Posterior regularization for Joint Modelling of Multiple Structured Prediction Tasks with Soft Constraints

Kartik Goyal
Carnegie Mellon University
Pittsburgh, PA 15213
kartikgo@cs.cmu.edu

Chris Dyer
Carnegie Mellon University
Pittsburgh, PA 15213
cdyer@cs.cmu.edu

Abstract

We propose a multi-task learning objective for training joint structured prediction models when no jointly annotated data is available. We use conditional random fields as the joint predictive model and train their parameters by optimizing the marginal likelihood of all available annotations, with additional posterior constraints on the distributions of the latent variables imposed to enforce agreement. Experiments on named entity recognition and part-of-speech tagging show that the proposed model outperforms independent task estimation, and the posterior constraints provide a useful mechanism for incorporating domain-specific knowledge.

1 Overview

Researchers working in applied sciences like natural language processing, bioinformatics, meteorology, etc. are often interested in modeling various facets of naturally occurring data, which are often inter-related. While a world in which data is annotated jointly and consistently is pleasant to imagine, in practice different annotation guidelines exist and different data is annotated in service of different practical goals.

This paper proposes a new technique, based on posterior regularization, to learn a joint model over several tasks from disjoint annotations. Single task learning which involves independent optimization routines over these disparate datasets can be effective if enough data is available, but in low data scenarios, it helps to incorporate inductive bias about the data and the task. Multi-task learning based approaches (Caruana, 1997) often incorporate this bias by exploiting the relatedness between various facets/tasks such that several disjointly annotated datasets for different tasks can be used for joint optimization over different tasks. However, most of the existing work on multi-task learning focuses on the case when the the tasks share both the input space and output space (Obozinski et al., 2010; Jebara, 2011), which makes approaches based upon parameter tying, feature selection, kernel selection etc. suitable for these scenarios. Some examples of 'common output space' formulation of multi-task learning are binary classification and regression over multiple datasets with common output space: [0,1] and $\mathbb{R}$ respectively for classification and regression. Importantly, in this work we focus on the case in which the tasks share the input space but their output spaces are disjoint. We approach this scenario by guiding the multi task learning according to some external world knowledge about the relationship between the output spaces of different tasks.

To illustrate, consider the scenario, in which we want to train a named entity recognizer (NER) and a part of speech (PoS) tagger for a low resource language which offers very small amount of disjoint training data for each of these tasks. Typically, both these tasks are treated as sequence labeling problems (Ratnaparkhi and others, 1996; Tjong Kim Sang and De Meulder, 2003), which are modeled by undirected Markov networks like linear chain conditional random fields (CRFs) (Lafferty et al., 2001). We focus on jointly modeling these tasks with features that pertain to the relationships between the tasks. Further, we wish to guide the learning of joint models by incorporating external knowledge about relationship between the two tasks which is independent of language and can be obtained by analyzing high resource languages or from domain experts. For example, we know that it is highly likely for a Part of Speech to be `Noun` if the Named Entity is tagged as `Person`, but it is highly unlikely for a word being tagged as a Named Entity if it is a `Verb`. In this work, we propose to learn a joint CRF from the disjoint datasets and influence the learning by incorporating biases about the posterior distribution, pertaining to the inter-relationship between the tasks. Given multiple tasks modeled by CRFs with different output/label space, which share structure and sufficient statistics derivable from the observed data, we perform multi-task learning by modelling the tasks by a *joint latent* CRF, which is trained to maximize the likelihood of the disjointly annotated heterogeneous training data for different tasks. Then, we influence the learning of the *joint latent* CRF by incorporating constraints over the posterior distribution of the joint CRF, which encode relationship between the tasks. We present experimental results of our approach on joint learning of PoS tagging and NER tagging in low data scenario, and compare them with the single-task approach and the unbiased/unregularized joint modelling approaches. The encouraging results suggest that our approach of biasing joint latent CRFs via Posterior Regularization is a principled and effective way of exploiting inter task relationships. In the description below, we describe our approach and experiments with respect to linear-chain CRFs parametrized by exponential families, but our approach is general and can be applied to any set of tasks that are modeled by arbitrary CRFs that share some structure.

2 Problem statement

We are given a collection of annotated datasets $\mathcal{D} = \mathcal{D}_1, \ldots, \mathcal{D}_M$ for M tasks and each task has its training set $\mathcal{D}_m$ of $\mathbf{T}_m$ input-output pairs $(\mathbf{x}_{m,1}, \mathbf{y}_{m,1}), \ldots, (\mathbf{x}_{m,\mathbf{T}_m}, \mathbf{y}_{m,\mathbf{T}_m})$. Particularly, we are interested in structured prediction tasks where $\mathbf{x}_{j,k}$ is a sequence and the output, $\mathbf{y}_{j,k}$ is modelled by a Conditional Random Field (Lafferty et al., 2001) conditioned on global information derivable from $\mathbf{x}_{j,k}$. Typically, the output space $\mathcal{Y}_m$ of each task for a sequence is very large and disjoint i.e. $\mathcal{Y}_{j,i} \cap \mathcal{Y}_{k,i} = \emptyset \ \ \forall j,k \in 1..M, k \neq j$. For example, the output space for a sequence $\mathbf{x}$ is a set of all valid parse trees ($\mathcal{Y}_{\texttt{parse},\mathbf{x}}$) for the task of parsing and for the task of NER based upon a linear chain CRF, it is a chain of named entity predictions($\mathcal{Y}_{\texttt{NER},\mathbf{x}}$). Also, our approach focuses on the case when the datasets for the different tasks are disjoint i.e. $\mathbf{x}_j \cap \mathbf{x}_k = \emptyset \ \forall j,k \in 1..M, k \neq j$. The probability distribution characterized by a CRF for a particular task can be expressed as:

$$p(\mathbf{y}_{m,i}|\mathbf{x}_{m,i}) = \frac{1}{Z(\mathbf{x}_{m,i})} \prod_{c \in \mathcal{C}_{m,i}} \psi(\mathbf{x}_{m,i}, \mathbf{y}_{m,i,c})$$

with $\mathcal{C}_{m,i} = (\mathbf{x}_{j,k,c}, \mathbf{y}_{j,k,c})$ set of cliques in a CRF, $\psi(\mathbf{x}_{m,i}, \mathbf{y}_{m,i,c})$ is the potential for a clique c, and Z in $\sum_{\mathbf{y}\in\mathcal{Y}} \prod_{c\in\mathcal{C}} \psi(\mathbf{x}, \mathbf{y}_c)$ is the global normalization factor. The potential is a function of the input and the relevant output variables in the clique. In our experiments, we work with the distribution parametrized as an exponential family distribution: $\psi(\mathbf{x}_{m,i}, \mathbf{y}_{m,i,c}) = \exp(\theta^T \mathbf{f}(\mathbf{x}_{m,i}, \mathbf{y_{m,i,c}}))$, where $\mathbf{f}(\mathbf{x}, \mathbf{y_c})$ is a vector of informative features that can be derived from $\mathbf{x}$, and θ is the parameter vector characterizing the distribution, which is estimated during the learning phase. Parameter estimation is performed by maximizing the likelihood of the observed labels given the training sequence. The derivative w.r.t. the parameter θ_k is:

$$\mathbb{E}_{data}\ f_k(\mathbf{x}_{m,i}, y_{m,i,c}) - \mathbb{E}_{model}\ f_k(\mathbf{x}_{m,i}, y'_{m,i,c})$$

Furthermore, we have a set of constraints $\mathcal{S}$ with the individual constraints $s(\mathcal{Y}_{ci,j}, \mathcal{Y}_{ci',k})$, for tasks j and k, defined over substructures(cliques) of the structured output spaces for different tasks, which exhibit some correlation between the tasks. For example, in joint modelling of NER and parsing, there can be constraints pertaining to correlations between preterminals of the parses and the NE labels assigned to the tokens in the sequence. In this work, we aim to learn a joint probabilistic graphical model that represents $p(\mathbf{y}|\mathbf{x})$, where $\mathbf{y} \in \prod_{i=1}^{M} \mathcal{Y}_i$ over all the tasks while respecting the constraint set, from the given disjoint single task training data for each task.

$$\max_{\boldsymbol{\theta}} \sum_{m=1}^{M} \sum_{i=1}^{T_m} \log p_{\boldsymbol{\theta}}(\mathbf{y}_{m,i}|\mathbf{x}_{m,i}) \ \ s.t.$$
$$\text{satisfy}(s) = 1 \ \forall s \in \mathcal{S}$$

While not a required condition for our approach, inference with our method becomes efficient if the cliques over which the constraints are defined, share structural similarities across the tasks.

Hence for NER and PoS, if $\mathbf{x}$ contains w words, then the output space for NER and PoS tagging is $\mathcal{Y}_{\text{NER},\mathbf{x}} = (\mathcal{T}_{NER})^w$ and $\mathcal{Y}_{\text{PoS},\mathbf{x}} = (\mathcal{T}_{PoS})^w$. It is important to note that $\mathcal{T}_{NER} \cap \mathcal{T}_{PoS} = \emptyset$. Also, since both the tasks are modeled as linear chain CRFs, for a given sequence $\mathbf{x}$, they share a similar clique structure, which makes the joint inference easier. The constraint set $\mathcal{S}$ consists of several constraints that exhibit the relationship between the two tasks. These constraints can be formulated by domain experts or can also be transferred from the large related domain corpora if the constraints are not sensitive to the domains. For this pair of tasks, constraints can be defined on all the cliques (node and edge based) without requiring any changes in the inference algorithm. However, even for two tasks with very different CRF structures (For eg. constituency parsing and NER), we can facilitate sharing of node based cliques at preterminals of the parse trees and the nodes of linear chain CRFs for NER. For simplicity of exposition, further discussion will assume $M = 2$, and CRF for each task is a first order linear chain CRF.

3 Unregularized Models

In this section, we'll first consider a fully supervised scenario, in which labels for both the tasks are available for each sequence $\mathbf{x}$ in the training data. After that, we will discus *latent joint* CRF, which will be used to model the actual scenario, in which we have output labels from only one of the tasks for each sequence $\mathbf{x}$ in the training data.

3.1 Fully supervised joint CRF

Full supervision requires that each input sequence $\mathbf{x}$ is annotated for all tasks. Our motivating assumption is that this is an ideal scenario, but not always available. Additionally, this model lays the foundation of the latent joint model we discus in the next section. The joint CRF is a simple modification of the single task CRF. For linear chain CRF models associated with the tagging tasks, we simply consider the expanded tag-space $\mathcal{T}_{joint} = \mathcal{T}_{task1} \times \mathcal{T}_{task2}$. Now, for a sequence $\mathbf{x}$ of length w, the size of the output space is $\mathcal{Y}_{joint} = \mathcal{T}^w_{joint}$ and the CRF distribution is parametrized as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x};\theta)} \times \sum_{t}^{w} \exp(\theta^T \mathbf{f}(\mathbf{x}, y_{joint,t}, y_{joint,t-1}, t)) \times \exp(\theta^T \mathbf{f}(\mathbf{x}, y_{joint,t}, t))$$

where $y_{joint,t} = (y_{task1,t}, y_{task2,t})$. It should be noted that for the joint model we have new kind of transition and label features based upon the task identities: $\mathbf{f}(\mathbf{x}, y_{joint,t}, y_{joint,t-1})$, $\mathbf{f}(\mathbf{x}, y_{task1,t}, y_{task1,t-1})$, $\mathbf{f}(\mathbf{x}, y_{task2,t}, y_{task2,t-1})$, $\mathbf{f}(\mathbf{x}, y_{joint,t})$, $\mathbf{f}(\mathbf{x}, y_{task1,t})$, $\mathbf{f}(\mathbf{x}, y_{task2,t})$. Hence this model is much larger than the single task model both in terms of output-space($\mathcal{Y}$) and the feature space.

Exact inference for parameter estimation and finding the best sequence can be performed by algorithms similar to the ones used for the single linear chain CRFs.

3.2 Joint Latent CRF

This model is very much similar to the model described in the previous section, but, in this case we work with the original data scenario i.e. several small single task datasets output labels for only one of the tasks provided. During parameter estimation, the joint CRF model observes only partial output, so marginalization over the latent output variables is required. The objective function in this case is to maximize the likelihood of the partial output, given the input sequence $\mathbf{x}$:

$$\max_{\theta} \sum_{m=1}^{2} \sum_{i=1}^{T_m} \log p_\theta(\mathbf{y}_{m,i}|\mathbf{x}_{m,i}) =$$

$$\max_{\theta} \sum_{m=1}^{2} \sum_{i=1}^{T_m} \log \sum_{k=1}^{\mathbf{y}_{m-,i}} p_\theta(\mathbf{y}_{k,i}|\mathbf{x}_{m,i}).$$

For the latent model, the gradient w.r.t. θ_k is: The derivative w.r.t. the parameter θ_k is:

$$\sum_{m-1}^{2} \sum_{j=1}^{T_m} \sum_{i=1}^{|\mathbf{x}|} \mathbb{E}_{p(\mathbf{y}_{m-,i}|\mathbf{y}_{m,i},\mathbf{x}_{m,i})} f_k(\mathbf{x}_{m,i}, y_i, y_{i-1}) - \mathbb{E}_{p(\mathbf{y}'|\mathbf{x}_{m,i})} f_k(\mathbf{x}_{m,i}, y'_i, y'_{i-1})$$

From the above equation, we notice that as far as inference is considered, the only change in this model, when compared to the completely supervised joint model, is that inference now involves marginalization over all the latent output labels. The inference for computing the expectation quantities and the marginal probabilities can still be done modifying the junction tree algorithm used in the supervised joint CRF accordingly. However, the objective now is non convex and parameter estimation is done via a discriminative EM procedure.

The advantage of this model is that now it allows us to train a joint model over both the datasets with informative features pertaining to both the tasks, which was not possible with single task CRF models. It is expected that this method will learn to incorporate certain correlations between the two tasks just by the virtue of looking at different training datasets and learning features pertaining to both the output labels. Moreover, this model also lays the basis for the model discussed in the next section which regularizes the posterior distribution of this latent model.

4 Constraint based regularization for Multi Task Learning

In this section, we describe our method to influence the parameter learning of the *latent joint* CRF described in the last section, according to the constraints pertaining to the relationship between the tasks that we are interested in. The motivation behind this approach is that often, varying sources of information about the tasks and we would like to expose our models to information beyond what is provided by the annotated training data.

The constraints could be in the form of biases based upon world knowledge that are provided by the domain experts, or they could determined empirically by analysis of related domains which expose relationships among the relevant tasks. For example, compatibility of part of speech tags and named entity tags is largely invariant across several languages. In scenarios, where the multiple tasks have non intersecting output spaces, these constraints can convey information about the relation between the output spaces. Now we discus our method to bias the joint latent CRF for multitask learning and we will also discus about the kinds of constraints and information our method easily allows to incorporate.

4.1 Posterior Regularization

Posterior Regularization(Ganchev et al., 2010; Zhu et al., 2014) is an effective technique to perform constraint based learning when the original model's parameters are learned via Expectation Maximization. (1998) showed that both $\mathbf{M}$ and $\mathbf{E}$ steps are maximization problems over a function that is dependent on the model parameters and the distribution over the latent variables respectively, and is also a lower bound for the log-likelihood of the observed data.

$$\mathcal{L}(\theta) = \mathbb{E}_{data}(log \sum_{y} p(x,y))$$

$$\geq \mathbb{E}_{data}(\sum_{y} q(y|x) log \frac{p_\theta(x,y)}{q(y|x)}) = F(q,\theta)$$

where x is the observed variable and y is the hidden variable. The standard EM procedure amounts to:

$$\mathbf{E} : q^{t+1}(\mathbf{y}|\mathbf{x}) = \arg \max_{q} F(q,\theta^t) =$$

$$\arg \min_{q} KL(q(\mathbf{y}|\mathbf{x}) \mid\mid p_{\theta^t}(\mathbf{y}|\mathbf{x})) = p_{\theta^t}(\mathbf{y}|\mathbf{x})$$

$$\mathbf{M} : \theta^{t+1} = \arg \max_{\theta} F(q^{t+1},\theta) =$$

$$\arg \max_{\theta} \mathbb{E}_{data}(\sum_{y} q^{t+1}(y|x) log p_\theta(x,y))$$

Posterior Regularization refers to modifying the E-step such that the q(y|x) distribution that is estimated in the E-step also respects certain linear Expectation based constraints belonging to the constraint set $\mathcal{S}$. The M-step is typically left unchanged. This has an effect of regularizing the expectations of the hidden variables. In the context of the joint latent CRF, the hidden task output variables are the latent variables and rest of the variables are observed. Formally, the E-step can be described as:

$$\begin{aligned} &\arg\min_{q} \ \mathrm{KL}(q(\mathbf{y} \mid \mathbf{x}) \mid\mid p_\theta(\mathbf{y} \mid \mathbf{x})) \\ &\quad \text{subject to} \\ &\quad \mathbb{E}_q(\phi(\mathbf{x},\mathbf{y})) - \mathbf{b} \leq \xi, \\ &\quad ||\xi||_\beta < \epsilon, \\ &\quad \sum_{\mathbf{y}} q(\mathbf{y} \mid \mathbf{x}) = 1 \end{aligned}$$

where $\phi(\mathbf{x}, \mathbf{y})$ are constraint features that can be computed from the input and the output, and $\mathbf{b}$ are respective expected values of the constraint features over a corpus. This framework can handle constraints based on the expected values of certain quantities over training data, under the model's distribution. ξ is the slack parameter, which relaxes the necessity for exactly matching the expectation of constraint features under model with $\mathbf{b}$. Assuming that a feasible $q(\mathbf{y})$ exists, this E-step optimization problem can be solved by solving by using Lagrangian duality and solving the dual problem. The solution of the dual problem results in the following form of the $q(\mathbf{y})$ distribution:

$$q^*(\mathbf{y} \mid \mathbf{x}) = \frac{p_\theta(\mathbf{y}|\mathbf{x}) \exp(-\lambda^{*T} \phi(\mathbf{x}, \mathbf{y}))}{Z(\lambda^*, \theta)} \tag{1}$$

where λ^* is the solution of the dual problem and $Z(\lambda^*, \theta) = \sum_{\mathbf{y}} p_\theta(\mathbf{y}|\mathbf{x}) \exp(-\lambda^{*T}.\phi(\mathbf{x}, \mathbf{x}))$, is the normalization constant for $q(\mathbf{Y})$.

The associated dual problem with parameter vector λ is:

$$\lambda^* = \arg\max_{\lambda \geq \mathbf{0}} -\mathbf{b}\lambda - log(Z(\lambda, \theta)) - \epsilon||\lambda||_{\beta^*} \tag{2}$$

where $||.||_{\beta^*}$ is the conjugate of norm $||.||_\beta$. In our experiments, we set $\beta = \infty$ such that $\beta^* = 1$. Hence, the dual optimization can be carried out by proximal gradient ascent with the following update for λ_k pertaining to the k^{th} constraint:

$$\lambda_k^{i+1} = S_{t\epsilon}(\lambda_k^i + t(-b_k - \frac{d \log\ Z(\lambda, \theta)}{d\lambda})) = \\ S_{t\epsilon}(\lambda_k^i + t(-b_k + \mathbf{E}_q(\phi_k(\mathbf{x}, \mathbf{y}))))$$

where t is the step size and $S_{t\epsilon}()$ is the soft thresholding operator.

An important observation to be made here is that if p_θ is modeled by a CRF parametrized by an exponential family distribution and the computation of $\phi(\mathbf{x}, \mathbf{y})$ decomposes according to the cliques of the CRF representing p_theta then the approximating q distribution has the form:

$$q(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x}; \theta, \lambda)} \prod_{c \in \mathcal{C}} \exp(\theta^T \mathbf{f}(\mathbf{x}, \mathbf{y_c}) - \lambda^T \phi(\mathbf{x}, \mathbf{y_c})) \tag{3}$$

This form enables us to perform inference with $q(\mathbf{y} \mid \mathbf{x})$ efficiently by using exactly same inference routine as the one used for carrying out inference with $p(\mathbf{y} \mid \mathbf{x})$. Therefore, in our experiments with *first-order linear-chain* CRFs, we work with the constraint features that can be computed locally along the nodes and edges of the CRF.

For example, 'ϕ = proportion of the label (Person,Noun)' can be computed incrementally by using marginal probability of the label at each node of the CRF, which is an artefact of the inference algorithm in linear chain CRFs.

Similarly, 'ϕ = proportion of the edge (Person,Noun) → (Not-NE,VERB)', also can be computed fairly easily by using the marginal probability of the edges of the CRF.

However, a constraint feature like 'ϕ = proportion of NE=Person, given PoS=Noun' does not decompose along the graph of a first order linear chain CRF and cannot be computed incrementally along the structure of the CRF. Therefore, incorporating this type of constraint will make inference harder and we don't address this problem in this work, and assume that the computation of the constraint features is decomposable according to the structure of the joint CRF. The EM procedure for the latent joint CRF becomes:

E-step: Compute the optimal dual parameters(λ^*) for the constraint features by optimizing Eqn 2. Then, use λ^* to compute $q^{i+1}(\mathbf{y}|\mathbf{x})$ using Eqn 3

M-step: Compute the optimal CRF feature parameters θ by maximizing the likelihood of the training data consisting of partially observed output, conditioned on the input sequence:

$$\theta^{i+1} = \arg\ \max\ \mathbb{E}_{data}(\mathbb{E}_{q(\mathbf{y}|\mathbf{x})}\ logp_\theta(\mathbf{y}|\mathbf{x}))$$

The above EM procedure can be interpreted as block co-ordinate descent over the parameters of a linear chain CRF that characterizes the distribution $q(\mathbf{y} \mid \mathbf{x})$. While this perspective leads us to view λ and θ as similar parameters, both of them are subtly different. θ is responsible for matching model expectation ($\mathbb{E}_{model}(f)$) of features with the empirical expectation of the features ($\mathbb{E}_{data}(f)$), λ on the other hand is responsible for matching model expectation of constraint features ($\mathbb{E}_{model}(\phi)$), with the external bias ($\mathbf{b}$).

5 Related Work

Posterior Regularization was proposed by (2010). It was also expressed in a more general form and extended to Bayesian non parametric models by (Zhu et al., 2014), who also show that the real expressive power of PR lies in modelling external constraints based upon corpus statistics in addition to the model parameters, which differentiates it from regular Bayesian treatment of external knowledge as parameter based priors. It is very closely related to the Bayesian Measurements framework (Liang et al., 2009) which is more abstract than Posterior Regularization, in which the constraint features ϕ are measured with noise as $\mathbf{b}$.

$$\mathbf{b} = \phi(\mathbf{x}, \mathbf{y}) + \text{noise}_{\phi}$$

The noise $\log(p(\mathbf{b}|\phi, \mathbf{X}, \mathbf{Y})) = -h_{\phi}(b - \phi(\mathbf{X}, \mathbf{Y}))$ with convex h, is modeled as a log concave noise so that over all MAP objective is convex. In particular, it is modeled as box noise ($b \leq \mathbb{I}_{\phi(\mathbf{X},\mathbf{Y})+/-\epsilon}$). According to this framework, not only the constraint features, but also, fully annotated training data examples themselves are considered Bayesian measurements.Assuming a Bayesian setting with a prior on θ, the model distribution is:

$$p(\theta, \mathbf{y}, \mathbf{b}|\mathbf{x}, \phi) = p(\theta)\, p(\mathbf{y}|\mathbf{x}; \theta)\, p(\mathbf{b}|\mathbf{x}, \mathbf{y}, \phi) \quad (4)$$

(2009) approximate to the posterior of $p(\mathbf{Y}, \theta|\mathbf{X}, \phi, \mathbf{b})$ by mean field factorization and further relaxing the problem to be able to leverage duality for the solution. With their approximation, they arrive at the objective of Posterior Regularization. The key to their model and optimization lies in the noise used to model the measurements and also the variational approximation procedure to optimize an approximate objective. In particular, box noise is responsible for the constraints in their model to be linear expectation based constraints. Other log concave noise distributions offer the potential to model other non-linear constraints as well.

There is a lot of work pertaining to semi-supervised learning using external biases in the form of either hard or soft constraints. Like Posterior Regularization, Generalized expectation(Druck et al., 2008) is able to incorporate soft constraints defined over a whole distribution of labels by adding the expectation based constraints to the objective(MLE) of the problem. Although this is an appealing method, it can be very expensive to run because the gradient calculations depend on the cross product of model feature space and model constraint space. In fact, Posterior Regularization can be seen as a variational approximation to the objective of GE criterion (Ganchev et al., 2010). PR and GE have been shown to be useful in incorporating soft constraints for various tasks like bilingual NER (Che et al., 2013), cross lingual projection of coreference(Martins, 2015) etc. There has been plenty of work to bias predictions/ learning of structured prediction models in presence of hard constraints, which incorporate discrete penalty associated with label combinations relevant to the constraint features. A key difference of these models from Posterior Regularization is that instead of working with expected counts of output labels, they work with hard count assignments. The constraint driven learning approach of (2007) adds a penalty term to the conditional log probability of the output that can be seen as adding cost deterministically for violating the constraints. Their approach is usually intractable practically and approximations like beam search are used. Also, (2010) show that dual decomposition methods can be very effective for different related tasks with hard constraints based upon the relatedness of the tasks. This method solves the joint objective of the different tasks and forces and agreement between predictions of different tasks according to the hard constraints that inter-relate their output spaces. This is an effective approach if the relationship between the output spaces of the two tasks is perfectly deterministic.However, this approach only improves joint inference and isn't very effective at learning parameters of the model w.r.t. the constraints. Another popular approach for constraint based inference is using Integer Linear Programming (Roth and Yih, 2005), but this too doesn't focus on guiding learning of joint models using the constraints. Both the hard count based approaches are unsuitable for modelling the problem described in this paper, which aims at using non-deterministic soft-constraints pertaining to the relationship between the tasks to bias the learning of the models.

Multi Task learning refers to a very broad array of problem scenarios and techniques(Caruana, 1997; Thrun and Pratt, 2012) which are motivated by a common hypothesis: Modelling multiple inter-related tasks enables us to work with a larger amount of data and has the potential to transfer statistical information across various tasks, domains and datasets, such that generalization performance of the predictive models improves for all of the tasks. Most of approaches (Obozinski et al., 2010; Jebara, 2011) assume that the multiple tasks have the same input space($x \in \mathbb{R}^d$) and also share the output space; eg. $\mathbb{R}$ for regression and 0,1 for classification based tasks. These multi-task learning techniques include sparse feature selection via group l1 regularization(Obozinski et al., 2010), feature transformation to jointly train over all the tasks(Evgeniou and Pontil, 2004), kernel selection (Jebara, 2011) etc. Crucially, in our work, we work with multiple tasks that have different output spaces. In fact, in our approach and experiments, the output label spaces are completely disjoint. Hence, we try to bias our probabilistic models by soft constraints encoding the relationship between the output spaces of different tasks.

Table 1: Sizes of the different training datasets.

Data set	#NER instances	#PoS instances
Base (1x)	219	223
Base×2 (2x)	442	444
Base×4 (4x)	886	873

Table 2: Constraints used for the experiments. UB and LB refer to the upper and lower bounds on the expectations

$\phi(proportion)$	$\mathbf{b}(UB)$	$\mathbf{b}(LB)$
(O,NOUN)	0.21	0.18
(I-PER,NOUN)	0.055	0.053
(I-ORG,ADJ)	0.046	0.44
(I-LOC,NOUN)	0.041	0.039
(I-MISC,NOUN)	0.016	0.013
(I-PER,NOUN)→ (I-PER,NOUN)	0.028	0.023
(I-ORG,NOUN)→ (I-ORG,NOUN)	0.018	0.015
(I-LOC,NOUN)→ (O,.)	0.017	0.014
(I-PER,NOUN)→ (O,.)	0.018	0.015
(I-ORG,NOUN)→ (O,NUM)	0.015	0.011
(O,.)→ (I-LOC,NOUN)	0.013	0.010

6 Experiments

We performed experiments on jointly modelling two tasks: 1) Named Entity Recognition(NER) and 2) Part of Speech (PoS) tagging. For NER, we follow the standard convention of ‘B-I-O tagging’ (Tjong Kim Sang and De Meulder, 2003; Sha and Pereira, 2003) where ‘B’ and ‘I’ help identify segments of named entities and ‘O’ identifies the words that are not named entities. For PoS, we used the ‘Universal’ PoS tagset, which is largely invariant across several languages (Petrov et al., 2011). The tagset for the two tasks was:

```
NER: [O, B-PER, I-PER, B-ORG,
I-ORG, B-LOC, I-LOC, B-MISC, I-MISC]

POS: [VERB, NOUN, PRON, ADJ, ADV,
ADP, CONJ, DET, NUM, PRT, X, .]
```

Since, we wish to study the effect of the size of training data, we used the standard English ConLL dataset (Tjong Kim Sang and De Meulder, 2003) for both NER and PoS tagging models and artificially impoverished the data by randomly sampling disjoint task specific datasets of varying sizes (Table 1). For training all of the CRF models (single, supervised joint, latent joint, and posterior regularized joint), we used a standard set of indicator features derivable from the input sequence.

For obtaining informative constraints, we used the statistics from a large Spanish NER dataset (Tjong Kim Sang and De Meulder, 2003). We specifically chose this setting to gauge the ease and efficacy of language invariant relationships between NER and PoS tagging tasks. Specifically, we focused on the expected proportions of the joint labels and the joint edges in the training corpus. We also used the performance on our development set to identify a small pool of constraintswhich are listed in Table 2. It should be noted that depending upon the specific data and task settings many other kinds of informative constraints, that also condition on observed sequence $\mathbf{x}$ can be easily incorporated as long as their computation decomposes along the cliques of our joint models. For numerical stability, the constraints in table 2 were scaled to be in the same range by scaling ϕ.

Table 3: Performance on NER and Part of Speech tagging. 'P', 'R', 'F1' stand for Precision, Recall and F1 score for Named Entity Recognition task, 'Acc' refers to part of speech tagging accuracy. 1x, 2x, and 3x refer to the data sets for the two tasks as described in table 1. 'Single-task' refers to independent training of CRFs for the two tasks, 'Latent CRF' refers to the joint CRF trained over partially observed data via EM, 'Posterior Reg.' refers to out approach of regularizing the output distribution of 'Latent CRF', 'Oracle' refers to the unrealistic case when both the datasets are annotated with both task outputs.

SZ	SINGLE-TASK				LATENT-CRF				**POSTERIOR REG.**				ORACLE (2X SUPERVISED)			
	P	R	F1	ACC	P	R	F1	ACC	**P**	**R**	**F1**	**ACC**	P	R	F1	ACC
1x	0.67	0.39	0.50	0.83	0.53	0.51	0.52	0.84	0.58	0.53	0.55	0.84	0.67	0.63	0.65	0.88
2x	0.69	0.55	0.62	0.87	0.63	0.62	0.63	0.87	0.66	0.62	0.64	0.87	0.73	0.71	0.72	0.90
4x	0.79	0.61	0.69	0.89	0.71	0.70	0.70	0.90	0.70	0.71	0.70	0.90	0.79	0.77	0.77	0.92

6.1 Results

Our experimental focus is on comparing our approach of regularizing the output distribution of a joint CRF with other approaches described in the paper: i) training a single CRF for each task with its respective data. ii) training a latent joint CRF over both the datasets jointly via EM. We also present results for the fully supervised joint CRF model, which was trained assuming the unrealistic scenario, in which we have annotations for both the tasks in our training data. This effective doubles the training data for the fully supervised joint CRF. This provides an effective upper bound on the performance of the joint CRF model. These results are reported over the CoNLL test set which consists of 3250 sequences. We notice in table 3 that for all the three data scenarios, the single task CRFs perform the worst on both the tasks. The latent CRF based approach consistently improves over the single task performance. The posterior regularization models further improve over the latent CRF performance. The improvement with posterior regularization is most pronounced for the smallest dataset. The part of speech tagging accuracy improves slightly for smallest data scenario with our approach, but it is comparable for the larger data scenarios. This might be because PoS tagging is a considerably easier problem and relies less on the 'structure' in the model than NER (Liang et al., 2008).

Another consistent pattern is that the 'Oracle' is always significantly better at both the tasks in all the data settings because it is trained on fully annotated dataset of both the tasks for a give data scenario. Interestingly, its performance is always slightly better than the single model scenario with 2x data. This suggests that joint CRF modelling is providing some gains over independent task training and empirically the effect on sample complexity due to the bigger CRF model doesn't seem to hurt at all.

7 Conclusion

We presented a multi task learning approach based upon jointly modelling structured prediction tasks when no jointly annotated data is available. We presented a latent CRF model to jointly model the two tasks, whose output posterior distribution is influenced by constraints that encode some external knowledge about the tasks and their interrelationships. Specifically, we assume that the output spaces of the different tasks do not necessarily intersect, and instead we only know about the tendencies of compatibility between the different output spaces. We bias the learning of our models by using this external knowledge about the tasks. We report experimental results on two Natural Language Processing tasks: i) Named Entity Recognition and ii) Part of speech tagging. Our results show that our method is very effective in low data scenarios and always is significantly better that training individual models on small datasets.

References

Rich Caruana. 1997. Multitask learning. *Machine learning*, 28(1):41–75.

Ming-Wei Chang, Lev Ratinov, and Dan Roth. 2007. Guiding semi-supervision with constraint-driven learning. In *Annual Meeting-Association for Computational Linguistics*, volume 45, page 280.

Wanxiang Che, Mengqiu Wang, Christopher D Manning, and Ting Liu. 2013. Named entity recognition with bilingual constraints. In *HLT-NAACL*, pages 52–62.

Gregory Druck, Gideon Mann, and Andrew McCallum. 2008. Learning from labeled features using generalized expectation criteria. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 595–602. ACM.

Theodoros Evgeniou and Massimiliano Pontil. 2004. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117. ACM.

Kuzman Ganchev, Joao Graça, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior regularization for structured latent variable models. *The Journal of Machine Learning Research*, 11:2001–2049.

Tony Jebara. 2011. Multitask sparsity via maximum entropy discrimination. *The Journal of Machine Learning Research*, 12:75–110.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Percy Liang, Hal Daumé III, and Dan Klein. 2008. Structure compilation: trading structure for features. In *Proceedings of the 25th international conference on Machine learning*, pages 592–599. ACM.

Percy Liang, Michael I Jordan, and Dan Klein. 2009. Learning from measurements in exponential families. In *Proceedings of the 26th annual international conference on machine learning*, pages 641–648. ACM.

André FT Martins. 2015. Transferring coreference resolvers with posterior regularization. ACL.

Radford M Neal and Geoffrey E Hinton. 1998. A view of the em algorithm that justifies incremental, sparse, and other variants. In *Learning in graphical models*, pages 355–368. Springer.

Guillaume Obozinski, Ben Taskar, and Michael I Jordan. 2010. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2011. A universal part-of-speech tagset. *arXiv preprint arXiv:1104.2086*.

Adwait Ratnaparkhi et al. 1996. A maximum entropy model for part-of-speech tagging. In *Proceedings of the conference on empirical methods in natural language processing*, volume 1, pages 133–142. Philadelphia, USA.

Dan Roth and Wen-tau Yih. 2005. Integer linear programming inference for conditional random fields. In *Proceedings of the 22nd international conference on Machine learning*, pages 736–743. ACM.

Alexander M Rush, David Sontag, Michael Collins, and Tommi Jaakkola. 2010. On dual decomposition and linear programming relaxations for natural language processing. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1–11. Association for Computational Linguistics.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 134–141. Association for Computational Linguistics.

Sebastian Thrun and Lorien Pratt. 2012. *Learning to learn*. Springer Science & Business Media.

Erik F Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics.

Jun Zhu, Ning Chen, and Eric P Xing. 2014. Bayesian inference with posterior regularization and applications to infinite latent svms. *The Journal of Machine Learning Research*, 15(1):1799–1847.