Structure Regularization for Structured Prediction

Authored by:

Xu Sun

Abstract

While there are many studies on weight regularization, the study on structure regularization is rare. Many existing systems on structured prediction focus on increasing the level of structural dependencies within the model. However, this trend could have been misdirected, because our study suggests that complex structures are actually harmful to generalization ability in structured prediction. To control structure-based overfitting, we propose a structure regularization framework via emph{structure decomposition}, which decomposes training samples into mini-samples with simpler structures, deriving a model with better generalization power. We show both theoretically and empirically that structure regularization can effectively control overfitting risk and lead to better accuracy. As a by-product, the proposed method can also substantially accelerate the training speed. The method and the theoretical results can apply to general graphical models with arbitrary structures. Experiments on well-known tasks demonstrate that our method can easily beat the benchmark systems on those highly-competitive tasks, achieving record-breaking accuracies yet with substantially faster training speed.

1 Paper Body

Structured prediction models are popularly used to solve structure dependent problems in a wide variety of application domains including natural language processing, bioinformatics, speech recognition, and computer vision. Recently, many existing systems on structured prediction focus on increasing the level of structural dependencies within the model. We argue that this trend could have been misdirected, because our study suggests that complex structures are actually harmful to model accuracy. While it is obvious that intensive structural dependencies can effectively incorporate structural information, it is less obvious that intensive structural dependencies have a drawback of increasing the generalization risk, because more complex structures are easier to suffer from overfitting. Since this type of overfitting is caused by structure complexity, it can hardly be solved by ordinary regularization methods such as L2 and L1

regularization schemes, which is only for controlling weight complexity. To deal with this problem, we propose a simple structure regularization solution based on tag structure decomposition. The proposed method decomposes each training sample into multiple minisamples with simpler structures, deriving a model with better generalization power. The proposed method is easy to implement, and it has several interesting properties: (1) We show both theoretically and empirically that the proposed method can effectively reduce the overfitting risk on structured prediction. (2) The proposed method does not change the convexity of the objective function, such that a convex function penalized with a structure regularizer is still convex. (3) The proposed method has no conflict with the weight regularization. Thus we can apply structure regularization together with weight regularization. (4) The proposed method can accelerate the convergence rate in training. The term structural regularization has been used in prior work for regularizing structures of features, including spectral regularization [1], regularizing feature structures for classifiers [20], and many 1

recent studies on structured sparsity in structured prediction scenarios [11, 8], via adopting mixed norm regularization [10], Group Lasso [22], and posterior regularization [5]. Compared with those prior work, we emphasize that our proposal on tag structure regularization is novel. This is because the term structure in all of the aforementioned work refers to structures of feature space, which is substantially different compared with our proposal on regularizing tag structures (interactions among tags). Also, there are some other related studies. [17] described an interesting heuristic piecewise training method. [19] described a ?lookahead" learning method. Our work differs from [17] and [19] mainly because our work is built on a regularization framework, with arguments and theoretical justifications on reducing generalization risk and improving convergence rate. Also, our method and the theoretical results can fit general graphical models with arbitrary structures, and the detailed algorithm is very different. On generalization risk analysis, related studies include [2, 12] on non-structured classification and [18, 7] on structured classification. To the best of our knowledge, this is the first theoretical result on quantifying the relation between structure complexity and the generalization risk in structured prediction, and this is also the first proposal on structure regularization via regularizing tag-interactions. The contributions of this work1 are two-fold: ? On the methodology side, we propose a structure regularization framework for structured prediction. We show both theoretically and empirically that the proposed method can effectively reduce the overfitting risk, and at the same time accelerate the convergence rate in training. Our method and the theoretical analysis do not make assumptions based on specific structures. In other words, the method and the theoretical results can apply to graphical models with arbitrary structures, including linear chains, trees, and general graphs. ? On the application side, for several important natural language processing tasks, our simple method can easily beat the benchmark systems on those highly-competitive tasks, achieving record-breaking accuracies as well as substantially faster training speed.

2 Structure Regularization A graph of observations (even with arbitrary

structures) can be indexed and be denoted by using an indexed sequence of observations O = {o1 , . . . , on }. We use the term sample to denote O = {o1 , . . . , on }. For example, in natural language processing, a sample may correspond to a sentence of n words with dependencies of tree structures (e.g., in syntactic parsing). For simplicity in analysis, we assume all samples have n observations (thus n tags). In a typical setting of structured prediction, all the n tags have inter-dependencies via connecting each Markov dependency between neighboring tags. Thus, we call n as tag structure complexity or simply structure complexity below. x(1) , . . . , x(n) }, where x(k) ? A sample is converted to an indexed sequence of feature vectors x = {x X is of the dimension d and corresponds to the local features extracted from the position/index k. x, y) ? Z denote We can use an n ? d matrix to represent x ? X n . Let Z = (X n , Y n) and let z = (x x1 , y 1), . . . , z m = (x xm , y m)}, a sample in the training data. Suppose a training set is S = {zz 1 = (x with size m, and the samples are drawn i.i.d. from a distribution D which is unknown. A learning algorithm is a function G : Z m 7? F with the function space F ? {X n 7? Y n }, i.e., G maps a training set S to a function GS : X n 7? Y n . We suppose G is symmetric with respect to S, so that G is independent on the order of S. Structural dependencies among tags are the major difference between structured prediction and nonstructured classification. For the latter case, a local classification of g based on a position k can be x(k?a) , . . . , x (k+a)), where the term {x x(k?a) , . . . , x (k+a) } represents a local winexpressed as g(x dow. However, for structured prediction, a local classification on a position depends on the whole x(1) , . . . , x (n) } rather than a local window, due to the nature of structural dependencies input x = {x among tags (e.g., graphical models like CRFs). Thus, in structured prediction a local classification x(1) , . . . , x (n) , k). To simplify the notation, we define on k should be denoted as g(x x, k) , g(x x(1) , . . . , x (n) , k) g(x 1

See the code at http://klcl.pku.edu.cn/member/sunxu/code.htm

2

y y

(1)

(2)

y

(2)

x

y

(3)

y

(3)

x

(4)

y

(5)

y

x

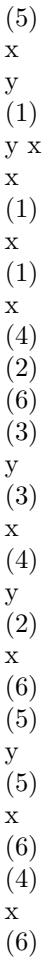

Figure 1: An illustration of structure regularization in simple linear chain case, which decompose a training sample z with structure complexity 6 into three mini-samples with structure complexity 2. Structure regularization can apply to more general graphs with arbitrary dependencies. x, k), y (k)], which measures the cost on We define point-wise cost function c : Y ?Y 7? R+ as c[GS (x x, k) and the gold-standard tag y (k) , and we introduce the point-wise a position k by comparing GS (x loss as x, k), y (k)] ?(GS , z , k) , c[GS (x Then, we define sample-wise cost function C : Y n ? Y n 7? R+ , which is the cost function with respect to a whole sample, and we introduce the sample-wise loss as x), y] = L(GS , z) , C[GS (x

n ?

?(GS , z , k) =

k=1

n ?

x, k), y (k)] c[GS (x

k=1

Given G and a training set S, what we are most interested in is the generalization risk in structured prediction (i.e., expected average loss) [18, 7]: [L(G , z)] S R(GS) = Ez n Since the distribution D is unknown, we have to estimate R(GS) by using the empirical risk: 1 ?? 1 ? L(GS , z i) = ?(GS , z i , k) mn i=1 mn i=1 m

Re (GS) =

m

n

k=1

To state our theoretical results, we must describe several quantities and assumptions following prior work [2, 12]. We assume a simple real-valued structured prediction scheme such that the class x, k) ? D.2 Also, we assume the point-wise cost predicted on position k of x is the sign of GS (x function c? is convex and ? -smooth such that ?y1 , y2 ? D, ?y ? ? Y —c? (y1 , y ?) ? c? (y2 , y ?)— ? ? —y1 ? y2 —

(1)

x, k) ? GS i (x x, k)— while changing a single Also, we use a value ? to quantify the bound of —GS (x sample (with size n? ? n) in the training set with respect to the structured input x . This ?-admissible assumption can be formulated as ?k, x, k) ? GS i (x x, k)— ? ?——GS ? GS i ——2 ? ——x x——2 —GS (x

(2)

where ? ? R+ is a value related to the design of algorithm G. 2.1 Structure Regularization Most existing regularization techniques are for regularizing model weights/parameters (e.g., a representative regularizer is the Gaussian regularizer or so called L2 regularizer), and we call such regularization techniques as weight regularization. Definition 1 (Weight regularization) Let N? : F 7? R+ be a weight regularization function on F with regularization strength ?, the structured classification based objective function with general weight regularization is as follows: R? (GS) , Re (GS) + N? (GS)

(3)

2 In practice, many popular structured prediction models have a convex and real-valued cost function (e.g., CRFs).

3

Algorithm 1 Training with structure regularization 1: Input: model weights w , training set S, structure regularization strength ? 2: repeat 3: S? ? ? 4: for i = 1 ? m do 5: Randomly decompose z i ? S into mini-samples N? (zz i) = {zz (i,1) , . . . , z (i,?) } 6: S ? ? S ? ? N? (zz i) 7: end for 8: for i = 1 ? —S ? — do w) 9: Sample z ? uniformly at random from S ? , with gradient ?gz ? (w w) 10: w ? w ? ??gz ? (w 11: end for 12: until Convergence 13: return w While weight regularization is normalizing model weights, the proposed structure regularization method is normalizing the structural complexity of the training samples. As illustrated in Figure 1, our proposal is based on tag structure decomposition, which can be formally defined as follows: Definition 2 (Structure regularization) Let N? : F 7? F be a structure regularization function on F with regularization strength ? with 1 ? ? ? n, the structured classification based objective function

with structure regularization is as follows3 : 1 ?? 1 ??? R? (GS) , Re [GN? (S)] = L[GS ? , z (i,j)] = ?[GS ? , z (i,j) , k] mn i=1 j=1 mn i=1 j=1 m

?

m

? n/?

(4)

k=1

where N? (zz i) randomly splits z i into ? mini-samples {zz (i,1) , . . . , z (i,?) }, so that the mini-samples have a distribution on their sizes (structure complexities) with the expected value n? = n/?. Thus, we get S ? = {zz (1,1) , z(1,2) , . . . , z (1,?) , . . . , z (m,1) , z (m,2) , . . . , z (m,?) } (5) — {z } — {z } ?

?

with m? mini-samples with expected structure complexity n/?. We can denote S ? more compactly as S ? = {zz ?1 , z ?2 , . . . , z ?m? } and R? (GS) can be simplified as 1 ? 1 ?? R? (GS) , L(GS ? , z ?i) = ?[GS ? , z ?i , k] mn i=1 mn i=1 m? n/?

m?

(6)

k=1

When the structure regularization strength ? = 1, we have S ? = S and R? = Re . The structure regularization algorithm (with the stochastic gradient descent setting) is summarized in Algorithm x(1) , . . . , x (n) } represents feature vectors. Thus, it should be emphasized that 1. Recall that x = {x the decomposition of x is the decomposition of the feature vectors, not the original observations. Actually the decomposition of the feature vectors is more convenient and has no information loss ? decomposing observations needs to regenerate features and may lose some features. The structure regularization has no conflict with the weight regularization, and the structure regularization can be applied together with the weight regularization. Definition 3 (Structure & weight regularization) By combining structure regularization in Definition 2 and weight regularization in Definition 1, the structured classification based objective function is as follows: R?,? (GS) , R? (GS) + N? (GS) (7) When ? = 1, we have R?,? = Re (GS) + N? (GS) = R? . Like existing weight regularization methods, currently our structure regularization is only for the training stage. Currently we do not use structure regularization in the test stage. 3 The notation N is overloaded here. For clarity throughout, N with subscript ? refers to weight regularization function, and N with subscript ? refers to structure regularization function.

4

2.2

Reduction of Generalization Risk

In contrast to the simplicity of the algorithm, the theoretical analysis is quite technical. In this paper we only describe the major theoretical result. Detailed analysis and proofs are given in the full version of this work [14]. Theorem 4 (Generalization vs. structure regularization) Let the structured prediction

objective function of G be penalized by structure regularization with factor ? ? [1, n] and L2 weight regularization with factor ?, and the penalized function has a minimizer f : m? (1 ?) ? (8) f = argmin R?,? (g) = argmin L? (g, z ?j) + ——g——22 mn j=1 2 g?F g?F Assume the point-wise loss ?? is convex and differentiable, and is bounded by ?? (f, z , k) ? ?. x, k) is ?-admissible. Let a local feature value be bounded by v such that x(k,q) ? v for Assume f (x q ? {1, . . . , d}. Then, for any ? ? (0, 1), with probability at least 1 ? ? over the random draw of the training set S, the generalization risk R(f) is bounded by ?) ? ln ? ?1 2d? 2 ?2 v 2 n2 ((4m ? 2)d? 2 ?2 v 2 n2 R(f) ? Re (f) + + (9) +? m?? m??2 2m Since ?, ?, and v are typically small compared with other variables, especially m, (9) can be approximated as follows by ignoring small terms: (dn2 ?ln ? ?1) ? (10) R(f) ? Re (f) + O ??1.5 m) (2? ln?? ?1 in (10) The proof is given in the full version of this work [14]. We call the term O dn 1.5 ?? m as ?overfit-bound", and reducing the overfit-bound is crucial for reducing the generalization risk bound. First, (10) suggests that structure complexity n can increase the overfit-bound on a magnitude of O(n2), and applying weight regularization can reduce the overfit-bound by O(?). Importantly, applying structure regularization further (over weight regularization) can additionally reduce the overfit-bound by a magnitude of O(?1.5). Since many applications in practice are based on sparse features, using a sparse feature assumption can further improve the generalization bound. The improved generalization bounds are given in the full version of this work [14]. 2.3

Accelerating Convergence Rates in Training

We also analyze the impact on the convergence rate of online learning by applying structure regularization. Following prior work [9], our analysis is based on the stochastic gradient descent (SGD) w) be the structured prediction objective function and w ? W is the with fixed learning rate. Let g(w weight vector. Recall that the SGD update with fixed learning rate ? has a form like this: wt) w t+1 ? w t ? ??gz t (w

(11)

w t) is the stochastic estimation of the objective function based on z which is randomly where gz (w drawn from S. To state our convergence rate analysis results, we need several assumptions following w , w ? ? W, (Nemirovski et al. 2009). We assume g is strongly convex with modulus c, that is, ?w c ? w ?) ? g(w w) + (w w ? ? w)T ?g(w w) + ——w w ? w ——2 g(w (12) 2 When g is strongly convex, there is a global optimum/minimizer w ? . We also assume Lipschitz w , w ? ? W, continuous differentiability of g with the constant q, that is, ?w w ?) ? ?g(w w)—— ? q——w w ? ? w —— ——?g(w

(13)

w) has almost surely positive correlation with It is also reasonable to assume that the norm of ?gz (w the structure complexity of z ,4 which can be quantified by a bound ? ? R+ : w)——2 ? ?—zz — almost surely for ?w w?W ——?gz (w

(14)

4 Many structured prediction systems (e.g., CRFs) satisfy this assumption that the gradient based on a larger sample (i.e., n is large) is expected to have

a larger norm.

5

where —zz — denotes the structure complexity of z . Moreover, it is reasonable to assume ?c ¡ 1

(15)

because even the ordinary gradient descent methods will diverge if ?c ¿ 1. Then, we show that structure regularization can quadratically accelerate the SGD rates of convergence: Proposition 5 (Convergence rates vs. structure regularization) With the aforementioned asc???2 sumptions, let the SGD training have a learning rate defined as ? = q? 2 n2 , where ? ¿ 0 is a convergence tolerance value and ? ? (0, 1]. Let t be a integer satisfying t?

q?2 n2 log (qa0 /?) ??c2 ?2

(16)

where n and ? ? [1, n] is like before, and a0 is the initial distance which depends on the initialization w 0 ? w ? ——2 . Then, after t updates of w it of the weights w 0 and the minimizer w ? , i.e., a0 = ——w w t) ? g(w w ?)] ? ?. converges to E[g(w The proof is given in the full version of this work [14]. As we can see, using structure regularization with the strength ? can quadratically accelerate the convergence rate with a factor of ?2 .

3

Experiments

Diversified Tasks. The natural language processing tasks include (1) part-of-speech tagging, (2) biomedical named entity recognition, and (3) Chinese word segmentation. The signal processing task is (4) sensor-based human activity recognition. The tasks (1) to (3) use boolean features and the task (4) adopts real-valued features. From tasks (1) to (4), the averaged structure complexity (number of observations) n is very different, with n = 23.9, 26.5, 46.6, 67.9, respectively. The dimension of tags —Y— is also diversified among tasks, with —Y— ranging from 5 to 45. Part-of-Speech Tagging (POS-Tagging). Part-of-Speech (POS) tagging is an important and highly competitive task. We use the standard benchmark dataset in prior work [3], with 38,219 training samples and 5,462 test samples. Following prior work [19], we use features based on words and lexical patterns, with 393,741 raw features5 . The evaluation metric is per-word accuracy. Biomedical Named Entity Recognition (Bio-NER). This task is from the BioNLP-2004 shared task [19]. There are 17,484 training samples and 3,856 test samples. Following prior work [19], we use word pattern features and POS features, with 403,192 raw features in total. The evaluation metric is balanced F-score. Word Segmentation (Word-Seg). We use the MSR data provided by SIGHAN-2004 contest [4]. There are 86,918 training samples and 3,985 test samples. The features are similar to [16], with 1,985,720 raw features in total. The evaluation metric is balanced F-score. Sensor-based Human Activity Recognition (Act-Recog). This is a task based on real-valued sensor signals, with the data extracted from the Bao04 activity recognition dataset [15]. The features are similar to [15], with 1,228 raw features in total. There are 16,000 training samples and 4,000 test samples. The evaluation metric is accuracy. We choose the CRFs [6] and structured perceptrons (Perc) [3], which

are arguably the most popular probabilistic and non-probabilistic structured prediction models, respectively. The CRFs are trained using the SGD algorithm,6 and the baseline method is the traditional weight regularization scheme (WeightReg), which adopts the most representative L2 weight regularization, i.e., a Gaussian prior.7 For the structured perceptrons, the baseline WeightAvg is the popular implicit regularization technique based on parameter averaging, i.e., averaged perceptron [3]. 5

Raw features are those observation features based only on x , i.e., no combination with tag information. In theoretical analysis, following prior work we adopt the SGD with fixed learning rate, as described in Section 2.3. However, since the SGD with decaying learning rate is more commonly used in practice, in experiments we use the SGD with decaying learning rate. 7 We also tested on sparsity emphasized regularization methods, including L1 regularization and Group Lasso regularization [8]. However, we find that in most cases those sparsity emphasized regularization methods have lower accuracy than the L2 regularization. 6

6
Bio?NER: CRF
POS?Tagging: CRF
Word?Seg: CRF
Act?Recog: CRF
97.4
72.4
97.2
72.2
72 StructReg WeightReg
97.15 5
10
15
71.8 0
20
5
5
10
97.3
71.8
97.2
5
10
15
20
Mini?Sample Size (n/?)
20
0
5
10

15
20
Mini?Sample Size (n/?) Act?Recog: Perc
93.5
97.1
93
97 StructReg WeightAvg
71.2 0
15
Word?Seg: Perc
72
71.6
StructReg WeightReg
93
Mini?Sample Size (n/?)
71.4
97.1 0
93.2
92.6 97.4 0
20
F?score (%)
97.15
F?score (%)
StructReg WeightAvg
93.4
92.8
97.42
Bio?NER: Perc
POS?Tagging: Perc
Accuracy (%)
15
97.44
Mini?Sample Size (n/?)
Mini?Sample Size (n/?)
97.2
10
97.46
Accuracy (%)
97.1 0
StructReg WeightReg
97.48
Accuracy (%)
97.25
93.6
97.5 F?score (%)
StructReg WeightReg

97.3
F?score (%)
Accuracy (%)
97.35
5
10
15
StructReg WeightAvg 96.9 0
20
Mini?Sample Size (n/?)
5
10
15
Mini?Sample Size (n/?)
20
StructReg WeightAvg 92.5 0
5
10
15
20
Mini?Sample Size (n/?)

Figure 2: On the four tasks, comparing the structure regularization method (StructReg) with existing regularization methods in terms of accuracy/F-score. Row-1 shows the results on CRFs and Row-2 shows the results on structured perceptrons. Table 1: Comparing our results with the benchmark systems on corresponding tasks. POS-Tagging (Acc%) Bio-NER (F1%) Word-Seg (F1%) Benchmark system 97.33 (see [13]) 72.28 (see [19]) 97.19 (see [4]) Our results 97.36 72.43 97.50 The rich edge features [16] are employed for all methods. All methods are based on the 1st-order Markov dependency. For WeightReg, the L2 regularization strengths (i.e., ?/2 in Eq.(8)) are tuned among values 0.1, 0.5, 1, 2, 5, and are determined on the development data (POS-Tagging) or simply via 4-fold cross validation on the training set (Bio-NER, Word-Seg, and Act-Recog). With this automatic tuning for WeightReg, we set 2, 5, 1 and 5 for POS-Tagging, Bio-NER, Word-Seg, and Act-Recog tasks, respectively. 3.1

Experimental Results

The experimental results in terms of accuracy/F-score are shown in Figure 2. For the CRF model, the training is convergent, and the results on the convergence state (decided by relative objective change with the threshold value of 0.0001) are shown. For the structured perceptron model, the training is typically not convergent, and the results on the 10?th iteration are shown. For stability of the curves, the results of the structured perceptrons are averaged over 10 repeated runs. Since different samples have different size n in practice, we set ? being a function of n, so that the generated mini-samples are with fixed size n? with n? = n/?. Actually, n? is a probabilistic distribution because we adopt randomized decomposition. For example, if n? = 5.5, it means the minisamples are a mixture of the ones with the size 5 and the ones with the

size 6, and the mean of the size distribution is 5.5. In the figure, the curves are based on n? = 1.5, 2.5, 3.5, 5.5, 10.5, 15.5, 20.5. As we can see, the results are quite consistent. It demonstrates that structure regularization leads to higher accuracies/F-scores compared with the existing baselines. We also conduct significance tests based on t-test. Since the t-test for F-score based tasks (Bio-NER and Word-Seg) may be unreliable8 , we only perform t-test for the accuracy-based tasks, i.e., POS-Tagging and Act-Recog. For POS-Tagging, the significance test suggests that the superiority of StructReg over WeightReg is very statistically significant, with p ¡ 0.01. For Act-Recog, the significance tests suggest that both the StructReg vs. WeightReg difference and the StructReg vs. WeightAvg difference are extremely statis8 Indeed we can convert F-scores to accuracy scores for t-test, but in many cases this conversion is unreliable. For example, very different F-scores may correspond to similar accuracy scores.

7
4
2.5
x 10
POS?Tagging: CRF
Bio?NER: CRF
Word?Seg: CRF
5000
Act?Recog: CRF
5000
5000
StructReg WeightReg 1
3000 StructReg WeightReg 2000
4000 3500 StructReg WeightReg
3000
Train?time (sec)
1.5
4000
Train?time (sec)
Train?time (sec)
Train?time (sec)
4500 2
4000
3000 StructReg WeightReg 2000
2500 0.5 0
5
10
15
1000 0
20
5
10
15

2000 0
20
Mini?Sample Size (n/?)
Mini?Sample Size (n/?)
5
Bio?NER: Perc
POS?Tagging: Perc
15
1000 0
20
600
350 300 250
StructReg WeightAvg
200
400
StructReg WeightAvg
150 400 0
5
10
15
Mini?Sample Size (n/?)
20
100 0
20
350
Train?time (sec)
StructReg WeightAvg
15
300 Train?time (sec)
Train?time (sec)
Train?time (sec)
800
10
Act?Recog: Perc
450
400 1000
5
Mini?Sample Size (n/?)
Word?Seg: Perc
450
1200
10
Mini?Sample Size (n/?)
5
10
15

350 0
20
Mini?Sample Size (n/?)
5
10
15
20
250 StructReg WeightAvg
200 150 100 0
Mini?Sample Size (n/?)
5
10
15
20
Mini?Sample Size (n/?)

Figure 3: On the four tasks, comparing the structure regularization method (StructReg) with existing regularization methods in terms of wall-clock training time. tically significant, with p ¡ 0.0001 in both cases. The experimental results support our theoretical analysis that structure regularization can further reduce the generalization risk over existing weight regularization techniques. Our method outperforms the benchmark systems on the three important natural language processing tasks. The POS-Tagging task is a highly competitive task, with many methods proposed, and the best report (without using extra resources) until now is achieved by using a bidirectional learning model in [13],9 with the accuracy 97.33%. Our simple method achieves better accuracy compared with all of those state-of-the-art systems. Furthermore, our method achieves as good scores as the benchmark systems on the Bio-NER and Word-Seg tasks. On the Bio-NER task, [19] achieves 72.28% based on lookahead learning and [21] achieves 72.65% based on reranking. On the Word-Seg task, [4] achieves 97.19% based on maximum entropy classification and our recent work [16] achieves 97.5% based on feature-frequency-adaptive online learning. The comparisons are summarized in Table 1. Figure 3 shows experimental comparisons in terms of wall-clock training time. As we can see, the proposed method can substantially improve the training speed. The speedup is not only from the faster convergence rates, but also from the faster processing time on the structures, because it is more efficient to process the decomposed samples with simple structures.

4

Conclusions

We proposed a structure regularization framework, which decomposes training samples into minisamples with simpler structures, deriving a trained model with regularized structural complexity. Our theoretical analysis showed that this method can effectively reduce the generalization risk, and can also accelerate the convergence speed in training. The proposed method does not change the convexity of the objective function, and can be used together with any existing weight regularization methods. Note that, the proposed method and

the theoretical results can fit general structures including linear chains, trees, and graphs. Experimental results demonstrated that our method achieved better results than state-of-the-art systems on several highly-competitive tasks, and at the same time with substantially faster training speed. Acknowledgments. This work was supported in part by NSFC (No.61300063). 9 See a collection of the systems at http://aclweb.org/aclwiki/index.php?title=POS_Tagging_(State_of_the_art)

8

2 References

[1] A. Argyriou, C. A. Micchelli, M. Pontil, and Y. Ying. A spectral regularization framework for multi-task structure learning. In Proceedings of NIPS?07. MIT Press, 2007. [2] O. Bousquet and A. Elisseeff. Stability and generalization. Journal of Machine Learning Research, 2:499?526, 2002. [3] M. Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In Proceedings of EMNLP?02, pages 1?8, 2002. [4] J. Gao, G. Andrew, M. Johnson, and K. Toutanova. A comparative study of parameter estimation methods for statistical natural language processing. In Proceedings of ACL?07, pages 824?831, 2007. [5] J. Gra?a, K. Ganchev, B. Taskar, and F. Pereira. Posterior vs parameter sparsity in latent variable models. In Proceedings of NIPS?09, pages 664?672, 2009. [6] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In ICML?01, pages 282?289, 2001. [7] B. London, B. Huang, B. Taskar, and L. Getoor. Pac-bayes generalization bounds for randomized structured prediction. In NIPS Workshop on Perturbation, Optimization and Statistics, 2007. [8] A. F. T. Martins, N. A. Smith, M. A. T. Figueiredo, and P. M. Q. Aguiar. Structured sparsity in structured prediction. In Proceedings of EMNLP?11, pages 1500?1511, 2011. [9] F. Niu, B. Recht, C. Re, and S. J. Wright. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In NIPS?11, pages 693?701, 2011. [10] A. Quattoni, X. Carreras, M. Collins, and T. Darrell. An efficient projection for l1,infinity regularization. In Proceedings of ICML?09, page 108, 2009. [11] M. W. Schmidt and K. P. Murphy. Convex structure learning in log-linear models: Beyond pairwise potentials. In Proceedings of AISTATS?10, volume 9 of JMLR Proceedings, pages 709?716, 2010. [12] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Learnability and stability in the general learning setting. In Proceedings of COLT?09, 2009. [13] L. Shen, G. Satta, and A. K. Joshi. Guided learning for bidirectional sequence classification. In Proceedings of ACL?07, 2007. [14] X. Sun. Structure regularization for structured prediction: Theories and experiments. In Technical report, arXiv, 2014. [15] X. Sun, H. Kashima, and N. Ueda. Large-scale personalized human activity recognition using online multitask learning. IEEE Trans. Knowl. Data Eng., 25(11):2551?2563, 2013. [16] X. Sun, W. Li, H. Wang, and Q. Lu. Feature-frequency-adaptive on-line training for fast and accurate natural language processing. Computa-

tional Linguistics, 40(3):563?586, 2014. [17] C. A. Sutton and A. McCallum. Piecewise pseudolikelihood for efficient training of conditional random fields. In ICML?07, pages 863?870. ACM, 2007. [18] B. Taskar, C. Guestrin, and D. Koller. Max-margin markov networks. In NIPS?03, 2003. [19] Y. Tsuruoka, Y. Miyao, and J. Kazama. Learning with lookahead: Can history-based models rival globally optimized models? In Conference on Computational Natural Language Learning, 2011. [20] H. Xue, S. Chen, and Q. Yang. Structural regularized support vector machine: A framework for structural large margin classifier. IEEE Transactions on Neural Networks, 22(4):573?587, 2011. [21] K. Yoshida and J. Tsujii. Reranking for biomedical named-entity recognition. In ACL Workshop on BioNLP, page 209?lC216, 2007. [22] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society, Series B, 68:49?67, 2006.

9