Text Clustering Method Based on Improved CFSFDP Algorithm and Its Applications Post-print

Authors: Zhan Chunxia, Wang Rongbo, Huang Xiaoxi, Chen Zhiqun

Date: 2017-11-08T00:00:00+00:00

Abstract

Objective: To address the issue that the CFSFDP (Clustering by Fast Search and Find of Density Peaks) algorithm produces unsatisfactory clustering results due to its utilization of the product of local density and distance for cluster center selection. Method: This paper proposes a CFSFDP algorithm based on particle swarm optimization, which employs particle swarm optimization to seek optimal thresholds for local density and distance in the CFSFDP algorithm, thereby obtaining cluster centers characterized by relatively high local density and distance values and reducing the impact of outlier points on cluster center selection, with experiments conducted on datasets randomly selected from a question bank of examinees provided by a college entrance examination consultation platform. Results: Experimental results indicate that across different datasets, the proposed algorithm exhibits significant improvements in accuracy, recall, and F-value compared to the basic CFSFDP algorithm. Limitations: Semantic relationships were not considered in the text processing stage. Conclusion: The proposed method demonstrates effective clustering performance, and its application to the college entrance examination consultation database can effectively reduce the workload of consultants and facilitate rapid responses to examinees' inquiries.

Full Text

Preamble

ChinaXiv Cooperative Journal

Text Clustering Method Based on Improved CFSFDP Algorithm and Its Application*

Zhan Chunxia, Wang Rongbo, Huang Xiaoxi, Chen Zhiqun
(School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China)

Abstract

[**Objective**] This paper addresses the unsatisfactory clustering results of the CFSFDP (Clustering by Fast Search and Find of Density Peaks) algorithm, which selects cluster centers based on the product of local density and distance. [**Methods**] We propose a CFSFDP algorithm based on particle swarm optimization that searches for optimal local density and distance thresholds to identify cluster centers with relatively high local density and distance, thereby reducing the influence of discrete points on center selection. The proposed method was evaluated on randomly selected datasets from a college entrance examination consultation platform' s question database. [**Results**] Experimental results demonstrate that our algorithm achieves significant improvements in accuracy, recall, and F-measure compared to the basic CFSFDP algorithm across different datasets. [**Limitations**] The semantic relationships were not considered during text processing. [**Conclusions**] The proposed method demonstrates effective clustering performance and can substantially reduce the workload of consultation staff while helping to quickly answer candidates' questions when applied to the college entrance examination consultation database.

Keywords: CFSFDP, Cluster Centers, Particle Swarm Optimization Algorithm
Classification Number: TP391

1 Introduction

With the advent of the information age, data on the Internet has grown explosively. How to extract useful information from these massive datasets and process them effectively has become a hot research topic. Clustering [2], a popular branch of data mining [1], is an unsupervised learning method that requires no prior knowledge. It identifies commonalities among data points based on some similarity metric and partitions datasets into distinct clusters. Data within the same cluster exhibit high similarity and low variation, while data across different clusters show low similarity. Research on clustering methods has spanned several decades, with widespread applications in medicine, pattern recognition, image processing, user interest recommendation, and other fields, driving societal development and improving people' s lives.

Currently, clustering algorithms are mainly divided into five categories [2-3]: hierarchical methods, partitioning methods, density-based methods, model-based methods, and network-based methods. Each category includes several classical

algorithms [3] that have been widely applied in text processing. However, given the diversity and complexity of data, no single clustering algorithm can be universally applicable to all datasets. Each method has its own advantages and disadvantages, and different clustering algorithms produce different results.

In our comparative experiments, Agglomerative Clustering and DBSCAN represent hierarchical and density-based methods, respectively. The basic CFSFDP algorithm, proposed by Rodriguez and Laio [4], is a novel density-based clustering algorithm capable of discovering clusters of arbitrary shape with simplicity and efficiency. Zhang Wenkai studied density-based hierarchical clustering algorithms [5], Mehmood et al. investigated fuzzy clustering based on CFSFDP [6], and Ma Chunlai et al. proposed a density peak clustering algorithm with an automatic cluster center selection strategy [7].

Since CFSFDP selects cluster centers based on the product of data point density and distance—where larger products indicate higher likelihood of being centers —data points with high density but small distance, or low density but large distance, may also have large products and be mistakenly identified as cluster centers. Therefore, this paper introduces particle swarm optimization to find a pair of density-distance thresholds. Data points whose density and distance both exceed these thresholds are selected as centers, reducing the influence of discrete points on center selection and enabling automatic determination of cluster centers with less manual intervention.

The experimental data in this study originates from a college entrance examination consultation platform' s automatic Q&A application, containing students' questions about university admission policies, basic school information, and related topics. Clustering these texts helps improve the robot knowledge base and enhance the accuracy of answering student inquiries. Applying the proposed algorithm to datasets extracted from this platform demonstrates its effectiveness.

2.1 CFSFDP Algorithm

The fundamental idea of the CFSFDP [4] clustering algorithm involves three steps: first, computing the density and distance of each data point; second, selecting cluster centers; and finally, assigning non-center points to clusters. The selection of cluster centers is the critical step in this algorithm. Cluster centers possess two important characteristics: they have relatively high local density, surrounded by data points with lower density, and they maintain relatively large distances from other data points with higher density.

The basic CFSFDP algorithm has significant drawbacks in selecting cluster centers: data points with high density but small distance, or low density but large distance, may also have large products and be mistakenly identified as cluster centers. Additionally, the number of cluster centers cannot be determined automatically and requires manual intervention.

Consider a dataset $\{x_1, x_2, ..., x_n\}$. Let $d_{i,j}$ denote the distance between data point x_i and data point x_j. For each data point x_i in dataset S, we characterize it using two variables: local density and distance. The local density ρ_i is calculated using Equation (1) [4]:

$$\rho_i = \sum_j \chi(d_{i,j} - d_c)$$

where $\chi(x) = 1$ if $x < 0$ and $\chi(x) = 0$ otherwise, and parameter $d_c > 0$ is the cutoff distance.

Equation (1) shows that each data point' s density equals the number of data points in dataset S whose distance to this point is less than d_c (excluding the point itself).

When data point x_i has the maximum local density, its distance δ_i is the distance to the farthest data point from x_i in S. For other data points without maximum density, the distance represents the minimum distance between x_i and any data point with higher local density. This is calculated using Equation (2) [4]:

$$\delta_i = \min_{j:\rho_j > \rho_i} (d_{i,j})$$

Decision Graph
Using local density ρ as the x-axis and distance δ as the y-axis, we plot a decision graph characterizing each data point' s local density and distance. Figure 1 [Figure 1: see original paper] shows a scatter plot containing 28 data points, and the corresponding decision graph is shown in Figure 2 [Figure 2: see original paper] [4].

2.2 Particle Swarm Optimization Algorithm

In particle swarm optimization [8-9], each particle in the swarm has a velocity that determines its direction and position, and a fitness value determined by the fitness function. Each particle dynamically adjusts its position by moving toward its own historically best position and the swarm's best position, obtaining the optimal solution through iteration.

Assume the swarm size is N and each particle' s dimension is D. Each particle has two attributes: current position x_i and flight velocity v_i, denoted as $x_i = (x_i^1, x_i^2, ..., x_i^D)$ and $v_i = (v_i^1, v_i^2, ..., v_i^D)$, where $i = 1, 2, ..., N$. P_i represents the position with the highest fitness value found by particle x_i during the search process; P_g denotes the globally optimal position achieved by the entire swarm, i.e., the position with maximum fitness among all P_i values. During each iteration, every particle adjusts its position and velocity based on these two extremal

values. The position and velocity updates are shown in Equations (3) and (4) [9]:

$$v_i(t+1) = w \times v_i(t) + c_1 \times r_1 \times (P_i - x_i(t)) + c_2 \times r_2 \times (P_g - x_i(t))$$

$$x_i(t+1) = x_i(t) + v_i(t+1)$$

where c_1 and c_2 are positive constants called acceleration factors; r_1 and r_2 are random numbers uniformly distributed in $[0, 1]$; w is the inertia weight factor; and v_{max} is the maximum particle velocity. When a particle' s flight velocity exceeds v_{max}, it is set to v_{max}.

---

3 Text Clustering Using the Improved CFSFDP Algorithm

Due to the aforementioned limitations of the basic CFSFDP algorithm, this paper introduces particle swarm optimization. The main idea of the improved CFSFDP algorithm is to use PSO to regulate the selection of cluster centers in CFSFDP. Specifically, PSO obtains a pair of density and distance thresholds; data points whose density and distance both exceed these thresholds are designated as cluster centers. Clustering is then performed based on these selected centers, and the fitness value calculated from the clustering results serves as the criterion for updating PSO. When applied to text clustering, the algorithm computes similarity between texts to calculate each text' s density and local distance, thereby achieving text clustering. The algorithm flow is illustrated in Figure 3 [Figure 3: see original paper].

For the entire text dataset, we employ the most widely used text processing method: representing texts using the Vector Space Model (VSM) [12] based on TF-IDF (Term Frequency-Inverse Document Frequency) [10-12], where each dimension of the vector represents the weight of a corresponding feature term in the text. Text similarity is measured using cosine distance [13-14]; larger values indicate greater similarity and closer proximity between two texts.

During iteration, PSO relies on the fitness function—higher fitness values indicate better adaptation of a particle, influencing the evolution of the next generation to produce optimal solutions. This paper uses the inverse of the Rand index [15] as the fitness function for PSO to evaluate clustering quality.

The algorithm proceeds as follows:

1. Preprocess text set S, compute feature term weights for each text, obtain vector representations, and calculate similarities between texts.
2. Treat each vector from step 1 as a data point, and compute local density and distance for each point using Equations (1) and (2).

3. Initialize PSO parameters, including swarm size m, inertia weight w, learning factors c_1 and c_2, maximum iterations t, etc. Randomly generate initial velocities and positions, set each particle' s initial position as its personal best position P_i, and determine the global best position P_g from all P_i (positions are defined by density and distance thresholds).
4. Compute clustering results for each particle. Pass each particle' s position (thresholds) to the CFSFDP algorithm: data points with local density and distance exceeding these thresholds are marked as cluster centers. Non-center points are then assigned using the point assignment method to complete clustering.
5. Calculate fitness values for each particle' s clustering results, update each particle' s personal best position, update the swarm' s global best position based on all personal bests, and update each particle' s position and velocity.
6. Check convergence conditions or maximum iteration count. If satisfied, proceed to step 7; otherwise, increment iteration count and return to step 4.
7. Select cluster centers based on the swarm' s global best position, complete clustering via point assignment, and obtain the final clustering results for the text set.

---

4 Experiments

The experimental data consists of questions posed by candidates to university admissions offices through a college entrance examination consultation platform APP. We randomly selected seven categories from the question database: school and major code inquiries, military training matters, college entrance exam bonus policies, score difference policies, admissions office phone inquiries, provincial control line information, and withdrawal policies. From each category, we randomly selected samples to construct three datasets containing seven classes: data1050, data3100, and data5000, comprising 1,050, 3,100, and 5,000 data entries, respectively. The distribution of categories in each dataset is shown in Table 1 . We preprocessed the datasets using "Jieba" word segmentation and stop-word removal, then conducted comparative experiments on different datasets. Clustering performance was evaluated using four metrics: Accuracy, Precision, Recall, and F-Measure [16-17].

4.2 Experimental Results Analysis

We compared the proposed algorithm against Agglomerative Clustering [18-19], DBSCAN [20], and basic CFSFDP on the three extracted datasets. Agglomerative Clustering is widely used in text clustering due to its applicability to datasets of arbitrary shape and attributes. For our experiments, we set the

number of clusters to 7 for Agglomerative Clustering, which yielded the best results across all three datasets. PSO parameters were set as follows: swarm size = 50, maximum iterations = 30, acceleration factor = 2, and inertia weight = 0.5. DBSCAN is a representative classical density-based clustering algorithm. Through multiple experiments, we selected the following parameters that produced relatively optimal results: for data1050, eps = 0.8 and minPts = 30; for data3100, eps = 0.8 and minPts = 70; and for data5000, eps = 0.8 and minPts = 110. The overall F-measure comparison among Agglomerative Clustering, DBSCAN, basic CFSFDP, and our proposed algorithm is shown in Figure 4 [Figure 4: see original paper], demonstrating that our algorithm achieves better clustering performance across different datasets. Detailed experimental results are presented in Table 2 , confirming that our algorithm outperforms the other three algorithms on the college entrance examination consultation text database.

The basic CFSFDP algorithm suffers from inaccuracy due to noise points causing two or more data points within the same cluster to become centers. DBSCAN is highly sensitive to its parameters eps (the maximum distance between texts considered to be in the same class) and minPts (a text is considered a cluster center if at least minPts other texts are within distance eps). When data distribution density within classes is non-uniform, a small eps value splits low-density classes into multiple similar clusters, while a large eps merges nearby high-density classes into a single large cluster, resulting in suboptimal performance. Agglomerative Clustering achieves better results than DBSCAN, but its computational complexity···

---

5 Conclusion

This paper addresses the arbitrary selection of cluster centers in the CFSFDP algorithm by proposing a CFSFDP algorithm based on particle swarm optimization. By introducing PSO to find a pair of thresholds and selecting data points exceeding both thresholds as cluster centers, we reduce the impact of discrete points on clustering results and improve accuracy. Applying this algorithm to randomly extracted questions from a college entrance examination consultation platform validates its effectiveness and accuracy. The method helps candidates obtain answers more accurately and efficiently while reducing the consultation workload and saving time for both parties. However, the algorithm has limitations: due to the inherent characteristics of PSO, high-dimensional problems require a larger number of particles, resulting in high computational complexity.

References

[1] Tan P N, Steinbach M, Kuma V. Introduce to Data Mining [M]. Addison-Wesley Professional, 1988.

[2] Sun Jigui, Liu Jie, Zhao Lianyu. Clustering Algorithms Research[J]. Journal of Software, 2008, 19(1): 48-61.
[3] Shi Mengjie. Summary of Text Clustering Algorithms[J]. Modern Computer, 2014(2): 3-6.
[4] Rodriguez A, Laio A. Clustering by Fast Search and Find of Density Peaks[J]. Science, 2014, 344(6191): 1492-1496.
[5] Zhang Wenkai. Research on Density-based Hierarchical Clustering Algorithm[D]. Hefei: University of Science and Technology of China, 2015.
[6] Mehmood R, Bie R, Dawood H, et al. Fuzzy Clustering by Fast Search and Find of Density Peaks[C]//Proceedings of the 2015 International Conference on Identification, Information, and Knowledge in the Internet of Things. 2015.
[7] Ma Chunlai, Shan Hong, Ma Tao. Improved Density Peaks Based Clustering Algorithm with Strategy Choosing Cluster Center Automatically[J]. Computer Science, 2016, 43(7): 255-258.
[8] Kennedy J, Eberhart R. Partical Swarm Optimization[C]//Proceeding of the 1995 IEEE International Conference on Neural Networks. 1995.
[9] Liu Jianhua. The Basic Theory of Particle Swarm Optimization and Its Improvement[D]. Changsha: Central South University, 2009.
[10] Huang Chenghui, Yin Jian, Hou Fang. A Text Similarity Measurement Combining Word Semantic Information with TF-IDF Method[J]. Chinese Journal of Computer, 2011, 34(5): 856-864.
[11] Aizawa A. An Information-treoretic Perspective of TF-IDF Measures[J]. Information Processing and Management, 2003, 39(1): 45-65.
[12] Salton G, Buckley C. Term Weight Approaches in Automatic Text Retrieval[J]. Information Processing and Management, 1988, 24(5): 513-523.
[13] Tan Jing. Research on Text Similarity Algorithm Based on Vector Space Model[D]. Chengdu: Southwest Petroleum University, 2015.
[14] Zhao Junjie, Hu Xuegang. Similarity Calculation Based on Text Classification[J]. Microcomputer Application, 2008, 24(12): 46-47.
[15] Halkidi M, Batistakis Y, Vazirgiannis M. On Clustering Validation Techniques[J]. Journal of Intelligent Information Systems, 2015, 17(2-3): 107-145.
[16] Liang J, Bai L, Dang C, et al. The K-Means-Type Algorithms Versus Imbalanced Data Distributions[J]. IEEE Transactions on Fuzzy Systems, 2012, 20(4): 728-745.
[17] Zhang Ming. Study on the Evaluation Index Symbol of Data Clustering[D]. Taiyuan: University of Shanxi, 2013.
[18] Franti P, Virmajoki O, Hautamaki V. Fast Agglomerative Clustering Using a K-nearest Neighbor Graph[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2006, 28(11): 1875-1881.
[19] Duan Mingxiu. Research and Application of Hierarchical Clustering Algorithm[D]. Changsha: Central South University, 2009.
[20] Feng Shaorong, Xiao Wenjun. An Improved DBSCAN Clustering Algorithm[J]. Journal of China University of Mining & Technology, 2008, 37(1): 106-111.

Author Contribution Statement

Zhan Chunxia: Conceived the research idea and collected/analyzed data;
Zhan Chunxia, Wang Rongbo: Conducted experiments and drafted/revised the manuscript;
Wang Rongbo, Huang Xiaoxi, Chen Zhiqun: Revised the manuscript.

Conflict of Interest Statement

The experimental data used in this study was provided by Dayan Company. The data is limited to scientific research purposes only and may not be disseminated online or used for other purposes.

Supporting Data

The supporting data is self-archived by the authors at http://pan.baidu.com/s/1bpL0WcJ.

[1] Wang Rongbo. data1050.rar. Data contained in dataset data1050.
[2] Wang Rongbo. data3100.rar. Data contained in dataset data3100.
[3] Wang Rongbo. data5000.rar. Data contained in dataset data5000.

Received: December 30, 2016
Revised: March 15, 2017

Note: Figure translations are in progress. See original paper for figures.