Learning Hidden Markov Models with Distributed State Representations for Domain Adaptation

First Author
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

Second Author
Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

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. The experimental results demonstrate the effectiveness of the proposed distributed state representation learning of 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 (POS) 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 as different genres of text may use very different vocabularies, which produces cross-domain feature distribution divergence and feature *sparsity* issue. A number of techniques have been developed in the literature to address this cross-domain feature divergence and 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 afterwards by a simple mapping with the state embedding matrix. We then use the distributed representations of the context aware words as their augmenting features to perform cross-domain 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 gen-

eralizable features. However, they used standard HMMs to produce discrete hidden state features for each observation word, we induce distributed state representation vectors, which is similar as the word embedding based method (Hovy et al., 2015). Our distributed HMMs can also be more space efficient than the standard HMMs. Moreover, we incorporate local context information into observation feature vectors to perform representation learning in a context-aware manner. Hence the induced distributed state representations have larger representing capacities and generalizing capabilities for cross-domain learning.

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. 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. Candito et al. (2011) empirically investigated using Brown clusters (Brown et al., 1992) for out-of-domain statistical parsing.

The *word embedding based representation learning* methods learn a dense real-valued representation vector over a 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 tackle NLP cross-domain problems as well. For example, Blitzer et al. (2006) proposed a structural correspondence learning (SCL) method for POS tagging, which first selects a set of pivot features (occurring frequently in the two domains) and then models the correlation between pivot features and non-pivot features to induce generalizable features.

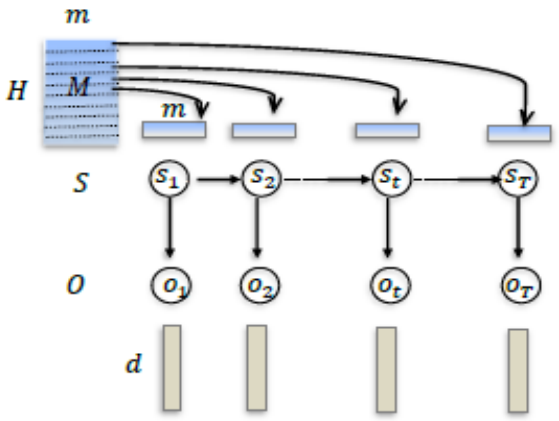

Figure 1: Hidden Markov Models with Distributed State Representations (dHMM).

3 Proposed Model

In this paper, we proposed 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 (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 the 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 dealing with 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 multi-dimensional 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 the 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: the initial state distribution, state transition distribution, and the observation emission distribution. We use Θ to denote the set of parameters involved.

First we use a multinomial *initial state distribution*,

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

where $\phi(s_t) \in \{0,1\}^H$ is a mapping function which produces a H-dim indicator vector with a single 1 value at its s_t-th entry, $\lambda \in [0,1]^H$ is the parameter vector such that $\lambda^T \mathbf{1} = 1$.

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

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

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

Finally, we assume that observation $\mathbf{o}_t \sim \mathcal{N}\left(\phi(s_t)^T MQ, \sigma I_d\right)$, and use a multivariate Gaussian model for the *emission distribution*,

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

where $\kappa(s_t, \mathbf{o}_t) = \phi(s_t)^T MQ - \mathbf{o}_t^T$,

with model parameters $Q \in \mathbb{R}^{m \times d}$ and $\sigma \in \mathbb{R}$.

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, which 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 $\Theta = \{M \in \mathbb{R}^{H \times m}, W \in \mathbb{R}^{H \times m}, Q \in \mathbb{R}^{m \times d}, \sigma \in \mathbb{R}, \lambda \in [0,1]^T\}$, where m is a very small constant.

3.2 Model Training

Let $\mathcal{D}$ denote the set of N sequences of data $\{O^1, \ldots, O^n, \ldots, O^N\}$, we can compute the log-likelihood on the data set (we drop the superscript n for the convenience of notation)

$$\begin{aligned}\mathcal{L}(\Theta) &= \log \sum_S P(O, S; \Theta) \\ &\geq \mathcal{L}(\Theta) - \mathrm{KL}(\mathcal{Q}(S)||P(S|O; \Theta)) \quad (1)\end{aligned}$$

where $\mathcal{Q}(S)$ is any non-zero distributions over the hidden state variables S and $\mathrm{KL}(.||.)$ denotes the Kullback-Leibler (KL) divergence. Let $\mathcal{F}(\mathcal{Q}, \Theta)$ denote the lower bound of the log-likelihood in (1), we then maximize it by using an iterative expectation-maximization (EM) algorithm (Dempster et al., 1977) until reach a local convergence. We first randomly initialize the model parameters while forcing λ in the feasible region ($\lambda^T \mathbf{1} = 1$). For the (k+1)-th iteration, given $\{\mathcal{Q}^{(k)}, \Theta^{(k)}\}$, we then sequentially update $\mathcal{Q}$ with an E-step (2) and update Θ with an M-step (3).

$$\mathcal{Q}^{(k+1)} = \arg\max_{\mathcal{Q}} \mathcal{F}(\mathcal{Q}, \Theta^{(k)}) \quad (2)$$

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

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) on the labeled source training sentences and target test sentences. The corresponding distributed state representation vectors can be obtained as $\{M^{*T}\phi(s_1^*), M^{*T}\phi(s_2^*), \ldots, M^{*T}\phi(s_T^*)\}$. 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.

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

Systems	POS Tagging (Accuracy)		NP Chunking (F1)	
	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**

4 Experiments

We conducted experiments on cross-domain part-of-speech (POS) tagging and noun-phrase (NP) chunking. 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 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 use a sliding window with the size of 3 to construct the d-dimensional feature vector ($d = 1500$) for each observation in a given sentence. We set the number of hidden states H to be 80 and the dimension $m = 20$. We used all the labeled and unlabeled training data in the two domains to train dHMM.

4.2 Results and Discussion

We used the induced distributed state representations of each observation as its augmenting features to train a Conditional Random Fields (CRF) based on 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 use the word id and orthographic features as the baseline features for POS tagging and add 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 OOV word).

From 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. Those 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 system. This suggests that by learning distributed representations in a context-aware manner, dHMMs can effectively bridge the domain divergence.

5 Conclusion

In this paper, we extended the standard HMMs to learn distributed state representations. We map each state variable to a distributed representation vector and simultaneously learn the state embedding matrix and other model parameters with an EM algorithm. The experimental results on cross-domain part-of-speech tagging task and noun-phrase chunking task demonstrate the effectiveness of the proposed approach for domain adaptation. In the future, we plan to apply our approach to other cross-domain 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.

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 International 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 International 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 the 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 Conference of the North American Chapter of the Association for Computational Linguistics (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 the Association for Computational Linguistics (NAACL)*.

F. Huang and A. Yates. 2009. Distributional representations for handling sparsity in supervised sequence-labeling. In *Proc. of the Joint Conference of the Annual Meeting of the ACL and the International Joint Conference on Natural Language Processing 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 the Association for Computational Linguistics (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 Computational Natural Language Learning (CoNLL)*.

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 the CoNLL Shared Task Session of EMNLP-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 Computational Linguistics (ACL)*.