Clustering: CLARA

Mitrica Octavian (IA-1B)

1. Importance and practical applications

CLARA (Clustering Large Applications) is a centroid-based clustering algorithm used for handling large sets of data.

It shares some similarities with K-means, but instead of using means as the center points, we switch to medoids. We will also use different types of distance functions instead of Euclidian as in means [1].

CLARA is a sampling-based approach to clustering, particularly developed to address the computational challenges associated with applying traditional clustering algorithms to large datasets. It is not a specific algorithm but rather an approach to combining sampling techniques with known clustering algorithms. So basically CLARA is a sampling version of PAM (Partitioning Around Medoids, or K-medoids).

When working with large datasets, it's essential to consider factors such as scalability, efficiency, and the ability to handle noise and outliers. CLARA is particularly useful in situations where the entire dataset cannot fit into memory or when computation time is a critical factor.

2. General presentation

CLARA is an extension of PAM, as mentioned earlier, that uses the sampling approach in order to process large datasets with a lower computational cost. Instead calculating the medoids for the entire data set, CLARA considers a sample with fixed size and applies the PAM algorithm to generate an optimal set of medoids for the sample [2].

Algorithm 4: CLARA

Inputs: X is a dataset having N points
k is the desired number of clusters
n is the chosen number of samples
s is the chosen size for each sample e.g. $40 + 2k$
Outputs: A is an N-vector of cluster assignments
C is a set of k cluster medoids
Procedure CLARA
for $i \leftarrow 1$ to n
do
$S_i \leftarrow \text{RandomSample}(X, s)$
$(a_i, c_i) \leftarrow \text{PAM}(S_i, k)$
end do
end for
$C \leftarrow \text{Best}(c_1, c_2, \ldots, c_n)$
$A \leftarrow \text{AssignPointsToNearestMedoid}(X, C)$
return A, C
end procedure

Fig1. Formal algorithm CLARA [3]

The above fig. is a formal presenation of the algorithm, as presented in [3]. As we can see, CLARA uses PAM, modifyig only the evidence assignation point and combining it with sampling at the top.

Below I tried to show a more friendly explanation of the algorithm.

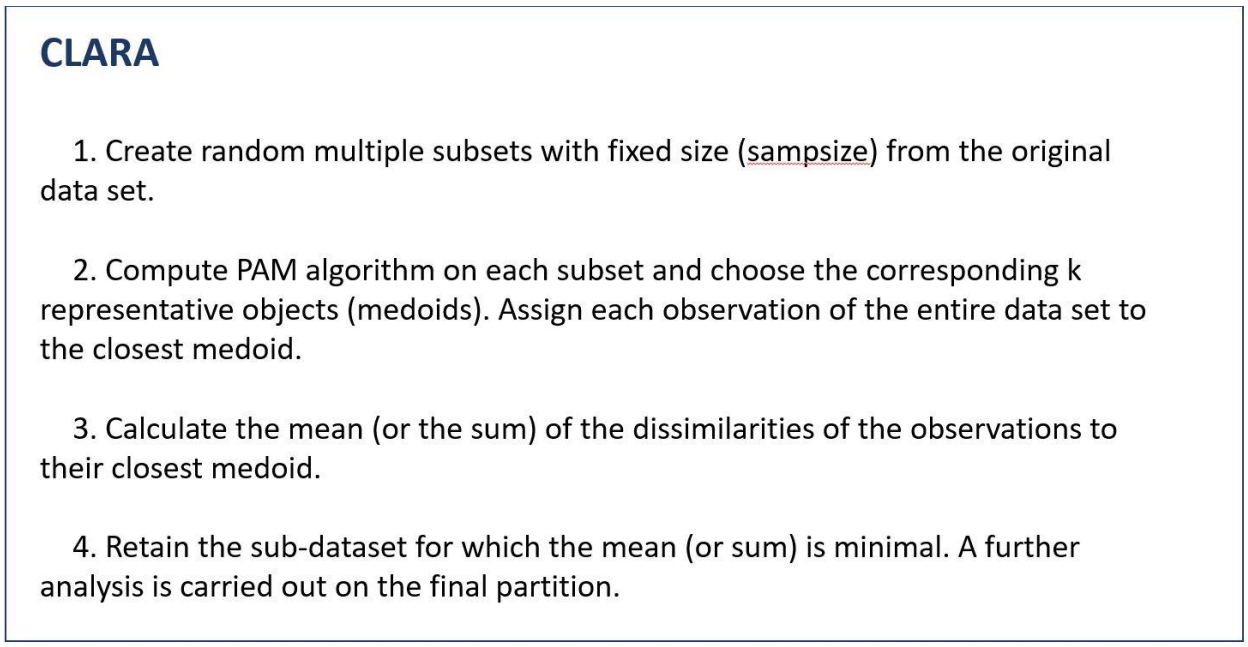

Fig1. CLARA algorithm explained

3. Known results and issues

Although it performs better than K-means, as we will show in the later experiments, and PAM also, CLARA leaves room for improvement as it is not the fastest. As seen in graph below, algorithms like K-means-lite and PAM-lite are faster once the data size increases [3].

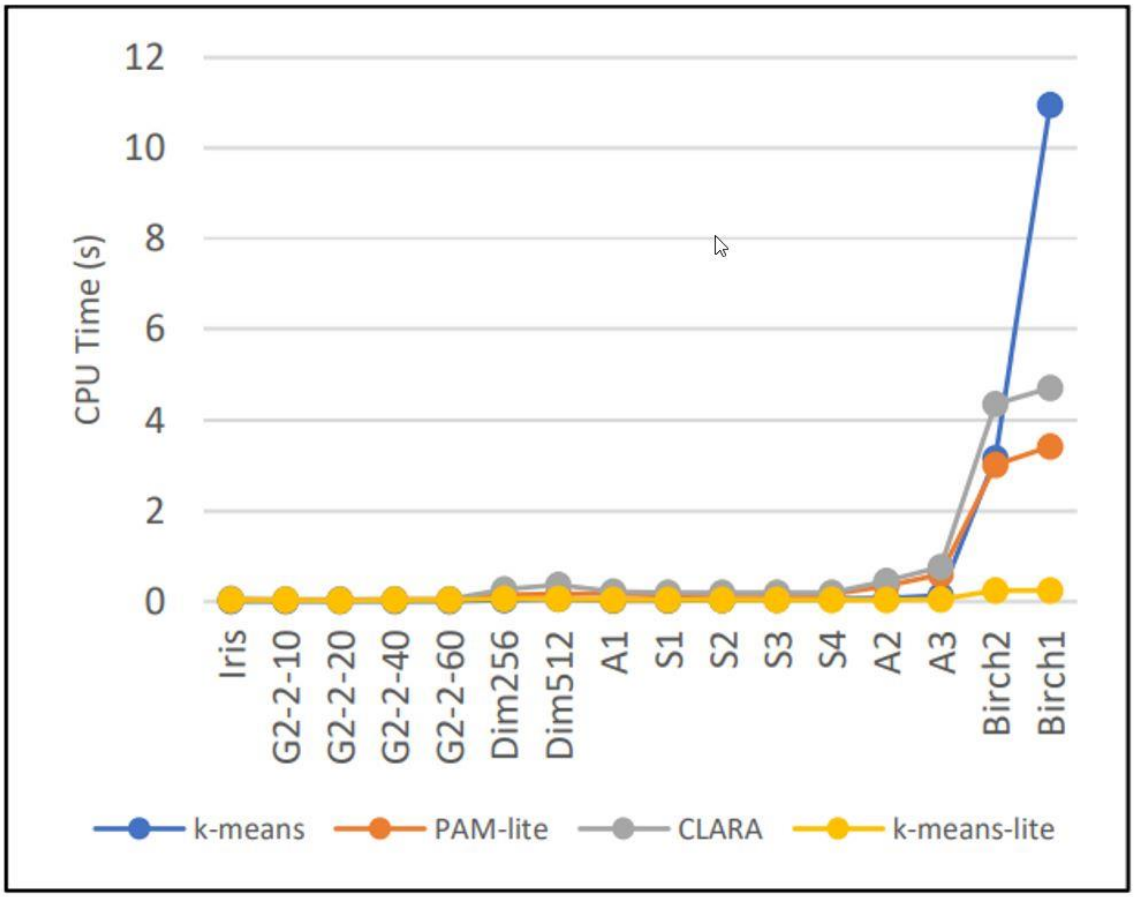

Fig2. CPU times comparison [3]

However, the algorithm may give wrong clustering results if one or more sampled medoids are away from the actual best medoids.

Furthermore, CLARANS takes care of the cons of CLARA, using random search [4]. There is also a way to make classic algorithms like CLARA and Kmeans faster by using a more relaxed 'SWAP' relaxation in the main function as shown in [5].

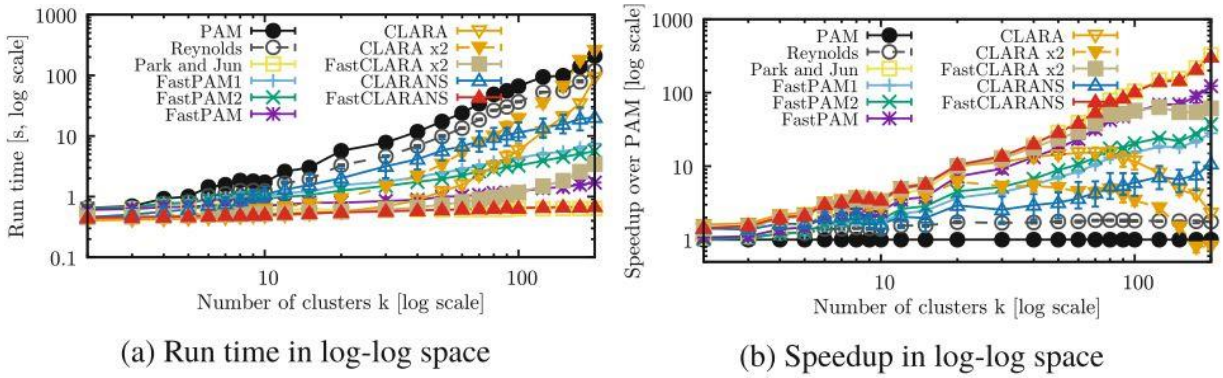

Fig3. CPU times between multiple algorithms and their 'Fast' versions [5]

4. Datasets used

For this experiments, I used 4 datasets of different sizes and shapes, in order to get a better look at how CLARA works and how it differs from Kmeans.

The datasets are open-source and have been downloaded from Kaggle mainly.

a. ***Cars*** – Demo purposes (50 x 2) – Link:
https://www.rdocumentation.org/packages/openintro/versions/1.7.1/topics/cars

b. ***US Violent Crimes*** – Small Dataset (1.35KB - 50 x 5) – Link:

https://www.kaggle.com/datasets/mathchi/violent-crime-rates-by-us-state

c. ***Adult*** – Medium Dataset (4MB - 32561 x 15) – Link:
https://www.kaggle.com/datasets/mlbysoham/adult-dataset

d. ***Ecommerce customer data*** – Big Dataset (115MB - 2019501 x 12) – Link:

https://www.kaggle.com/datasets/iabdulw/ecommerce-customer-data

5. Results

a. Demo

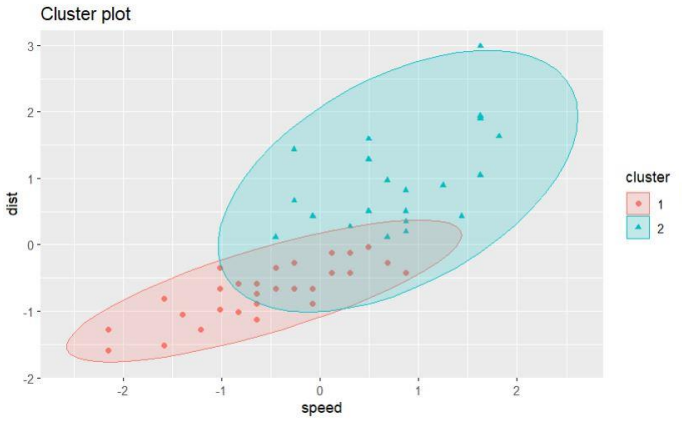

Fig4. Clara plot on 'cars'

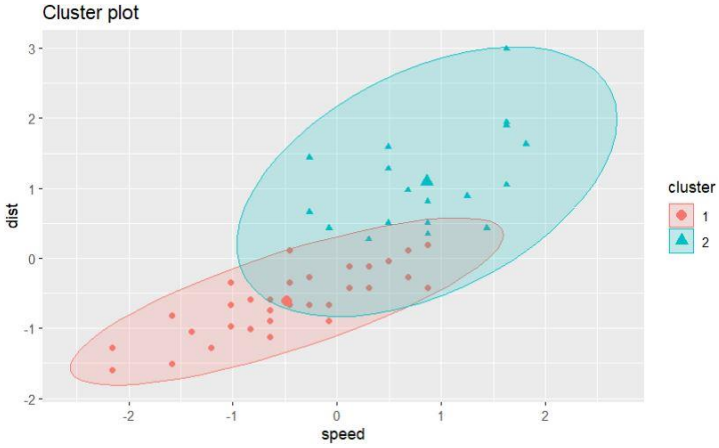

Fig5. Kmeans plot on 'cars'

b. Small Dataset

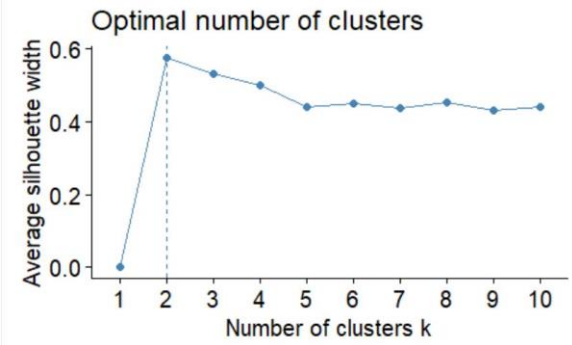

Fig6. Optimal number of clusters for Clara clustering on 'US Violent Crimes'

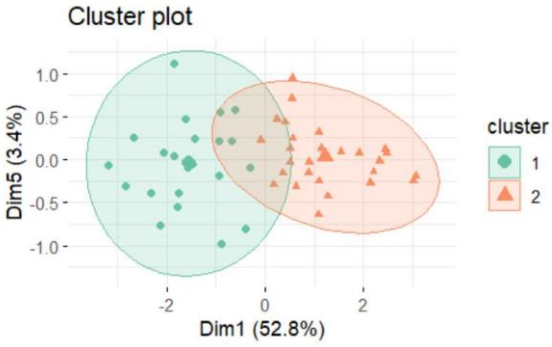

Fig7. Clara plot on 'US Violent Crimes'

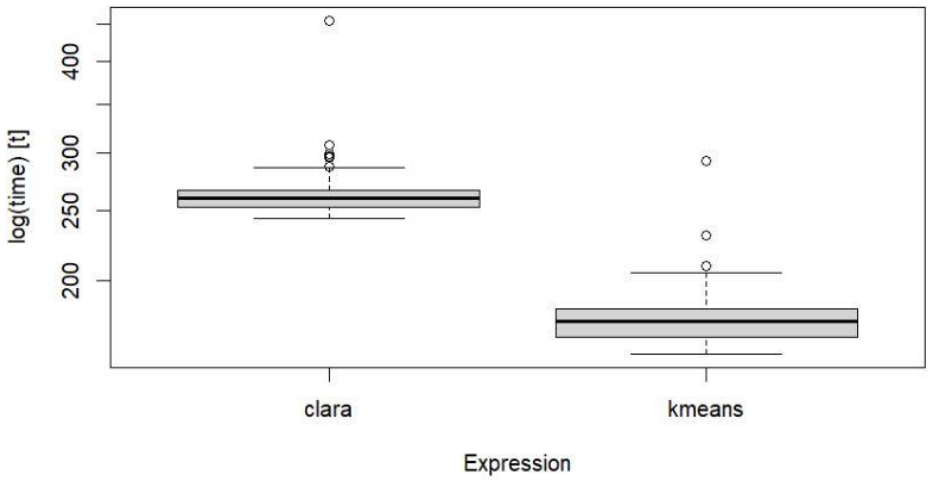

Fig8. CPU times Clara vs. kmeans

c. Medium Dataset

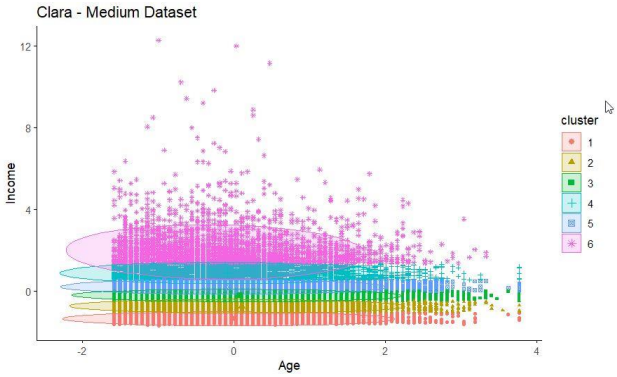

Fig9. Clara plot on 'Adult'

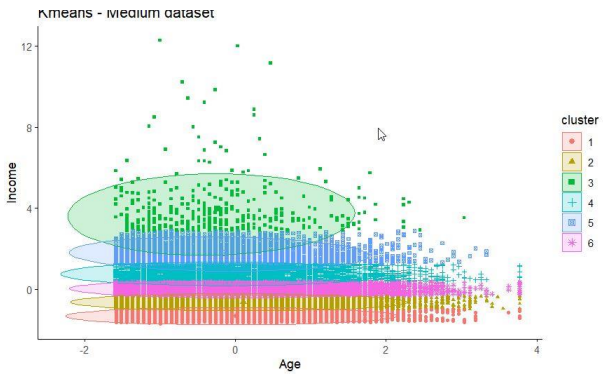

Fig10. Kmeans plot on 'Adult'

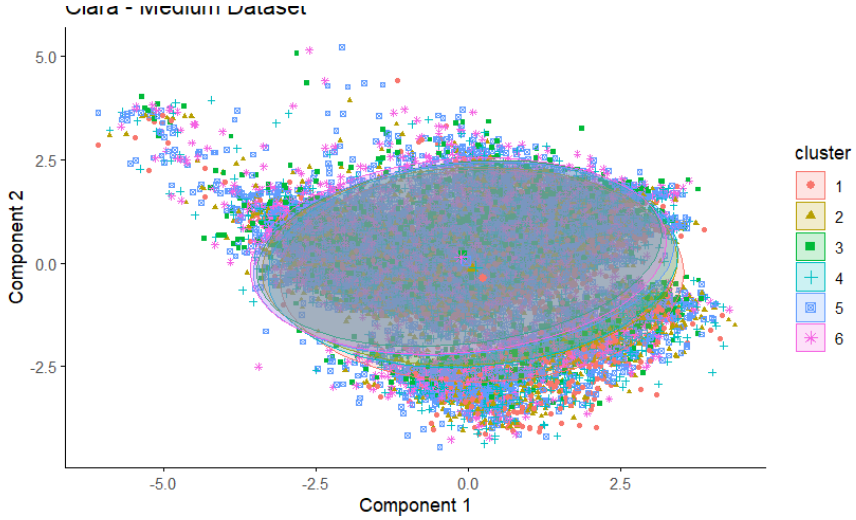

Fig11. Clara plot on 'Adult' with all rows

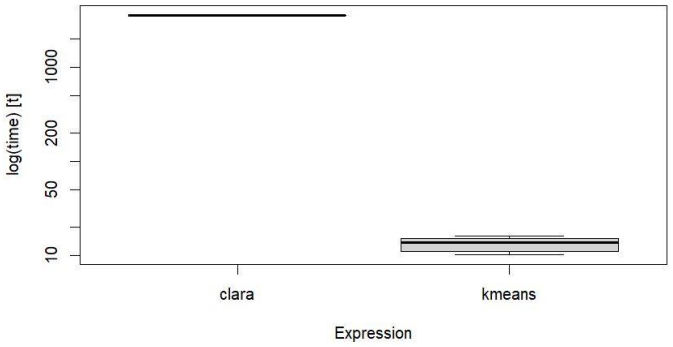

Fig12. CPU times clara vs kmeans on few rows

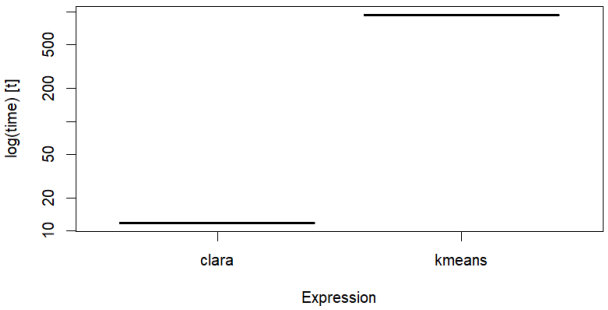

Fig13. CPU times clara vs kmeans on all rows

d. Big Dataset

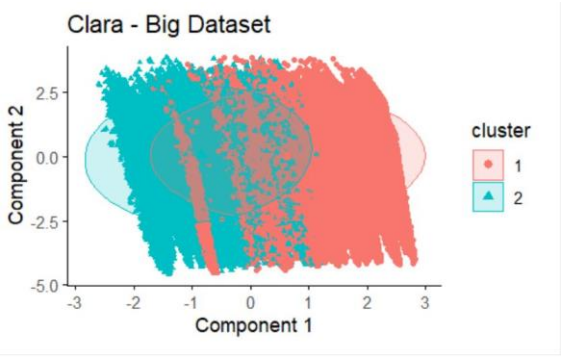

Fig14. Clara plot on 'Ecommerce'

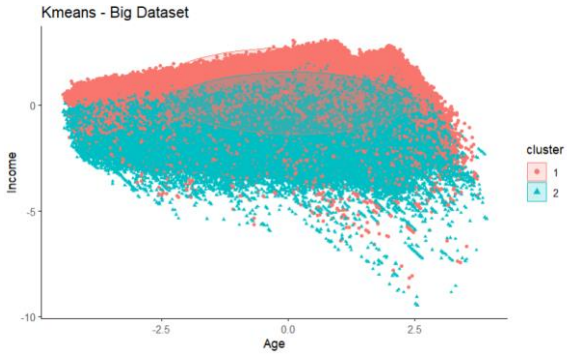

Fig15. Kmeans plot on 'Ecommerce'

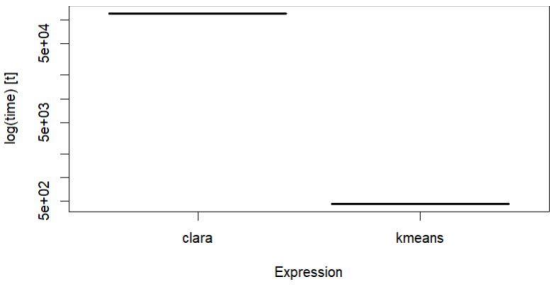

Fig16. CPU times clara vs kmeans with few rows

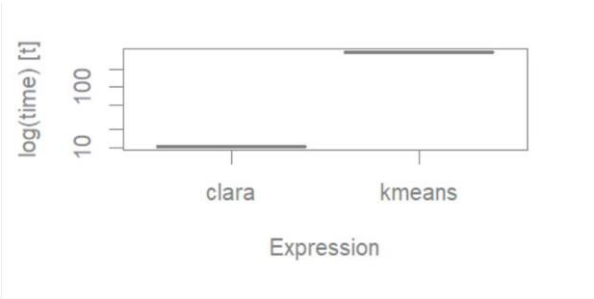

Fig17. CPU times clara vs kmeans with all rows

6. References

[1] Gupta, T., & P. Panda, S. (2019). A Comparison of K-Means Clustering Algorithm and CLARA Clustering Algorithm on Iris Dataset. International Journal of Engineering & Technology, 7(4), 4766–4768. https://doi.org/10.14419/ijet.v7i4.21472

[2] CLARA in R : Clustering Large Applications (https://datanovia.com/en/lessons/clara-in-r-clustering-large-applications/) - Accessed on 20.11.2023

[3] P. O. Olukanmi, F. Nelwamondo and T. Marwala, "Performance evaluation of sampling-based large-scale clustering algorithms," 2019 Southern African Universities Power Engineering Conference/Robotics and Mechatronics/Pattern Recognition Association of South Africa (SAUPEC/RobMech/PRASA), Bloemfontein, South Africa, 2019, pp. 194-199, doi: 10.1109/RoboMech.2019.8704854

[4] Comprehensive Guide To CLARANS Clustering Algorithm (https://analyticsindiamag.com/comprehensive-guide-to-clarans-clustering-algorithm/) – Accessed on 20.11.2023

[5] Schubert, E., Rousseeuw, P.J. (2019). Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms. In: Amato, G., Gennaro, C., Oria, V., Radovanović , M. (eds) Similarity Search and Applications. SISAP 2019. Lecture Notes in Computer Science(), vol 11807. Springer, Cham. https://doi.org/10.1007/978-3-030-32047-8_16