HAL
open science

Domain adaptation for sequence labeling using hidden Markov models

Edouard Grave, Guillaume Obozinski, Francis Bach

To cite this version:

Edouard Grave, Guillaume Obozinski, Francis Bach. Domain adaptation for sequence labeling using hidden Markov models. New Directions in Transfer and Multi-Task: Learning Across Domains and Tasks (NIPS Workshop), Dec 2013, Lake Tahoe, United States. hal-00918371

HAL Id: hal-00918371

https://inria.hal.science/hal-00918371

Submitted on 13 Dec 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Domain adaptation for sequence labeling using hidden Markov models

Édouard Grave
Inria - Sierra Project-Team
École Normale Supérieure
Paris, France

Guillaume Obozinski
Université Paris-Est, LIGM
École des Ponts - ParisTech
Marne-la-Vallée, France

Francis Bach
Inria - Sierra Project-Team
École Normale Supérieure
Paris, France

Abstract

Most natural language processing systems based on machine learning are not robust to domain shift. For example, a state-of-the-art syntactic dependency parser trained on Wall Street Journal sentences has an absolute drop in performance of more than ten points when tested on textual data from the Web. An efficient solution to make these methods more robust to domain shift is to first learn a word representation using large amounts of unlabeled data from both domains, and then use this representation as features in a supervised learning algorithm. In this paper, we propose to use hidden Markov models to learn word representations for part-of-speech tagging. In particular, we study the influence of using data from the source, the target or both domains to learn the representation and the different ways to represent words using an HMM.

1 Introduction

Nowadays, most competitive natural language processing systems are based on supervised machine learning. Despite the great successes obtained by those techniques, they unfortunately still suffer from important limitations. One of them is their sensitivity to domain shift: for example, a state-of-the-art part-of-speech tagger trained on the Wall Street Journal section of the Penn treebank achieves an accuracy of 97% when tested on sentences from the Wall Street Journal, but only 90% when tested on textual data from the Web [1]. This drop in performance can also be observed for other tasks such as syntactic parsing or named entity recognition.

One of the explanations for this drop in performance is the big lexical difference that exists accross domains. This results in a lot of out-of-vocabulary words (OOV) in the test data, *i.e.*, words of the test data that were not observed in the training set. For example, more than 25% of the tokens of the test data from the Web corpus [1] are unobserved in the training data from the WSJ. By comparison, only 11.5% of the tokens of the test data from the WSJ are unobserved in the training data from the WSJ. Part-of-speech taggers make most of their errors on those out-of-vocabulary words.

Labeling enough data to obtain a high accuracy for each new domain is not a viable solution. Indeed, it is expensive to label data for natural language processing, because it requires expert knowledge in linguistics. Thus, there is an important need for transfer learning, and more precisely for domain adaptation, in computational linguistics. A common solution consists in using large quantities of unlabeled data, from both source and target domains, in order to learn a good word representation. This representation is then used as features to train a supervised classifier that is more robust to domain shift. Depending on how much data from the source and the target domains are used, this method can be viewed as performing semi-supervised learning or domain adaptation. The goal is to reduce the impact of out-of-vocabulary words on performance. This scheme was first proposed to reduce data sparsity for named entity recognition [2, 3], before being applied to domain adaptation for part-of-speech tagging [4, 5] or syntactic parsing [6, 7, 8, 9].

Hidden Markov models have already been considered in previous work to learn word representations for domain adaptation [4, 5] or semi-supervised learning [10]. Our contributions in this paper are mostly experimental: we compare different word representations that can be obtained from an HMM and study the effect of training the unsupervised HMM on source, target or both domains. While previous work mostly use Viterbi decoding to obtain word representations from an HMM, we empirically show that posterior distributions over latent classes give better results.

2 Method

In this section, we describe the method we use to perform domain adaptation for word sequence labeling. We suppose that we have large quantities of unlabeled word sequences from both the source domain and the target domain. In addition, we suppose that we have labeled sentences only for the source domain. Our method is a two-step procedure: first, we train an unsupervised model on the unlabeled data, second, we use this model to obtain word representations used within a supervised classifier trained on labeled data. We now describe in greater details each step of the method.

2.1 First step: unsupervised learning of an HMM

We start by training a hidden Markov chain model using unlabeled data. The words forming a sequence are denoted by the K-tuple $\mathbf{w} = (w_1, ..., w_K) \in \{1, ..., V\}^K$, where K is the length of the sequence and V is the size of the vocabulary. Similarly, we note the corresponding latent classes by the K-tuple $\mathbf{c} = (c_1, ..., c_K) \in \{1, ..., N\}^K$, where N is the number of latent classes. The joint probability distribution on words and classes thus factorizes as

$$p(W = \mathbf{w},\ C = \mathbf{c}) = \prod_{k=1}^{K} p(C_k = c_k \mid C_{k-1} = c_{k-1})\, p(W_k = w_k \mid C_k = c_k).$$

We follow the method described by [10], which consists in using the online variant of the expectation-maximization algorithm proposed by [11] in order to scale to datasets with millions of word sequences, and to use an approximate message passing algorithm [12] to perform inference. Its complexity is $O(N \log N)$, where N is the number of latent classes, compared to the $O(N^2)$ complexity of the exact message passing algorithm.[1] This allows us to train models with hundreds of latent classes on tens of millions of sequences on a single core in a day.

2.2 Second step: supervised learning of a CRF

We then use the learnt unsupervised hidden Markov model to obtain features that are used to train a conditional random field [13] using the labeled sequences. We consider three kinds of features that can be obtained using an HMM. The first one, referred to as VITERBI, consists in computing the most probable sequence of latent classes corresponding to the sentence $\mathbf{w}$ by using Viterbi decoding:

$$\hat{\mathbf{c}} = \operatorname*{argmax}_{\mathbf{c}} p(C = \mathbf{c} \mid W = \mathbf{w}) = \operatorname*{argmax}_{\mathbf{c}} p(W = \mathbf{w},\ C = \mathbf{c}).$$

Then we use the latent class $\hat{\mathbf{c}}_k$ as a categorical feature for the kth word of the sequence. The second one, referred to as POSTERIOR-TOKEN, consists in computing for the kth word of the sentence $\mathbf{w}$ its posterior distribution $\mathbf{u}^{(k)}$ over the N latent classes:

$$u_c^{(k)} = \mathbb{E}\left[\mathbf{1}\{C_k = c\} \mid W = \mathbf{w}\right].$$

This distribution is then used as N continuous features. Finally, the last one, referred to as POSTERIOR-TYPE, consists in computing for each word type $\tilde{w}$ of the vocabulary its posterior distribution $\mathbf{v}^{(\tilde{w})}$ over the N latent classes, averaged over all the utterances of this word type in the unlabeled data $\mathcal{U}$:

$$\mathbf{v}^{(\tilde{w})} = \frac{1}{Z_{\tilde{w}}} \sum_{i \in \mathcal{U}\ :\ w_i = \tilde{w}} \mathbf{u}^{(i)}$$

where $Z_{\tilde{w}}$ is the number of utterances of the word type $\tilde{w}$.

[1] We use the *k-best* approximation described by [10], setting k to $3 \log_2(N)$.

	Biomedical			Twitter		
	SOURCE	BOTH	TARGET	SOURCE	BOTH	TARGET
VITERBI	87.8	90.2	89.9	64.5	66.5	67.4
POSTERIOR-TOKEN	90.4	91.5	91.4	67.6	68.5	70.4
POSTERIOR-TYPE	90.4	89.9	92.2	68.5	69.4	71.5
POSTERIOR-BOTH	91.7	92.0	**92.9**	69.4	70.3	**72.4**
BASELINE		84.6			62.4	

Table 1: Domain adaptation results for part-of-speech tagging. Unsupervised models were trained using sentences only from the news domain (SOURCE), from both domains (BOTH) and only from the target domain (TARGET). The labeled data used to train the CRF *only* come from the *source* domain.

Let us briefly discuss the differences between those three word representations. First, the VITERBI representation is categorical, while both POSTERIOR representations are vectorial and continuous. They thus capture much finer information, but on the other hand, lead to a slower learning algorithm for CRF, since they are not sparse. Another difference is the fact that both the VITERBI and POSTERIOR-TOKEN representations depend on the context, while the POSTERIOR-TYPE representation is independent of the context, and thus all tokens with the same word type have the same representation. VITERBI, POSTERIOR-TOKEN and POSTERIOR-TYPE representations have been previously used for weakly supervised learning (see [4, 5, 10]), but were never compared.

3 Experiments

In this section, we describe the experiments we carried out on part-of-speech tagging. We use the universal part-of-speech tagset, introduced by [14], which comprises twelve coarse tags. As a baseline, we use a CRF trained on the source domain, without any word representation or other features but the word type. In the following, we consider HMM with 128 latent classes.

3.1 Datasets

In all our experiments, the source domain corresponds to news articles. We use the $2,000$ first sentences of the Wall Street Journal section of the Penn treebank as labeled data for training the CRF. The first ten years of the New York Times corpus (1987-1997) are used as unlabeled data to train the hidden Markov model. This corresponds to approximately 15 millions sentences and 300 millions tokens.

The first target domain we consider corresponds to textual data from abstracts of biomedical articles. We use the first $5,000$ sentences of the Genia treebank [15] to evaluate our method. In this test set, approximately 40% of the tokens were not observed in the training set. For the unlabeled data used to train the HMM, we have collected approximately 8 millions sentences using the PubMed online library,[2] by performing a search with the keyword *cancer*.

The second target domain we consider corresponds to textual data from Twitter. We use the 547 sentences of the DAILY547 test set of [16] to evaluate our method. In this test set, approximately 35% of the tokens were not observed in the training set. For the unlabeled data used to train the HMM, we have collected approximately 2 millions Twitter messages in English.

3.2 Comparison of word representations

In this section, we perform experiments to compare the different word representations we have introduced in section 2.2. In particular, we compare the three representations, VITERBI, POSTERIOR-TOKEN and POSTERIOR-TYPE. We also consider a fourth representation, called POSTERIOR-BOTH, which is just the concatenation of the two posterior representations introduced in section 2.2.

[2] http://www.ncbi.nlm.nih.gov/pubmed

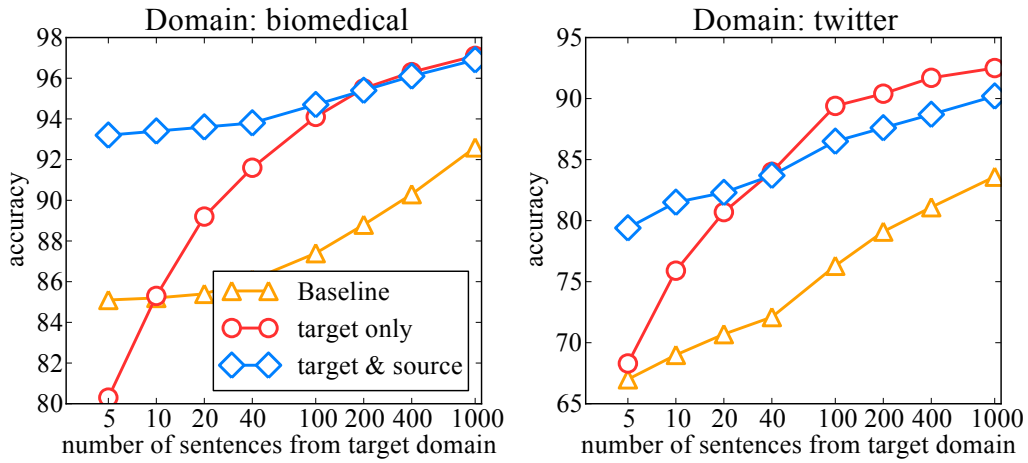

Figure 1: Domain adaptation for part-of-speech tagging. In this experiment, we consider adding labeled sentences from the target domain to the training set. BASELINE is a CRF without word representation, trained on the *labeled* sentences from both domains. TARGET ONLY is the accuracy obtained by training a CRF with word representation using only the *labeled* sentences from the target domain. The scale of the x-axis is logarithmic.

We also compare unsupervised HMM trained by using sentences only from the source domain, only from the target domain and from both domains, in order to determine the importance of the domain from which the unlabeled data come from. Indeed, as far as we now, in previous work, people have always considered using unlabeled data coming from both domains, and never investigated changing the weight of a sentence based on the domain from which it comes from. We recall that all the labeled sentences used to train the CRF come from the *source* domain. We report results for adaptation to both domains in table 1.

First, we observe that features obtained by Viterbi decoding are outperformed by posterior based representations. This is not really surprising, since the latter capture more information. We also note that using both posterior representations gives better results than using only one of them. Finally, the best results are attained for word representations trained only on unlabeled sentences from the target domain.

3.3 Using labeled data from the target domain

In this section, we assume that we also have some labeled sentences from the target domain. We compare the performance obtained by training a CRF using only this small amount of labeled data with training a CRF on labeled sentences from both the source and target domains. We report results for adaptation to the biomedical domain and Twitter in figure 1.

We note that adding sentences from the target domain to the training set improves the results differently for the two domains: we observe a 0.5 point improvement for biomedical domain and 9.5 points improvement for Twitter. Moreover we observe that when enough sentences from the target domain are available, using data from source domain for training is not useful (biomedical domain), or even gives worst results (Twitter). We have not investigated weighting sentences differently whether they come from the source or the target domain.

4 Discussion

In this paper, we presented a very simple yet efficient method for domain adaptation for sequence tagging, and conduct experiments on part-of-speech tagging on two domains. We demonstrated that this method gives very good results when no labeled data from target domain is considered. In the case where a large enough quantity of labeled data from the target domain is used, it is not necessarily useful to use data from the source domain. Whether it is possible to still leverage data from the source domain is an interesting question for future work.

References

[1] S. Petrov and R. McDonald. Overview of the 2012 shared task on parsing the web. In *Notes of the First Workshop on Syntactic Analysis of Non-Canonical Language (SANCL)*, 2012.

[2] D. Freitag. Trained named entity recognition using distributional clusters. In *EMNLP*, 2004.

[3] S. Miller, J. Guinness, and A. Zamanian. Name tagging with word clusters and discriminative training. In *HLT-NAACL*, 2004.

[4] F. Huang and A. Yates. Distributional representations for handling sparsity in supervised sequence labeling. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2009.

[5] F. Huang, A. Yates, A. Ahuja, and D. Downey. Language models as representations for weakly supervised NLP tasks. In *Computational Natural Language Learning (CoNLL)*, 2011.

[6] M. Candito, E. Henestroza Anguiano, and D. Seddah. A word clustering approach to domain adaptation: effective parsing of biomedical texts. In *Proceedings of the 12th International Conference on Parsing Technologies*, 2011.

[7] D. Seddah, B. Sagot, and M. Candito. The Alpage architecture at the SANCL 2012 shared task: robust pre-processing and lexical bridging for user-generated content parsing. In *Notes of the First Workshop on Syntactic Analysis of Non-Canonical Language (SANCL)*, 2012.

[8] K. Hayashi, S Kondo, K. Duh, and Y. Matsumoto. The NAIST dependency parser for SANCL 2012 shared task. In *Notes of the First Workshop on Syntactic Analysis of Non-Canonical Language (SANCL)*, 2012.

[9] X. Wu and D.A. Smith. Semi-supervised deterministic shift-reduce parsing with word embeddings. In *Notes of the First Workshop on Syntactic Analysis of Non-Canonical Language (SANCL)*, 2012.

[10] E. Grave, G. Obozinski, and F. Bach. Hidden Markov tree models for semantic class induction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, 2013.

[11] O. Cappé and E. Moulines. On-line expectation–maximization algorithm for latent data models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2009.

[12] C. Pal, C. Sutton, and A. McCallum. Sparse forward-backward using minimum divergence beams for fast training of conditional random fields. In *Proceedings of ICASSP.*, 2006.

[13] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the 18th International Conference on Machine Learning*, 2001.

[14] S. Petrov, D. Das, and R. McDonald. A universal part-of-speech tagset. In *Proceedings of LREC*, 2012.

[15] Y. Tateisi, A. Yakushiji, T. Ohta, and J. Tsujii. Syntax annotation for the Genia corpus. In *Proceedings of IJCNLP*, 2005.

[16] O. Owoputi, B. O'Connor, C. Dyer, K. Gimpel, N. Schneider, and N. A. Smith. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of NAACL*, 2013.