Unsupervised Deep Belief Features for Speech Translation

Sameer Maskey, Bowen Zhou

IBM T.J. Watson Research Center
Yorktown Heights, NY, USA
smaskey@us.ibm.com, zhou@us.ibm.com

Abstract

We present a novel formalism for introducing deep belief features to Hierarchical Machine Translation Model. The deep features are generated by unsupervised training of a deep belief network built with stacked sets of Restricted Boltzmann Machines. We show that our new deep feature based hierarchical model is better than the baseline hierarchical model with gains for two different languages pairs in two different data size settings. We obtain absolute BLEU score improvement of +1.13 on Dari-to-English and +0.66 on English-to-Dari Transtac Evaluation task. We also observe gains on English-to-Chinese translation task.

1. Introduction

Many different features have been explored for Machine Translation such as syntactic features [1], language model features [2] and reordering features [3]. Many of these features are manually designed based on linguistic phenomena that relate two language pairs. Instead of designing new features based on intuition, linguistic knowledge and domain, we explore the possibility of inducing new features in an unsupervised fashion using Deep Learning (DL) paradigm.

There has been growing interest in use of Deep Learning for various NLP and speech processing tasks. [4] proposed a unified framework based on DL for suite of NLP tasks such as chunking, parts of speech tagging, named entity tagging and semantic role labeling. They map the words into continuous space features and build deep networks to perform classification. One of the more closely related relevant work for Machine Translation is by [5]. They present a deep network structure which can perform Machine Transliteration. They train their network by mapping characters into deep network features and tie the source and target transliteration by a large hidden layer on the top of the network.

2. Deep Learning for MT

Hierarchical Machine Translation Model (Hiero) has been widely used since it's introduction by [6]. We employ our implementation of Hiero as the baseline model. Hiero is based on Synchronous Context Free grammar such that the MT model is a set of grammar rules that can be used to parse the source sentence. Each rule has a set number of features that are combined in log linear fashion for a decoding process. Using Deep Belief Networks we would like to induce new features for each rule.

2.1. Deep Belief Networks

Deep Belief Network (DN) is a type of generative model that is composed of multiple layers of latent variables with the first layer representing the visible feature vectors. Deep Belief Network D consisting of l layers models a joint distribution between the hidden nodes in all layers $h^k, k = 1, ..., l$ and all the visible nodes v_j.

A key parameter of the model that needs to be estimated by training algorithm for DN is the weight matrix, W_{ik}, that defines the relationship between layers h_i^k and h_i^{k+1}. Since the DN does not allow same layer node connections, the factorial conditional distribution can be given by

$$p(h^k|h^{k+1}) = \prod_i p(h_i^k|h^{k+1}) \tag{1}$$

where

$$p(h_i^k|h^{k+1}) = sig(b_i^k + \sum_j W_{ij}^k h_j^{k+1}) \tag{2}$$

and b_i^k are biases for hidden node i in layer k and

$$sig(h_i^k) = \frac{1}{1 + exp(-h_i^k)} \tag{3}$$

The joint distribution between layer pairs can be modeled using Restricted Boltzmann Machines (RBM). RBM with v visible units and h hidden units has the following joint distribution

$$p(v, h) = \frac{1}{Z} exp(-E(v, h)) \tag{4}$$

where

$$E(v, h) = h'Wv + b'v + c'h \tag{5}$$

Z is a normalization constant for the given distribution given by $\sum_v \sum_h e^{-E(v,h,\theta)}$ where model parameter $\theta = (W, b, c)$ and b is biases for visible units; and c is biases for hidden units. $E(v, h)$ is also known as energy of the state (v, h). Since we do not allow visible-visible and hidden-hidden connections the conditional distributions $p(v|h)$ and $p(h|v)$ for RBM are factorials, i.e. $p(v|h) = \prod_j p(v_j|h)$ such that

$$p(v_j = 1|h; \theta) = \sigma(\sum_i W_{ij} h_i + b_j) \quad (6)$$

and

$$p(h_i = 1|v; \theta) = \sigma(\sum_i W_{ij} v_j + c_i) \quad (7)$$

2.2. Deep Features based Hierarchical MT

Let us describe our Deep Features based MT model. Let (s_q, t_q) be a source-target sentence pair in our training corpus with a total of T number of sentences. Let the final rule set R include rules with 0, 1 and 2 non-terminals. Each rule in R consists of 4 static features which we described earlier in Section 2. An example rule with 2 non-terminals is shown below where f_n is a feature for the rule estimated from the phrase and alignment table.

$$X -> X_1 \; the \; X_2, X_2 \; de \; X_1, \; f_1, f_2, f_3, f_4 \quad (8)$$

We build DN such that the first layer with visible nodes equal the number of features in our baseline Hiero model. Hence, the total number of visible nodes $V_T = 4$ in our Deep Network D. Each visible node v_i corresponds to our original feature f_i of our Hiero model. We then design our network with various layer sizes and various number of layers. There is no easy standard way of inducing network structure for deep networks [7]. For our models, we explore four different types of network structures shapes which are Parallel (R), Pyramid (P), Inverted Pyramid (I) and Diamond (D) shaped structures. All of the structures have visible layer with 4 nodes for 4 baseline features.

We first experimented with deep networks with a single layer of hidden nodes, effectively Restricted Boltzmann Machine (RBM). We consider RBM with Gaussian visible units with a fixed variance of 1. Hence, the update Equation 6 needs to be modified to Equation 9.

$$p(v_j = 1|h; \theta) = \mathcal{N}(\sum_i W_{ij} h_i + b_j) \quad (9)$$

We maximize the log-likelihood of the feature distribution using Equation 10.

$$\frac{\partial log p(v,h)}{\partial \theta} = -\langle \frac{\partial log E(v,h)}{\partial \theta} \rangle_0 + \langle \frac{\partial log E(v,h)}{\partial \theta} \rangle_\infty \quad (10)$$

where $< \; . \; >_0$ denotes an average with respect to the product of the data distribution (Hiero feature distributions) and $p(h|v)$. On the other hand $< \; . \; >_\infty$ represents an average with respect to the model distribution. [8] proposed an efficient training 'contrastive divergence' algorithm where $< \; . \; >_\infty$ is estimated by $< \; . \; >_k$ where k can be as small as 1. This results in a very simple gradient estimator for weight W_{ij} simply by getting $p(h_i = 1|\boldsymbol{v})v_j - p(h_i = 1|\boldsymbol{v'})v'_j$ where v' is a sample from reconstructed data distribution. Hence, to train our deep network for Hiero features we first take all the rules to form a rule set R_T. We divide the rule set into B number of batches where $B = \frac{R_T}{N}$; N is the number of rules in each batch.

For each batch the derivative of log likelihood is computed as shown in Equation 10 and update the weights. The updates are averaged across all the rules in the batch. One full iteration of training includes B number of updates to RBM parameters θ which are W, b, c. We keep track of reconstruction error $p(h_i = 1|\boldsymbol{v})v_j - p(h_i = 1|\boldsymbol{v'})v'_j$ and iterate until it converges. Once the training is completed the trained weights W_{ik} and biases b are used to generate RBM features F_d for all of our rules in rule set R such that $|F_d| = |R|$. We should note that the training is an unsupervised training algorithm and we do not assume presence of any human labels.

3. Experiments and Results

3.1. Data and Baseline Model

We first performed evaluations on a task of Dari-English bi-directional speech translation task. The training data consisted of about 150K parallel sentences. A held out set of randomly sampled sentences were selected to prepare held out development and test set. The held out test sets contained 1569 and 1275 sentences for E2F and F2E directions respectively. We also performed our experiments on a larger data set size for a different language pair: Chinese-English. For the larger set we had 0.878 million sentences with 1500 sentences for E2F and 1000 sentences for F2E test set respectively. The corpus is a part of conversational MT corpus and is a superset of IWSLT corpus [9].

		E2F	F2E
Baseline	Dev	0.1355	0.1425
	Test	0.1337	0.1422

Table 1: Baseline Results for Dari-English MT model (Hiero) 1-reference

We first built our baseline hierarchical MT (Hiero) model following [6]. We obtained development and test set BLEU scores as shown in Table 1 which were 0.1425 and 0.1422 respectively for Foreign to English (F2E) direction. Similarly, BLEU scores were 0.1355 and 0.1337 for E2F direction respectively.

3.2. Deep Feature Hierarchical Model

Deep Model		E2F	F2E
4x4	Dev	0.1354	0.1504
	Test	0.1342	0.1481
4x4x4	Dev	0.1382	0.1473
	Test	0.1390	0.1503
4x4x4x4	Dev	0.1382	0.1497
	Test	0.1388	0.1522
4x4x4x4x4	Dev	0.1373	0.1466
	Test	0.1367	0.1456
4x4x4x4x4x4	Dev	0.1377	0.1455
	Test	0.1385	0.1471
4x4x4x4x4x4x4	Dev	0.1342	0.1455
	Test	0.1358	0.1507
4x4x4x4x4x4x4x4	Dev	0.1261	0.1445
	Test	0.1254	0.1453

Table 2: Deep Models with Additional Layers of Hidden Nodes : Dari-English (1-reference)

Using the 4 features of baseline Hiero model as input features to 4 visible nodes we then built Deep Belief Network of various sizes and types. First, a single layer network (effectively RBM Model) with 1 layer of 4 hidden nodes was trained. We ran 50 iterations of contrastive divergence with 0.001 for learning rate and 0.0002 for weight decay. Learning rate and weight decay were decided empirically. After obtaining the trained weight W_{ik} and biases b we generated the features by passing all R_T rules through the network. Each set of resulting features were normalized by an average for that feature set and were appended as extra features to the model. The results are shown in Table 2. We can see that the deep model with 1 layer (4x4 RBM Model) does better on both tune and test sets for both direction with BLEU scores of 0.1504 and 0.1481 respectively which is +0.79 and +0.59 absolute better than baseline for F2E direction. We see that with 1 layer RBM Model we do not see much gain in E2F direction.

3.2.1. Discussion : Network Depth and Structure vs. MT Performance

[10] has shown that adding more hidden layers always maximizes the log-likelihood of the generative deep network model. In order to explore the best depth of the deep network we built deep networks of various sizes. In total we build eight deep networks $D_1^4, D_2^4, ..., D_8^4$ with $1, 2, ..., 8$ layers of hidden nodes respectively. D_3^4 with 3 layers and 4 hidden nodes on 3rd layer is represented by 4x4x4 in Table 2. The effect of increasing the depth of our deep network can be seen in Table 2. We should note that the deep network starts to overtrain when we add too many layers. The best network for E2F direction is D_3^4 with depth level of 3 and for F2E direction D_4^4 with depth level of 4. The test set BLEU score for D_3^4 for E2F direction is 0.1390 which is +0.53 absolute better than the baseline model. Similarly, test set BLEU score for D_4^4 for F2E direction is 0.1522 which +1.00 absolute better than the baseline as shown in Table 2. We would like to emphasize that we obtained above improvement in a completely unsupervised fashion of inducing features.

		E2F	F2E
4x4x8x12x12	Dev	0.1365	0.1469
	Test	0.1392	0.1464
4x4x8x12x12x16	Dev	0.1399	0.1474
	Test	0.1388	0.1449
4x8x12x16	Dev	0.1357	0.1559
	Test	0.1375	0.1538
4x4x8x12x12x8x4	Dev	0.1403	0.1499
	Test	0.1403	0.1399
4x3x2	Dev	0.1329	0.1522
	Test	0.1340	0.1521

Table 3: Various Deep Network Structure based Features for MT: Dari-English (1-reference)

Network structure of DN can also make a significant difference in the MT performance. We also experimented with different types of Network structure such as Diamond, Pyramid, Inverse Pyramid and Parallel shaped structure. Results in Table 3 show that we can obtain the highest F2E dev and test set performance with Inverse Pyramid structure 4x8x12x16 (4 layers with 16 nodes at the top layer) with a BLEU score of 0.1559 and 0.1538 respectively.

Although general trend of improvement with the use of DN applies for both direction as seen in result Tables 2 and 3 we observe that different network structures may work better for different directions of translations. For example, 4x8x12x16 model was the best for F2E while 4x4x8x12x12 worked the best for E2F direction. For both direction we see that instead of having only 4 hidden nodes in each layer having more nodes in higher level layers is helpful; but again this property also seem direction dependent. This can be noted by observing that 4x4x8x12x16x4 structure which is worse than 4x4x8x12x12 on F2E but better on E2F. In fact, we see for this diamond like deep network structure, E2F does the best with +0.66 absolute BLEU score improvement over the the baseline. Hence, finding the optimal structure of deep network is important when we use deep network feature for machine translation.

		E2F	F2E
4x4x8	Dev	0.0950	0.1347
	Test	0.0969	0.1403

Table 4: Deep Hiero with No Baseline Features; Dari-English (1-reference)

		E2F	F2E
Baseline	Dev	0.1821	0.2621
	Test	0.1808	0.2428

Table 5: Baseline Chinese-English Hiero Model (1-reference)

In order to validate our deep feature based MT model further we also experimented on a larger data set for a different language. We experimented on Chinese-English translation task with more than 0.878 million sentence pairs of training data as described in 3.1. Our baseline model resulted in BLEU scores of 0.1808 for E2F direction and 0.2428 for F2E direction (1 reference) on the dev set as shown in Table 5. Even though we did not see significant gains on F2E direction, our best deep network model for Chinese-English resulted in BLEU scores of 0.1865 for E2F (Table 6) which is +0.57 absolute better than than the baseline models showing that unsupervised DN features are useful for larger data set as well.

In both languages different network structures provided the best performance for different directions. For future work, we would like to automatically induce the optimal network structure for given data size and language pair instead of performing an exhaustive search. One should also note that our formalism has no constraints tied to hierarchical model and one can use the same technique to generate deep belief features for phrase based model as well.

		E2F	F2E
4x4	Dev	0.1887	0.2626
	Test	0.1865	0.2430
4x4x4	Dev	0.1855	0.2627
	Test	0.1844	0.2418
4x8x12	Dev	0.1849	0.2557
	Test	0.1825	0.2467

Table 6: Results with Deep Features in Chinese-English Model (1-reference)

4. Conclusion

We presented a new formalism of using deep learning for introducing unsupervised features based on deep belief networks to Hierarchical Machine Translation model and presented results for Dari-English and Chinese-English translation tasks. Our method produced deep feature Hiero model which was better than the baseline by +1.13 on F2E direction and +0.66 on E2F direction for Dari-English language pair. We also showed gains for Chinese-English translation. We experimented with effect of increasing the depth of networks and its effect on BLEU; and we saw that after a certain depth, adding layers may not necessarily help translation. We also experimented with different types of deep network structure including parallel (R), Pyramid (P), Inverse Pyramid (InvP), Diamond shaped (D) structures. We noted that structure had a significant influence on the deep feature Hiero model performance, and in general increasing the number of hidden nodes at higher layers provided better deep MT model. We believe the deep belief features are adding discriminating power to the translation decoder such that the noisy hypothesis are pruned. To our best knowledge, this is the first paper to show the use of unsupervised deep belief features that improves statistical machine translation.

5. References

[1] K. Yamada and K. Knight, "A syntax-based statistical translation model," in *Proceedings of ACL*. Toulouse, France: ACL, July 2001, pp. 523–530.

[2] T. Brants, A. C. Popat, P. Xu, F. J. Och, J. Dean, and G. Inc, "Large language models in machine translation," in *Empirical Methods for Natural Language Processing*, 2007, pp. 858–867.

[3] S. Maskey and B. Zhou, "Rapid integration of parts of speech information to improve reordering model for english-farsi speech to speech translation," in *Proceedings of ICASSP*, March 2010.

[4] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of NIPS*, 2008.

[5] T. Deselaers, O. Bender, and H. Ney, "A deep learning approach to machine transliteration," in *Proceedings of the Fourth Workshop on Statistical Machine Translation*, 2007.

[6] D. Chiang, "A hierarchical phrase-based model for statistical machine translation," in *Proceedings of ACL*, 2005.

[7] Y. Lecun, S. Chopra, R. Hadsell, F. J. Huang, G. Bakir, T. Hofman, B. Schlkopf, A. Smola, and B. Taskar, "A tutorial on energy-based learning," 2006.

[8] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, August 2002.

[9] M. Paul, "Overview of the iwslt 2006 evaluation campaign," in *Proceedings of IWSLT*, 2006.

[10] N. Le Roux and Y. Bengio, "Representational power of restricted boltzmann machines and deep belief networks," *Neural Comput.*, vol. 20, pp. 1631–1649, June 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1374176.1374187