Linear Time Methods for Propagating Beliefs

Min Convolution, Distance Transforms and Box Sums

Daniel Huttenlocher
Computer Science Department
December, 2004
Problem Formulation

- Find good assignment of labels $x_i$ to sites $i$
  - Set $\mathcal{L}$ of $k$ labels
  - Set $S$ of $n$ sites
  - Neighborhood system $\mathcal{N} \subseteq S \times S$ between sites

- Undirected graphical model
  - Graph $\mathcal{G} = (S, \mathcal{N})$
  - Hidden Markov Model (HMM), chain
  - Markov Random Field (MRF), arbitrary graph
  - Consider first order models
    - Maximal cliques in $\mathcal{G}$ of size 2
Problems We Consider

- Labels $x = (x_1, \ldots, x_n)$, observations $(o_1, \ldots, o_n)$
- Posterior distribution $P(x|o)$ factors
  $P(x|o) \propto \prod_{i \in S} \Psi_i(x_i) \prod_{(i, j) \in N} \Psi_{ij}(x_i, x_j)$
- Sum over labelings
  $\sum_x \left( \prod_{i \in S} \Psi_i(x_i) \prod_{(i, j) \in N} \Psi_{ij}(x_i, x_j) \right)$
- Min cost labeling
  $\min_x (\sum_{i \in S} \Psi'_i(x_i) + \sum_{(i, j) \in N} \Psi'_{ij}(x_i, x_j))$
  - Where $\Psi'_i = -\ln(\Psi_i)$ and $\Psi'_{ij} = -\ln(\Psi_{ij})$
Computational Limitation

- Not feasible to directly compute clique potentials when large label set
  - Computation of $\Psi_{ij}(x_i, x_j)$ requires $O(k^2)$ time
  - Issue both for exact HMM methods and heuristic MRF methods
- Restricts applicability of combinatorial optimization techniques
  - Use variational or other approaches
- However, often can do better
  - Problems where pairwise potential based on differences between labels $\Psi_{ij}(x_i, x_j) = \rho_{ij}(x_i - x_j)$
Applications

- Pairwise potentials based on difference between labels
  - Low-level computer vision problems such as stereo, and image restoration
    - Labels are disparities or true intensities
  - Event sequences such as web downloads
    - Labels are time varying probabilities
Fast Algorithms

- Summing posterior (sum product)
  - Express as a convolution
  - $O(k \log k)$ algorithm using the FFT
  - Better linear-time approximation algorithms for Gaussian models

- Minimizing negative log probability cost function (corresponds to max product)
  - Express as a min convolution
  - Linear time algorithms for common models using distance transforms and lower envelopes
Message Passing Formulation

- For concreteness consider local message update algorithms
  - Techniques apply equally well to recurrence formulations (e.g., Viterbi)
- Iterative local update schemes
  - Every site in parallel computes local estimates
    - Based on $\Psi$ and neighboring estimates from previous iteration
    - Exact (correct) for graphs without loops
    - Also applied as heuristic to graphs with cycles (loopy belief propagation)
Message Passing Updates

- At each step $j$ sends neighbor $i$ a message
  - Node $j$’s “view” of $i$’s labels

- Sum product
  \[
  m_{j 	o i}(x_i) = \sum_{x_j} (\Psi_j(x_j) \rho_{ji}(x_j-x_i)) \prod_{k \in \mathcal{N}(j) \setminus i} m_{k \to j}(x_j)
  \]

- Max product (negative log)
  \[
  m'_{j \to i}(x_i) = \min_{x_j} (\Psi'_j(x_j) + \rho'_{ji}(x_j-x_i) + \sum_{k \in \mathcal{N}(j) \setminus i} m'_{k \to j}(x_j))
  \]
Sum Product Message Passing

- Can write message update as convolution
  \[ m_{j \rightarrow i}(x_i) = \sum_{x_j}(\rho_{ji}(x_j - x_i) \cdot h(x_j)) \]
  \[ = \rho_{ji} \ast h \]
  - Where \( h(x_j) = \Psi_j(x_j) \prod_{k \in \mathcal{N}(j) \setminus i} m_{k \rightarrow j}(x_j) \)

- Thus FFT can be used to compute in \( O(k \log k) \) time for \( k \) values
  - Can be somewhat slow in practice

- For \( \rho_{ji} \) a (mixture of) Gaussian(s) do faster
Fast Gaussian Convolution

- A box filter has value 1 in some range
  \[ b_w(x) = \begin{cases} 1 & \text{if } 0 \leq x \leq w \\ 0 & \text{otherwise} \end{cases} \]

- A Gaussian can be approximated by repeated convolutions with a box filter
  - Application of central limit theorem, convolving pdf’s tends to Gaussian
  - In practice, 4 convolutions [Wells, PAMI 86]
    \[ b_{w_1}(x) \ast b_{w_2}(x) \ast b_{w_3}(x) \ast b_{w_4}(x) \approx G_\sigma(x) \]
  - Choose widths \( w_i \) such that \( \sum_i (w_i^2 - 1)/12 \approx \sigma^2 \)
Fast Convolution Using Box Sum

- Thus can approximate \( G_\sigma(x) \ast h(x) \) by cascade of box filters
  \[
  b_{w_1}(x) \ast (b_{w_2}(x) \ast (b_{w_3}(x) \ast (b_{w_4}(x) \ast h(x))))
  \]
- Compute each \( b_w(x) \ast f(x) \) in time independent of box width \( w \) – sliding sum
  - Each successive shift of \( b_w(x) \) w.r.t. \( f(x) \) requires just one addition and one subtraction
- Overall computation just 4 add/sub per label, \( O(k) \) with very low constant
Fast Sum Product Methods

- Efficient computation without assuming parametric form of distributions
  - \( O(k \log k) \) message updates for arbitrary discrete distributions over \( k \) labels
    - Likelihood, prior and messages
    - Requires prior to be based on differences between labels rather than their identities
- For (mixture of) Gaussian clique potential linear time method that in practice is both fast and simple to implement
  - Box sum technique
Max Product Message Passing

- Can write message update as
  \[ m'_{j \rightarrow i}(x_i) = \min_{x_j} (\rho'_{ji}(x_j-x_i) + h'(x_j)) \]
  - Where \( h'(x_j) = \Psi'_j(x_j) \sum_{k \in \mathcal{N}(j) \setminus i} m'_{k \rightarrow j}(x_j) \)
  - Formulation using minimization of costs, proportional to negative log probabilities

- Convolution-like operation over min,+ rather than \( \sum, \times \) [FH00,FHK03]
  - No general fast algorithm like FFT
  - Certain important special cases in linear time
Commonly Used Pairwise Costs

- Potts model \( \rho'(x) = \begin{cases} 0 \text{ if } x=0 \\ d \text{ otherwise} \end{cases} \)
- Linear model \( \rho'(x) = c|x| \)
- Quadratic model \( \rho'(x) = cx^2 \)
- Truncated models
  - Truncated linear \( \rho'(x)=\min(d,c|x|) \)
  - Truncated quadratic \( \rho'(x)=\min(d,cx^2) \)
- Min convolution can be computed in linear time for any of these cost functions
Potts Model

- Substituting in to min convolution
  \[ m'_{j \to i}(x_i) = \min_{x_j}(\rho'_{ji}(x_j - x_i) + h'(x_j)) \]
  can be written as
  \[ m'_{j \to i}(x_i) = \min(h'(x_i), \min_{x_j} h'(x_j) + d) \]
- No need to compare pairs \( x_i, x_j \)
  - Compute min over \( x_j \) once, then compare result with each \( x_i \)
- \( O(k) \) time for \( k \) labels
  - No special algorithm, just rewrite expression to make alternative computation clear
Linear Model

- Substituting in to min convolution yields
  \[ m'_{j \rightarrow i}(x_i) = \min_{x_j} (c|x_j-x_i| + h'(x_j)) \]

- Similar form to the L_1 distance transform
  \[ \min_{x_j} (|x_j-x_i| + 1(x_j)) \]
  - Where \( 1(x) = \begin{cases} 0 & \text{when } x \in P \\ \infty & \text{otherwise} \end{cases} \)
    is an indicator function for membership in \( P \)

- Distance transform measures L_1 distance to nearest point of \( P \)
  - Can think of computation as lower envelope of cones, one for each element of \( P \)
Using the $L_1$ Distance Transform

- Linear time algorithm
  - Traditionally used for indicator functions, but applies to any sampled function

- Forward pass
  - For $x_j$ from 1 to $k-1$
    $$m(x_j) \leftarrow \min(m(x_j), m(x_j-1)+c)$$

- Backward pass
  - For $x_j$ from $k-2$ to 0
    $$m(x_j) \leftarrow \min(m(x_j), m(x_j+1)+c)$$

- Example, $c=1$
  - $(3,1,4,2)$ becomes $(3,1,2,2)$ then $(2,1,2,2)$
Quadratic Model

- Substituting in to min convolution yields
  \[ m'_{j \rightarrow i}(x_i) = \min_{x_j} (c(x_j-x_i)^2 + h'(x_j)) \]
- Again similar form to distance transform
  - However algorithms for L_2 (Euclidean) distance do not directly apply as did in L_1 case
- Compute lower envelope of parabolas
  - Each value of \( x_j \) defines a quadratic constraint, parabola rooted at \((x_j, h(x_j))\)
  - Comp. Geom. O(klogk) but here parabolas are ordered
Lower Envelope of Parabolas

- Quadratics ordered $x_1 < x_2 < \ldots < x_n$
- At step $j$ consider adding $j$-th one to LE
  - Maintain two ordered lists
    - Quadratics currently visible on LE
    - Intersections currently visible on LE
  - Compute intersection of $j$-th quadratic with rightmost visible on LE
    - If right of rightmost intersection add quadratic and intersection
    - If not, this quadratic hides at least rightmost quadratic, remove and try again
Running Time of Lower Envelope

- Consider adding each quadratic just once
  - Intersection and comparison constant time
  - Adding to lists constant time
  - Removing from lists constant time
    - But then need to try again

- Simple amortized analysis
  - Total number of removals $O(k)$
    - Each quadratic, once removed, never considered for removal again

- Thus overall running time $O(k)$
Overall Algorithm (1D)

```c
static float *dt(float *f, int n) {
    float *d = new float[n], *z = new float[n];
    int *v = new int[n], k = 0;
    v[0] = 0; z[0] = -INF; z[1] = +INF;
    for (int q = 1; q <= n-1; q++) {
        float s = ((f[q]+c*square(q)) (f[v[k]]+c*square(v[k])))
            /(2*c*q-2*c*v[k]);
        while (s <= z[k]) {
            k--; s = ((f[q]+c*square(q))-(f[v[k]]+c*square(v[k])))
                /(2*c*q-2*c*v[k]);
        }
        k++; v[k] = q; z[k] = s; z[k+1] = +INF; }
    for (int q = 0; q <= n-1; q++) {
        while (z[k+1] < q)
            k++;
        d[q] = c*square(q-v[k]) + f[v[k]];
    }
    return d;
}
```
Combined Models

- **Truncated models**
  - Compute un-truncated message $m'$
  - Truncate using Potts-like computation on $m'$ and original function $h'$
    \[ \min(m'(x_i), \min_{x_j} h'(x_j) + d) \]

- **More general combinations**
  - Min of any constant number of linear and quadratic functions, with or without truncation
    - E.g., multiple “segments”
Illustrative Results

- Image restoration using MRF formulation with truncated quadratic clique potentials
  - Simply not practical with conventional techniques, message updates $256^2$
- Fast quadratic min convolution technique makes feasible
  - A multi-grid technique can speed up further
- Powerful formulation largely abandoned for such problems
Illustrative Results

- Pose detection and object recognition
  - Sites are parts of an articulated object such as limbs of a person
  - Labels are locations of each part in the image
    - Millions of labels, conventional quadratic time methods do not apply
  - Compatibilities are spring-like
Summary

- Linear time methods for propagating beliefs
  - Combinatorial approach
  - Applies to problems with discrete label space where potential function based on differences between pairs of labels
- Exact methods, not heuristic pruning or variational techniques
  - Except linear time Gaussian convolution which has small fixed approximation error
- Fast in practice, simple to implement
Readings