Learning Message-Passing Inference Machines for Structured Prediction

The Robotics Institute, Carnegie Mellon University
Stephane Ross, Daniel Munoz, Martial Hebert & J. Andrew Bagnell

Overview: Structured Prediction for Scene Labeling
Predict a class label at every site (pixel/superpixel/segment) in a scene:

- 3D Point Cloud Classification [2]
- Surface Layout Estimation [6]

Naive Approach: Independent Predictions

- Features → Predictor

Typical Approach: Graphical Models

- Features → Predictor

Model

- Exploits context (models spatial relationships between neighbors)
- Intractable without approximations (e.g. Tree model/Loopy BP/Graph-cut with limited class of potentials)
- Optimization with approximations can lead to poor performance [5]

Pairwise CRF Model
- Models the conditional joint distribution over labelings
- \( P(Y | X) \propto \prod_{i,j \in E} \phi(Y_i, Y_j) \prod_{i \in X} \psi(Y_i) \)
- Learn node \( \phi \) and edge potentials \( \psi \) from training data.

Loopy Belief Propagation (BP)
- Approximate inference procedure to find most likely labeling.
- Iteratively visits all nodes & edges in the graph to update neighbors’ messages until convergence:

\[
P(Y_i = y_i | x) = \phi(y_i, x_i) \prod_{N(i)} m_{ij}(y_i | y_j)
\]

\[
m_{ij}(y_i | y_j) = \sum_{y_i'} \phi(y_i', x_i) \prod_{N(i)} m_{i'j}(y_i')
\]

General Learning Procedures for Inference Machines

- Learning Synchronous Message-Passing Inference
  - Forward Training: Learn a separate predictor for each inference pass:
    - Predictions pass through graph:
    - Learning synchronous message-passing algorithm (e.g. Mean-Field, BP):
      - Given any message-passing algorithm (e.g. Mean-Field, BP):
        - Keep the same algorithm structure.
        - Message updates are replaced by a learned predictor \( h \)’s outputs:
        - e.g., logistic regressor
        - \( h \) outputs messages (a distribution over labels) to send to neighbors, given input features and neighbors’ messages.
        - \( h \) minimizes the loss of the inference’s output prediction.
  - Learning Asynchronous Message-Passing Inference
    - For each message pass:
      - Learn one predictor for all inference passes:
      - Dataset Aggregation (DAgger) [1]

Proposed Approach: Inference Machines

Inference procedure treated as a black box function to optimize, instead of learning a graphical model ([3,4] are special cases of our approach):

- Iterate many times over graph:
  - Features → Predictor
  - Features
  - Neighbors’ Predictions
- Exploits context
- Exploits discriminative power of classifiers
- Strong theoretical guarantees
- Accuracy/speed trade-off via more/less complex predictors

Theoretical Guarantees
- Forward Training:
  - If the predictors have \( \varepsilon \) avg. loss on the supervised learning task, then the sequence will have \( \varepsilon \) avg. loss during inference.
  - DAgger:
    - In \( N \) iterations, if we achieve \( \varepsilon \) avg. loss on the aggregate dataset, we are guaranteed to find a predictor \( h \) that has avg. loss of \( \varepsilon + O(1/N) \) during inference.
    - For both: Performance at inference is guaranteed to be at least as good as naive independent predictions.

Experimental Results

- Surface Layout Estimation [6]:
  - Label each superpixel into 7 classes
  - Adjacent pairwise structure + segments
  - Logistic Regressor as base predictor

- Point Cloud Classification [2]:
  - Label each 3D point into 5 classes
  - 5 NN pairwise structure + segments
  - Logistic Regressor as base predictor

- Learning Procedures:
  - Supervised Learning
  - Aggregate Dataset
  - Inference on Training Scenes
  - Initial predictor
  - New data
  - New predictor

- Prediction for C

- Features

Ground Truth

Inference on Training Scenes

New data

New predictor

Inference on Test Scenes

New predictor

[1][S. Ross, G. Gordon & J. A. Bagnell. A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. AISTATS 2011.]