Resource Constrained Structured Prediction

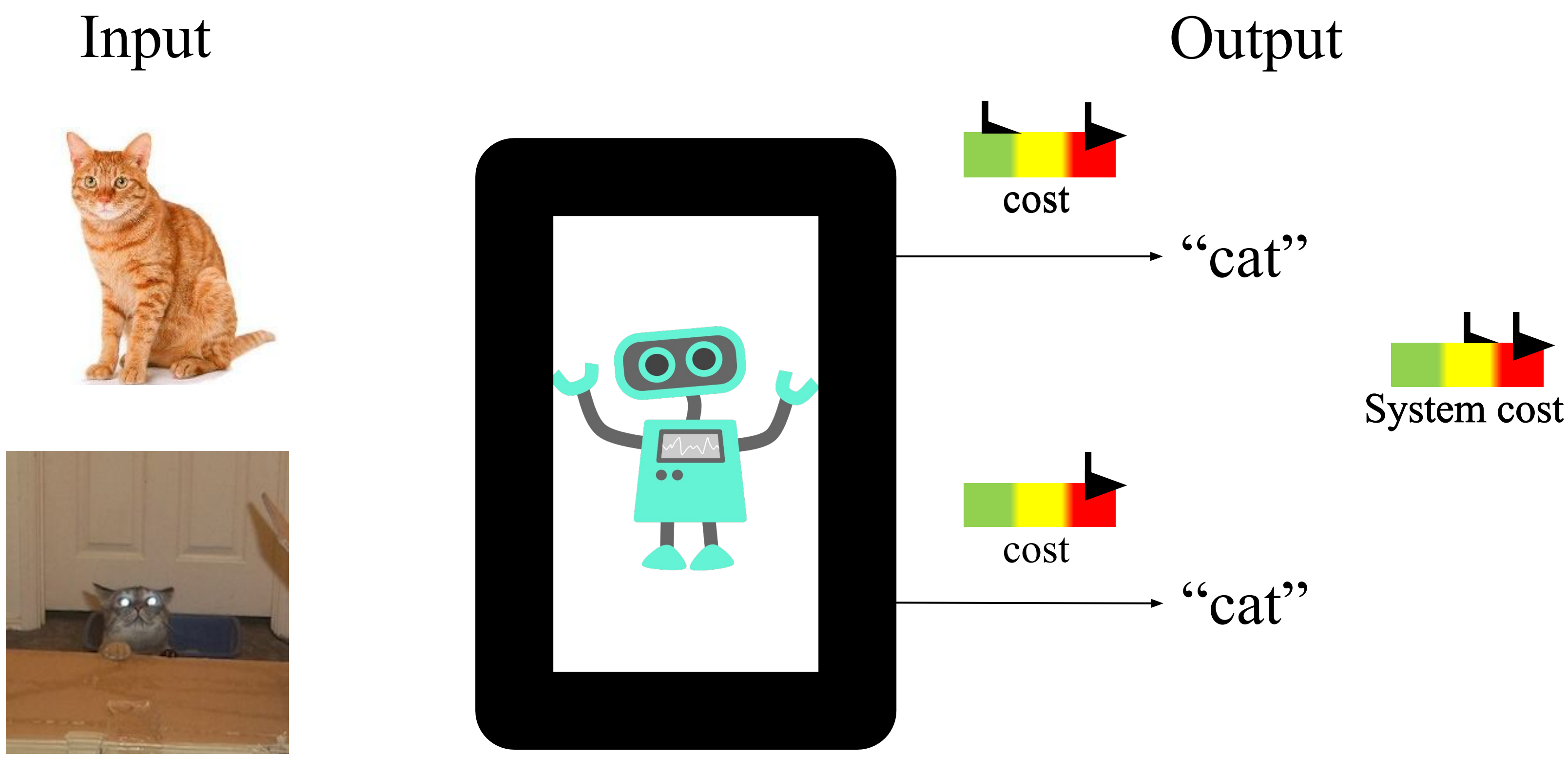
Tolga Bolukbasi, Kai-Wei Chang, Joseph Wang,
Venkatesh Saligrama

AAAI 2017

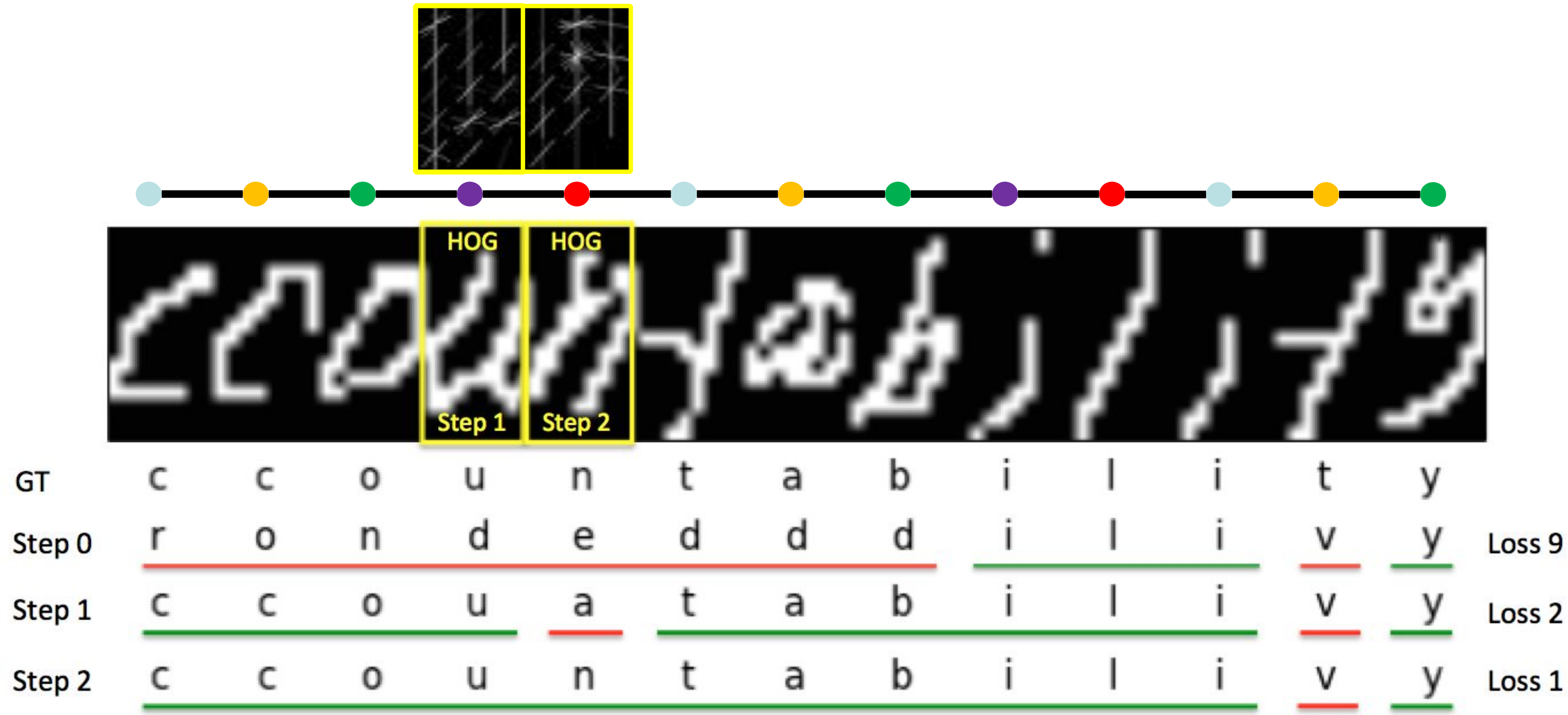
BOSTON UNIVERSITY

UNIVERSITY of VIRGINIA

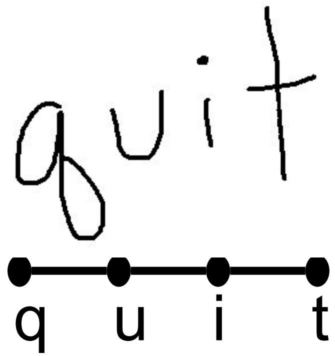
Learning with adaptive budget



Exploiting the structure for budget



Budgeted structured learning – setup



$$X = \begin{bmatrix} x_1^1 & \cdots & x_m^1 \\ \vdots & \ddots & \vdots \\ x_1^K & \cdots & x_m^K \end{bmatrix}$$

Part features

$$Y = [y_1 \quad \cdots \quad y_n]$$

Structured label

$$R(X,Y) = \underbrace{\mathcal{L}(F(X),Y)}_{\text{Prediction error}}$$

Modified Risk

$$R(X,Y,S) = \overbrace{\mathcal{L}(F(X,S),Y)}^{\text{Prediction error}} + \underbrace{\sum_{i=1}^{m} \sum_{k \in s_i} c_k}_{\text{Feature costs}}$$

Feature subset for each part

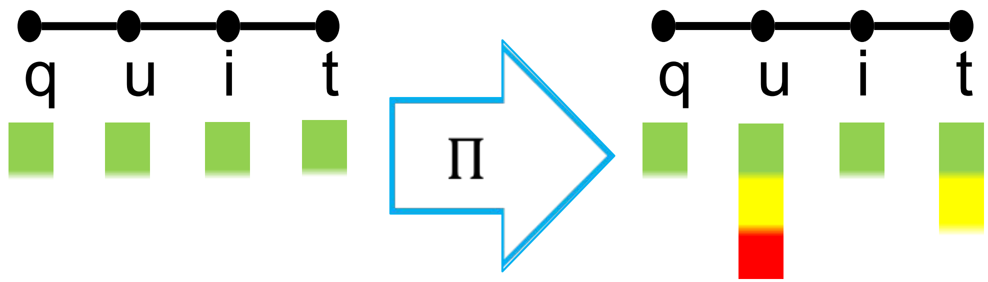
$$S = [s_1 \quad \cdots \quad s_m]$$

Budgeted structured learning – policy

Adaptive Structured Feature Selector

$$\Pi : \mathcal{X} \rightarrow \mathcal{S}$$

Maps from “initial” part features to subsets of features adaptively



Goal: find Π to minimize expected loss

$$E_{X,Y}[R(X,Y,\Pi)]$$

$$R(X,Y,\Pi) = \sum_{S \in \mathcal{S}} R(X,Y,S) 1_{\Pi(X)=S}$$

$$R(X,Y,\Pi) \leq R_{max}(X,Y) 1_{\Pi(X) \neq S^*(X,Y)}$$

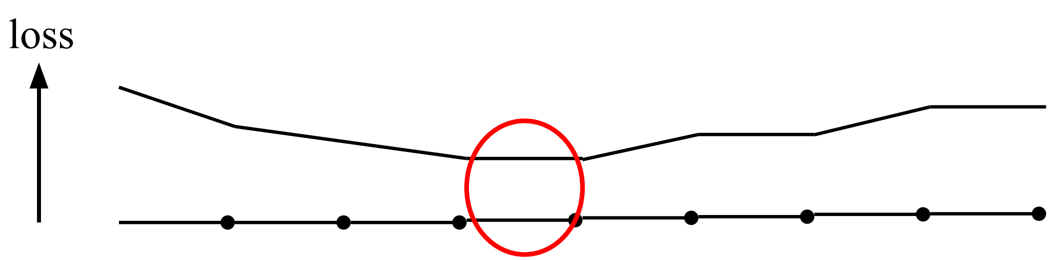
Challenges

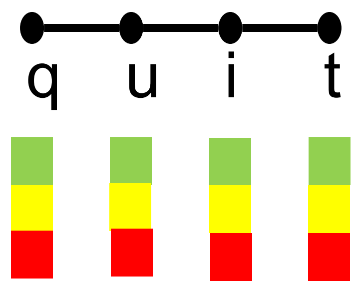
- How to find the best feature states for training?
  - Combinatorial search space
- What should be the form of policy?
  - Varying sample shape/structure
  - Combinatorial output space

Finding the best feature states

$$S^*(X,Y) = \min_{S \in \mathcal{S}} R(X,Y,S)$$

- Greedy approximation with trajectory search
  - Start with initial state
  - Do greedy search over all single feature acquisitions
  - Pick the one that minimizes immediate loss
  - Repeat until all features are acquired





1-step policy

- Upper bound the risk $R(X,Y,\Pi) = \sum_{S\in\mathcal{S}} R(X,Y,S) 1_{\Pi(X)=S}$ with the worst case one $R_{max}(X,Y) = \max_{S_1,S_2\in\mathcal{S}} \left(R(X,Y,S_1) - R(X,Y,S_2)\right)$

$$\Pi^* = \min E_{X,Y}\left[R_{max}(X,Y) 1_{\Pi(X)\neq S^*(X,Y)}\right]$$

Theorem: 1-step system is equivalent to cost-sensitive structured learning problem

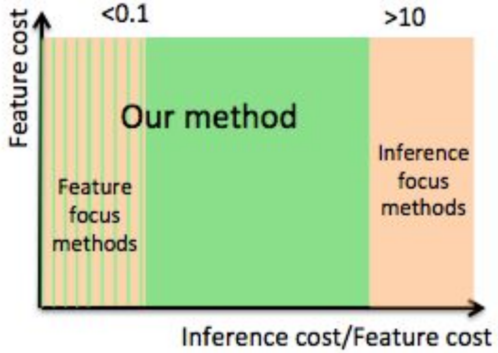
Finding $R_{max}(X,Y)$ is also combinatorial, approximate using greedy search as before

Anytime policy (sequential)

Anytime policy: Greedily find trajectory of feature subsets and train sequence of 1-step systems

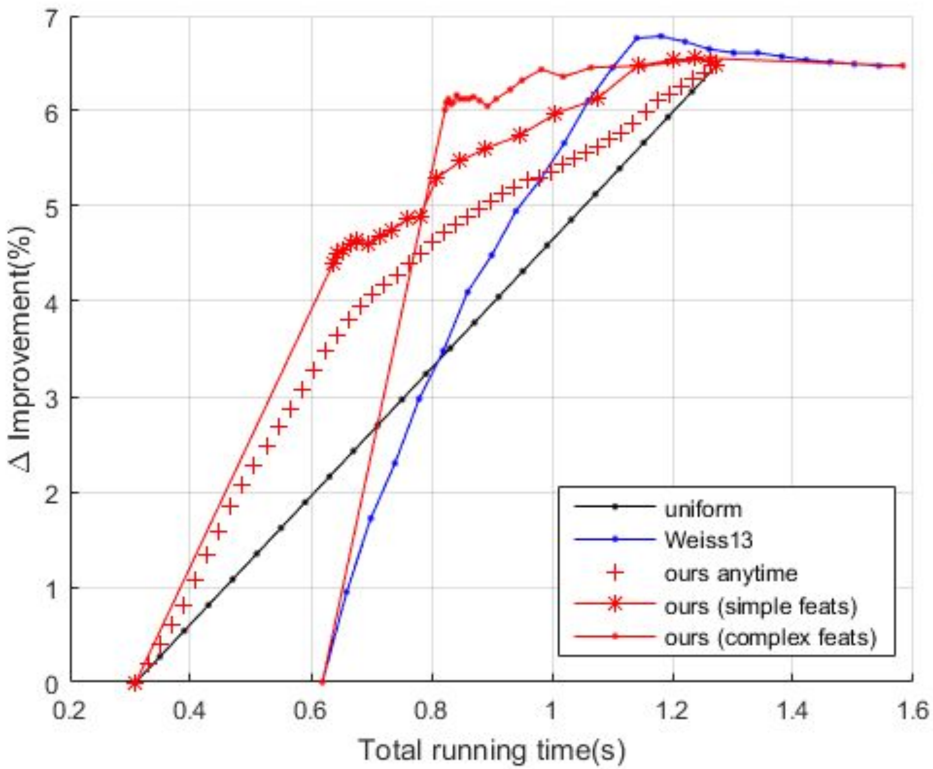
Relevant work

- Feature cost dominates
- Reduction of structured prediction to sequential multiclass prediction
  - (emma)
- Input shape/structure is fixed
  - our own eccv paper
- Inference cost dominates



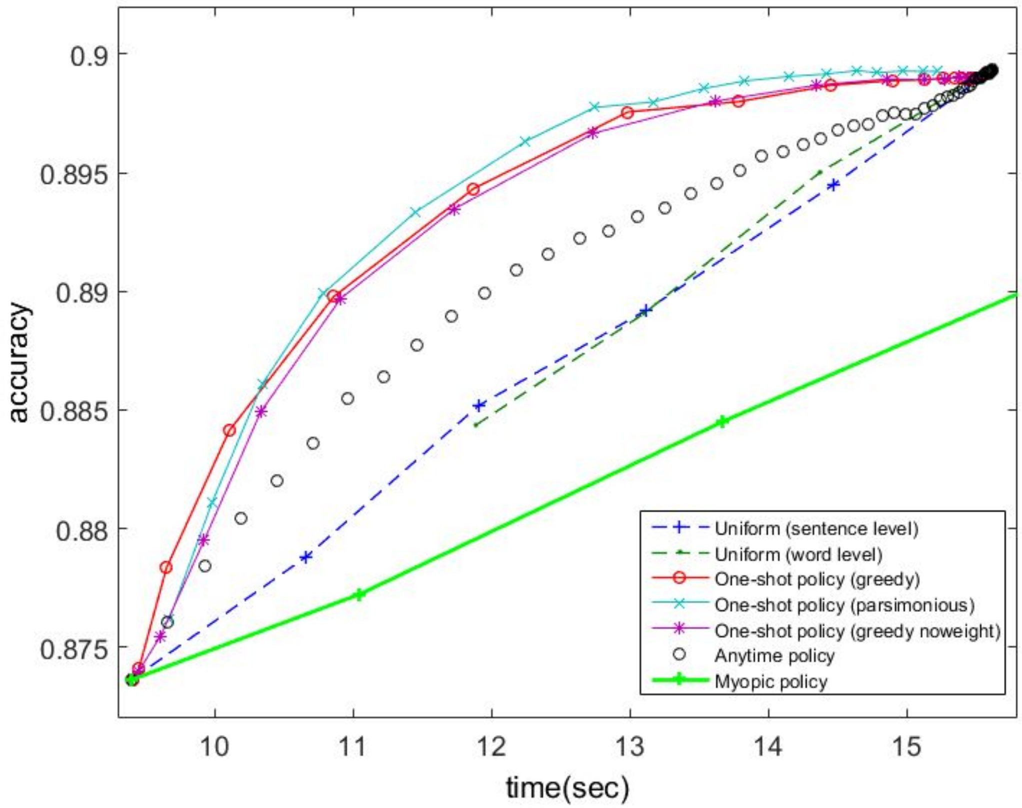
<u>Previous work</u>:
•Weiss and Taskar, 2013
•**JW** et al. 2014

Experiments – OCR



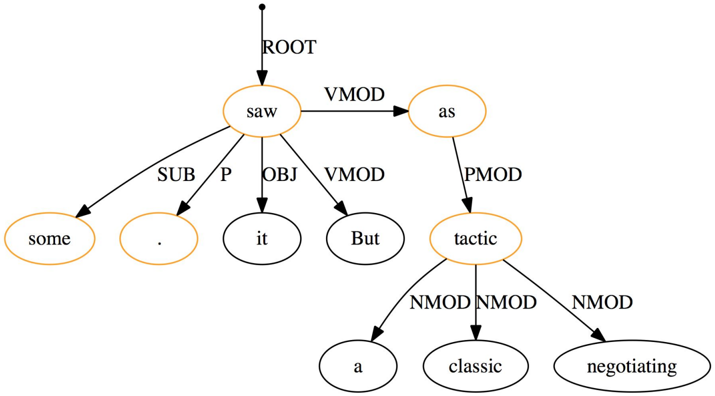
Approximately same performance, ~30% time reduction

Experiments – Dependency Parsing



Features:

- Part-of-speech tags
- Lexicons (Lemmas)
- Relationships to surrounding words



Words using relationship features

- Nan, **JW**, V. Saligrama, “Feature-Budgeted Random Forests.“ ICML 2015.
- **JW**, Bolukbasi, Trapeznikov, Saligrama, “Model Selection by Linear Programming," ECCV 2014.
- **JW**, Trapeznikov, Saligrama, “An LP for Sequential Learning Under Budgets," AISTATS 2014.

Thanks for listening!

Questions?

Features and outputs

Structured labels	Features for each part
•Words	•RGB histograms •HOG features
•Parse tree	•Part-of-speech tags •Lemmas •Word relationships