A NOVEL META-LEARNING SYSTEM FOR CLUSTERING ALGORITHM RECOMMENDATION BASED ON META-FEATURES

[1]S.SRINIVAS REDDY, [2]Dr. SUNITA GOND
[1]Research Scholar, Dept. of C.S.E, Dr. A.P. J. Abdul Kalam University, Indore (M.P), India.
[2]Research Guide, Dept. of C.S.E, Dr. A.P. J. Abdul Kalam University, Indore (M.P), India.

ABSTRACT: Clustering a novel unsupervised problem usually necessitates prior knowledge of a factor that has a significant impact on the results achieved. Since clustering is one of the most special areas where Machine Learning (ML) algorithms are used, it is very important to understand high quality clustering algorithms. Trial and Blunder can be used to determine high-quality rules for modern datasets by experimenting with large algorithms with some bias, but this approach comes with high computational costs. With different biases, however this method has a significant computing cost. A unique meta-learning system based on meta-features is described in this research for clustering algorithm recommendation. This method may use meta-features to rank algorithms based on their suitability for a new dataset, and the resulting ranking can be used to successfully select the best ML algorithm for calculating the number of clusters in unsupervised data. To review and analyze datasets and extract a set of meta-features, this system employs a meta-learning method. These meta-features can then be used to intelligently select an effective clustering approach, eliminating the need to manually execute the method. The experiment can be carried out on datasets to identify expected results based solely on their meta-features for clustering algorithm selection utilizing meta-learning. After then the accuracy and runtime of the results are ranked individually.

KEYWORDS: Machine Learning (ML), Meta-learning, clustering algorithm, Recommendation and meta-features.

I. INTRODUCTION

Data mining techniques aimed at revealing potential knowledge patterns in datasets are currently used in areas ranging from genomics to big data applications [1].

Depending on the characteristics of the data addressed, these methods are categorized into two main groups: supervised and unsupervised. Each object in the dataset is linked to the output target variable of the monitored task. However, most of the information obtained from the actual data collection process, such as the information emitted by the sensor, is not monitored because it has no accompanying output [2]. Unsupervised methods are primarily used to discover structures or relationships between various input objects in a dataset [3]. Due to the lack of labelled data for these tasks, unsupervised algorithms are beneficial for situations where gathering labelled data is difficult, time consuming, or monetarily costly [4]. Clustering is the most commonly researched on unsupervised learning [5], despite the fact that there are several forms of unsupervised methods such as Anomaly detection or Hebb learning.

The purpose of clustering is to see if the data can be divided into different groups or clusters. The clustering technique divides a set of objects into clusters so that related things are grouped together and different objects are grouped together [6]. Proposing the right algorithms to solve a specific task is critical in Machine Learning (ML) in order to obtain a model with good predictive or descriptive power [7]. Each dataset has a unique set of properties that support various learning biases. The best algorithm for a new dataset can be found through trial and error or by leveraging the user's prior experience. These methods are arbitrary and can incur significant computational costs. Using predictive models for algorithm selection has proven

successful as a solution to these problems. This method additionally called auto ML or meta-learning, the usage of Machine Learning (ML) algorithms to study predictive Meta functions which is capable to recommend the best method for a new dataset [8].

As a result, this paper presents a comprehensive study of novel dataset characterization for the purpose of clustering algorithm suggestion employing ranking learning approaches based on multiple Meta features metrics [9]. A collection of meta-characteristics is defined for a particular number of datasets and their values are determined. Hence then this uses the datasets to apply some properties of clustering techniques to evaluate their performance, accuracy and execution time. These insights are reflected in neural networks that use the same algorithms to predict the performance and execution time of new datasets. Without having to run the algorithms, a recommendation for which approach would maximize performance and which would optimize the runtime which may be produced.

II. LITERATURE SURVEY

Edenhofer et al. [10] cluster data instances using k-means for a recommendation task. Four base algorithms are used for each data instance, and their performance according to their recall in a top-N set is noted. Each cluster is assigned a best-algorithm according to majority voting. They cluster new instances and identify a predicted best-algorithm according to the training data. Clustering is the least effective of all metalearning approaches (decision tree, gradient boosting, stacking using decision trees), and is 22.8% worse than the best base algorithm (content-based filtering using TF-IDF). Nechaev et al. [11] use the HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) algorithm to cluster user features and item features in 5 recommendation datasets. They identify the cost that is incurred when a meta-learner selects a non-optimal algorithm. Their meta-learner therefore chooses between the predicted-best algorithm per-user, and the overall-best algorithm, according to a utility function that factors in errors caused by their meta-learner.

"Hierarchical Clustering Procedures in Data Mining" was designed by Z. Abdullah .et .al in 2015. The introduced method generates a solution by assigning each point to its own cluster and selecting and merging pairs of clusters until only one cluster remains. The method used to determine the pair of clusters to be merged at each stage is important in the agglutination method. The efficiency of the suggested different width cluster approach [12] is demonstrated by experimental results acquired on synthetic and actual datasets. The term "meta-learning" was originally used in the field of psychology. There are numerous definitions available. By examining existing algorithms and meta-learning, Zeng Zilin.et.al [13] On the basis of meta-learning ideology, a unified framework for algorithm selection was introduced. This framework is useful for computer science, Artificial Intelligence, operations research, and statistics. Keep in mind that Machine Learning, bioinformatics, other sciences, and metal learning all have a lot of capacity for development.

In all related work mentioned so far, meta-learning and clustering are used in predicting a single algorithm for future datasets (e.g., Lee and Giraud-Carrier.et.al [14] use 85 datasets, and therefore have 85 instances for cross-validation of their meta-learner). Per-instance meta-learning is not common for recommendation systems. Meta-learning is based on learning results and subsequent learning, or a large amount of learning to get at the end result [15]. Meta-learning

is, in a nutshell, learning about learning. The term "Meta-Learning" became initially used with inside the area of Machine Learning in 1992 to explain the mapping among the different techniques of machine learning processes (meta-algorithms) and relative measurements will set the rules for performance. Form the learning method for meta-knowledge.

III. A NOVEL META-LEARNING SYSTEM FOR CLUSTERING ALGORITHM RECOMMENDATION

The methods used in this investigation are outlined in this part, beginning with data pretreatment and feature extraction. This system of training and timing every set of rules in every dataset, in addition to documenting outcomes, continues. The neural network chooses a fixed k training datasets which are the maximum similar to the new dataset in terms of the meta-learning environment`s facts properties (meta-function). After you've chosen your datasets, you'll need to combine their rankings. The Average Ranking method can be utilized for this method. Consider the subsequent definition which is great advantage to a higher of the Average Ranking method. Let $r_j=(r_{(1,j)},\ldots,r_{(i,j)},\ldots,r_{(k,j)})$ be the rating of algorithm j (j = 1, . . . , a) for k datasets, wherein a is the range of algorithms. For each algorithm j, the average rank r_j is calculated as follows:

$$\bar{r}_j = \frac{\sum_{i=1}^{k} r_{i,j}}{k}$$

The ranking is obtained by reassigning the rank places with the lowest value of r_j being assigned first and the highest value being assigned last. After that, the data is visualized and evaluated.

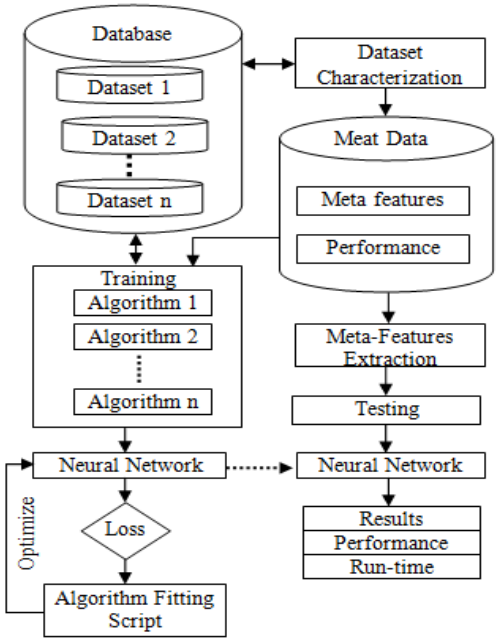

Fig. 1: ENTIRE PROCESS OF META-LEARNING BASED CLUSTERING ALGORITHM RECOMMENDATION SYSTEM

Figure 1 shows a diagram of the entire process of meta-learning system based clustering algorithm recommendation. The dataset characterization method is carried out from database comprising several datasets, and the meta-functions are calculated. Algorithms are run to decide their overall performance. Metadata is used to keep each meta-functions and performances. Meta-learning may be completed the usage of metadata with the aid of using and combining the meta-characteristics of every dataset with the overall performance of every set of rules. You can use neural network generation to indicate scores primarily based totally on saved meta-characteristics, new datasets, and set of rules based on overall performance.

3.1 Dataset Characterization

The meta-functions utilized in this method are based on Open ML datasets that cover a wide range of topics, such as medicine, biology, climate, and social issues. The procedure begins by normalizing all values from 0 to 1. Then find and delete the column that is the calculation, or a column that is exactly unique to the other columns. If two columns are linear combinations of each other, they are said to be collinear. When performing multi-variate analysis, this will result in errors; hence all such columns must be eliminated. It's vital to note that eliminating these columns will have no effect on this model's predictive power because they're effectively duplicate characteristics. If there are one or more small values that can be rounded to zero, the set is considered computationally singular. That is, it is a singular matrix. It is not possible to perform a set of algorithms because the singular matrix is not reversible. Caret includes several functions to pre-process the predictor data. It assumes that all of the data are numeric.

3.2 Meta data

In this context, you should use data called metadata that describes the performance and problem properties of the algorithm. Data performance is important because it is used to calculate the ranking of the algorithm. The property in question is called a meta-function and represents an aspect of a particular task. These meta-features do not require you to extract class information from issues related to clustering algorithm recommendations. Meta-learning involves building a database of meta-knowledge and mapping meta-characteristics to the relative performance of an algorithm in the context of a Machine Learning method.

3.3 Meta-Features Extraction

Meta-feature characterization's major goal is to capture a dataset's differentiating qualities and utilize that information to categories additional datasets that are comparable. To make intelligent recommendations, this work will primarily rely on dataset meta-features. As a result, the features selected and how they are calculated becomes important. This work is based on previous metrics by integrating 6 statistical features with 19 distance-based metrics. Each record is represented by a vector of 25 elements.

Statistical-Based Meta-features: These are the dataset's macro-level observations. You can assess the data's normality, variance, and overall distribution, as well as quantifying information such as the size of the dataset (both the number of entries and the number of parameters for each entry). These features give a basic understanding of each dataset's size, quality, and behavior.

1) Total Number of Entries (NE): NE=n, where n is the total number of entries. This is the size

of the dataset.
2) Number of Entries per Attribute (NEA): NEA=n/p, where n is the number of entries and p is the number of attributes. This metric indicates how robust the dataset is or how descriptive it is.
3) Missing Values Percentage (PMV): PMV=m/t.100, where t denotes the total number of entries and m denotes the number of missing entries. It assesses the dataset's consistency.
4) Multivariate Normality (MN): A measure of a dataset's similarity to its normal distribution. This value is computed using R's MVN package and Royston's technique.
5) Skewness (SK): A measure of how far to the left or right a distribution has been pushed. This statistic assesses the asymmetry of the dataset. This value is computed to determine multi-variate skewness using R's MVN package and Mardia's Test.
6) Percentage of Outliers (PO): PO= o/t.100, where t is the number of entries flagged as outliers and t is the total number of entries.

Distance-Based Meta-features: To calculate the pair wise Euclidean distance between entries (rows). The following formula is used to calculate the distance d between entries I and j of dataset X with n entries described by the p variable.

$$d(X_i, X_j) = \sqrt{\sum_{c=1}^{P}(x_{i,c} - x_{j,c})^2}$$

Create a vector of size n(n−1)/2 listing all pair wise distances:

$$d = [d_{1,2}, d_{1,3,} d_{1,4}, \dots, d_{2,3}, d_{2,4}, \dots, d_{n-1,n}]$$

Min-Max to normalise the vector on the interval [0, 1], feature scaling is used.

3.3 Training Algorithms

Each dataset has been normalized to the range [0, 1]. A well-known approach to dealing with an uncertain number of clusters is to make the number of clusters equal to the number of instructions contained within the dataset. This method is used for algorithms that require a specific range of clusters. As figuring out the right range of clusters is a tough challenge in and of itself. Choosing too many clusters could make effects extra complicated, whilst deciding on too few clusters can result in facts loss and over-generalization. The Scikit-study package deal in Python is used to run 8 algorithms. Each algorithm will be judged on its performance as well as its execution time.

Table 1: CLUSTERING ALGORITHMS IN TRIANING

Label	Algorithm
AA	Average Agglomerative
CA	Complete Agglomerative Clustering
FCM	Fuzzy C-Means
GMf	Gaussian Mixture with full matrix
KM	K-Means
GMd	Gaussian Mixture with diagonal matrix
KKM	Kernel K-Means
MK	Mini Batch K-Means

3.4 Neural Net Training/Testing

Following the introduction of the metadata with the meta-functions and set of rules have overall performance and able to know device is capable of gathering statistics from this metadata which must be used. This form of getting to know device is referred to as a meta-leaner. Its purpose is to take a hard and fast of those meta-functions as enter and examine from them so as to be expecting the overall performance of algorithms for brand new troubles and the use of meta-learning method. The method of ML strategies may be used to rank clustering algorithms. For this method, the k-Nearest Neighbors set of rules may be used. The k-NN set of rules reveals datasets which can be the maximum much like the only at hand. A set of meta-functions is used to calculate the gap among datasets. The k-NN set of rules is an instance-primarily based totally getting to know set of rules, because of this that it's miles a lazy-getting to know set of rules due to the fact that the induction or generalization method is behind schedule till classification. During the education phase, lazy-getting to know algorithms require much less computation time than eager-getting to know algorithms.

Then, using leave-one-out cross validation, two neural networks are created. The meta-feature and the one-shot coding of the desired algorithm are used as both inputs. The result is one performance prediction and the other runtime prediction.

3.5 Algorithm fitting script

There are 33 characteristics in each of the 1080 input tensors (135 datasets, 8 algorithms). Each time the neural network is tested on the held out set, it is trained with 134 datasets. Because leave one out cross validation has a significant variance, ten iterations are done and the average is obtained, with the variance kept within a reasonable range. As mentioned earlier, the process of designing hidden layers in neural networks is a fairly inaccurate science. The runtime network is set up with three hidden layers of sizes 32, 24, and 8 after rigorous testing and tuning based on feedback and data from test runs. The Leaky activation function is used in the first two hidden layers. Sigmoid activation is used in the third hidden layer. As Sigmoid creates output in the range, it cannot produce negative values by definition (0, 1). Tanh activation is used in the power neural network's three hidden layers, which are 32, 24 and 16 in size. Manually start your network training and keep track of the average training and test losses after each run. Then, using a set of weight loss and learning rate values, keep track of each weight loss value. Eight numbers [106, 105, 104, 0.001, 0.01, 0.1, 1, 10] are chosen for learning rate, and five for weight decay [104, 0.001, 0.01, 0.1, 1]. Despite the fact that they should be similar given the same settings, both training and testing losses are recorded.

IV. RESULTS

The process of analysing the results of each dataset in aggregate is covered here. After all numerical values for each clustering algorithm that were used during training are ranked, utilize this information to calculate the accuracy rate and provide visualizations to determine the best recommendations for your clustering algorithm. In this case, eight clustering algorithms were selected as the best clustering algorithm recommendations to provide results from different perspectives. Their names and abbreviations are listed below: Fuzzy CMeans (FCM), Gaussian Mixing with Diagonal Matrix (GMd), Gaussian Mixing with Complete Matrix (GMf), Kernel KMeans (KKM), KMeans (KM) and mini-batch K Means (MK). Figure (2) depicts the order of each clustering. As previously stated, the AA (Adaptive Additive) algorithm is highly

recommended as the best because it is ranked first, whereas the Gmf (Gaussian Mixture Model) algorithm is rarely recommended because it is ranked eighth.

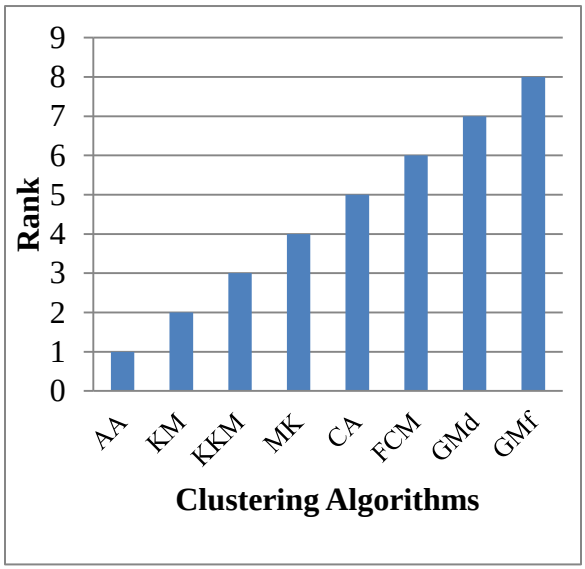

Fig. 2: RANKING OF EACH ALGORITHM FOR RECOMMENDATION

The mean, standard deviation, and rank of each clustering algorithm are shown in Table 2.

Table 2: MEAN, STANDARD DEVIATION OF EACH CLUSTERING ALGORITHM

Algorithm	Mean	STD
AA	2.8173	2.2459
CA	5.6940	2.6625
FCM	5.9566	2.1425
GMd	6.8789	2.3535
GMf	9.2602	1.4737
KKM	4.5639	2.1481
KM	2.9748	1.4929
KMD	6.8881	2.2675
MK	4.5821	1.9249
WA	5.3835	2.2060

Performance Data: To calculate performance (accuracy), ten clustering metrics are used. Because this is not monitored, an internal index is used to assess performance. In other words, the quality of the clustering structure is determined by the features present in the dataset. Because each metric has its own scale and target, the data must be normalized and averaged to guarantee that each of the ten metrics is equally weighted. After the neural network produces all ten results, the join and averaging procedures are executed. To make all measures equal, invert metrics with a minimization goal by multiplying by 1. The objective is to compare and contrast

eight algorithms. The numerical results are unimportant as long as all eight are consistent, and you can rate them.

Runtime Data: The exact time required to train each set of rules to calculate runtime is measured using a dedicated CPU (Intel Xeon E5-1603 V3 @ 2.80GHz, four cores, four threads, 8GB RAM running Ubuntu 18.04). The device has no community connection and only some concurrent strategies to dispose of any unrelated factors. Because the runtime of a set of rules may be stimulated via way of means of how efficaciously a package deal is implemented, the scikit-analyze Python package deal is used for all to make sure consistency. Perform a total of 10 runs, averaging, and minimizing the variance of each run. If the run-time results differ significantly, you need to address external conditions. As indicated in Table 3, the anticipated and actual values obtained during the run are used to rank the performance and run-time results.

Table 3: RESULT OF TOP 3 RANKED ALGORITHMS ON A SINGLE DATASET

Parameter		AA	KM	KKM
Performance	Predicted	1	2	2
	Actual	3	1	3
Runtime	Predicted	1	3	4
	Actual	1	4	3

It is clear from this that the predicted topper forming algorithm was AA (Adaptive Additive); it was the third-best performance. Mean aggregation was the projected top runtime algorithm (red), as well as the actual top runtime algorithm. Because the goal of this project is to find the best algorithm for each goal, the first three outcomes have been highlighted. Figure 3 compares the best algorithm predictions with the actual results obtained by running the algorithm. I would like to think that the chart with the 1st arrangement peaks at the 1st and the chart with the 2nd arrangement peaks at the 2nd.

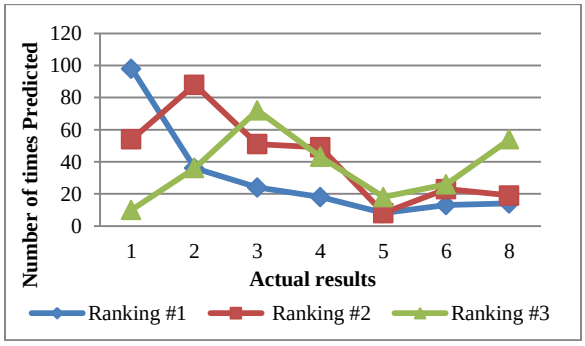

Fig. 3: PERFORMANCE PREDICTION RESULTS

V. CONCLUSION

In this paper, a meta-learning system is presented using Meta features to intelligently recommend clustering algorithms. Each meta-definition feature's and calculation process is described. The statistical and distance-based methods are both used. The datasets were characterized using meta characteristics. The performance of eight clustering techniques was calculated using performance and runtime validations. The experiments then consist of evaluating a recommended system in order to recommend the best algorithm based on a ranking method using a meta-knowledge database. In terms of runtime, our meta-learning system predicted the best algorithm. With a

performance accuracy of about 85%, it was able to select one of the top two methods. This system was successful in recommending one of the top three algorithms with high performance values. Moving the suggestion process further back in the AutoML chain could be a specializeda strategy. While it can provide recommendations for which algorithm to employ, changing and creating neural networks is still a trial and error process.

VI. REFERENCES

[1] Jidapa Thadajarassiri, Cansu Sen, Thomas Hartvigsen, Elke Rundensteiner, "Learning Similarity-Preserving Meta Embedding for Text Mining", 2020 IEEE International Conference on Big Data (Big Data), 2000

[2] Gabriel Maicas, Cuong Nguyen, Farbod Motlagh, Gustavo Carneiro, "Unsupervised Task Design to Meta-Train Medical Image Classifiers", IEEE 17th International Symposium on Biomedical Imaging (ISBI), 2020

[3] S. D. L. H. Maheeshanake, M. W. A. R. Weerasinghe, M. A. I. Perera, "Unsupervised Techniques for Meta-Analysis of Cancer Genomic Data", 20th International Conference on Advances in ICT for Emerging Regions (ICTer), 2020

[4] T. Zoppi, A. ceccarelli, T. Capecchi and A. Bondavalli, "Unsupervised anomaly detectors to detect intrusions in the current threat landscape", *arXiv:2012.11354*, 2020,

[5] T. Zoppi, A. Ceccarelli, L. Salani and A. Bondavalli, "On the educated selection of unsupervised algorithms via attacks and anomaly classes", *J. Inf. Secur. Appl.*, vol. 52, Jun. 2020.

[6] .B. A. Pimentel and A. C. de Carvalho, "A new data characterization for selecting clustering algorithms using meta-learning", *Information Sciences*, vol. 477, pp. 203-219, 2019.

[7] A. R. Ali, M. Budka and B. Gabrys, "A Meta-Reinforcement Learning Approach to Optimize Parameters and Hyper-parameters Simultaneously", *PRICAI 2019: Trends in Artificial Intelligence*, pp. 93-106, 2019.

[8] Z. C. Xiao and Mengchu Zhou, "Learning Locomotion Skills via Model-based Proximal Meta-Reinforcement Learning", *2019 IEEE International Conference on Systems*, pp. 1545-1550, 2019.

[9] Mitchell Wortsman, Kiana Ehsani, Mohammad Rastegari, Ali Farhadi and Roozbeh Mottaghi, "Learning to learn how to learn: Self-adaptive visual navigation using meta-learning", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6750-6759, 2019.

[10] Edenhofer, G.Collins, A Aizawa, A. and J. Beel, "Augmenting the DonorsChoose.org Corpus for Meta-Learning", 1st Interdisciplinary Workshop on Algorithm Selection and Meta-Learning in Information Retrieval (AMIR), (2019).

[11] Nechaev, A., Zhukova, N. and V. Meltsov, "Utilizing Metadata to Select a Recommendation Algorithm for a User or an Item", 11th Majorov International Conference on Software Engineering and Computer Systems (MICSECS 2019), (2019)

[12] Z. Abdullah et al "Hierarchical Clustering Algorithms in Data Mining" World Academy of Science, Engineering and Technology International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol:9, No:10, 2015.

[13] Zilin Zeng, Hongjun Zhang,Rui Zhang, Zhiteng Wang, "Summary of algorithm selection based on meta learning thought. Control and Decision", 2014, 29 (6): 961-968.

[14] Lee, J.W. and Giraud-Carrier, C., "Automatic selection of classification learning algorithms for data mining practitioners". Intelligent Data Analysis. 17, 4 (2013), 665–678.

[15] Prodromidis, Andreas, Philip Chan, and Salvatore Stolfo. "Meta-learning in distributed data mining systems: Issues and approaches", Advances in distributed and parallel knowledge discovery 3 (2000): 81-114.