Multiple Instance Learning with Manifold Bags

Boris Babenko¹  Nakul Verma¹  Piotr Dollar²  Serge Belongie³

¹ Computer Science and Engineering
University of California, San Diego
{nnaverma, sjb}@cs.ucsd.edu

² Computer Science
California Institute of Technology
{pdollar}@caltech.edu

ABSTRACT

• Multiple Instance Learning is a relaxed form of supervised learning
  » Learner receives labeled bags rather than labeled instances
  » Reduces burden of collecting labeled data

• Existing analysis assumes bags have a finite size
  » For many applications, bags are modeled better as manifolds in feature space; thus existing analysis is not appropriate

• In this setting we show:
  » geometric structure of manifold bags affects PAC learnability
  » A MIL algorithm that learns from finite sized bags can be trained with manifold bags
  » a simple heuristic algorithm for reducing memory requirements

DEFINITION

MIL: relaxed form of supervised learning
  » set of examples, label pairs provided
  » MIL lingo: set of examples = bag of instances
  » Bag labeled positive if at least one instance in bag is positive

EXISTING ANALYSIS

Data model (bottom up)
  » Draw r instances and their labels from fixed distribution P
  » Create bag from instances, determine its label (max of instance labels)
  » Return bag & bag label to learner
Blum & Kalai (1998)
  » If: access to noise tolerant instance learner, instances drawn independently
  » Then: bag sample complexity linear in r
Sabato & Tishby (2009)
  » If: can minimize empirical error on bags
  » Then: bag sample complexity logarithmic in r

APPLICATIONS

Object Detection (images)
  » Instance: image patch
  » Instance label: is face?
  » Bag: whole image
  » Bag label: contains face?

Phoneme Detection (audio)
  » Instance: audio clip
  » Instance label: is V?
  » Bag: whole audio
  » Bag label: contains V?

Event Detection (video)
  » Instance: video clip
  » Instance label: is true
  » Bag: whole video
  » Bag label: contains [keep, delete]

OBSERVATIONS

Top down process: draw entire bag from a bag distribution, then get instances
  » Instances of a bag lie on a manifold
  » Potentially infinite number of instances per bag = existing analysis inappropriate
  » Expect sample complexity to scale with manifold parameters (curvature, dimension, volume, etc)

MANIFOLD BAGS

FORMULATION

• Manifold bag g drawn from bag distribution P

  h ∈ H, h : B → [0, 1]''
  • Corresponding bag hypotheses:
  h ∈ H, h : B → [0, 1]''
  I(″) = max h(x)

MANIFOLD BAGS

FORMULATION

• Manifold bag g drawn from bag distribution P

  h ∈ H, h : B → [0, 1]''
  • Corresponding bag hypotheses:
  h ∈ H, h : B → [0, 1]''
  I(″) = max h(x)

GENERALIZATION BOUNDS

VC DIMENSION

A way of relating empirical error E(y) to generalization error E(γ)

Standard bound:

ε ≤ O(√r(m/n) log(1/γ))

VC Dimension of bag hypothesis class

Want to relate VC(P) ≤ VC(P(C)) ≤ VC(P(C′))

For finite sized bags (Sabato & Tishby 2009):

VC(P(C)) ≤ VC(P) log γ

Turns out VC(P) is unbounded even for arbitrary smooth bags

TAMING THE RICHNESS

Bag hypothesis class too powerful: for positive bag,
  need to only classify one instance as positive

Infinitely many instances = too much flexibility for bag hypothesis

Would like to ensure a non-negligible portion of positive bags is labeled positive

Solution:
  » Switch to real valued hypothesis class
  » h ∈ H, h : B → [0, 1]
  » h_0 must be Lipschitz smooth w.r.t. Z
  » h_0 must label bags with a margin

FAT SHATTERING DIMENSION

Fat shattering dimension relates empirical error at margin γ to generalization error

Standard bound:

ε ≤ c_f √ r(m/n) log(1/γ)

Fat shattering dimension of bag hypothesis class

Unlike VC, we can relate P(C) ≤ P(C′)

Key quantities: empirical error at margin γ, the number of training bags (m), manifold bag dimension (d'), manifold bag volume (V), smoothness (q)

Filip

Orient

3D

NUMERICAL RESULTS

EXPERIMENTS

SYNTHETIC DATA

smooth

Error scales with curvature and volume

REAL DATA

INRIA Heads (Dalal et al. ’05)

TIMIT Phonemes (Garofolo et al., ’93)

p4w-16

p4w-32

Error scales with number of training bags, volume, number of queried instances and number of IQH iterations.

TRAINING WITH MANY INSTANCES

• Problem: want many instances/bag, but have computational limits
  » Solution: Iterative Querying Heuristic (IQH)
  » Grab small number of instances/bag, run standard MIL algorithm
  » Query more instances from each bag, only keep the ones that get high score from classifier

At each iteration, train with small # of instances

TAKE-HOME MESSAGE

• Increasing r reduces complexity term
• Increasing γ reduces failure probability
  » Seems to contradict previous results (smaller bag size r is better)
  » Important difference between r and γ !
  » If γ is small, may only get negative instances from a positive bag
  » Increasing r requires extra labels, increasing γ does not