

Structured Prediction and PyStruct

Andreas Müller

April 10, 2018

Motivation

Multi-Label Classification

	Politics	Sports	Finance	Domestic	Religion
News Story1	1	0	0	1	1
News Story2	0	1	0	1	0
News Story3	0	0	1	0	0

Multi-Label Classification

	Politics	Sports	Finance	Domestic	Religion
News Story1	1	0	0	1	1
News Story2	0	1	0	1	0
News Story3	0	0	1	0	0

	Owns Car	Smokes	Married	Self-Employed	Has Kids
Customer1	1	0	1	0	1
Customer2	1	1	0	1	0
Customer3	0	1	1	0	0

Sequence Tagging



Sequence Tagging



Stroke cat.



Stroke cat.



Stroke cat.



Open trash can.



Put cat in trash can.

Sequence Tagging



Stroke cat.



Stroke cat.



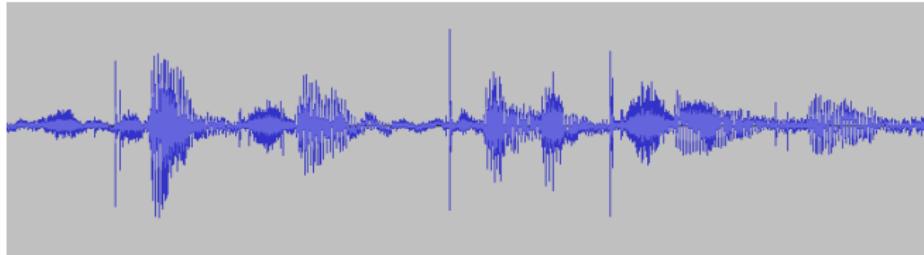
Stroke cat.



Open trash can.



Put cat in trash can.



Sequence Tagging



Stroke cat.



Stroke cat.



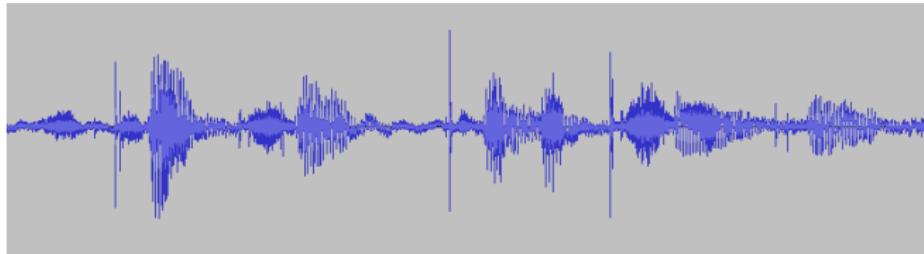
Stroke cat.



Open trash can.

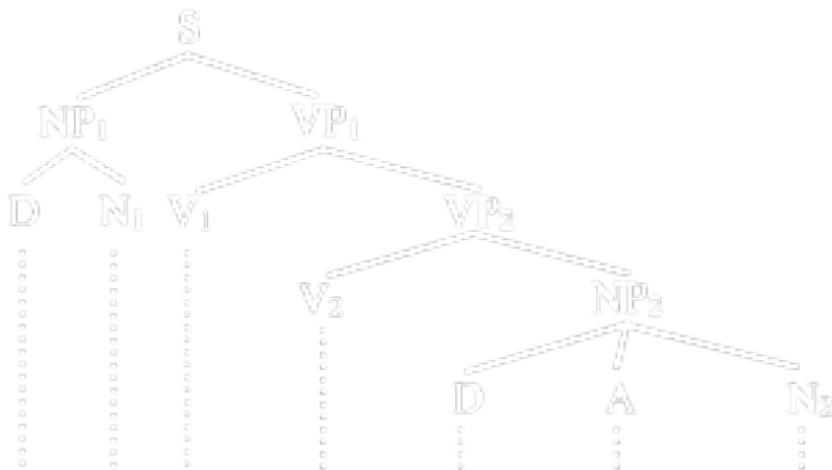


Put cat in trash can.



Struc-tured pre-dic-tion in Py-thon

Predicting Parse Trees



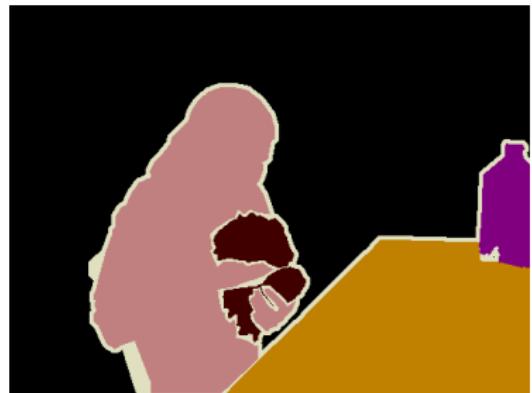
This tree is illustrating the constituency relation.

Constituency relation (PSG)

Semantic Images Segmentation



Semantic Images Segmentation



Predicting Structured Objects

$$f(x, w) := \arg \max_{y \in \mathcal{Y}} g(x, y, w)$$

Predicting Structured Objects

$$f(x, w) := \arg \max_{y \in \mathcal{Y}} g(x, y, w)$$

If you like:

$$\arg \max_{y \in \mathcal{Y}} p(y|x, w)$$

Predicting Structured Objects

$$f(x, w) := \arg \max_{y \in \mathcal{Y}} g(x, y, w)$$

If you like:

$$\arg \max_{y \in \mathcal{Y}} p(y|x, w)$$

$$f(x, w) := \arg \max_{y \in \mathcal{Y}} w^T \psi(x, y)$$

Inference and Factor Graphs

Predicting discrete vectors

$$\mathbf{y} = (y_1, y_2, \dots, y_{n_i})$$

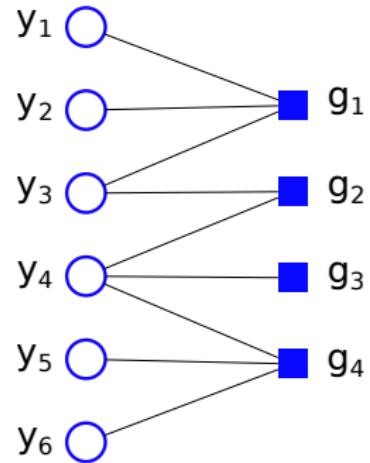
Predicting discrete vectors

$$y = (y_1, y_2, \dots, y_{n_i})$$

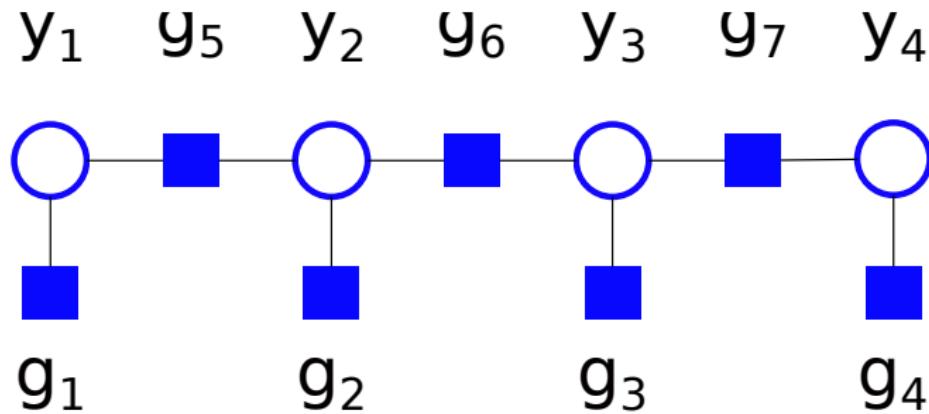
$$\begin{aligned} f(x, w) &= \arg \max_{y \in \mathcal{Y}} w^T \psi(x, y) \\ &= \arg \max_{y_1, y_2, \dots, y_{n_i}} w^T \psi(x, y) \end{aligned}$$

Factor Graphs

$$g(x, y) = g_1(x, y_1, y_2, y_3) + g_2(x, y_3, y_4) \\ + g_3(x, y_4) + g_4(x, y_4, y_5, y_6)$$

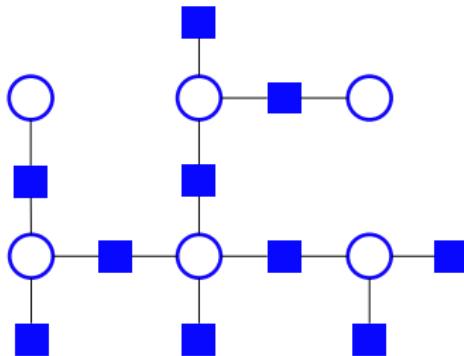


Factor Graph for HMM

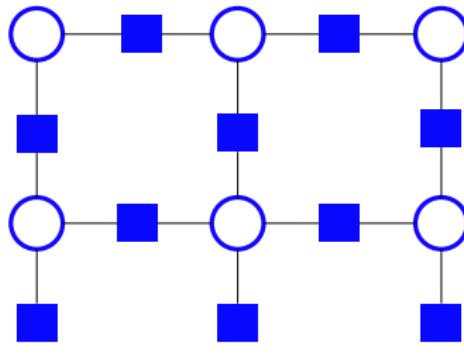


Inference

Easy



Tricky



Learning

Probabilistic Learning

$$p(y|x, w) = \frac{1}{Z} \exp(w^T \psi(x, y))$$

$$Z = \sum_{y' \in \mathcal{Y}} \exp(w^T \psi(x, y'))$$

Probabilistic Learning

$$p(y|x, w) = \frac{1}{Z} \exp(w^T \psi(x, y))$$

$$Z = \sum_{y' \in \mathcal{Y}} \exp(w^T \psi(x, y'))$$

Objective

$$\begin{aligned} & \max_w \sum_i \log(p(y^i|x^i, w)) \\ &= \max_w \sum_i w^T \psi(x^i, y^i) - \log(Z) \end{aligned}$$

Max-Margin Learning

$$\min_w \frac{1}{2} \|w\|^2 + C \sum_i \ell(x^i, y^i, w)$$

$$\ell(x^i, y^i, w) = [\max_{y \in \mathcal{Y}} \Delta(y^i, y) + w^T \psi(x^i, y) - w^T \psi(x^i, y^i)]_+.$$

PyStruct

Simple structured prediction

Estimator = Learner + Model + Inference

Simple structured prediction

Estimator = Learner + Model + Inference

- Learner: SubgradientSSVM, StructuredPerceptron, OneSlackSSVM, LatentSSVM
- Model: BinaryClf, MultiLabelClf, ChainCRF, GraphCRF, EdgeFeatureGraphCRF
- Inference: Linear Programming, QPBO (PyQPBO), Dual Decomposition (AD3), Message Passing (OpenGM), Everything (OpenGM)

Example OCR

```
from pystruct.datasets import load_letters
from pystruct.models import ChainCRF
from pystruct.learners import OneSlackSSVM

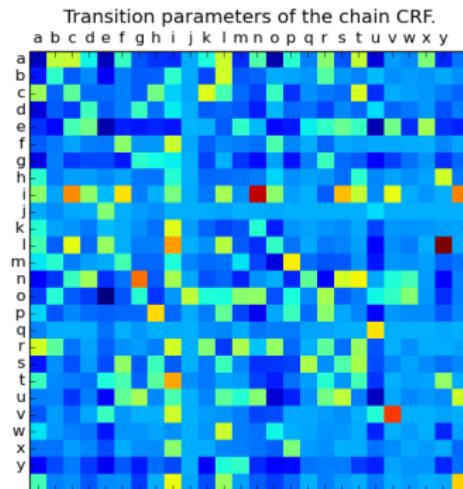
abc = "abcdefghijklmnopqrstuvwxyz"

letters = load_letters()
X, y, folds = letters['data'], letters['labels'], letters['folds']
# we convert the lists to object arrays, as that makes slicing much more
# convenient
X, y = np.array(X), np.array(y)
X_train, X_test = X[folds == 1], X[folds != 1]
y_train, y_test = y[folds == 1], y[folds != 1]

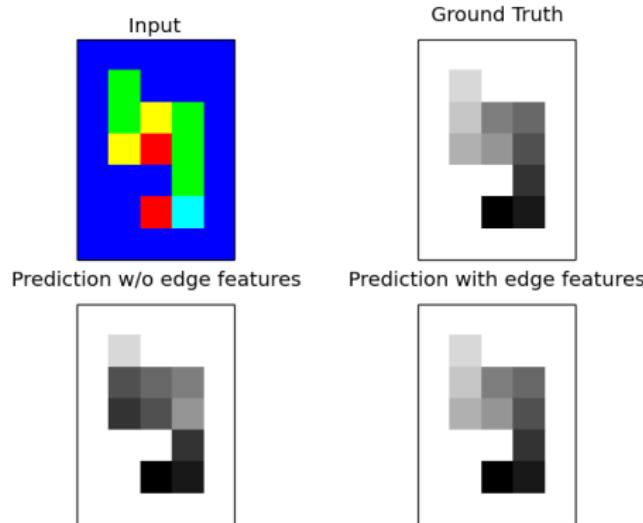
# Train linear SVM
svm = LinearSVC(dual=False, C=.1)
# flatten input
svm.fit(np.vstack(X_train), np.hstack(y_train))

# Train linear chain CRF
model = ChainCRF()
ssvm = OneSlackSSVM(model=model, C=.1, inference_cache=50, tol=0.1, verbose=3)
ssvm.fit(X_train, y_train)
```

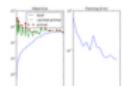
Example OCR



Example Snake



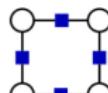
Examples



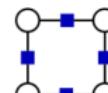
Plotting the objective and constraint caching in 1-slab SSVM



Efficient exact learning of 1-slab SSVMs



SVM as CRF



Semantic Image Segmentation on Pascal VOC



Latent Dynamics CRF



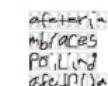
SVM objective values



Learning directed interactions on a 2d grid



Learning interactions on a 2d grid



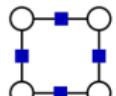
OCR Letter sequence recognition



Crammer-Singer Multi-Class SVM



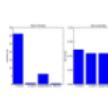
Latent SVM for odd vs. even digit classification



Mult-label classification



Latent Variable Hierarchical CRF



Binary SVM as SSVM



Comparing PyStruct and SVM-Struct