ARTIFICIAL INTELLECT TECHNOLOGIES BASED ON WEB DOCUMENT OBJECTS CLUSTERING PROGRESSIVE METRIC MODELS AND ALGORITHMS

Mominov B.B, Husanov Sh.A.

Tashkent University of Information Technologies named after Muhammad al-Khwarizmi

Abstract. *This in the article artificial intellect from technologies used without web document objects clustering progressive metric models and algorithms working The study was published in web of documents internal structure , semantic connections and link networks analysis made, their each other similarity level metric distance functions using is determined. Proposal done approach web documents rating automatic evaluation, semantic analysis deepening and information classification process optimization opportunity gives.*

Keywords: *artificial intelligence, clustering, web document, object, similarity, metric, distance function, algorithm.*

The two web document objects are respectively represented by two

metrics for measuring similarity in terms of features. For this, special cDist A library is created and its main function is as follows:

$$d = \text{cDist}(x_a, x_b, \text{metric}_{\text{name}}, \quad \text{outType}, \quad \text{rw})$$

Here x_a, x_b– feature vector of two matching objects , $\text{metric}_{\text{name}}$- metric type, - outTypesimilarity output type, rw- various settings.

The following distance calculation functions can be used for metric methods to calculate the distance between two features of these two objects : 'braycurtis', 'canberra', 'chebyshev', 'cityblock', 'correlation', 'cosine', 'dice', 'euclidean', 'hamming', 'jaccard', 'jensenshannon', 'kulczynski1', 'mahalanobis', 'match', 'minkowski', 'rogerstanimoto', 'russellrao', 'seuclidean', 'sokalmichener', 'sokalsneath', 'sqeuclidean', 'yule'.

The results of these metrics are presented in the form of a zoomed-in maritza.

All possible arguments to a metric object as additional parameters to this metric are:

p is the n -normal used for Minkowski , with weighted and unweighted values. The current value of the parameter is p= 2.

w - Weight vector for indicators that support weights

v - Standard Euclid for dispersion vector . The current value of the parameter

v = var(vstack ([XA, XB]), axis=0, ddof =1)

v_i - Mahalanobis for covariance matrix inverse . The current value of the parameter is : inv(cov (vstack ([XA, XB].T))).T

as a result M_b by M_a distance matrix is returned . For each i and j, the metric is calculated as follows

$$\text{dist}(u = XA[i], v = XB[i])$$

Some features and parameters of the proposed methods for calculating the similarity between two features of two objects have been developed:

1. $Y = \text{cdist}(XA, XB, 'euclidean')$ Uses Euclidean distance (2nd norm) as a measure of distance between features corresponding to a point and calculates the distance between points m . The points are in the matrix X mxn dimensional row vectors as is taken .

$Y = \text{cdist}(XA, XB, 'minkowski', p = 2.)$- Calculates distances using the Minkowski distance, where $\|u - v\|_p$ (p-norm)(p >0). (Note that this is only for quasimetrics, where 0< p <1).

2. $Y = \text{cdist}(XA, XB, 'cityblock')$- Points between cityblock or Manhattan distance counts .

3. $Y = \text{cdist}(XA, XB, 'seuclidean', V = None)$- Standard Euclid distance Two n - vectors she is between and standardized Euclid distance :

$$d = \sqrt{\sum (u_i - v_i)^2 / (x_i)}$$

V - dispersion vector ; V[i] - of points all i-components according to calculated dispersion . If not passed If it is , it is automatic. accordingly is considered .

4. $Y = \text{cdist}(XA, XB, 'sqeuclidean')$- vectors between $\|u - v\|_2^2$ Euclid distance square counts .

5. $Y = \text{cdist}(XA, XB, 'cosine')$- she is and v are vectors between cosine calculates :

$$1 - \frac{u \cdot v}{\|u\|_2 \|v\|_2}$$

6. $Y = \text{cdist}(XA, XB, 'correlation')$- she is and v are vectors between correlation calculates :

$$1 - \frac{(u - \bar{u}) \cdot (v - \bar{v})}{\|(u - \bar{u})\|_2 \|(v - \bar{v})\|_2}$$

$\bar{v}$ v is the mean of the vector

7. $Y = \text{cdist}(XA, XB, 'hamming')$ - Normalized Hemming distance or two n - vector she is and v to each other suitable unreachable vector of elements ratio calculates . In memory storage matrix X for logical kind of to be possible .

8. $Y = \text{cdist}(XA, XB, 'jaccard')$- Points Jaccard distance between calculates . Two vectors given she is and v, Jacquard the distance u[i] and v[i] from each other suitable unreachable of elements ratio .

9. $Y = \text{cdist}(XA, XB, 'jensenshannon')$- Calculates the Jensen-Shannon distance between two probability arrays. Given two probability vectors, p and q, the Jensen-Shannon distance

$$\sqrt{\frac{D(p \parallel m) + D(q \parallel m)}{2}}$$

Here m is the average of the points p and q, and D is the Kullback-Leiber divergence.

10. Y = cdist(XA, XB, 'chebyshev')- Calculates the Chebyshev distance between points. The Chebyshev distance between two n-vectors u and v is the maximum 1-norm distance between their corresponding elements. More precisely, the distance is given by:

$$d(u, v) = \max_i |u_i - v_i|$$

11. Y = cdist(XA, XB, 'canberra')- Calculates the Canberra distance between points. It 's two and the distance between point v and Canberra:

$$d(u, v) = \sum_i \frac{|u_i - v_i|}{|u_i| + |v_i|}$$

12. Y = cdist(XA, XB, 'braycurtis')- Calculates the Bray-Curtis distance between points. Two u The Bray-Curtis distance between point a and v is:

$$d(u, v) = \frac{\sum_i |u_i - v_i|}{\sum_i |u_i + v_i|}$$

13. Y = cdist(XA, XB, 'mahalanobis', VI = Noni)- Calculates the Mahalanobis distance between points. Two u The Mahalanobis distance between point a and v is:

$$\sqrt{(u - v)(1/V)(u - v)^T}$$

Here v $_i$ is the inverse covariance. If v $_i$ returns a value, v $_i$ is used as the inverse covariance matrix.

14. Y = cdist(XA, XB, 'yule')- Calculates the Yule distance between each pair of logical vectors.

15. Y = cdist(XA, XB, 'matching')- synonym for hammin g metric i

16. Y = cdist(XA, XB, 'dice')- Calculates the Dice distance between each logical vector pair.

17. Y = cdist(XA, XB, 'kulczynski1')- Calculates the kulczynski1 distance between each pair of logical vectors.

18. Y = cdist(XA, XB, 'rogerstanimoto')- Each logical vector couple between Rogers -Tanimoto distance counts .

19. Y = cdist(XA, XB, 'rusellrao')- Each logical vector couple Russell-Rao distance between counts .

20. Y = cdist(XA, XB, 'sokalmichener')- Each logical vector couple Sokal-Michener distance between counts .

21. Y = cdist(XA, XB, 'sokalsneath')- Each logical vector couple between Sokal-Snea z distance counts .

22. $Y = cdist(XA, XB, 'f')$- Calculates the distance between all pairs of vectors in X using a user-provided function f.

These distance metrics are widely used in extended models. The relationships between objects in web documents determine their similarity, and it is desirable to define these metrics in a basic way. It is necessary to combine all these mathematical expressions and create a tool library.

REFERENCES

1. Tan, PN, Steinbach, M., & Kumar, V. (2019). Introduction to Data Mining. Pearson Education.

2. Bishop, CM (2021). Pattern Recognition and Machine Learning. Springer.

3. Manning, CD, Raghavan, P., & Schütze, H. (2018). Introduction to Information Retrieval. Cambridge University Press.

4. Jain, AK, Murty, MN, & Flynn, PJ (1999). Data Clustering: A Review. ACM Computing Surveys.

5. Alpaydin , E. (2020). Machine Learning: The New AI. MIT Press.

6. Aggarwal, CC, & Zhai, C. (2022). Mining Text Data Using AI Algorithms. Springer.

7. Hu, D., & Tian, Y. (2021). A Comprehensive Survey of Clustering Algorithms. Annals of Data Science.

8. Leskovec, J., Rajaraman, A., & Ullman, JD (2020). Mining of Massive Datasets. Cambridge University Press.