Stochastic Structured Prediction under Bandit Feedback

Artem Sokolov[◇,‡] Julia Kreutzer[‡] Chris Lo[†] Stefan Riezler[‡,*]
Amazon[◇] Heidelberg University[‡] Broad Institute[†] IWR[‡]
{sokolov, kreutzer, riezler}@cl.uni-heidelberg.de

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Highlights

Stochastic approximation for structured prediction

- ✔ stochastic first-order optimization with bandit feedback on complete structures
- ✔ 1 convex & 2 non-convex objectives
- ✔ applications to machine translation & chunking
- ✔ empirically, **pairwise loss** found to be the best and fastest to converge
- ✔ numerical analysis to explain such result

Bandit Structured Prediction

Input: learning rates γ_t, loss $\mathcal{L}$ ← we evaluate 3 objectives
Initialize parameters w_0
for $t = 0, \ldots, T$ **do**
  Observe input x_t
  Sample structure $\tilde{y}_t$ from a model distribution $p_{w_t}(y|x_t)$
  Obtain feedback $\Delta(\tilde{y}_t)$
  Update $w_{t+1} = w_t - \gamma_t\, s_t$, where $\mathbb{E}[s_t] = \nabla\mathcal{L}$
Choose a solution $\hat{w}$ from the list $\{w_0, \ldots, w_T\}$

Objectives

➡ assume log-linear model $p_w(y|x) := e^{w^\top \phi(x,y)}/Z_w(x)$

➡ cross-entropy uses an unnormalized reward function $g(y)$, e.g. $1 - \Delta(y)$

	expected loss (EL)	pairwise loss (PR) ← new	cross-entropy loss (CE)
loss $\mathcal{L}$	$\mathbb{E}_{p(x)p_w(y\|x)}[\Delta(y)]$	$\mathbb{E}_{p(x)p_w(\langle y_i,y_j\rangle\|x)}[\Delta(\langle y_i, y_j\rangle)]$	$\mathbb{E}_{p(x)g(y)}[-\log p_w(y\|x)]$
distribution	$p_w(y\|x)$	$p_w(y_i\|x)p_{-w}(y_j\|x)$	$p_w(y\|x)$
update s_t	$\Delta(\tilde{y}_t)\left(\phi(x_t,\tilde{y}_t) - \mathbb{E}_{p_{w_t}}[\phi(x_t,y)]\right)$	$\Delta(\langle\tilde{y}_i,\tilde{y}_j\rangle_t)\left(\phi(x_t,\langle\tilde{y}_i,\tilde{y}_j\rangle_t) - \mathbb{E}_{p_{w_t}(\langle y_i,y_j\rangle\|x_t)}[\phi(x_t,\langle y_i,y_j\rangle)]\right)$	$\frac{g(\tilde{y}_t)}{p_{w_t}(\tilde{y}_t\|x_t)}\left(-\phi(x_t,\tilde{y}_t) + \mathbb{E}_{p_{w_t}}[\phi(x_t,y)]\right)$

Experiments

	Machine Translation	Chunking
data	FR-EN, Europarl→News	CoNLL'00, shallow parsing
task structure	SCFG hypergraph	bigram CRF
train/dev/test	38k/1k/2k	8k/1k/2k
score	BLEU	F1

Two type of experiments:

I performance and empirical convergence comparison:
- convergence criteria based on early stopping on dev set
- dev-tuned: #iterations, ℓ_2 regularization, clipping k, learning rate γ
- binary/continuous feedback for PR treated as hyperparameter

II numerical convergence analysis:
- Lipschitz constant L, variance σ^2, update norm $\|s_T\|$
- fixed learning rate γ and horizon T
- PR uses binary and continuous feedback

[Ghadimi&Lan'12]: iterations to reach $\|\nabla\mathcal{L}\|^2 < \varepsilon$ is $\mathcal{O}(\frac{L^2}{\varepsilon} + \frac{L^2\sigma^2}{\varepsilon^2})$ ⟵

Results

I **performance:**
- **SMT:** all improve over out-of-domain full-info baseline (BLEU 0.265); PR(bin) is 2-4 times faster than EL/CE
- **Chunking:** all close to full-info baseline (F1 0.935); PR(cont) is fastest (but EL has the best F1)
- why does non-convex PR converge faster?

convergence speed

	Algorithm	Iterations	Score	γ
SMT	**CE**	281k	0.271	1e-6
SMT	**EL**	370k	0.267	1e-5
SMT	**PR**(bin)	**115k**	0.273	1e-4
Chunk	**CE**	5.9M	0.891	1e-6
Chunk	**EL**	7.5M	0.923	1e-4
Chunk	**PR**(cont)	**4.7M**	0.914	1e-4

II **estimated constants:**
- **SMT:** $\|s_T\|^2$ is much smaller for PR than for EL/CE; L and σ^2 smallest for PR too;
- **Chunking:** PR's $\|s_T\|^2$, L and σ^2 are smaller than CE's but similar to EL

	Algorithm	$\|s_T\|^2$	L	σ^2
SMT	**CE**	3.04	0.54	35
SMT	**EL**	0.02	1.63	3.13e-4
SMT	**PR**(bin)	**2.88e-4**	0.08	3.79e-5
Chunk	**CE**	4.20	1.60	4.88
Chunk	**EL**	**1.21e-3**	1.16	0.01
Chunk	**PR**(cont)	5.99e-3	1.11	0.03

{ Since iteration complexity increases w.r.t. L, σ^2, smaller constants imply faster convergence for PR