Hier	***96.4***	***+0.1***	***93.3***	***+0.1***	***92.8***	***+0.1***	***95.3***	***+0.2***	***95.9***	***+0.1***	***96.3***	***+0.2***

Table 2: Tagging accuracies (%) on UD test sets. For each language, we show test accuracy ("acc.") when only using labeled data and the change in test accuracy ("ULΔ") when adding unlabeled data. Results for NCRF and NCRF-AE are from Zhang et al. (2017), though results are not strictly comparable because we used pretrained word embeddings for all languages on Wikipedia. Bold is highest in each column, excluding the NCRF variants. Italic is the best accuracy including the unlabeled data.

Experiments

	Twitter		NER	
	acc.	no VR	F_1	no VR
BiGRU baseline	90.8	-	87.6	-
VSL-G	91.1	90.9	87.8	87.7
VSL-GG-Flat	91.4	90.9	88.0	87.8
VSL-GG-Hier	91.6	91.0	88.4	87.9

Table 4: Results on Twitter and NER dev sets. For each model, we show supervised results for the models with variational regularization ("acc." or F_1) and results when replacing variational components with their deterministic counterparts ("no VR").

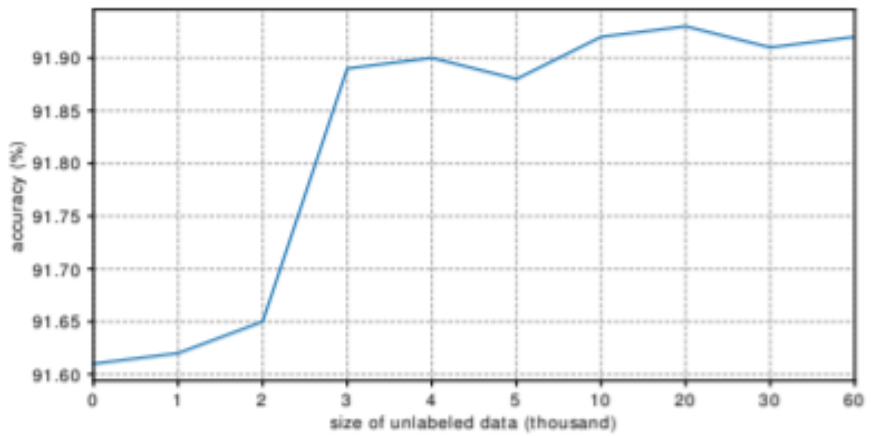


Figure 4: Twitter dev accuracies (%) when varying the amount of unlabeled data.