Efficient Trajectory Retrieval

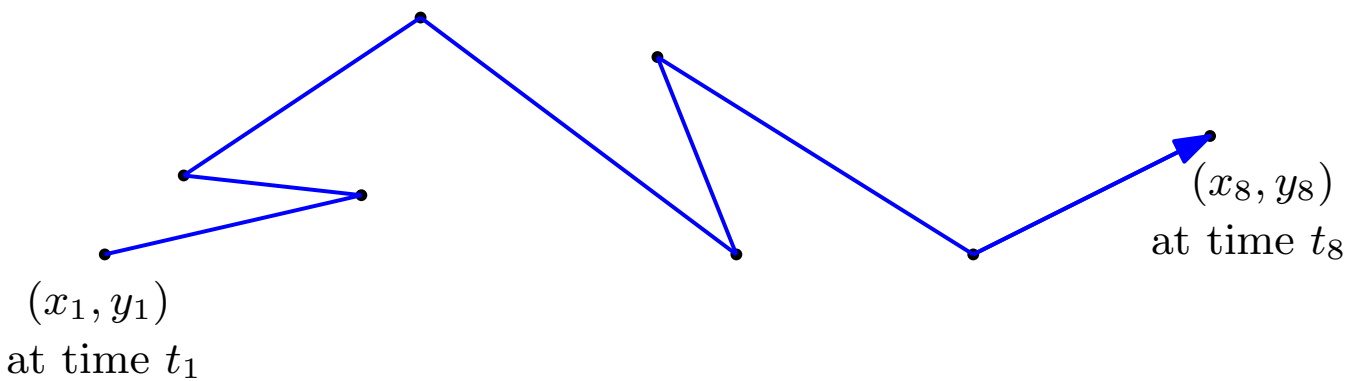
By Hasan Pourmahmood

December 6, 2021

Trajectory

A spatial trajectory (also called a piece-wise linear curve) is a sequence of waypoints along with a time stamp, i.e.

$$T = \{(x_1, y_1, t_1), \ldots, (x_n, y_n, t_n)\}.$$



Popular Distances

- Dynamic Time Warping distance (DTW)
- Fréchet distance
- Discrete Fréchet distance

Popular Distances

- Dynamic Time Warping distance (DTW)
- Fréchet distance
- Discrete Fréchet distance

Complexity: Quadratic in number of waypoints $O(n^2)$.

Popular Distances

- Dynamic Time Warping distance (DTW)
- Fréchet distance
- Discrete Fréchet distance

Complexity: Quadratic in number of waypoints $O(n^2)$.

Queryies

- k-NN queries: Find k nearest trajectories from data to a given query trajectory Q.
- Range queries: Find all trajectories within distance r from data to a given query trajectory Q.

Popular Distances

- Dynamic Time Warping distance (DTW)
- Fréchet distance
- Discrete Fréchet distance

Complexity: Quadratic in number of waypoints $O(n^2)$.

Queryies

- k-NN queries: Find k nearest trajectories from data to a given query trajectory Q.
- Range queries: Find all trajectories within distance r from data to a given query trajectory Q.

Applications

Route extraction, route recommendation, ...

Approach and Related Work

Related work

We are going to apply a similar method introduced in the following paper:

Apostolos N. Papadopoulos. Trajectory retrieval with latent semantic analysis. In SAC08 March 16–20, pages 1089–1094.

Approach and Related Work

Related work

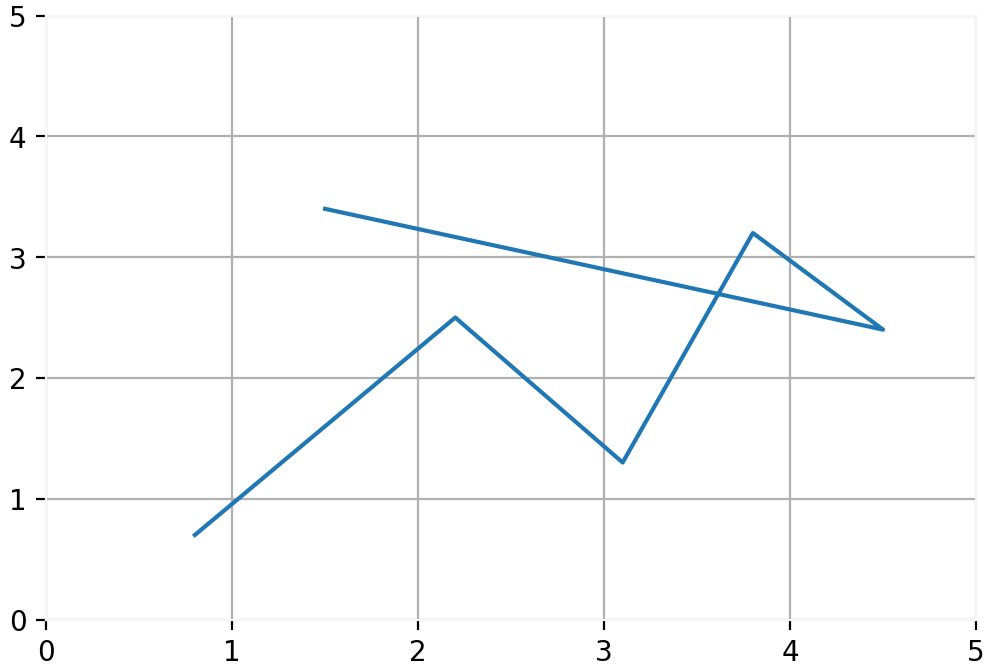
We are going to apply a similar method introduced in the following paper:

Apostolos N. Papadopoulos. Trajectory retrieval with latent semantic analysis. In SAC08 March 16–20, pages 1089–1094.

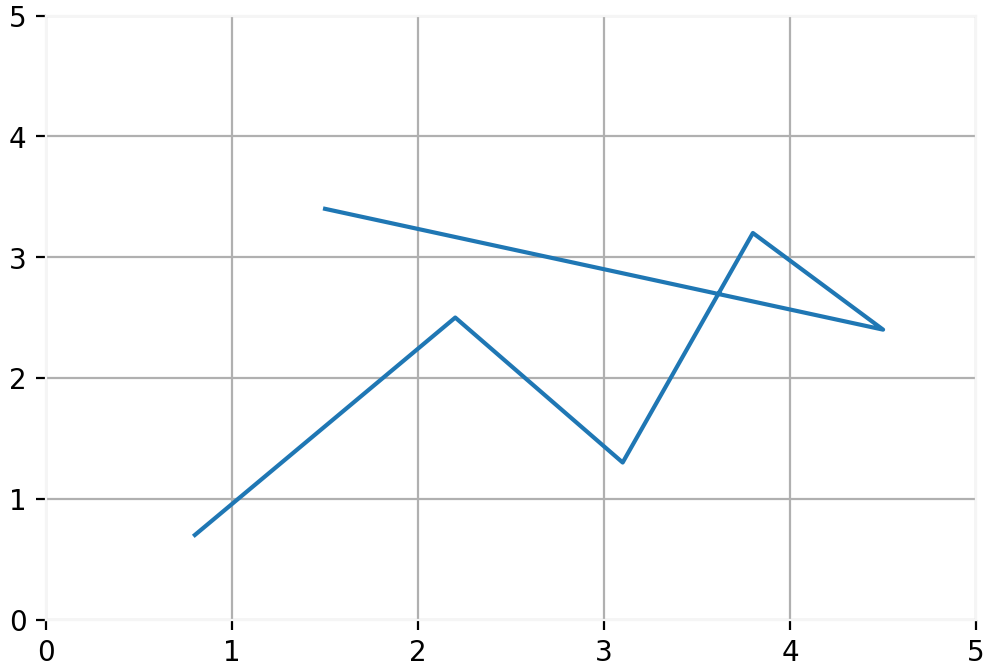
Novelty

However, among other things, we will do some modifications and apply novel vectorization techniques. Moreover, we will evaluate our approach on 3 real world datasets not only on Precision and Recall but also based on MAP and nDCG which will be adapted from their definitions in IR on text data.

Vector space model using grid structure

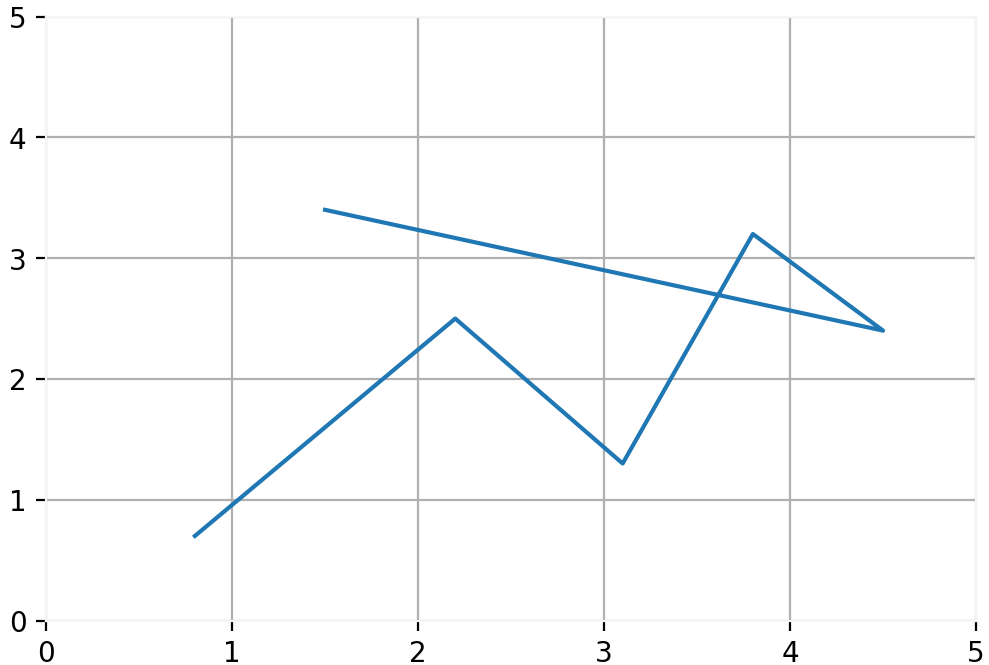


Vector space model using grid structure



- **Vocabulary:** the whole set of grids $\{c_1, \ldots, c_n\}$
- **Document:** trajectory
- **Term:** a grid

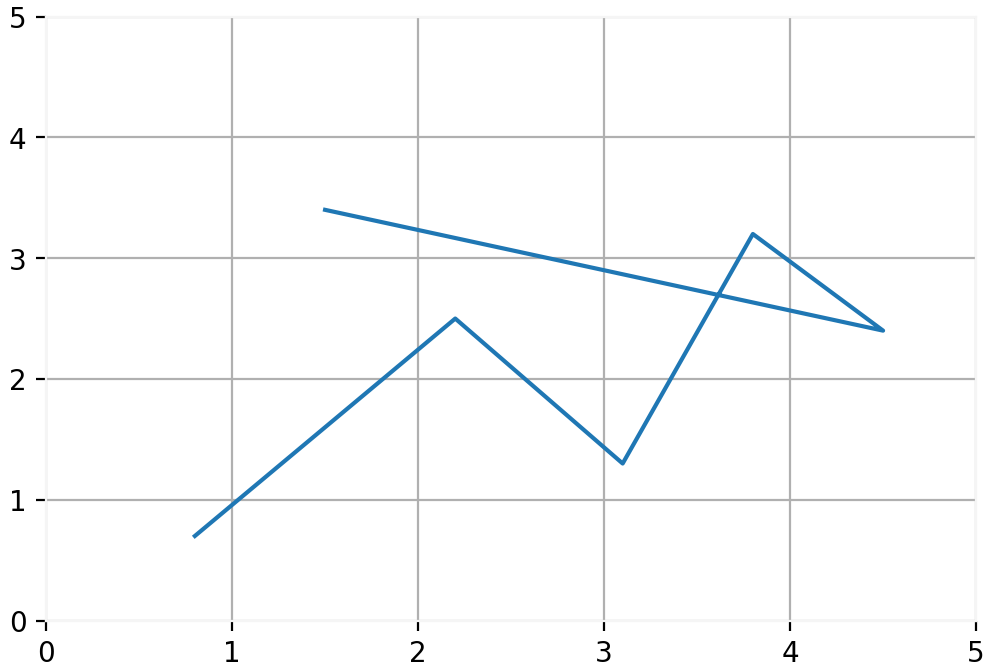
Vector space model using grid structure



Binary representation:

$$\tilde{T} = [1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0]$$

Vector space model using grid structure



Binary representation:

$$\tilde{T} = [1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0]$$

Frequency representation:

$$\tilde{T} = [1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, \underbrace{2, 2}_{13,14}, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0]$$

Grid aggregate distance (GAD)

Consider a cell similarity matrix $C_{n\times n} = (sim(c_i, c_j))_{i,j=1}^{n}$ (c_i is the center of cell i), like

$$sim(c_i, c_j) = e^{-\|c_i - c_j\|^2/\sigma^2}$$

Grid aggregate distance (GAD)

Consider a cell similarity matrix $C_{n\times n} = (sim(c_i, c_j))_{i,j=1}^{n}$ (c_i is the center of cell i), like

$$sim(c_i, c_j) = e^{-\|c_i - c_j\|^2/\sigma^2}$$

which is applied in experiments. Then

$$GAD(T_1, T_2) = \sqrt{(\tilde{T}_1 - \tilde{T}_2)^t \cdot C \cdot (\tilde{T}_1 - \tilde{T}_2)}$$

Grid aggregate distance (GAD)

Consider a cell similarity matrix $C_{n\times n} = (sim(c_i, c_j))_{i,j=1}^{n}$ (c_i is the center of cell i), like

$$sim(c_i, c_j) = e^{-\|c_i - c_j\|^2/\sigma^2}$$

which is applied in experiments. Then

$$GAD(T_1, T_2) = \sqrt{(\tilde{T}_1 - \tilde{T}_2)^t \cdot C \cdot (\tilde{T}_1 - \tilde{T}_2)}$$

Problem: The number of grids n is high (in experiments it is set to 2500), and so GAD computation would be inefficient.

Grid aggregate distance (GAD)

Consider a cell similarity matrix $C_{n\times n} = (sim(c_i, c_j))_{i,j=1}^{n}$ (c_i is the center of cell i), like

$$sim(c_i, c_j) = e^{-\|c_i - c_j\|^2/\sigma^2}$$

which is applied in experiments. Then

$$GAD(T_1, T_2) = \sqrt{(\tilde{T}_1 - \tilde{T}_2)^t \cdot C \cdot (\tilde{T}_1 - \tilde{T}_2)}$$

Problem: The number of grids n is high (in experiments it is set to 2500), and so GAD computation would be inefficient.

How to address this problem?

Dimensionality reduction technique

- Decompose C to $C = P\Delta P^t$,

Dimensionality reduction technique

- Decompose C to $C = P\Delta P^t$,
- Set $\tilde{\tilde{D}} = \Delta^{1/2} P^t \tilde{D}$, where $\tilde{D}$ is the binary/frequency-type representation of data D,

Dimensionality reduction technique

- Decompose C to $C = P\Delta P^t$,
- Set $\tilde{\tilde{D}} = \Delta^{1/2} P^t \tilde{D}$, where $\tilde{D}$ is the binary/frequency-type representation of data D,
- Use SVD for $\tilde{\tilde{D}}$ to reduce the dimension (in experiments 2500 is reduced to 25 and 100),

Dimensionality reduction technique

- Decompose C to $C = P\Delta P^t$,
- Set $\tilde{\tilde{D}} = \Delta^{1/2} P^t \tilde{D}$, where $\tilde{D}$ is the binary/frequency-type representation of data D,
- Use SVD for $\tilde{\tilde{D}}$ to reduce the dimension (in experiments 2500 is reduced to 25 and 100),
- This reduced dimensional data would be treated as our index,

Dimensionality reduction technique

- Decompose C to $C = P\Delta P^t$,
- Set $\tilde{\tilde{D}} = \Delta^{1/2} P^t \tilde{D}$, where $\tilde{D}$ is the binary/frequency-type representation of data D,
- Use SVD for $\tilde{\tilde{D}}$ to reduce the dimension (in experiments 2500 is reduced to 25 and 100),
- This reduced dimensional data would be treated as our index,
- Employ Euclidean distance for new short vectors as an approximation for GAD between trajectories,

Dimensionality reduction technique

- Decompose C to $C = P\Delta P^t$,
- Set $\tilde{\tilde{D}} = \Delta^{1/2} P^t \tilde{D}$, where $\tilde{D}$ is the binary/frequency-type representation of data D,
- Use SVD for $\tilde{\tilde{D}}$ to reduce the dimension (in experiments 2500 is reduced to 25 and 100),
- This reduced dimensional data would be treated as our index,
- Employ Euclidean distance for new short vectors as an approximation for GAD between trajectories,
- When a query comes, do the same and consult the index to retrieve appropriate trajectories.

Data Sets

Data	Size	Cleaned	Sample	Grids	Reduced dim
Car-Bus	163	105	105	2500	25
Characters	2858	2858	300	2500	25
Geolife	17621	14187	1000	2500	100

Table: Overview of data sets

Data Sets

Data	Size	Cleaned	Sample	Grids	Reduced dim
Car-Bus	163	105	105	2500	25
Characters	2858	2858	300	2500	25
Geolife	17621	14187	1000	2500	100

Table: Overview of data sets

Figure: Car-Bus (left), Geolife (middle), Characters: letters n, u (right).

kNN Queries (Geolife)

k	Precision/Recall	MAP	nDCG@5	nDCG@10
5	0.8676	0.8632	0.9182	0.9443
10	0.8921	0.8751	0.7822	0.9394
20	0.9074	0.8904	0.5823	0.7989

Table: Performance with binary representation

k	Precision/Recall	MAP	nDCG@5	nDCG@10
5	0.9480	0.9597	0.9822	0.9930
10	0.9590	0.9577	0.8233	0.9886
20	0.9653	0.9604	0.6029	0.8330

Table: Performance with frequency representation

Range Queries (Geolife)

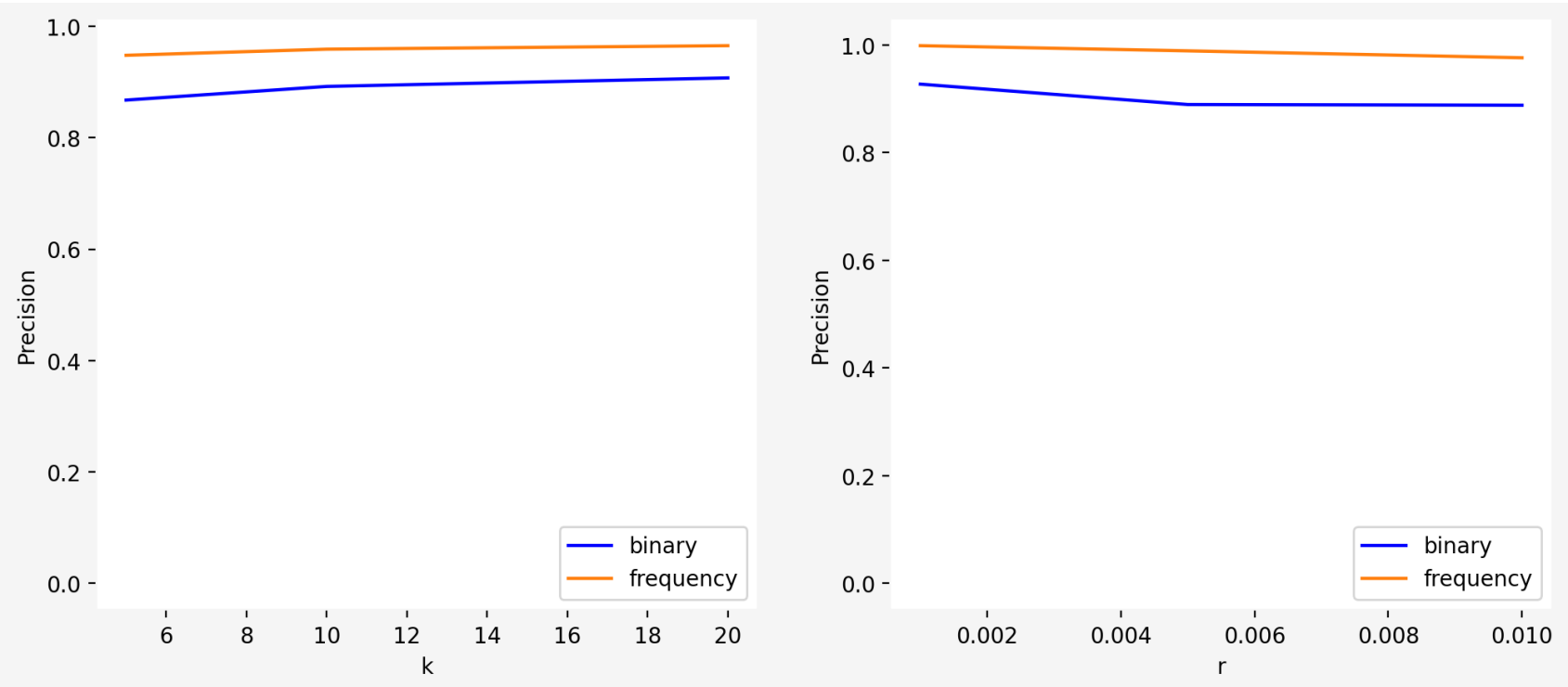
r	Precision	Recall	MAP	nDCG@5	nDCG@10
0.001	0.9275	0.9549	0.8116	0.7260	0.6559
0.005	0.8896	0.9023	0.2025	0.8893	0.8952
0.01	0.8883	0.8892	0.0376	0.9182	0.9390

Table: Performance with binary representation

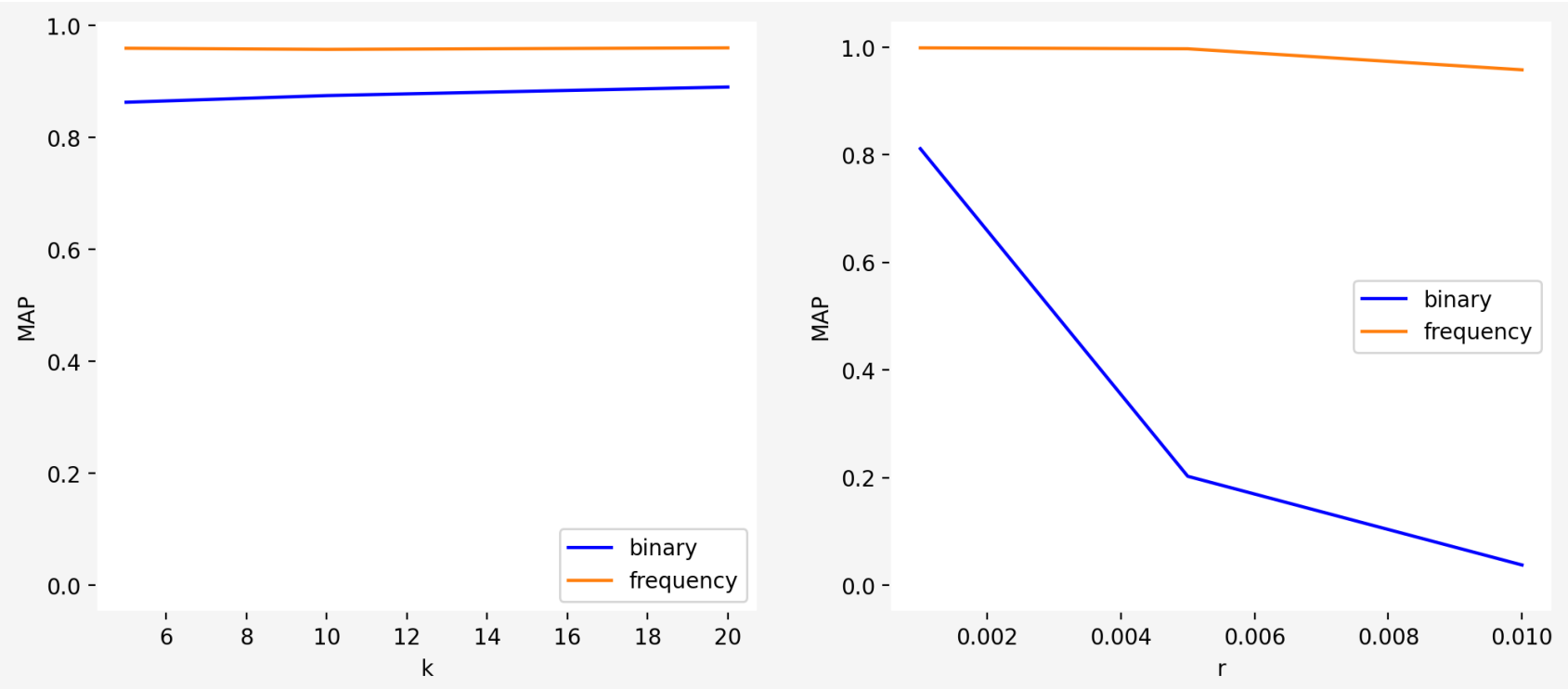
r	Precision	Recall	MAP	nDCG@5	nDCG@10
0.001	0.9990	1	0.9990	0.4906	0.3367
0.005	0.9894	1	0.9973	0.5291	0.3748
0.01	0.9765	0.9978	0.9582	0.5735	0.4276

Table: Performance with frequency representation

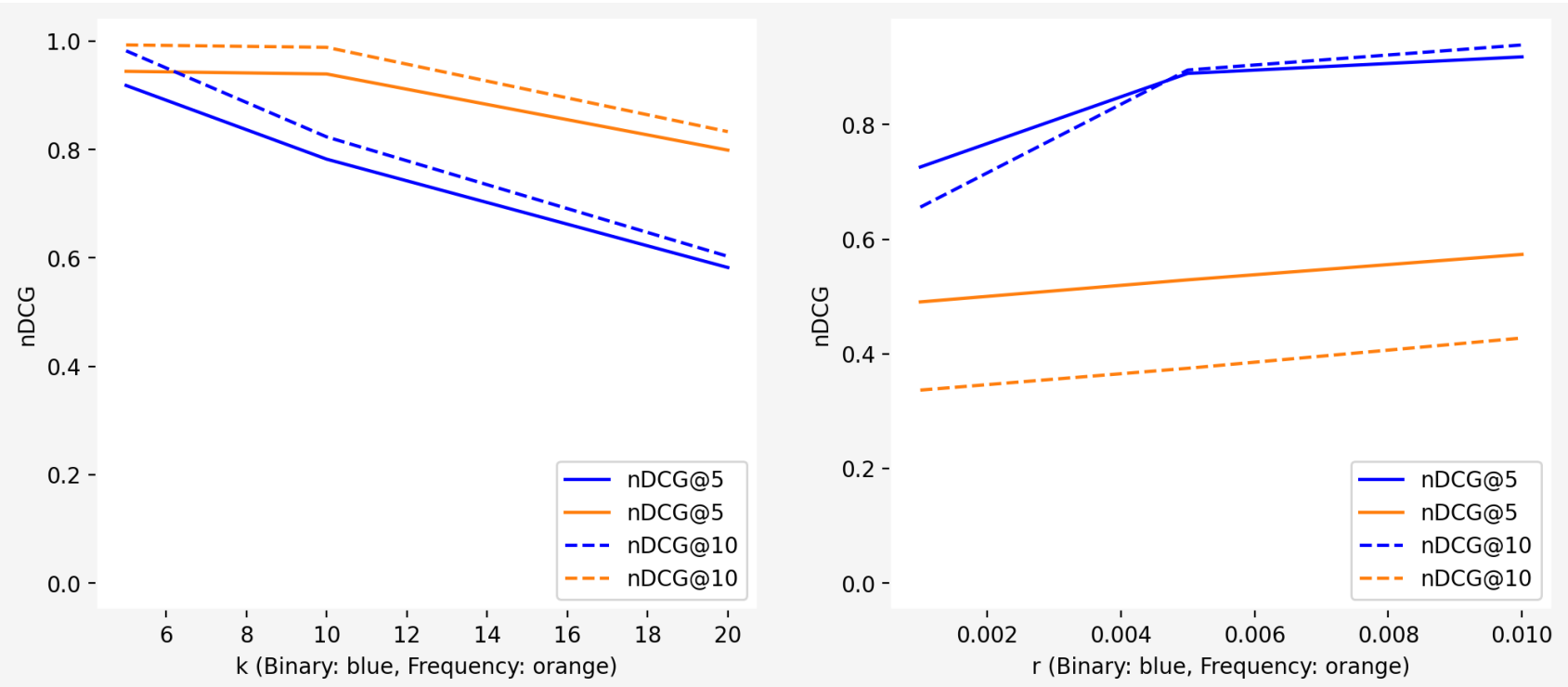
Precision


1.0
0.8
0.6
0.4
0.2
0.0
Precision
6
8
10
12
14
16
18
20
k
binary
frequency
1.0
0.8
0.6
0.4
0.2
0.0
Precision
0.002
0.004
0.006
0.008
0.010
r
binary
frequency

MAP


1.0
0.8
0.6
0.4
0.2
0.0
MAP
6
8
10
12
14
16
18
20
k
binary
frequency
0.002
0.004
0.006
0.008
0.010
r

nDCG


1.0
0.8
0.6
0.4
0.2
0.0
nDCG
6
8
10
12
14
16
18
20
k (Binary: blue, Frequency: orange)
nDCG@5
nDCG@5
nDCG@10
nDCG@10
0.8
0.6
0.4
0.2
0.0
nDCG
0.002
0.004
0.006
0.008
0.010
r (Binary: blue, Frequency: orange)
nDCG@5
nDCG@5
nDCG@10
nDCG@10

Conclusion

- LSI technique for trajectory retrieval is a quite successful one,

Conclusion

- LSI technique for trajectory retrieval is a quite successful one,
- The term-frequency-type representation tended to perform better in terms of AP, Recall and MAP,

Conclusion

- LSI technique for trajectory retrieval is a quite successful one,
- The term-frequency-type representation tended to perform better in terms of AP, Recall and MAP,
- The binary representation tended to perform better in terms of nDCG for range queries,

Conclusion

- LSI technique for trajectory retrieval is a quite successful one,
- The term-frequency-type representation tended to perform better in terms of AP, Recall and MAP,
- The binary representation tended to perform better in terms of nDCG for range queries,
- LSI technique works better with GAD in comparison to DTW distance,

Conclusion

- LSI technique for trajectory retrieval is a quite successful one,
- The term-frequency-type representation tended to perform better in terms of AP, Recall and MAP,
- The binary representation tended to perform better in terms of nDCG for range queries,
- LSI technique works better with GAD in comparison to DTW distance,
- Increasing the number of grids leads to a better performance,

Conclusion

- LSI technique for trajectory retrieval is a quite successful one,
- The term-frequency-type representation tended to perform better in terms of AP, Recall and MAP,
- The binary representation tended to perform better in terms of nDCG for range queries,
- LSI technique works better with GAD in comparison to DTW distance,
- Increasing the number of grids leads to a better performance,
- Increasing the number of dimensions in dimensionality reduction technique leads to a better performance.

My Contributions

- Applying a kernel-based similarity measure,
- Using 3 different datasets as benchmark including a big data,
- Utlizing binary vectorization of trajectories,
- Using term-frequency type trajectory vectorization,
- Applying DTW as a ground truth distance,
- Adjusting the definition of Precision, Recall, MAP and nDCG for trajectories,
- Creating a user-friendly interactive search tool for kNN and range queries of trajectories.

Questions?