Scalable Nearest Neighbor Search for Optimal Transport

Arturs Backurs
TTIC

Yihe Dong
Microsoft Research

Piotr Indyk
MIT

Ilya Razenshteyn
Microsoft Research

Tal Wagner
MIT

ICML 2020
We introduce **Flowtree**:  

- Fast nearest neighbor search algorithm for Optimal Transport  
  - a.k.a. Earth Mover Distance, Wasserstein-1 distance  

- **Analytically:** Linear running time, worst-case approximation bound  

- **Empirically:** Speeds up SOTA by up to 7.4 times  

- **Code** publicly available on github:  
  [https://github.com/ilyaraz/ot_estimators](https://github.com/ilyaraz/ot_estimators)
Optimal Transport

Distance between points $x$ and $y$: Euclidean, Manhattan, ...
Optimal Transport

Distance between point sets $X$ and $Y$?
Optimal Transport

Distance between point sets $X$ and $Y$?

Choose distributions $\mathcal{D}_X$ on $X$ and $\mathcal{D}_Y$ on $Y$

- For this talk: uniform distributions

$OT(X, Y) = \text{value of minimum-cost flow}$

from $X$ to $Y$ with demands $\mathcal{D}_X$ and $\mathcal{D}_Y$

$$= \min_F \sum_{x \in X, y \in Y} ||x - y|| \cdot F(x, y)$$

s.t. $F$ is a distribution on $X \times Y$

with marginals $\mathcal{D}_X$ and $\mathcal{D}_Y$
Motivation: “Word Mover Distance”

Kusner et al. (2015): Use OT as distance between text documents

Document 1

Obama speaks to the media in Illinois

Document 2

The President greets the press in Chicago

Word embedding
OT Nearest Neighbor Search

Exact computation does not scale
Approximate algorithms:

Linear time
Crude approximation

Means [Kusner et al. ‘15]
TF-IDF [Luhn ‘57]
Quadtree [Charikar ‘02, Indyk-Thaper ‘03]

Best of both worlds?

Flowtree:
“Slower” linear time
Fine approximation

R-WMD [Kusner et al. ‘15]
ACT [Atasu-Mittelholzer ‘19]
Sinkhorn [Cuturi ‘13]

Quadratic time
Fine approximation
Algorithm
Starting Point: Quadtree

Side length $\Delta$

$\Delta / 2$

$\Delta / 4$

$\Delta / 8$
Optimal Transport on a Quadtree

**Compute:** Optimal flow on tree
Optimal Transport on a Quadtree

**Compute:** Optimal flow on tree

**Return:** Flow cost in tree distance

\[ \sum_{\text{Tree edge } e} \text{weight}(e) \cdot F_T(e) \]

**Even faster:** $\ell_1$-embedding!

[Kleinberg-Tardos ’00, Charikar ’02, Indyk-Thaper ’03, Le et al. ’19, ...]
Our Algorithm: Flowtree

Evaluate optimal tree flow in original metric space

Return: $\sum_{x \in X, y \in Y} \|x - y\| \cdot F_T(x, y)$
Flowtree: Properties

- **Running time:**
  - **Quadtree:** Linear, $\ell_1$ embedding
  - **Flowtree:** Linear, does not give embedding

- **Nearest neighbor search approximation:**
  - **Quadtree:** $O(\log(d \cdot \Delta) \cdot \log(s \cdot n))$-approx.
    - Dependence on $n$ is necessary
  - **Flowtree:** $O(\log(d \cdot \Delta) \cdot \log s)$-approx.
  - **Flowtree in uniform case:** $O(\log^2 s)$-approx.
Experiments

20newsgroups dataset
Individual Algorithm Evaluation

Fast (milliseconds)
Crude approximation

Slower (seconds)
Fine approximation

recall@\(k\) = % queries whose true nearest neighbor is ranked in top-\(k\) returned points
Pipeline Experiments

Fast (milliseconds)
Crude approximation

Slow (seconds)
Fine approximation

Dataset ➔ Fast crude approx. ➔ Slow fine approx. ➔ Exact ➔ Nearest neighbor
Pipeline Experiments: Recall\(\@1\)

1st: Quadtree

2nd: R-WMD [Kusner et al. ‘15]

ACT-1 [Atasu-Mittelholzer’19]

Sinkhorn-1 [Cuturi’13]

Sinkhorn-3

3rd: Exact

New: Flowtree

\(x3.7\) speed up
Pipeline Experiments: Recall@5

1st: Quadtree
2nd: R-WMD [Kusner et al. ’15], ACT-1 [Atasu-Mittelholzer’19], Sinkhorn-1 [Cuturi’13], Sinkhorn-3
3rd: Exact
New: Flowtree

x7.4 speed up
Conclusion

We introduce **Flowtree**:  

- Fast *nearest neighbor search* algorithm for **Optimal Transport**  
  - a.k.a. Earth Mover Distance, Wasserstein-1 distance  

- **Analytically**: *Linear* running time, *worst-case* approximation bound  

- **Empirically**: Speeds up SOTA by up to 7.4 times  

- **Code** publicly available on github:  
  https://github.com/ilyaraz/ot_estimators

Thank you