Approximate Aggregation Queries on Geospatial Big Data

Dr. **Isam Mashhour Al Jawarneh**[1], Dr. Rebecca Montanari [2], Prof. Antonio Corradi [3]

[1] *Assistant Professor*, Department of Computer Science, University of Sharjah, UAE (ijawarneh@sharjah.ac.ae)
[2] Associate Professor, Department of Computer Science and Engineering – DISI, University of Bologna, Italy (rebecca.montanari@unibo.it)
[3] Professor, Department of Computer Science and Engineering – DISI, University of Bologna, Italy (Antonio.corradi@unibo.it)

IEEE GLOBECOM 2023

GC 2023 Workshop - SRINetworks

WS15-1: DIGITAL TWIN IN INDUSTRIAL NETWORKS

12/2023

Outline

- Introduction
  - Motivating scenario
  - Spatial data challenges & requirements
- Approximating geospatial aggregate queries
  - Overview
  - ApproxGeoAgg
- Results and Discussion
  - Deployment: baselines & testing setup
  - ApproxGeoAgg Vs. baseline
- Summary & future research

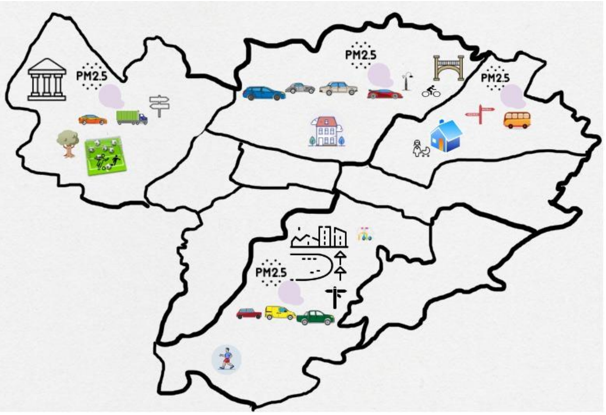
Motivating scenario

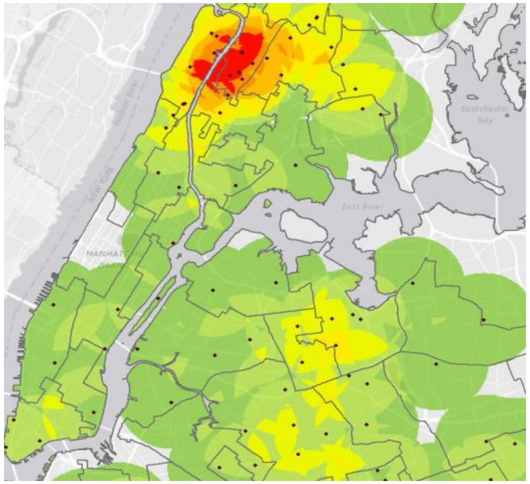
- Billions of GPS-enabled handheld devices collect massive data amounts
  - Urban planning and transportation data from smart cities
- Analyze data to explore the opportunities for better decision making
  - Urban planning and transportation design decisions

  •**Geospatial aggregation**

  •Air pollutants **density** in each **zone**,

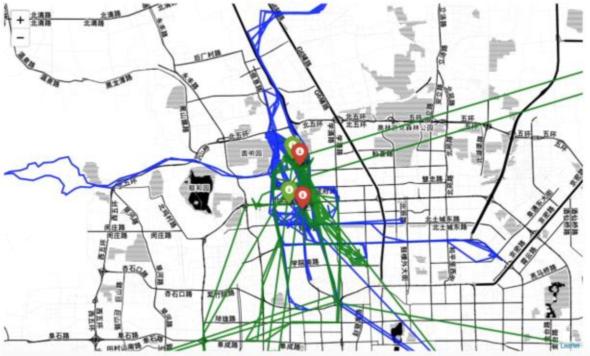
  •Autocorrelation between nearness and pollution





Visualizing georeferenced data requires aggregation

- **line-based**
  - time-series trajectory visualization of spatial data
  - Requires **aggregations** and **group-by**



- **region-based**
  - Tessellating geographic regions into grid cells, then, **grouping** data by region-based **aggregations**
  - e.g., **Choropleth** maps generation

Outline

- Geospatial big data analytics: Background and Motivating scenario
  - Motivating scenario
  - Spatial data challenges & requirements
- Approximating geospatial aggregate queries
  - Overview
  - ApproxGeoAgg
- Results and Discussion
  - Deployment: baselines & testing setup
  - ApproxGeoAgg Vs. baseline
- Summary & future research

Spatial data analytics challenges

Polygons

Shapefile, NYC

	LocationID	borough	geometry	zone
0	1	EWR	POLYGON ((-74.18445299999996 40.6949959999999,...	Newark Airport
1	2	Queens	(POLYGON ((-73.82337597260663 40.6389870471767...	Jamaica Bay
2	3	Bronx	POLYGON ((-73.84792614099985 40.87134223399991...	Allerton/Pelham Gardens
3	4	Manhattan	POLYGON ((-73.97177410965318 40.72582128133705...	Alphabet City
4	5	Staten Island	POLYGON ((-74.17421738099989 40.56256808599987...	Arden Heights

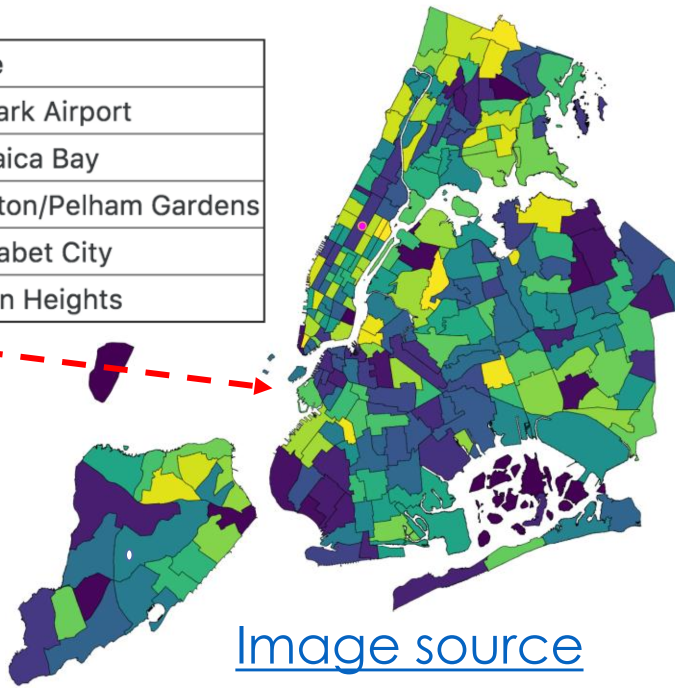


Image source

taxi dataset

	tpep_pickup_datetime	tpep_dropoff_datetime	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude
0	2016-05-01 00:00:00	2016-05-01 00:17:31	-73.985901	40.768040	-73.983986	40.730099
1	2016-05-01 00:00:00	2016-05-01 00:07:31	-73.991577	40.744751	-73.975700	40.765469
2	2016-05-01 00:00:00	2016-05-01 00:07:01	-73.993073	40.741573	-73.980995	40.744633
3	2016-05-01 00:00:00	2016-05-01 00:19:47	-73.991943	40.684601	-74.002258	40.733002
4	2016-05-01 00:00:00	2016-05-01 00:06:39	-74.005280	40.740192	-73.997498	40.737564

Points (parametrized) Projected Coordinate System (PCS)

Before **aggregations** → we need to assign trips pickups to city zones (districts) → an example of a **spatial join (expensive)**

	geometry	index_right	LocationID	borough	zone
0	POINT (-73.96599999999999 40.78)	42	43	Manhattan	Central Park

QoS Tension

Spatial (Approximate) Query Processing (S(A)QP)

Tension

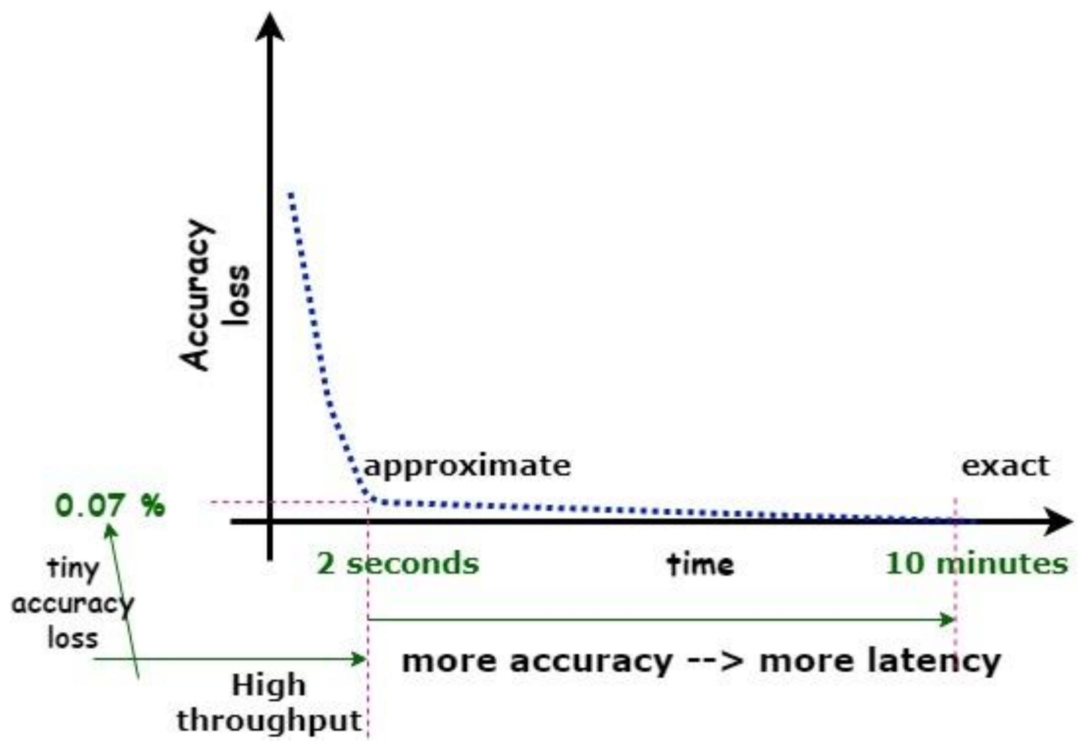
latency

Throughput

Accuracy

Spatial Approximate Query Processing

- Challenges
  - Data streams arrive very fast
  - Skewness and arrival rates fluctuate
- Decision makers accept tiny loss in accuracy in exchange for a throughput gain



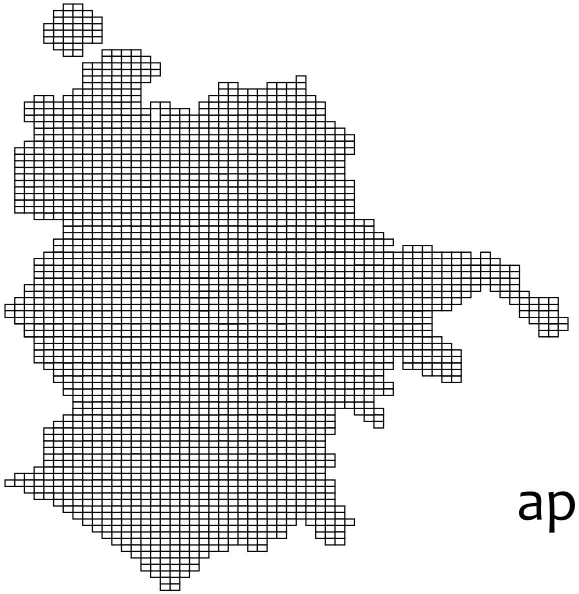
Outline

- Geospatial big data analytics: Background and Motivating scenario
  - Motivating scenario
  - Spatial data challenges & requirements
- Approximating geospatial aggregate queries
  - Overview
  - ApproxGeoAgg
- Results and Discussion
  - Deployment: baselines & testing setup
  - ApproxGeoAgg Vs. baseline
- Summary & future research

Geospatial aggregation Process

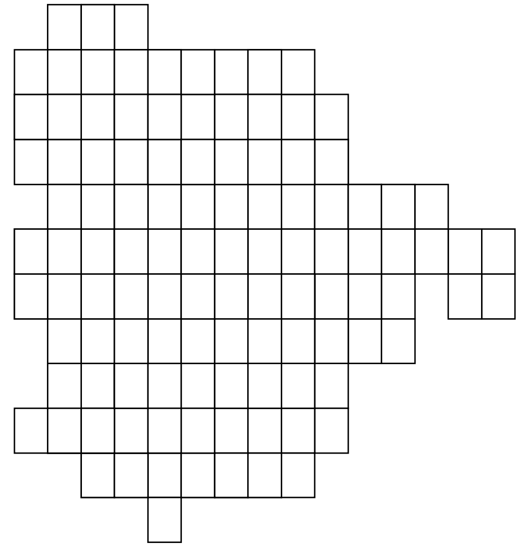
- Geospatial data **pre-processing**
  - **Clusters** (polygons) need to be defined
    - Granular level (Minimum Bounding Rectangle MBR, such as geohash)
    - Coarser level (neighborhood, district, borough, etc.,)
- Geospatial data **aggregation**
  - Join spatial data points with polygons
  - Geospatial stateful aggregation queries such as grouping-by

SAQP: Geohash encoding



With geohash precision 6

quick-and-
approximate filter



With geohash precision 5

Geohash cover, city of Rome, Italy

- Can be used for **stratified-like** sampling
  - Captures the reality
  - Each geohash is a **strata**
  - All geohash covering the area are **stratum**

Challenges in **stateful spatial** aggregations

- Stateful spatial data **aggregation**
  - Computationally **expensive** in real data stream settings
    - Georeferenced data is typically **parametrized**
    - Brining them into their original forms, is a kind of **geospatial join** (computationally **costly**)
  - Out-of-service during spikes in arrival rates

- Geospatial data **preprocessing** (including aggregation) is the **dominating** component for most spatial analytical pipelines , such as those encompassing a process of generating geospatial region-based maps (such as choropleth and heatmaps)

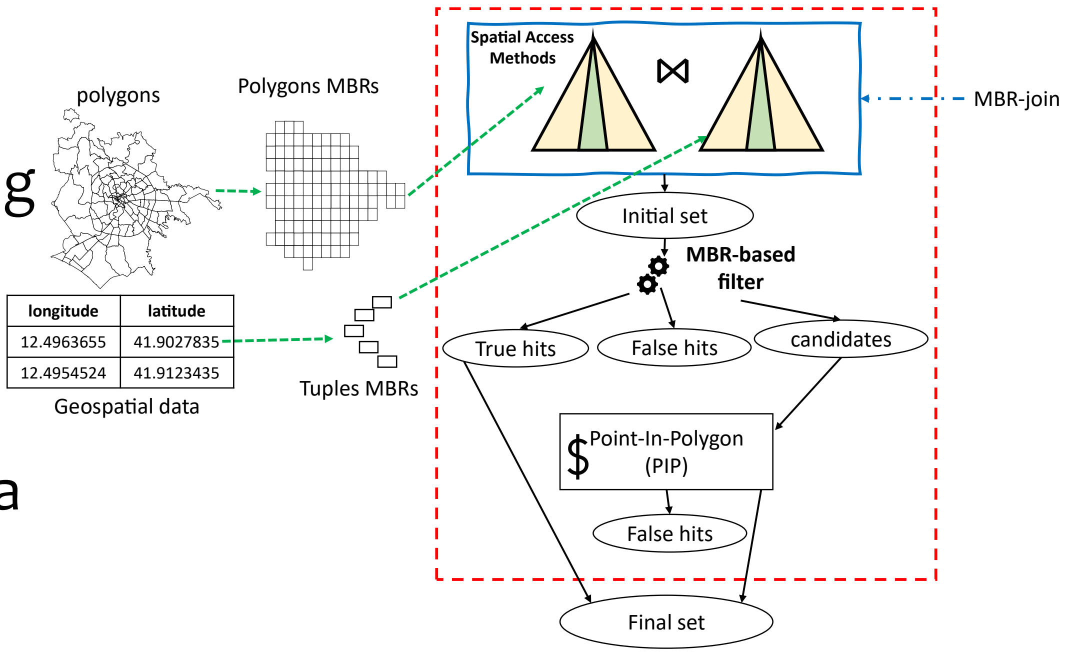
Coping up with geo-data loads

- **Scalability**
  - Hardware scalability. **Overprovisioning** resources
  - Scaling **up/out**
- **Approximate Query Processing** (**AQP**). Data reduction
  - **Spatial** Approximate Query Processing (**SAQP**)
  - e.g., dimensionality **reduction**, load shedding and geospatial **sampling**

Our focus!

Approximate spatial join: Plain Filter-and-refine

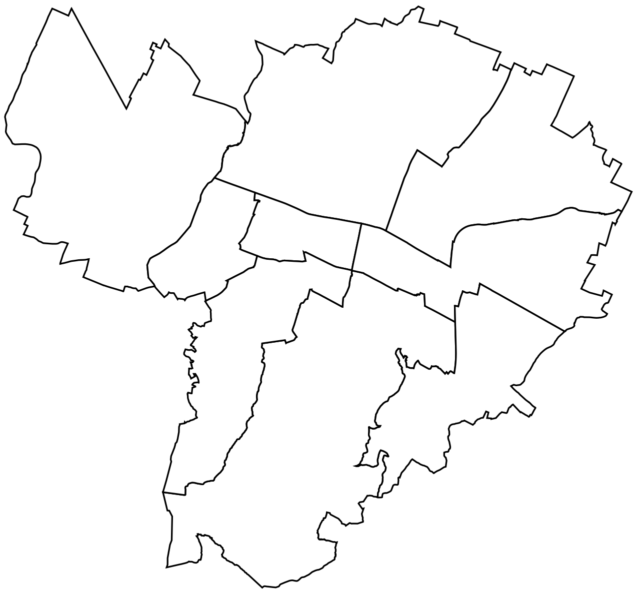
- Based on **dimensionality reduction**
  - Compute geohash for every point
  - Compute geohash covering of the embedding area
  - Perform a cheap equijoin to find which points fall within the embedding area (**filter**)
  - Use the ray casting algorithm to exclude false positives (**refine**)



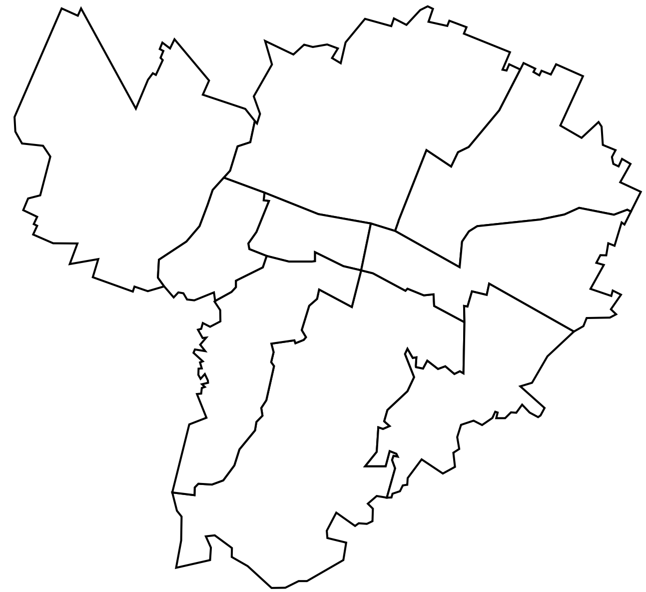
Outline

- Geospatial big data analytics: Background and Motivating scenario
  - Motivating scenario
  - Spatial data challenges & requirements
- Approximating geospatial aggregate queries
  - Overview
  - ApproxGeoAgg
- Results and Discussion
  - Deployment: baselines & testing setup
  - ApproxGeoAgg Vs. baseline
- Summary & future research

Boundary simplifier



Original polygons



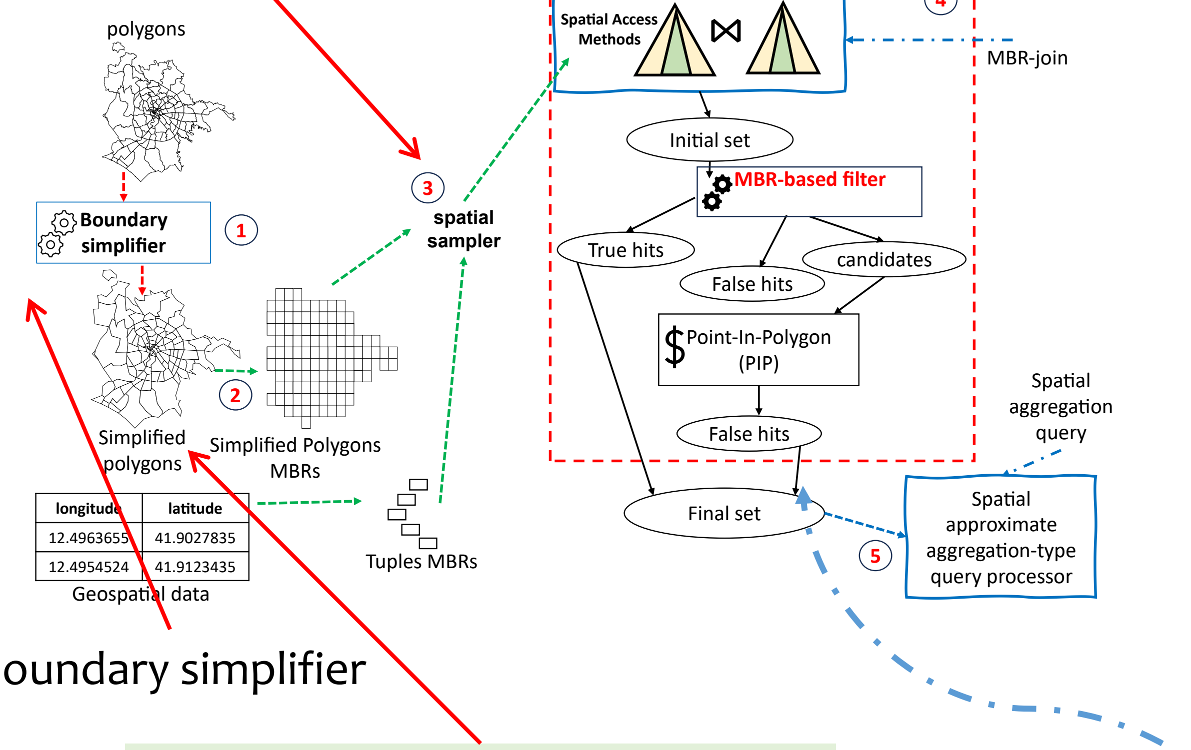
‘percentage of BPK’ 5%

Boundary **simplifier** function applied to polygons representing **Bologna** city, **Italy**

ApproxGeoAggr : Geospatial aggregation at Scale with QoS Guarantees

Plain Filter-and-refine

Stratified-like sampler



Six components

(1) Geospatial data **modelling** and **representation**

(2) Stratified-like geospatial **sampler**

(3) boundary **simplifier**

- Adapted version of the Douglas-Peucker (DP) algorithm

(4) geohash cover **generator** , and

(5) **aggregator**

(6) Error estimator

Simplified polygons: a compact representation

Mean Absolute Percentage Error (**MAPE**), a measure of prediction accuracy, for geo-statistic group-by queries (specifically 'mean' queries).

$$MAPE = {}^{100\%}/_{n} \sum_{i=1}^{n} \left| {}^{AC_i - P_i}/_{AC_i} \right|$$

Outline

- Geospatial big data analytics: Background and Motivating scenario
  - Motivating scenario
  - Spatial data challenges & requirements
- Approximating geospatial aggregate queries
  - Overview
  - ApproxGeoAgg
- Results and Discussion
  - Deployment: baselines & testing setup
  - ApproxGeoAgg Vs. baseline
- Summary & future research

UNIVERSITY OF SHARJAH

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA DIPARTIMENTO DI INFORMATICA - SCIENZA E INGEGNERIA

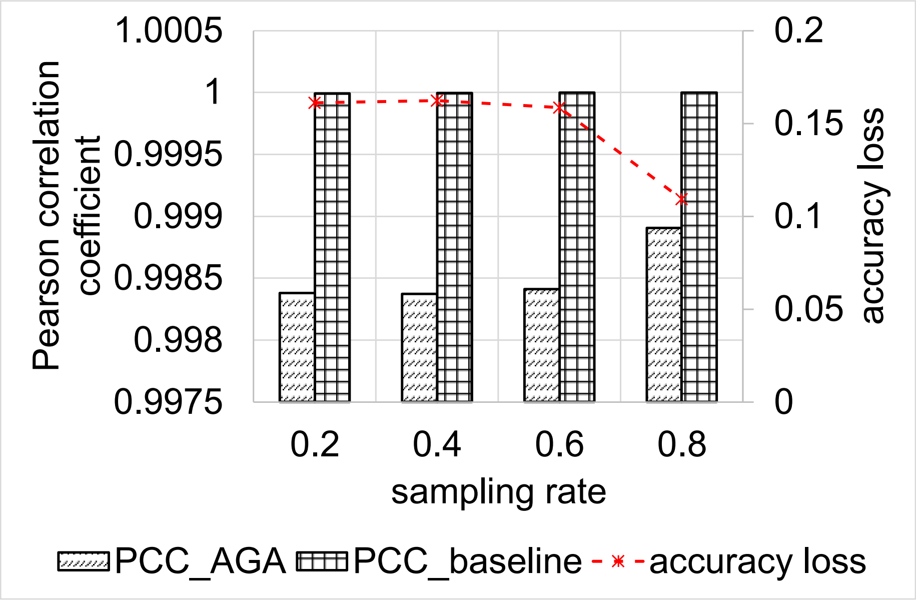
Experimental setup

- **Evaluation metrics**
  - Pearson Correlation Coefficient (PCC) for Top-N queries
  - Mean Absolute Percentage Error (MAPE) for geo-statistic group-by queries (specifically 'mean' queries)
- **Baselines**
  - Plain aggregator without the simplifier
- **Testbed**
  - We have deployed **ApproxGeoAgg** on a Microsoft Azure virtual machine hosting Python
  - **Datasets**
    - Vehicle mobility dataset
      - New York City taxicab trip datasets, USA
      - anonymized GPS coordinates (longitudes/latitudes) of taxi trips forming around one million and 1 Million and 400k tuples

Outline

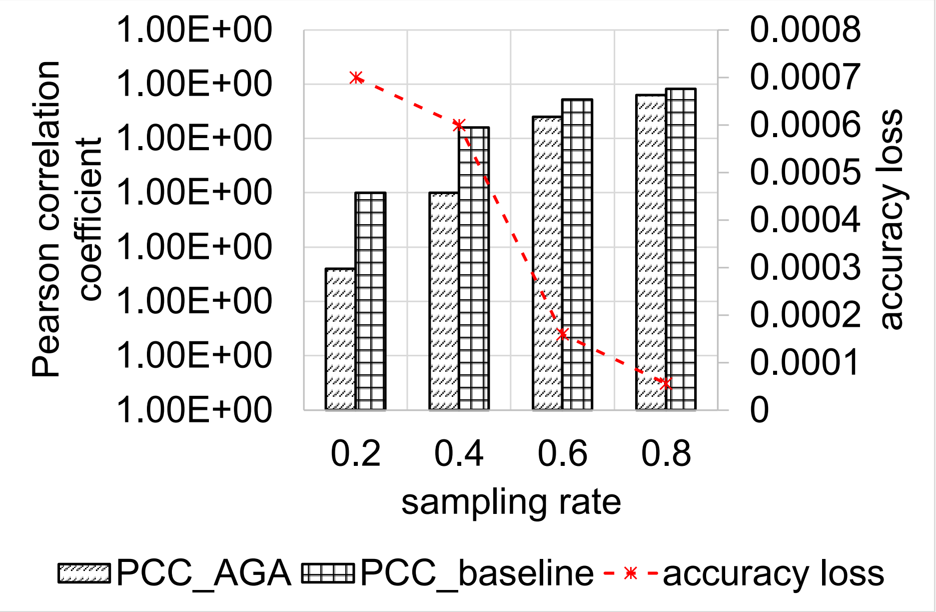
- Geospatial big data analytics: Background and Motivating scenario
  - Motivating scenario
  - Spatial data challenges & requirements
- Approximating geospatial aggregate queries
  - Overview
  - ApproxGeoAgg
- Results and Discussion
  - Deployment: baselines & testing setup
  - ApproxGeoAgg Vs. baseline
- Summary & future research

Pearson Correlation Coefficient (PCC) ApproxGeoAgg (AGA) **against** plain baseline,



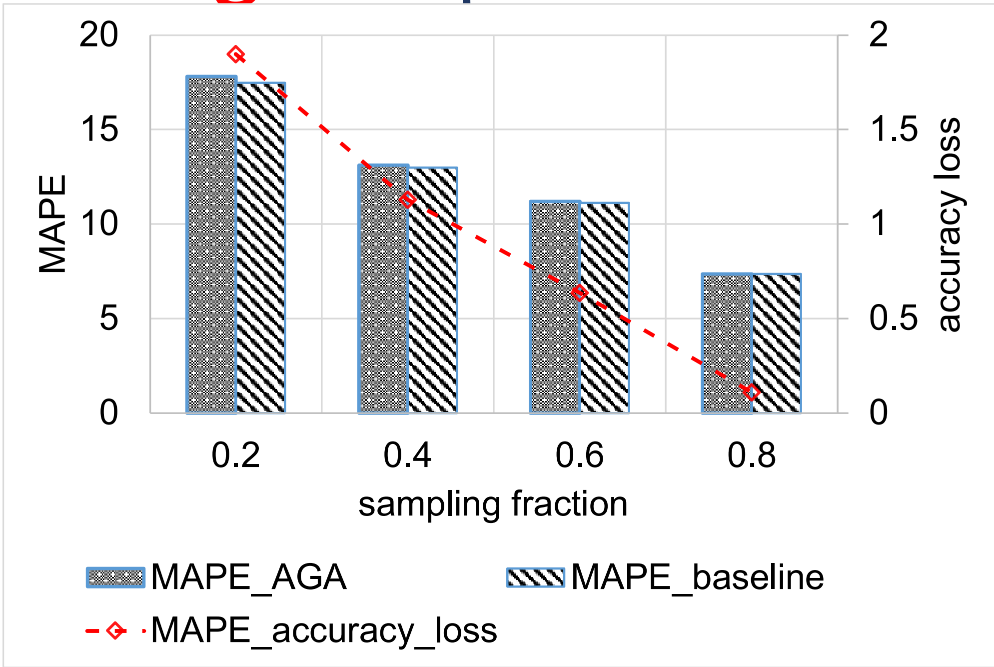
- measuring the performance on **Top-N** queries
- Geohash size 6, percentage of BPK 5%, NYC data
- **Varying** the geohash precision and sampling rate and
- **computing** PCC to test performance of both systems (AGA Vs. baseline)
- we obtain roughly a loss in **accuracy** that equals to 0.0147 %, on average

Pearson Correlation Coefficient (PCC) ApproxGeoAgg (AGA) **against** plain baseline



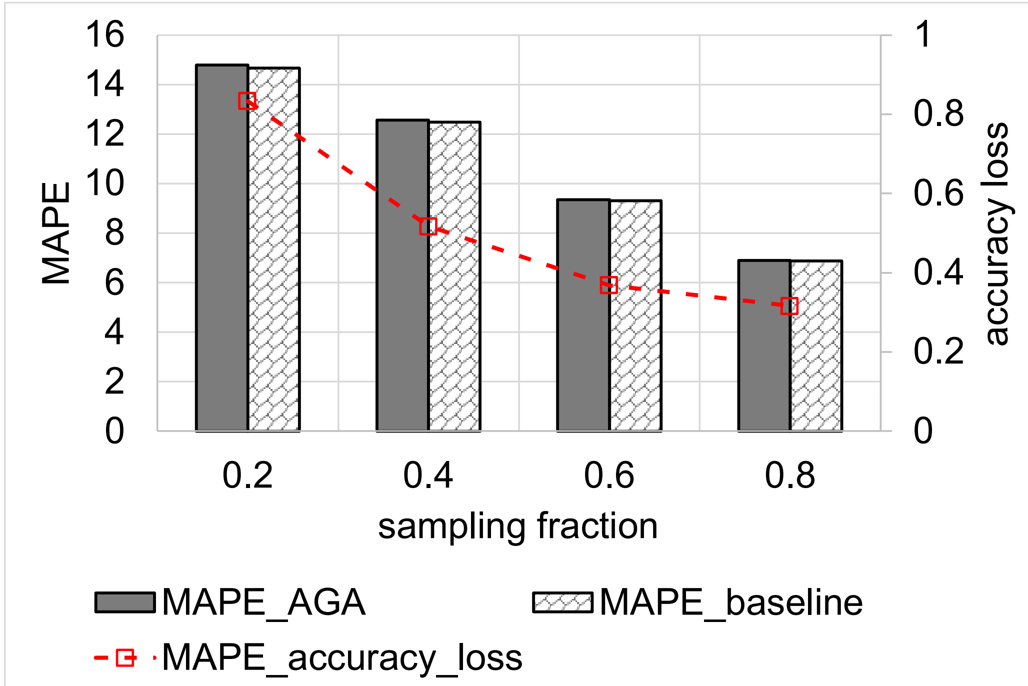
- measuring the performance on **Top-N** queries
- Geohash size 6, percentage of BPK 90%, NYC data
- **Varying** the geohash precision and sampling rate and
- **computing** PCC to test performance both systems (AGA Vs. baseline)
- Accuracy improves as we increase the 'percentage of BPK' to a generous 90%, where we obtain roughly 0.00038% loss in accuracy, extremely tiny and statistically insignificant

MAPE for ApproxGeoAgg (AGA) **against** plain baseline



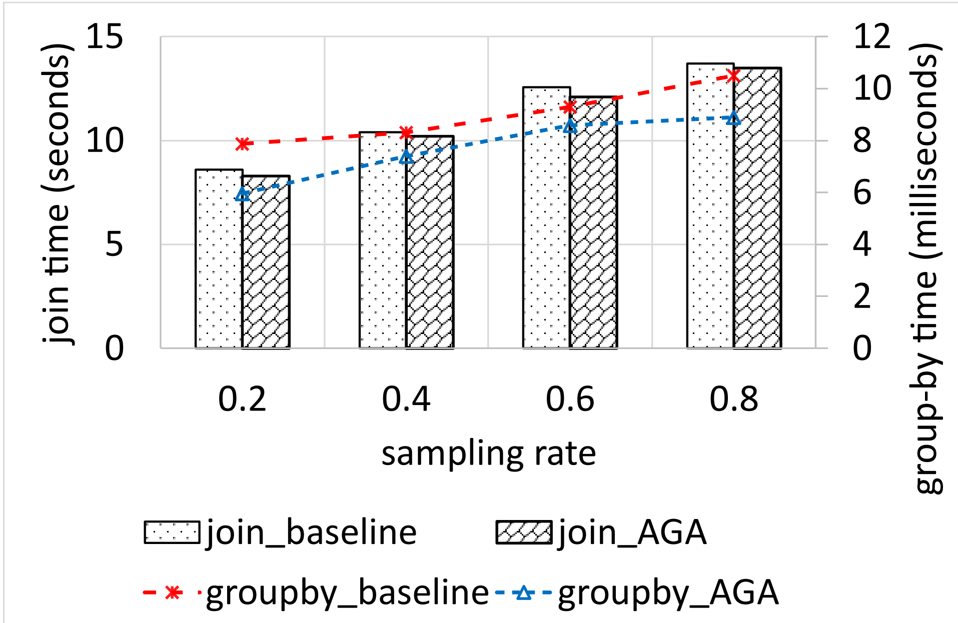
- measuring the performance on **geo-statistics** aggregate queries
- For a geohash precision 6 and 'percentage of BPK' that are equals to 5%
- **Varying** the geohash precision, the sampling fraction and the 'percentage of BPK' and
- **computing** MAPE to test performance of both systems (AGA Vs. baseline)
- The loss in accuracy equals roughly to 0.94%, on average

MAPE for ApproxGeoAgg (AGA) **against** plain baseline



- **Increasing** the ‘percentage of BPK’ to permissive **90%** and a geohash precision 6
- we obtain **higher** accuracy as the accuracy loss is on par with 0.43%, on average
- we obtain **higher** accuracy by either **increasing** the **sampling** fraction with same ‘percentage of BPK’ or increasing ‘percentage of BPK’ themselves

Spatial join and group-by running times for ApproxGeoAgg (AGA) against plain baseline



- For ‘percentage of BPK’ that equals 90% on geohash precision 6,
  - we obtain a gain in **running** time for the aggregation queries that equals to roughly 2.6%, on average
- We obtain higher gain for the same geohash precision with aggressive ‘percentage of BPK’ that equals to 5%, where we obtain a running time gain that equals to 12%, on average

Outline

- Geospatial big data analytics: Background and Motivating scenario
  - Motivating scenario
  - Spatial data challenges & requirements
- Approximating geospatial aggregate queries
  - Overview
  - ApproxGeoAgg
- Results and Discussion
  - Deployment: baselines & testing setup
  - ApproxGeoAgg Vs. baseline
- Summary & future research

Concluding remarks

- **ApproxGeoAgg** is a novel system devoted for smart city scenarios which require running geospatial **aggregate** queries over tremendous amounts of georeferenced data streams
  - Includes an adapted version of the **filter-refinement** approach for geospatial join processing
  - A front-stage filter, based on the **Douglas-Peucker** algorithm for reducing number of vertices polygons boundaries
- **Future research,** to investigate and test other methods for simplifying polygons **boundaries** and reducing the number of vertices of the boundary
  - Currently, an adapted version of Douglas-Peucker only

Q&A and Contacts

Thanks for your attention!

Question's time...

Dr. Isam Mashhour Al Jawarneh[1],

Dr. Rebecca Montanari[2],

Prof. Antonio Corradi[3]

[1] *Assistant Professor*, Department of Computer Science, University of Sharjah, UAE (ijawarneh@sharjah.ac.ae)

[2] Associate Professor, Department of Computer Science and Engineering – DISI, University of Bologna, Italy (rebecca.montanari@unibo.it)

[3] Professor, Department of Computer Science and Engineering – DISI, University of Bologna, Italy (Antonio.corradi@unibo.it)