Language modeling with neural trans-dimensional random fields

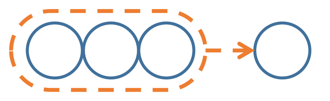
Bin Wang, Zhijian Ou

Speech Processing and Machine Intelligence (SPMI) Lab, Tsinghua University, Beijing China

wangbin12@mails.Tsinghua.edu.cn, ozj@Tsinghua.edu.cn

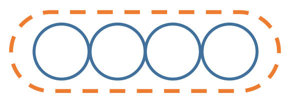
Introduction

Directed graphical language models:



$p(x_i|x_1,\dots,x_{i-1})$

Trans-dimensional random field (TRF) language models:



$p(x_1,x_2,\dots,x_l)$

- Being able to flexibly integrate rich features – discrete features and neural network features
- Computationally more efficient in inference than LSTM LMs.

Training objectives

$p(l,x^l;\theta,\zeta)$ an TRF LM with parameters θ,ζ

$q(l,x^l;\mu)$ an auxiliary LM with parameter μ

1. For θ. Maximize the likelihood.

$$E_D\left[\frac{\partial\phi}{\partial\theta}\right]-E_{p(l,x^l;\theta,\zeta)}\left[\frac{\partial\phi}{\partial\theta}\right]=0$$

The expectation on the training set D | The expectation under the TRF model distribution

2. For ζ. Optimize the length distribution

$$\sum_{x^l}p(l,x^l;\theta,\zeta)=\pi_l$$

The marginal length probability

3. For μ. Minimize the KL divergence between p and q

$$\frac{\partial}{\partial\mu}KL\left(p(l,x^l;\theta,\zeta)||q(l,x^l;\mu)\right)=0$$

Three objectives induce the following update operations

SA updates

θ

$$\theta^{(t)}=\theta^{(t-1)}+\gamma_{\theta,t}Adam\left\{E_{D^{(t)}}\left[\frac{\partial\phi}{\partial\theta}\right]-\frac{1}{K_B}\sum_{(l,x^l)\in B^{(t)}}\frac{\partial\phi(x^l;\theta)}{\partial\theta}\right\}$$

ζ

$$\begin{cases}\zeta_l^{(t-\frac{1}{2})}=\zeta_l^{(t-1)}+\frac{\gamma_{\zeta,t}}{\pi_l}\frac{1}{K_B}\sum_{(j,x^j)\in B^{(t)}}1(j==l)\text{ , for }l=1,\dots,m\\ \zeta^{(t)}=\zeta^{(t-\frac{1}{2})}-\zeta_1^{(t-\frac{1}{2})}\end{cases}$$

μ

$$\mu^{(t)}=\mu^{(t-1)}+\gamma_{\mu,t}\sum_{(l,x^l)\in B^{(t)}}\frac{\partial}{\partial\mu}\log q(l,x^l;\mu)$$

where:

- $D^{(t)}$ is the mini-batch of training data at iteration t
- $B^{(t)}$ is the sample set at iteration t, and $K_B=|B^{(t)}|$.
- $\gamma_{\theta,t},\gamma_{\zeta,t},\gamma_{\mu,t}$ are the learning rates for θ,ζ,μ respectively.
- $Adam$ is the Adam method

Trans-dimension random field LMs

Model Definition

x^l is the a word sequence of length l, ranging from 1 to m

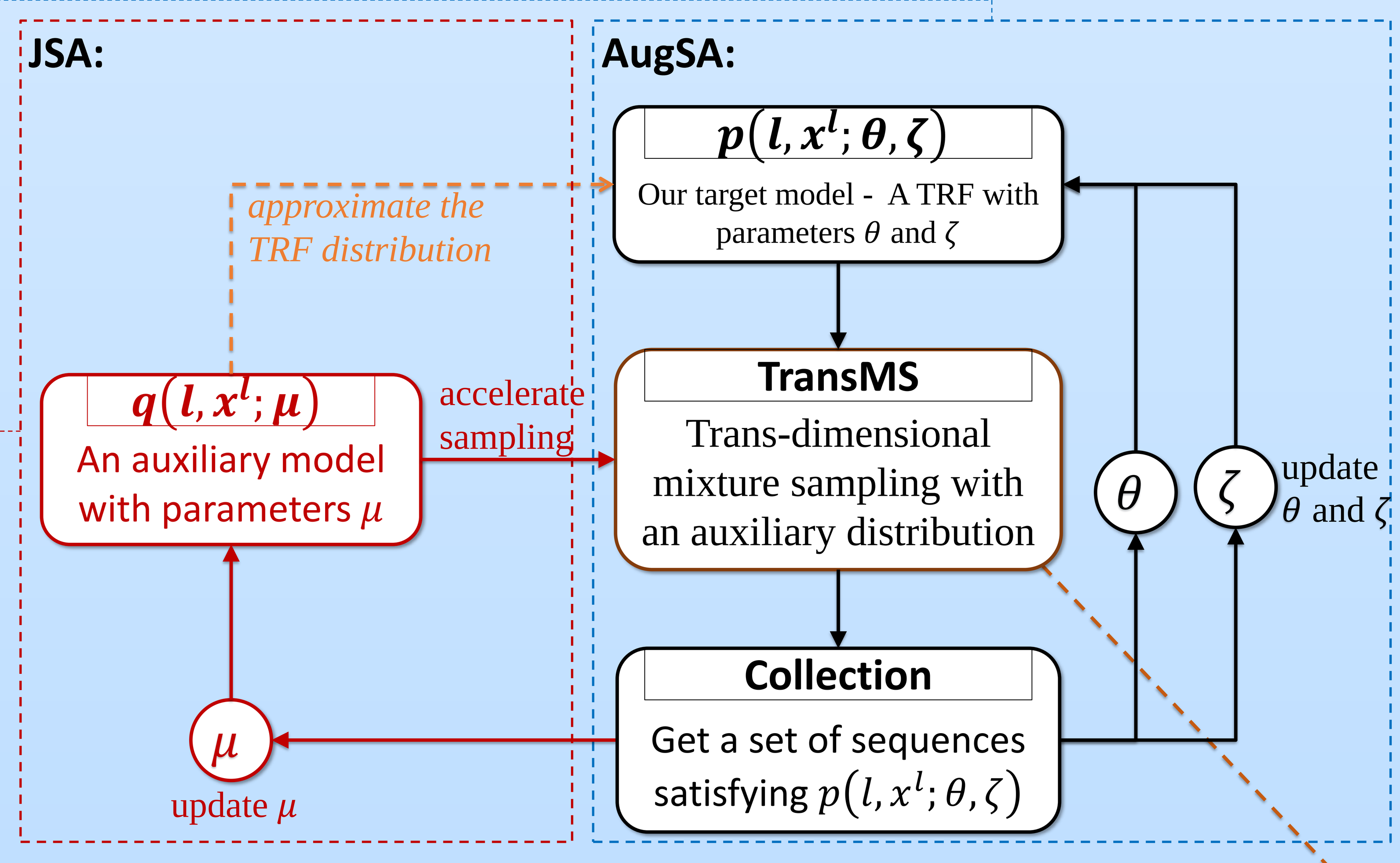
Variables need to be estimated:

- θ: the model parameters.
- $\zeta=(\zeta_1,\zeta_2,\dots,\zeta_m)$: normalization constants.

$$p(l,x^l;\theta,\zeta)=\pi_l\cdot\frac{1}{Z_0}e^{\phi(x^l;\theta)-\zeta_l}$$

The joint probability of sequence x^l and l | The empirical length probability | $\triangleq p_l(x^l;\theta,\zeta)$ the probability of sequence x^l

Model Training



Model Evaluation

LMs trained on Penn Treebank (PTB) training set are applied to rescore the 1000-best lists from recognizing WSJ'92 test data (330 utterances).

Model	PPL	WER(%)	#param	Training Time	Inference Time
KN5	141.2	8.78	2.3 M	22 s (1 CPU)	0.06 s (1 GPU)
LSTM-2x200	113.9	7.96	4.6 M	1.7 h (1 GPU)	6.36 s (1 GPU)
LSTM-2x650	84.1	7.66	19.8 M	7.5 h (1 GPU)	6.36 s (1 GPU)
LSTM-2x1500	78.7	7.36	66.0 M	1 day (1 GPU)	9.09 s (1 GPU)
discrete TRF	≥130	7.92	6.4 M	1 day (8 CPUs)	0.16 s (1 GPU)
neural TRF	≥37.4	7.60	4.0 M	3 days (1 GPU)	0.40 s (1 GPU)
KN5 + LSTM-2x1500		7.47			
neural TRF + LSTM-2x1500		**7.17**			

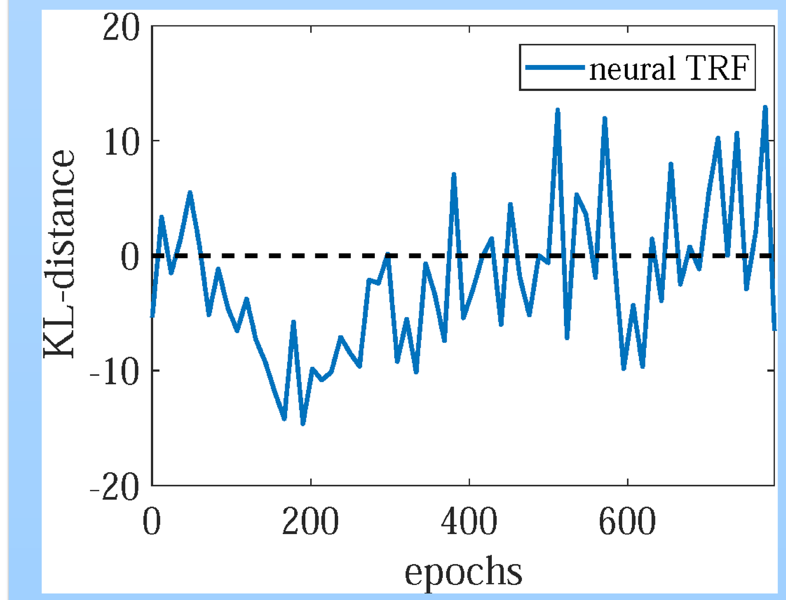


Fig.3. The KL-divergence $KL(p||q)$

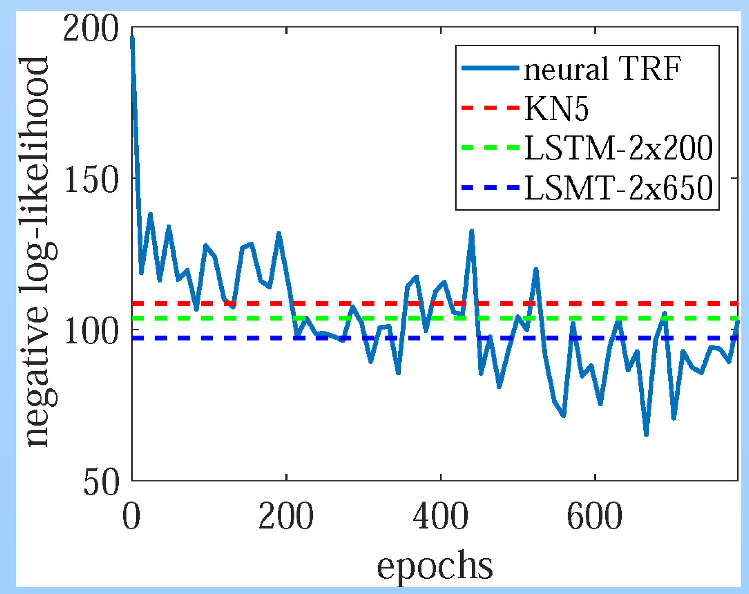


Fig.4. The negative log-likelihood on PTB test set

Deep CNN Architecture

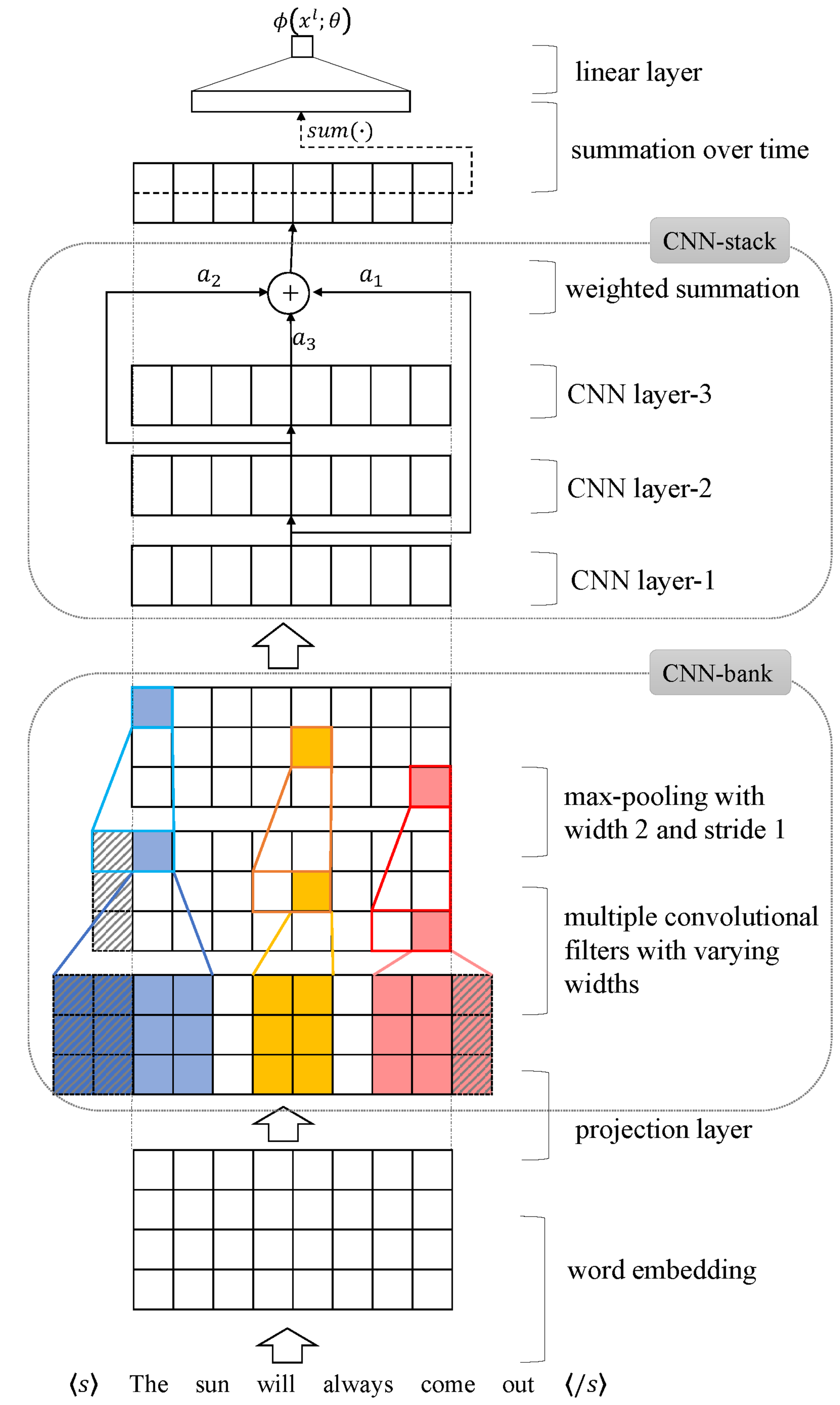


Fig. 1. The deep CNN architecture used to define the potential function $\phi(x^l;\theta)$. Shadow areas denote the padded zeros.

Trans-dimensional mixture sampling

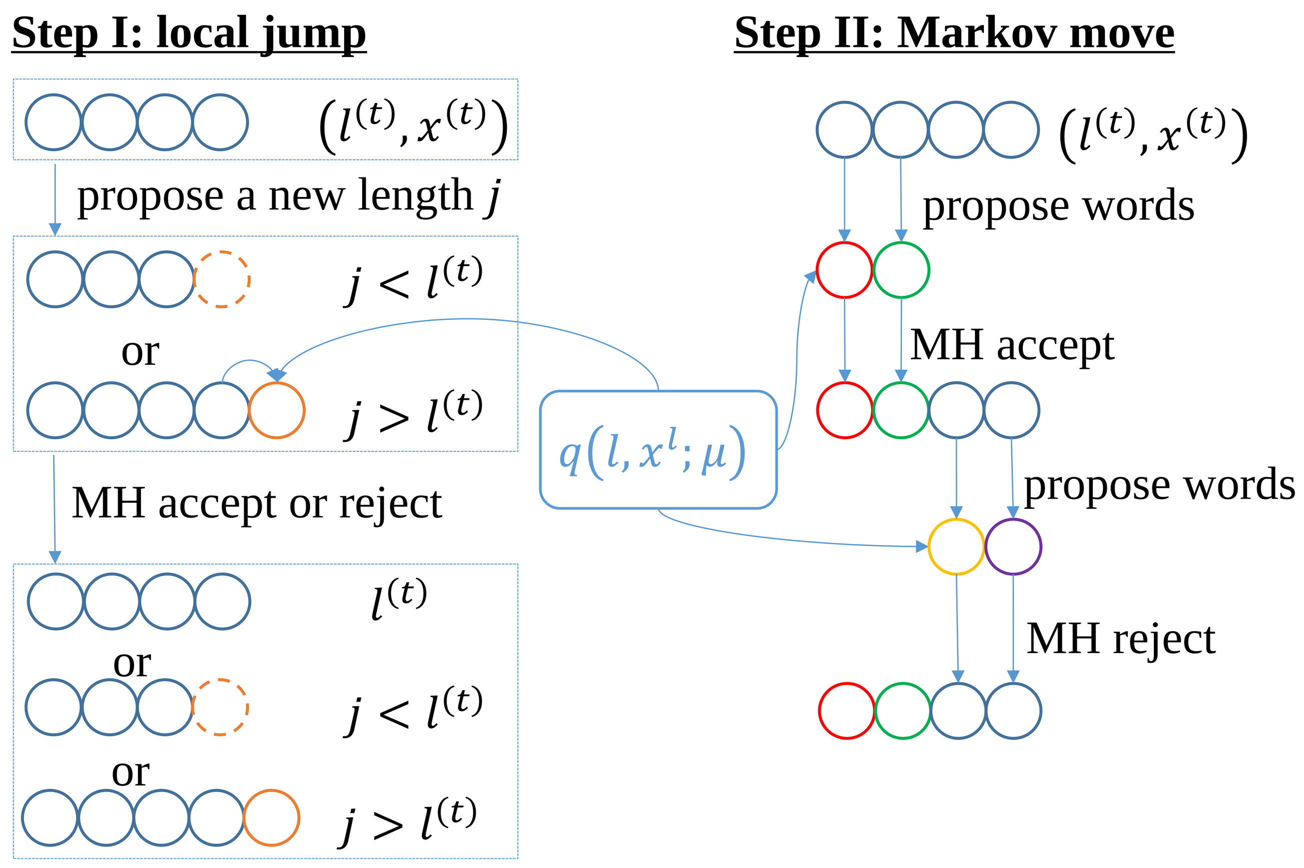


Fig 2. Trans-dimensional mixture sampling with an auxiliary distribution $q(l,x^l;\mu)$. Step I (left) changes the length of the input sequence and Step II (right) draws the words at each positions. Metropolis-Hasting (MH) method is used at both steps with $q(l,x^l;\mu)$ served as the proposal distribution.