**INTRODUCTION**

- **Background**
  - DBSCAN algorithm is one of the most widely used clustering algorithms.
  - The benefit is to capture arbitrary shape of clusters.
  - Key idea is to find dense region and to expand the regions in order to form clusters.
  - However, it is unlikely that a single machine support a typical size of big data.
  - Distributed processing has been adopted to increase the usability of DBSCAN.

- **Motivation**
  - Common procedures of existing parallel DBSCAN algorithms.

**OUR METHODOLOGY**

- **Key Idea**
  - Random Partitioning
  - It naturally solves the three limitations of region-based partitioning.

**EVALUATION METHODOLOGY**

- **Experimental Setting**
  - **Algorithms**
    - ESP-DBSCAN: even-split w. $\varepsilon$-approximation
    - CBP-DBSCAN: cost-based w. $\varepsilon$-approximation
    - RBP-DBSCAN: reduced-boundary w. $\varepsilon$-approximation
    - NG-DBSCAN: graph-based
    - RP-DBSCAN: proposed algorithm
  - **Real Data Sets**
    - GeoLife: 24,876,978 points, 3 dimensions
    - Cosmo50: 315,086,245 points, 3 dimensions
    - OpenStreetMap: 2,770,238,904 points, 2 dimensions
    - TeraClickLog: 4,373,472,329 points, 13 dimensions

**EVALUATION RESULT**

- **Efficiency Result**
  - **Efficiency Detail: Load Imbalance and Total Number of Points**

**ACCURACY RESULT**

- **Table 4: Accuracy of RP-DBSCAN in the Rand index**

**SIZE OF SUMMARY**

- The size was only ranging from 0.04% to 8.20% of the data set.

---

**REFERENCES**

- Hwanjun Song, Jae-Gil Lee

---

**RP-DBSCAN: A Superfast Parallel DBSCAN Algorithm Based on Random Partitioning**

Hwanjun Song†, Jae-Gil Lee‡*

† Graduate school of Knowledge Service Engineering, KAIST
‡ Corresponding Author

**Algorithm**

- RP-DBSCAN: proposed algorithm

**Datasets**

- GeoLife, Cosmo50, OpenStreetMap, TeraClickLog

**Performance**

- Efficiency
  - Load Imbalance
  - Total Number of Points

**Results**

- Accuracy
  - Rand index

**Summary**

- Size of the two-level cell dictionary.