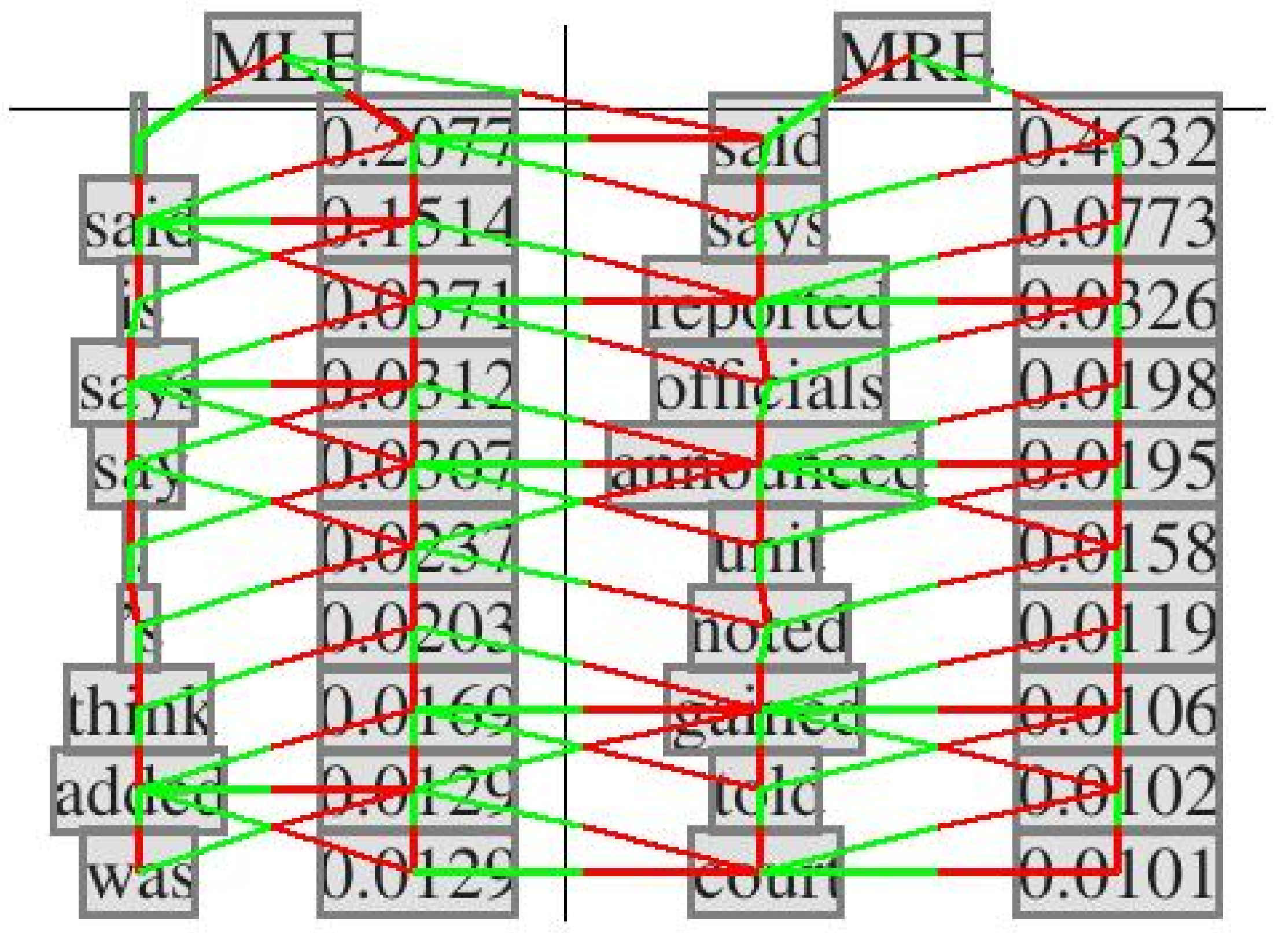
# sent.	MLE		MRE	
	accuracy	VI	accuracy	VI
10,000	0.5087	3.3471	0.5825	2.9018
20,000	0.5390	3.2387	0.5874	2.9217
30,000	0.5556	3.0764	0.6000	2.7904
40,000	0.5800	3.0117	0.6112	2.7403

Table 2: Effect of training corpus size.



MLE		MRE	
,	0.2077	said	0.4632
said	0.1514	says	0.0773
is	0.0371	reported	0.0326
says	0.0312	officials	0.0198
say	0.0307	announced	0.0195
:	0.0237	unit	0.0158
's	0.0203	noted	0.0119
think	0.0169	gained	0.0106
added	0.0129	told	0.0102
was	0.0129	court	0.0101

Table 3: Example emission probabilities for the POS tag "VBD" (verb past tense).

# state	CRF Autoencoders		MRE	
	accuracy	VI	accuracy	VI
10	0.4059	2.7145	0.3881	2.9322
20	0.4657	2.7462	0.5203	2.8879
30	0.5479	2.9585	0.5653	2.8199
40	0.5377	3.1048	0.6191	2.9255
50	0.5662	2.8450	0.6739	2.7522

Table 4: Comparison between CRF Autoencoders and MRE on unsupervised part-of-speech induction.

training corpus size for both MLE and MRE. The case for VI is similar. We find that our approach outperforms MLE consistently.

Table 3 shows example emission probabilities (e.g., $p(x|z)$) for the POS tag "VBD" (verb past tense). We follow Johnson (2007) to deterministically map hidden states to POS tags based on co-occurrence. As shown in Table 3, we find that MLE is prone to learn common but irrelevant correlations in the data (e.g., frequent words such as ",", ":", and "is"). In contrast, MRE is capable of identifying "said", "reported", "announced", "noted", "gained", and "told" correctly, suggesting that MRE enables HMMs to better discover intended correlations in the data.

Comparison with CRF Autoencoders We also compare our approach with CRF Autoencoders (Ammar, Dyer, and Smith 2014), which also builds on an encoding-reconstruction framework but allows for incorporating features. A surprising finding is that CRF Autoencoders achieves the highest accuracy with 50 states but obtains the lowest VI with 10 states. Our approach achieves the best accuracy and VI both with 50 states. While our approach slightly lags behind CRF Autoencoders in terms of VI, the improvements in terms of accuracy are statistically significant ($p < 0.01$).

criterion	model	C → E	E → C
MLE	Model 1	43.07	45.89
	Model 2	40.28	42.38
MRE	Model 1	41.90	45.39
	Model 2	38.33	41.73

Table 5: Comparison between MLE and MRE on IBM translation models for unsupervised word alignment. The evaluation metric is alignment error rate (AER).

MLE		MRE	
article	0.4932	article	0.5428
the	0.1924	articles	0.0995
says	0.0586	says	0.0624
points	0.0293	published	0.0497
an	0.0263	points	0.0349

Table 6: Example translation probabilities of the Chinese word "wenzhang".

Evaluation on Word Alignment

Setting We used the FBIS corpus as the training corpus, which contains 240K Chinese-English parallel sentences with 6.9M Chinese words and 8.9M English words. We used the TsinghuaAligner development and test sets (Liu and Sun 2015), which both contain 450 sentence pairs with gold-standard annotations. The evaluation metric is *alignment error rate* (AER) (Och and Ney 2003). Both MLE and MRE use the following training scheme: 5 iterations for IBM Model 1 and 5 iterations for IBM Model 2. As IBM Model 1 is a simplified version of IBM Model 2, the parameters of Model 1 at iteration 5 are used to initialize Model 2. We distinguish between two translation directions: Chinese-to-English (C → E) and English-to-Chinese (E → C).

Comparison with MLE Table 5 shows the comparison between MLE and MRE. We find that MRE outperforms MLE for both translation directions. All the differences are statistically significant ($p < 0.01$).

Table 6 shows example translation probabilities (i.e., $p(x|y)$) of the Chinese word "wenzhang" (i.e., "article"). We find that MLE tends to identify frequent words such as "the" and "an" as candidate translations while MRE finds more relevant candidate translations. This finding further confirms that MRE is more robust to common but irrelevant correlations.

Comparison with CRF Autoencoders On the same dataset, CRF Autoencoders achieve much lower AERs: 32.54 for C → E and 29.81 for E → C, respectively. The reason is that CRF Autoencoders are a discriminative latent-variable model capable of including more expressive IBM Model 4 as features. In contrast, our approach focuses on providing better training criterion for generative latent-