Deep Metric Learning to Rank

Fatih Cakir*  Kun He*  Xide Xia  Brian Kulis*  Stan Sclaroff*
* FirstFuel  Facebook Reality Labs  Boston University (equal contribution)

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank
Reduce loss mis-specification
• Pair-based: point-wise LTR
• Triplet-based: pair-wise LTR

Deep Euclidean embeddings, optimized wrt. Average Precision

Discrete sorting: gradients are zero almost everywhere
AP: non-decomposable over individual examples

Approximation by quantization: well-behaved gradients
Stochastic (minibatch) backprop
Minibatch sampling strategy

Probabilistic interpretation of AP (AUC-PR)
• Parametric forms of precision and recall
• Change-of-variable + distance quantization
• Simple histogram-based formula

Motivation: Optimize the true "quantity of interest" for retrieval
✓ list-wise learning to rank

Batch IR setup: fixed query set, fixed database - infeasible
Minibatch setup:
• Each example is treated as the query once.
• Optimize mAP over minibatch.

FastAP : Average Precision Loss

FastAP:

$\hat{\text{FastAP}} = \frac{2}{M} \sum_{i=1}^{M} \left( F_i^+ B_i^- + B_i^- F_i^+ + F_i^+ B_i^- + B_i^- F_i^+ \right)_{i \neq j}$

Challenges

Discrete sorting: gradients are zero almost everywhere
AP: non-decomposable over individual examples

Approximation by quantization: well-behaved gradients
Stochastic (minibatch) backprop
Minibatch sampling strategy

Probabilistic interpretation of AP (AUC-PR)
• Parametric forms of precision and recall
• Change-of-variable + distance quantization
• Simple histogram-based formula

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank

Batch IR setup: fixed query set, fixed database - infeasible
Minibatch setup:
• Each example is treated as the query once.
• Optimize mAP over minibatch.

FastAP : Formulation

Fast AP: Average Precision Loss

How to Train with Minibatches?

• Batch IR setup: fixed query set, fixed database - infeasible
• Minibatch setup:
  • Each example is treated as the query once.
  • Optimize mAP over minibatch.

How Well Does It Work?

• Larger batches — longer lists
  • harder retrieval problems
  • Overcoming GPU mem. limit
  • Gather gradients wrt. embedding matrix
  • Also works on single GPU!

Related work
Hashing as Tie-Aware Learning to Rank. K. He et al., CVPR 18
Efficient Optimization of Rank-based Loss Functions. P. Mohapatra et al., CVPR 18

Deep Metric Learning to Rank

Fatih Cakir*  Kun He*  Xide Xia  Brian Kulis*  Stan Sclaroff*
* FirstFuel  Facebook Reality Labs  Boston University (equal contribution)

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank
Reduce loss mis-specification
• Pair-based: point-wise LTR
• Triplet-based: pair-wise LTR

Deep Euclidean embeddings, optimized wrt. Average Precision

Discrete sorting: gradients are zero almost everywhere
AP: non-decomposable over individual examples

Approximation by quantization: well-behaved gradients
Stochastic (minibatch) backprop
Minibatch sampling strategy

Probabilistic interpretation of AP (AUC-PR)
• Parametric forms of precision and recall
• Change-of-variable + distance quantization
• Simple histogram-based formula

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank

Batch IR setup: fixed query set, fixed database - infeasible
Minibatch setup:
• Each example is treated as the query once.
• Optimize mAP over minibatch.

FastAP : Average Precision Loss

FastAP:

$\hat{\text{FastAP}} = \frac{2}{M} \sum_{i=1}^{M} \left( F_i^+ B_i^- + B_i^- F_i^+ + F_i^+ B_i^- + B_i^- F_i^+ \right)_{i \neq j}$

Challenges

Discrete sorting: gradients are zero almost everywhere
AP: non-decomposable over individual examples

Approximation by quantization: well-behaved gradients
Stochastic (minibatch) backprop
Minibatch sampling strategy

Probabilistic interpretation of AP (AUC-PR)
• Parametric forms of precision and recall
• Change-of-variable + distance quantization
• Simple histogram-based formula

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank

Batch IR setup: fixed query set, fixed database - infeasible
Minibatch setup:
• Each example is treated as the query once.
• Optimize mAP over minibatch.

FastAP : Formulation

Fast AP: Average Precision Loss

How to Train with Minibatches?

• Batch IR setup: fixed query set, fixed database - infeasible
• Minibatch setup:
  • Each example is treated as the query once.
  • Optimize mAP over minibatch.

How Well Does It Work?

• Larger batches — longer lists
  • harder retrieval problems
  • Overcoming GPU mem. limit
  • Gather gradients wrt. embedding matrix
  • Also works on single GPU!

Related work
Hashing as Tie-Aware Learning to Rank. K. He et al., CVPR 18
Efficient Optimization of Rank-based Loss Functions. P. Mohapatra et al., CVPR 18

Deep Metric Learning to Rank

Fatih Cakir*  Kun He*  Xide Xia  Brian Kulis*  Stan Sclaroff*
* FirstFuel  Facebook Reality Labs  Boston University (equal contribution)

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank
Reduce loss mis-specification
• Pair-based: point-wise LTR
• Triplet-based: pair-wise LTR

Deep Euclidean embeddings, optimized wrt. Average Precision

Discrete sorting: gradients are zero almost everywhere
AP: non-decomposable over individual examples

Approximation by quantization: well-behaved gradients
Stochastic (minibatch) backprop
Minibatch sampling strategy

Probabilistic interpretation of AP (AUC-PR)
• Parametric forms of precision and recall
• Change-of-variable + distance quantization
• Simple histogram-based formula

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank

Batch IR setup: fixed query set, fixed database - infeasible
Minibatch setup:
• Each example is treated as the query once.
• Optimize mAP over minibatch.

FastAP : Average Precision Loss

FastAP:

$\hat{\text{FastAP}} = \frac{2}{M} \sum_{i=1}^{M} \left( F_i^+ B_i^- + B_i^- F_i^+ + F_i^+ B_i^- + B_i^- F_i^+ \right)_{i \neq j}$

Challenges

Discrete sorting: gradients are zero almost everywhere
AP: non-decomposable over individual examples

Approximation by quantization: well-behaved gradients
Stochastic (minibatch) backprop
Minibatch sampling strategy

Probabilistic interpretation of AP (AUC-PR)
• Parametric forms of precision and recall
• Change-of-variable + distance quantization
• Simple histogram-based formula

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank

Batch IR setup: fixed query set, fixed database - infeasible
Minibatch setup:
• Each example is treated as the query once.
• Optimize mAP over minibatch.

FastAP : Formulation

Fast AP: Average Precision Loss

How to Train with Minibatches?

• Batch IR setup: fixed query set, fixed database - infeasible
• Minibatch setup:
  • Each example is treated as the query once.
  • Optimize mAP over minibatch.

How Well Does It Work?

• Larger batches — longer lists
  • harder retrieval problems
  • Overcoming GPU mem. limit
  • Gather gradients wrt. embedding matrix
  • Also works on single GPU!

Related work
Hashing as Tie-Aware Learning to Rank. K. He et al., CVPR 18
Efficient Optimization of Rank-based Loss Functions. P. Mohapatra et al., CVPR 18

Deep Metric Learning to Rank

Fatih Cakir*  Kun He*  Xide Xia  Brian Kulis*  Stan Sclaroff*
* FirstFuel  Facebook Reality Labs  Boston University (equal contribution)

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank
Reduce loss mis-specification
• Pair-based: point-wise LTR
• Triplet-based: pair-wise LTR

Deep Euclidean embeddings, optimized wrt. Average Precision

Discrete sorting: gradients are zero almost everywhere
AP: non-decomposable over individual examples

Approximation by quantization: well-behaved gradients
Stochastic (minibatch) backprop
Minibatch sampling strategy

Probabilistic interpretation of AP (AUC-PR)
• Parametric forms of precision and recall
• Change-of-variable + distance quantization
• Simple histogram-based formula

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank

Batch IR setup: fixed query set, fixed database - infeasible
Minibatch setup:
• Each example is treated as the query once.
• Optimize mAP over minibatch.

FastAP : Average Precision Loss

FastAP:

$\hat{\text{FastAP}} = \frac{2}{M} \sum_{i=1}^{M} \left( F_i^+ B_i^- + B_i^- F_i^+ + F_i^+ B_i^- + B_i^- F_i^+ \right)_{i \neq j}$

Challenges

Discrete sorting: gradients are zero almost everywhere
AP: non-decomposable over individual examples

Approximation by quantization: well-behaved gradients
Stochastic (minibatch) backprop
Minibatch sampling strategy

Probabilistic interpretation of AP (AUC-PR)
• Parametric forms of precision and recall
• Change-of-variable + distance quantization
• Simple histogram-based formula

Motivation: Optimize the true “quantity of interest” for retrieval
✓ list-wise learning to rank

Batch IR setup: fixed query set, fixed database - infeasible
Minibatch setup:
• Each example is treated as the query once.
• Optimize mAP over minibatch.

FastAP : Formulation

Fast AP: Average Precision Loss

How to Train with Minibatches?

• Batch IR setup: fixed query set, fixed database - infeasible
• Minibatch setup:
  • Each example is treated as the query once.
  • Optimize mAP over minibatch.

How Well Does It Work?

• Larger batches — longer lists
  • harder retrieval problems
  • Overcoming GPU mem. limit
  • Gather gradients wrt. embedding matrix
  • Also works on single GPU!

Related work
Hashing as Tie-Aware Learning to Rank. K. He et al., CVPR 18
Efficient Optimization of Rank-based Loss Functions. P. Mohapatra et al., CVPR 18