Spectral Learning of Sequence Taggers over Continuous Sequences

Adrià Recasens and Ariadna Quattoni
Universitat Politècnica de Catalunya

— Tagging Continuous Sequences —

Examples: Gesture Recognition, Robot Navigation.
Setting: We are given aligned sequences \( < x_1 \ldots x_i, y_1 \ldots y_i > \).
Goal: Learn a model of \( P(x,y) \) and use it to make predictions, i.e. compute \( \arg\max_y P(x,y) \).

— Spectral Background —

HMMs [HKZ09]
- m states \( S \in \{1, \ldots , m\} \)
- k symbols \( x_i \in \{\sigma_1, \ldots , \sigma_k\} \)
- Forward-backward equations with \( A \in \mathbb{R}^{m\times m} \):
  \[ P(x) = \alpha_1 A_{x_1} \cdots A_{x_n} I \]
- Observable statistics:
  \[ H(i,j) = P(x_{i-1} = \sigma_1, x_i = \sigma_j) \]
  \[ H_t(i,j) = P(x_{i-1} = \sigma_1, x_i = \sigma_j, x_{i+1} = \sigma_j) \]
- Algorithm: Compute SVD \( H = UDV^T \) and take top m right singular vectors \( V_m A = (HV_m)^+H V_m \)

Finite State Taggers (FST)
- Input alphabet \( \Lambda \) and output alphabet \( \Sigma \)
- Operators \( A^t \in \mathbb{R}^{\Lambda \times \Lambda} \) depend on input and output.
- \( P(x,y) = \alpha(x) A^1(x) \cdots A^n(x) \alpha(y) \)
- Algorithm: Balle et al, [ECML 2011].

— Continuous Sequence Taggers (CFST) —

A CFST over \( (\Phi(X) \times \Sigma)^* \) with m states is a tuple:
- \( A = (\Phi, \alpha, \alpha_0, \Omega^0) \)
- \( \Phi \) is a set of k feature functions: \( \phi_i : X \rightarrow \mathbb{R} \)
- \( \Omega^0 \in \mathbb{R}^{\Sigma \times \Lambda} \) are the \( k \times \Sigma \) operators and
  \( A(\Phi(x)), y_i) = \sum_{i=1}^k \phi_i(x) \Omega^0 \)
- The function \( f_A \) realised by the CFST is defined by:
  \[ f_A(x,y) = \alpha(x) A(\Phi(x), y_1) \cdots A(\Phi(x), y_i) \alpha(y) \]

— Example: Transitions as Mixture Models —
- \( P(x,y) = \sum_h \pi(h) \prod_{i=1}^n \pi(h_{i+1} | x_i, y_i | h_i) \)
- \( P(h_{i+1} | x_i, y_i | h_i) = \sum_z \pi(z | h_{i+1}, y_i | h_i) P(z | h_i, x_i) \)
- \( \phi_i(x) = P(z = i, x) \cdots P(z = k, x) \)
- \( \Omega_\Sigma = \prod_i \pi(h_{i+1} | h_i) \)

— The Algorithm —

Algorithm LearnCWFST(\( \mathbb{X}, \Phi, \Sigma, S, m \))
1. For every pair of sequences \( (x,y) \in S \) and every index \( 1 \leq t < |x| \) compute \( \phi_i(x) = (\phi_i(x_1), \ldots , \phi_i(x_t)) \)
2. Use S to estimate matrix statistics \( H_1 \in \mathbb{R}^{|S|}, H_2 \in \mathbb{R}^{|S| \times |S|}, H_3 \in \mathbb{R}^{|S| \times |S|} \), and covariance matrix \( C \in \mathbb{R}^{|S| \times |S|} \)
3. Compute the m rank compact SVD of \( H_1 = U A V^T \)
4. Compute the inverse of \( C^T = (H_1 V)^T H_1 V \)
5. Compute the start and ending parameters of the CWFST as:
   \( a_1 = H_1 V \alpha_0 = (H_2 V)^T \alpha_1 \)
6. Compute the transition matrices \( O^t \)

— Experimental Results —

Task Robot Navigation:
- Input: Sequence of Sensor Readings.
- Output: Sequence of Optimal Actions.

Features:
- Select \( k_1 \) points in \( \mathbb{R}^k \) (e.g. via kmeans)
- Define \( \phi_i(x) = \exp \frac{x_i}{d} \) for some distance function \( D \).

Inference:
- Max marginals.

Compare:
- FST spectral learning (discretized inputs).
- Different Feature Functions.

— Abstract —

We generlizate the class of FSTs over discrete input-output sequences to a class where transitions are linear combinations of elementary transitions.

Observable Statistics:
- \( H_1(i) = E[\phi(x)] \rightarrow \) Input unigram expectations.
- \( H_2(i,j) = E[\phi(x_1)\phi(x_{i+1})] \rightarrow \) Input bigram expectations.
- \( H_3(i,j) = E[y_1 y_{i+1}] \phi(x_1)\phi(x_{i+1}) \rightarrow \) Input trigram expectations conditioned on \( y_i \).
- \( C(i,j) = E[\phi(x_i)\phi(x_j)] \rightarrow \) Covariance.

— Duality: CFST and factorizations of \( H_2 —

Theorem: Minimal CFST \( A \iff \) Rank factorization of \( H_2 \).
Remarks \( \Rightarrow \):
- \( \text{Hypothesis: minimal CFST} \)
- \( H_2 = \text{FB} \), a rank factorization.
- \( A = (\Phi, \alpha, \alpha_0, \Omega^0) \) can be defined as:
  \( \alpha = F H \alpha_0 \alpha_1 = H \alpha B^\top \)
  \( Q^1 = F^\top H B = [Q^1(i,1), \ldots , Q^1(i,n)] = C^{-1}[Q(i,1), \ldots , Q(i,n)] \)

— Continuous Sequence Taggers —

Accuracy

\begin{array}{|c|c|c|c|c|c|c|}
\hline
\text{Accuracy} & 45 & 50 & 55 & 60 & 65 & 70 & 75 & 80 \\
\hline
\text{FST Euclidean} & & & & & & & & \\
\text{FST Cosine} & & & & & & & & \\
\text{FST Correlation} & & & & & & & & \\
\text{CWFST Euclidean} & & & & & & & & \\
\text{CWFST Cosine} & & & & & & & & \\
\text{CWFST Correlation} & & & & & & & & \\
\hline
\end{array}

Finding 

Hypothesis: minimal CFST.

We derive a spectral learning algorithm for this model that is both simple and fast.