Learning Hidden Markov Models with Distributed State Representations for Domain Adaptation

Min Xiao and **Yuhong Guo**
Department of Computer and Information Sciences
Temple University, Philadelphia, PA 19122, USA
{minxiao,yuhong}@temple.edu

Abstract

Recently, a variety of representation learning approaches have been developed in the literature to induce latent generalizable features across two domains. In this paper, we extend the standard hidden Markov models (HMMs) to learn distributed state representations to improve cross-domain prediction performance. We reformulate the HMMs by mapping each discrete hidden state to a distributed representation vector and employ an expectation-maximization algorithm to jointly learn distributed state representations and model parameters. We empirically investigate the proposed model on cross-domain part-of-speech tagging and noun-phrase chunking tasks. The experimental results demonstrate the effectiveness of the distributed HMMs on facilitating domain adaptation.

1 Introduction

Domain adaptation aims to obtain an effective prediction model for a particular target domain where labeled training data is scarce by exploiting labeled data from a related source domain. Domain adaptation is very important in the field of natural language processing (NLP) as it can reduce the expensive manual annotation effort in the target domain. Various NLP tasks have benefited from domain adaptation techniques, including part-of-speech tagging (Blitzer et al., 2006; Huang and Yates, 2010a), chunking (Daumé III, 2007; Huang and Yates, 2009), named entity recognition (Guo et al., 2009; Turian et al., 2010), dependency parsing (Dredze et al., 2007; Sagae and Tsujii, 2007) and semantic role labeling (Dahlmeier and Ng, 2010; Huang and Yates, 2010b).

In a typical domain adaptation scenario of NLP, the source and target domains contain text data of different genres (*e.g.*, newswire vs biomedical (Blitzer et al., 2006)). Under such circumstances, the original lexical features may not perform well in cross-domain learning since different genres of text may use very different vocabularies and produce cross-domain feature distribution divergence and feature sparsity issue. A number of techniques have been developed in the literature to tackle the problem of cross-domain feature divergence and feature sparsity, including clustering based word representation learning methods (Huang and Yates, 2009; Candito et al., 2011), word embedding based representation learning methods (Turian et al., 2010; Hovy et al., 2015) and some other representation learning methods (Blitzer et al., 2006).

In this paper, we extend the standard hidden Markov models (HMMs) to perform distributed state representation learning and induce context-aware distributed word representations for domain adaptation. Instead of learning a single discrete latent state for each observation in a given sentence, we learn a distributed representation vector. We define a state embedding matrix to map each latent state value to a low-dimensional distributed vector and reformulate the three local distributions of HMMs based on the distributed state representations. We then simultaneously learn the state embedding matrix and the model parameters using an expectation-maximization (EM) algorithm. The hidden states of each word in a sentence can be decoded using the standard Viterbi decoding procedure of HMMs, and its distributed representation can be obtained by a simple mapping with the state embedding matrix. We then use the context-aware distributed representations of the words as their augmenting features to perform cross-domain part-of-speech (POS) tagging and noun-phrase (NP) chunking.

The proposed approach is closely related to the clustering based method (Huang and Yates,

2009) as we both use latent state representations as generalizable features. However, they use standard HMMs to produce discrete hidden state features for each observation word, while we induce distributed state representation vectors. Our distributed HMMs share similarities with the word embedding based method (Hovy et al., 2015), and can be more space-efficient than the standard HMMs. Moreover, our model can incorporate context information into observation feature vectors to perform representation learning in a context-aware manner. The distributed state representations induced by our model hence have larger representing capacities and generalizing capabilities for cross-domain learning than standard HMMs.

2 Related Work

A variety of representation learning approaches have been developed in the literature to address NLP domain adaptation problems. The *clustering based word representation learning* methods perform word clustering within the sentence structure and use word cluster indicators as generalizable features to address domain adaptation problems. For example, Huang and Yates (2009) used the discrete hidden state of a word under HMMs as augmenting features for cross-domain POS tagging and NP chunking. Brown clusters (Brown et al., 1992), which was used as latent features for simple in-domain dependency parsing (Koo et al., 2008), has recently been exploited for out-of-domain statistical parsing (Candito et al., 2011).

The *word embedding based representation learning* methods learn a dense real-valued representation vector for each word as latent features for domain adaptation. Turian et al. (2010) empirically studied using word embeddings learned from hierarchical log-bilinear models (Mnih and Geoffrey, 2008) and neural language models (Collobert and Weston, 2008) for cross-domain NER tasks. Hovy et al. (2015) used the word embeddings learned from the Skip-gram Model (SGM) (Mikolov et al., 2013) to develop a POS tagger for Twitter data with labeled newswire training data.

Some other representation learning methods have been developed to tackle NLP cross-domain problems as well. For example, Blitzer et al. (2006) proposed a structural correspondence learning method for POS tagging, which first selects a set of pivot features (occurring frequently in the two domains) and then models the correlations between pivot features and non-pivot features to induce generalizable features.

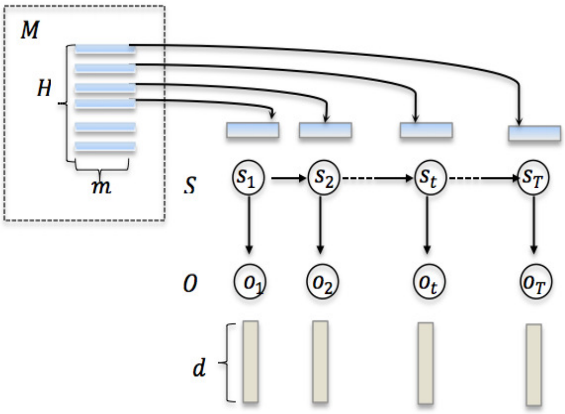

Figure 1: Hidden Markov models with distributed state representations (dHMM).

In terms of performing distributed representation learning for output variables, our proposed model shares similarity with the structured output representation learning approach developed by Srikumar and Manning (2014), which extends the structured support vector machines to simultaneously learn the prediction model and the distributed representations of the output labels. However, the approach in (Srikumar and Manning, 2014) assumes the training labels (i.e., output values) are given and performs learning in the standard supervised in-domain setting, while our proposed distributed HMMs address cross-domain learning problems by performing unsupervised representation learning. There are also a few works that extended standard HMMs in the literature, including the observable operator models (Jaeger, 1999), and the spectral learning method (Stratos et al., 2013). But none of them performs representation learning to address cross-domain adaptation problems.

3 Proposed Model

In this paper, we propose a novel distributed hidden Markov model (dHMM) for representation learning over sequence data. This model extends the hidden Markov models (Rabiner and Juang, 1986) to learn distributed state representations. Similar as HMMs, a dHMM (shown in Figure 1) is a two-layer generative graphical model, which generates a sequence of observations from a sequence of latent state variables using Markov properties. Let $O = \{\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_T\}$ be the sequence of observations with length T, where each observation $\mathbf{o}_t \in \mathbb{R}^d$ is a d-dimensional feature vector. Let $S = \{s_1, s_2, \ldots, s_T\}$ be the sequence of T hidden states, where each hidden state s_t has a discrete state value from a total H hidden states $\mathcal{H} = \{1, 2, \ldots, H\}$. Besides, we assume that there is a low-dimensional distributed representation vector associated with each hidden state. Let $M \in \mathbb{R}^{H \times m}$ be the state embedding matrix where the i-th row $M_{i:}$ denotes the m-dimensional representation vector for the i-th state. Previous works have demonstrated the usefulness of discrete hidden states induced from a HMM on addressing feature sparsity in domain adaptation (Huang and Yates, 2009). However, expressing a semantic word by a single discrete state value is too restrictive, as it has been shown in the literature that words have many different features in a multidimensional space where they could be separately characterized as number, POS tag, gender, tense, voice and other aspects (Sag and Wasow, 1999; Huang et al., 2011). Our proposed model aims to overcome this inherent drawback of standard HMMs on learning word representations. Given a set of observation sequences in two domains, the dHMM induces a distributed representation vector with continuous real values for each observation word as generalizable features, which has the capacity of capturing multi-aspect latent characteristics of the word clusters.

3.1 Model Formulation

To build the dHMMs, we reformulate the standard HMMs by defining three main local distributions based on the distributed state representations, i.e., the initial state distribution, the state transition distribution, and the observation emission distribution. Below we introduce them by using Θ to denote the set of parameters involved and using $\mathbf{1}$ to denote a column vector with all 1s.

First we use the following multinomial distribution as the *initial state distribution*,

$$P(s_1; \Theta) = \phi(s_1)^\top \lambda,$$

where $\phi(s_t) \in \{0, 1\}^H$ is a one-hot vector with a single 1 value at its s_t-th entry, and $\lambda \in \mathbb{R}^H$ is the parameter vector such that $\lambda \geq 0$ and $\lambda^\top \mathbf{1} = 1$.

We then define a multinomial logistic regression model for the *state transition distribution*,

$$P(s_{t+1}|s_t; \Theta) = \frac{\exp\left\{\phi(s_{t+1})^\top W M^\top \phi(s_t)\right\}}{Z(s_t; \Theta)}$$

where $W \in \mathbb{R}^{H \times m}$ is the regression parameter matrix and $Z(s_t; \Theta)$ is the normalization term.

Finally, we assume the observation vector is generated from a multivariate Gaussian distribution, i.e., $\mathbf{o}_t \sim \mathcal{N}\left(\phi(s_t)^\top MQ, \sigma I_d\right)$, and use the following model for the *emission distribution*,

$$P(\mathbf{o}_t|s_t; \Theta) = \frac{\exp\left\{\frac{-1}{2\sigma}\kappa(s_t, \mathbf{o}_t)\kappa(s_t, \mathbf{o}_t)^\top\right\}}{(2\pi)^{d/2}\sigma^{d/2}},$$

with $\kappa(s_t, \mathbf{o}_t) = \phi(s_t)^\top MQ - \mathbf{o}_t^\top$, where $Q \in \mathbb{R}^{m \times d}$ and $\sigma \in \mathbb{R}$ are the model parameters. Different from the standard HMMs which have discrete hidden states and discrete observations, the multivariate Gaussian model here generates each observation $\mathbf{o}_t$ as a d-dimensional continuous feature vector. This type of emission distribution provides us the flexibility to incorporate local context information or statistical global information for inducing distributed state representations. For example, we can use the concatenation of the one-hot word vectors within a sliding window around the target word as the observation vector. Moreover, we can also use the globally preprocessed continuous word vectors as the observation vectors, which we will describe later in our experiments.

The standard HMMs (Rabiner and Juang, 1986) use conditional probability tables for the state transition distribution, which grows quadratically with respect to the number of hidden states, and the emission distribution, which grows linearly with respect to the observed vocabulary size that is usually very large in NLP tasks. Instead, the dHMMs can significantly reduce the sizes of these conditional probability tables by introducing the low-dimensional state embedding vectors, and the dHMM is much more efficient in terms of memory storage. In fact, the complexity of dHMMs can be independent of the vocabulary size by using flexible observation features. We represent the dHMM parameter set as $\Theta = \{M \in \mathbb{R}^{H \times m}, W \in \mathbb{R}^{H \times m}, Q \in \mathbb{R}^{m \times d}, \sigma \in \mathbb{R}, \lambda \in [0, 1]^H\}$, where m is a small constant.

3.2 Model Training

Given a data set of N observed sequences $\{O^1, \ldots, O^n, \ldots, O^N\}$, its regularized log-

Table 1: Test performance for cross-domain POS tagging and NP chunking.

Systems	POS Tagging (Accuracy (%))		NP Chunking (F1-score)	
	All Words	OOV Words	All NPs	OOV NPs
Baseline	88.3	67.3	0.86	0.74
SGM (Hovy et al., 2015)	89.0	71.4	0.88	0.78
HMM (Huang and Yates, 2009)	90.5	75.2	0.91	0.85
dHMM	**91.1**	**76.0**	**0.93**	**0.88**

likelihood can be written as follows

$$\mathcal{L}(\Theta) = \sum_n \log P(O^n;\Theta) - \frac{\eta}{2}\mathcal{R}(W,Q,M) \quad (1)$$

where the regularization function is defined with Frobenius norms such as $\mathcal{R}(W,Q,M) = \|W\|_F^2 + \|Q\|_F^2 + \|M\|_F^2$. Moreover, each log-likelihood term has the following lower bound

$$\begin{aligned} \log P(O^n;\Theta) &= \log \sum_{S^n} P(O^n, S^n;\Theta) \\ &\geq \log P(O^n;\Theta) - \mathrm{KL}(\mathcal{Q}(S^n)||P(S^n|O^n;\Theta)) \quad (2) \end{aligned}$$

where $\mathcal{Q}(S^n)$ is any valid distribution over the hidden state variables S^n and KL(.||.) denotes the Kullback-Leibler divergence. Let $\mathcal{F}(\mathcal{Q},\Theta)$ denote the regularized lower bound function obtained by plugging the lower bound (2) back into the objective function (1). We then perform training by using an expectation-maximization (EM) algorithm (Dempster et al., 1977) that iteratively maximizes $\mathcal{F}(\mathcal{Q},\Theta)$ to reach a local optimal solution. We first randomly initialize the model parameters while enforcing λ to be in the feasible region ($\lambda \geq 0, \lambda^\top \mathbf{1} = 1$). In the (k+1)-th iteration, given $\{\mathcal{Q}^{(k)}, \Theta^{(k)}\}$, we then sequentially update $\mathcal{Q}$ with an E-step (3) and update Θ with a M-step (4).

$$\mathcal{Q}^{(k+1)} = \arg\max_{\mathcal{Q}} \mathcal{F}(\mathcal{Q}, \Theta^{(k)}) \quad (3)$$

$$\Theta^{(k+1)} = \arg\max_{\Theta} \mathcal{F}(\mathcal{Q}^{(k+1)}, \Theta) \quad (4)$$

3.3 Domain Adaptation with Distributed State Representations

We use all training data from the two domains to train dHMMs for local optimal model parameters $\Theta^* = \{M^*, W^*, Q^*, \sigma^*, \lambda^*\}$. We then infer the latent state sequence $S^* = \{s_1^*, s_2^*, \ldots, s_T^*\}$ using the standard Viterbi algorithm (Rabiner and Juang, 1986) for each labeled source training sentence and each target test sentence. The corresponding distributed state representation vectors can be obtained as $\{M^{*\top}\phi(s_1^*), M^{*\top}\phi(s_2^*), \ldots, M^{*\top}\phi(s_T^*)\}$. We then train a supervised NLP system (*e.g.*, POS tagging or NP chunking) on the labeled source training sentences using the distributed state representations as augmenting input features and perform prediction on the augmented test sentences.

4 Experiments

We conducted experiments on cross-domain part-of-speech (POS) tagging and noun-phrase (NP) chunking tasks. We used the same experimental datasets as in (Huang and Yates, 2009) for cross-domain POS tagging from Wall Street Journal (WSJ) domain (Marcus et al., 1993) to MEDLINE domain (PennBioIE, 2005) and for cross-domain NP chunking from CoNLL shared task dataset (Tjong et al., 2000) to Open American National Corpus (OANC) (Reppen et al., 2005).

4.1 Representation Learning

We first built a unified vocabulary with all the data in the two domains. We then conducted *latent semantic analysis* (LSA) over the sentence-word frequency matrix to get a low-dimensional representation vector for each word. We used a sliding window with size 3 to construct the d-dimensional feature vector ($d = 1500$) for each observation in a given sentence. We used $\eta = 0.5$, set the number of hidden states H to be 80 and the dimensionality $m = 20$. We used all the labeled and unlabeled training data in the two domains to train dHMMs.

4.2 Test Results

We used the induced distributed state representations of each observation as augmenting features to train conditional random fields (CRF) with the CRFSuite package (Okazaki, 2007) on the labeled source sentences and perform prediction on the target test sentences. We compared with the following systems: a *Baseline* system without representation learning, a SGM based word embedding system (Hovy et al., 2015), and a discrete hidden state based clustering system (Huang and Yates, 2009). We used the word id and orthographic features as the baseline features for POS tagging and added POS tags for NP chunking. We reported the POS tagging accuracy for all words and out-of-vocabulary (OOV) words (which appear less than three times in the labeled source training sentences), and NP chunking F1 scores for all NPs and only OOV NPs (whose beginning word is an OOV word) in Table 1.

We can see that the *Baseline* method performs poorly on both tasks especially on the OOV words/NPs, which shows that the original lexical based features are not sufficient to develop a robust POS tagger/NP chunker for the target domain with labeled source training sentences. By using unlabeled training sentences from the two domains, all representation learning approaches increase the cross-domain test performance, especially on the OOV words/NPs. These improvements over the *Baseline* method demonstrate that the induced latent features do alleviate feature sparsity issue across the two domains and help the trained NLP system generalize well in the target domain. Between these representation learning approaches, the proposed distributed state representation learning method outperforms both of the word embedding based and discrete HMM hidden state based systems. This suggests that by learning distributed representations in a context-aware manner, dHMMs can effectively bridge domain divergence.

4.3 Sensitivity Analysis over the Dimensionality of State Embeddings

We also conducted experiments to investigate how does the dimensionality of the distributed state representations, m, in our proposed approach affect cross-domain test performance given a fixed state number $H = 80$. We tested a number of different m values from $\{10, 20, 30, 40\}$ and used the same experimental setting as before for each m value. The POS tagging accuracy on all words of the test sentences and the chunking F1 score on all NPs with different m values are reported in Figure 2. We can see that the performance of both POS tagging and NP chunking has notable improvements with m increasing from 10 to 20. The POS tagging performance improves very slightly from $m = 20$ to $m = 30$ and is very stable from $m = 30$ to $m = 40$. The NP chunking performance is very stable from $m = 20$ to $m = 40$. These results suggest that the distributed state representation vectors only need to have a succinct length to capture useful information. The proposed distributed HMMs are not sensitive to the dimensionality of the state embeddings as long as m reaches a reasonable small value.

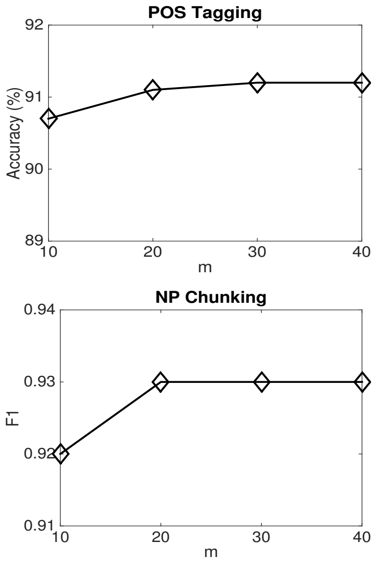

Figure 2: Cross-domain test performance with respect to different dimensionality values (m) of the hidden state representation vectors.

5 Conclusion

In this paper, we extended the standard HMMs to learn distributed state representations and facilitate cross-domain sequence predictions. We mapped each state variable to a distributed representation vector and simultaneously learned the state embedding matrix and the model parameters with an EM algorithm. The experimental results on cross-domain POS tagging and NP chunking tasks demonstrated the effectiveness of the proposed approach for domain adaptation. In the future, we plan to apply this approach to other cross-domain prediction tasks such as named entity recognition or semantic role labeling. We also plan to extend our method to learn cross-lingual representations with auxiliary resources such as bilingual dictionaries or parallel sentences.

Acknowledgments

This research was supported in part by NSF grant IIS-1065397.

References

J. Blitzer, R. McDonald, and F. Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

P. Brown, P. deSouza, R. Mercer, V. Pietra, and J. Lai. 1992. Class-based n-gram models of natural language. *Compututal Linguistics*, 18(4):467–479.

M. Candito, E. Anguiano, and D. Seddah. 2011. A word clustering approach to domain adaptation: Effective parsing of biomedical texts. In *Proc. of the Inter. Conference on Parsing Technologies (IWPT)*.

R. Collobert and J. Weston. 2008. A unified architecture for natural language processing: deep neural networks with multitask learning. In *Proc. of the Inter. Conference on Machine Learning (ICML)*.

D. Dahlmeier and H. Ng. 2010. Domain adaptation for semantic role labeling in the biomedical domain. *Bioinformatics*, 26(8):1098–1104.

H. Daumé III. 2007. Frustratingly easy domain adaptation. In *Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL)*.

A. Dempster, N. Laird, and D. Rubin. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38.

M. Dredze, J. Blitzer, P. Talukdar, K. Ganchev, J. Graça, and O. Pereira. 2007. Frustratingly hard domain adaptation for dependency parsing. In *Proc. of CoNLL Shared Task Session of EMNLP-CoNLL*.

H. Guo, H. Zhu, Z. Guo, X. Zhang, X. Wu, and Z. Su. 2009. Domain adaptation with latent semantic association for named entity recognition. In *Proc. of Human Language Technologies: The Annual Conf. of North American Chapter of ACL (HLT-NAACL)*.

D. Hovy, B. Plank, H. Alonso, and A. Søgaard. 2015. Mining for unambiguous instances to adapt pos taggers to new domains. In *Proc. of the Conference of the North American Chapter of ACL (NAACL)*.

F. Huang and A. Yates. 2009. Distributional representations for handling sparsity in supervised sequence-labeling. In *Proc. of the Annual Meeting of the ACL and the IJCNLP of the AFNLP (ACL-AFNLP)*.

F. Huang and A. Yates. 2010a. Exploring representation-learning approaches to domain adaptation. In *Proc. of the Workshop on Domain Adaptation for Natural Language Processing (DANLP)*.

F. Huang and A. Yates. 2010b. Open-domain semantic role labeling by modeling word spans. In *Proc. of the Annual Meeting of ACL (ACL)*.

F. Huang, A. Yates, A. Ahuja, and D. Downey. 2011. Language models as representations for weakly-supervised nlp tasks. In *Proc. of the Conference on Comput. Natural Language Learning (CoNLL)*.

H. Jaeger. 1999. Observable operator models for discrete stochastic time series. *Neural Computation*, 12:1371–1398.

T. Koo, X. Carreras, and M. Collins. 2008. Simple semi-supervised dependency parsing. In *Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

M. Marcus, M. Marcinkiewicz, and B. Santorini. 1993. Building a large annotated corpus of english: The Penn treebank. *Computational Linguistics*, 19(2):313–330.

T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*.

A. Mnih and E. Geoffrey. 2008. A scalable hierarchical distributed language model. In *Advances in Neural Information Processing Systems (NIPS)*.

N. Okazaki. 2007. CRFsuite: a fast implementation of conditional random fields (CRFs). http://www.chokkan.org/software/crfsuite/.

PennBioIE. 2005. Mining the bibliome project. http://bioie.ldc.upenn.edu.

L. Rabiner and B. Juang. 1986. An introduction to hidden Markov models. *IEEE ASSP Magazine*, 3(1):4–16.

R. Reppen, N. Ide, and K. Suderman. 2005. American national corpus (anc) second release. Linguistic Data Consortium.

I. Sag and T. Wasow. 1999. *Syntactic theory : a formal introduction*. CSLI publications.

K. Sagae and J. Tsujii. 2007. Dependency parsing and domain adaptation with lr models and parser ensembles. In *Proc. of CoNLL Shared Task Session of EMNLP-CoNLL*.

V. Srikumar and C. Manning. 2014. Learning distributed representations for structured output prediction. In *Advances in Neural Information Processing Systems (NIPS)*.

K. Stratos, A. Rush, S. Cohen, and M. Collins. 2013. Spectral learning of refinement HMMs. In *Proc. of the Conference on Computational Natural Language Learning (CoNLL)*.

K. Tjong, E. Sang, , and S. Buchholz. 2000. Introduction to the CoNLL-2000 shared task: Chunking. In *Proc. of the Conference on Computational Natural Language Learning (CoNLL)*.

J. Turian, L. Ratinov, and Y. Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proc. of the Annual Meeting of the Association for Comput. Linguistics (ACL)*.