Running Ex: POS Tagging

**Given**: Data: $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$

- $x^{(i)} = \text{flies like a plant}$
- $y^{(i)} = N V D N$

- $\tilde{x}^{(i)} \in \mathcal{X} \rightarrow \text{input space}$
- $y^{(i)} \in Y^{(i)} \rightarrow \text{output space (specific to } \tilde{x}^{(i)})$

**Loss**: $L(y^{*}, \tilde{y}) : Y^{(i)} \times \tilde{Y}^{(i)} \rightarrow \mathbb{R}$
- Possible outputs for given input

**Hypothesis Space**: $H \ni \theta \rightarrow \mathbb{R}$

**Goal**: Minimize Empirical Risk

\[
\hat{h} = \arg\min_{h \in H} \frac{1}{N} \sum_{i=1}^{N} L(y^{(i)}, h(x^{(i)}))
\]

\[
\hat{\theta} = \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} L(y^{(i)}, h_{\theta}(\tilde{x}^{(i)}))
\]

**Structured Prediction as Search**

$X_{1:T} = \text{flies like a plant}$

$Y_{1:T} = \text{Induced search space for } \hat{x}$

- Hamming Loss

$X_{1:T} = \text{flies like a plant}$

$Y_{1:T} = \text{Induced search space for } \hat{x}$

- Hamming Loss

**Def**: Trajectory is a path through the search space which is a sequence of output labels $\tilde{y}_{1:T}$
Def: a policy maps from (observation sequence $x_{1:T}$ and a partial trajectory $y_{1:t}$) a state $s_t$ to an action $a_t$ (i.e. next label $y_{t+1}$)

Def: an optimal policy is some $\pi^*$ induced by a labeled training example $(x, y)$ s.t.

\[
\pi^*(s_t) = \underset{s' \in N_{st}}{\arg \min} \ c^*(s')
\]

Goal of "Learning to Search": Find a good model policy

\[
\hat{y}_{1:T} \sim \pi_{\theta,x}(\text{start})
\]

\[
\hat{\pi} = \underset{\pi}{\arg \min} \ \mathbb{E}_{i=1}^{N} \ \text{loss}(y(i), \hat{y}_{1:t} \sim \pi_{\theta,x}(\text{start}))
\]

How? Define a scoring function over states $f_{\theta,x} : V_x \rightarrow \mathbb{R}$

Hopefully $f_{\theta,x} \approx c^*_{x,y}$, but we don't know $y$ at test

Def: a greedy policy for scoring function $f_{\theta,x}$ is
Def: a greedy policy for scoring function $f_{\theta}$ is

$$T_{\theta}(s_t) = \arg\max_{s' \in \mathcal{N}(s_t)} f_{\theta}(s')$$

Ex: scoring functions (1) Linear Model (2) NN (3) RNN-LM (4) seq2seq

e.g., probabilities, or log-probabilities