ANT-BASED DOCUMENT CLUSTERING AND VISUALIZATION

Yan Yang, Fan Jin, and Yongquan Jiang
School of Computer and Communication Engineering, Southwest Jiaotong University, Chengdu, P. R. China

Abstract: This paper discusses the document clustering and visualization process: analyzing documents index, clustering document, and visualizing exploration. It focuses on ant-based clustering algorithm and some significant improvements. Clusterings are formed on the plane by ants walking, picking up or dropping down projected document vectors with different probability. It is shown that the similar documents appear in spatial proximity, whereas unrelated documents are clearly separated in visual space.

Key words: ant-based algorithm, document clustering, visualization

1. INTRODUCTION

Document clustering is valuable tool in exploratory data mining, data analysis and web search. In order to help users orient the returning of thousands of documents in web search, it has proven useful to cluster documents according to contents-similarity and to visualize the clustered document data.

The ant-based clustering algorithm is inspired by the behavior of ant colonies in clustering their corpses and sorting their larvae. Deneubourg et al. [1] first proposed a basic model that allowed ants to randomly move, pick up and drop objects according to the number of similar surrounding objects so as to cluster them. Then Lumer and Faieta [2] extended Deneubourg's model from robotic implementation to exploratory data analysis (LF algorithm). Some improvements have later been proposed, such as the CSIM （a document clustering algorithm based on swarm intelligence and k-means）

by Wu and Shi [3], and the ant–based clustering ensemble algorithm by Yang and Kamel [4].

Clustering visualization aims at representing the clusterings using graphics to help users visually perceive clusterings. The Self-Organizing Map (SOM) is widely used as a data visualization method that performs a non-linear mapping from a high-dimensional data space to a lower dimensional visualization space. A color coding method to express the approximate cluster structure of the SOM model vectors was presented by Himberg [5]. Distance mapping, a new visualization method to envision the results trained by SOM, was studied by Liao et al. [6]. The ant-based algorithm also projects data objects onto 2-dimensional visual space. Clusterings are visually formed on the plane. Handl and Meyer proposed an improved ant-based clustering and sorting as the core of a visual document retrieval system for web searches [7].

In this paper, an improved ant-based clustering and visualization for document is used to meet users' search needs. Documents are clustered according to contents-similarity. The similar documents appear in spatial proximity, whereas unrelated documents are clearly separated in visualization space.

Section 2 of this paper describes steps of the documents index analysis. Section 3 discusses the ant-based document clustering algorithm and its modification. Finally, Section 4 gives an example of visualization and follows by a conclusion in Section 5.

2. DOCUMENT INDEX ANALYSIS

In order to cluster a document collection, the key task is to represent the documents. One of the most common representation techniques in information retrieval is vector space model, also called document indexing [8]. It usually is of the following step:

2.1 Cleaning

An important part of any text processing is the cleaning of a document. Cleaning a document is to get rid of unwanted elements of the document. The procedure for document cleaning in this algorithm includes removing tags, removal of stop-words, and stemming of words.

After removing tags, the textual contents are extracted ignoring the textual structure and organization. Stop-words are frequent words that carry no information such as "the", "and", "of", etc. It is often useful to eliminate these words. Finally, word stemming is the process of converting different

forms of a word into one canonical form, called terms. Words like "walk", "walker", "walked", "walking" are all converted to a single word "walk". The Porter stemming [9] is a popular algorithm for this task.

2.2 Indexing

In the vector space model, each document is represented by a vector of words d. Each element of the vector reflects a particular word, or term, associated with the given document. In the term space, $d_i = \{w_{i1}, w_{i2}, ..., w_{in}\}$, where $w_{ij}, j = 1,..., n$ is the weight of term j in document i. The most common ways of determining the weight w_{ij} are based on the assumption that the best indexing terms are those that occur frequently in individual documents but rarely in the remainder of the collection. A well-known approach for computing term weights is the TF-IDF-weighting. The weight of a term j in a document i is given by:

$$w_{ij} = tf_{ij} \times \log(N / df_j) \tag{1}$$

where tf_{ij} is the term j frequency in the document i, or the number of occurrences of the term j in a document i. N is the total number of documents and df_j is the document frequency, that is the number of documents in which the term j occurs at least once. The inverse document frequency (*idf*) factor of this type is given by $\log(N / df_j)$. In order to account for documents of lengths, each document vector is normalized.

Once the documents are represented as vectors, the similarity between two documents can be measured. The most common measure of similarity is the cosine measure, which is defined as:

$$Sim(d_i, d_j) = \frac{\sum_{k=1}^{n} (w_{ik} \cdot w_{jk})}{\sqrt{\sum_{k=1}^{n} (w_{ik})^2 \cdot \sum_{k=1}^{n} (w_{jk})^2}}. \tag{2}$$

2.3 Reducing Dimensionality

When documents are represented as vectors, as described above, they belong to a very high-dimensional feature space because of one dimension for each unique term in the collection of documents. In order to reduce the dimension of the feature vector, the Document Frequency Thresholding is performed. Some terms whose document frequency are less than the predetermined threshold or appear in over 90% of the documents are

removed. Further, only a small number of n terms with the highest weights in each document are chosen as indexing terms.

3. IMPROVED ANT-BASED DOCUMENT CLUSTERING

The ant-based document clustering algorithm is based on the basic LF model proposed by Lumer and Faieta [2] and the CSIM model given by Wu and Shi [3]. First, document vector are randomly projected onto a plane. Second, each ant chooses a vector at random, and picks up or drops down the vector according to picking-up or dropping probability with respect to the similarity of current document within the local region by probability conversion function. Finally, clusters are collected from the plane.

Let us assume that an ant is located at site r at time t, and finds a document vector $\mathbf{d}_i$ at that site. A measure of the average similarity of document vector $\mathbf{d}_i$ with the other vector $\mathbf{d}_j$ present in its neighborhood is given by:

In order to cluster a document collection, the key task is to represent the documents. One of the most common representation techniques in information retrieval is vector space model, also called document indexing [8]. It usually is of the following step:

$$f(d_i) = \max\left\{0, \frac{1}{s^2} \sum_{d_j \in Neigh_{s\times s}(r)} \left[1 - \frac{1 - Sim(d_i, d_j)}{\alpha(1 + (v-1)/v_{\max})}\right]\right\} \tag{3}$$

where $Neigh_{s\times s}(r)$ denotes a square of $s\times s$ sites surrounding site r. α is a factor that defines the scale of similarity between document vectors. Too Large values of α will result in making the similarity between the vectors larger and forces vectors to lay the same clusters. When α is too small, the similarity will decrease and may in the extreme result in too many separate clusters. On the other hand, the parameter α also adjusts the cluster number and the speed of convergence. The bigger α is, the smaller the cluster number, while the faster the algorithm converges.

The parameter v denotes the speed of the ants. Fast moving ants form clusters roughly on large scales, while slow ants group document vectors at smaller scales by placing vectors with more accuracy. In our algorithm, v is chosen as random value between 1 and $v_{\max}$, where $v_{\max}$ is the ants' maximum speed.

Probability conversion function is a function of $f(\mathbf{d}_i)$ that converts the average similarity of a document vector into the probability of picking-up or

dropping for an ant. The picking-up probability for a randomly moving ant that is currently not carrying a document to pick up a document is given by:

$$P_p = 1 - sigmoid(f(\mathrm{d}_i)) \tag{4}$$

The dropping probability for a randomly moving loaded ant to deposit a document is given by:

$$P_d = sigmoid(f(\mathrm{d}_i)) \tag{5}$$

Instead of using the quadric function by Lumer and Faieta [2], here we use the sigmoid function, which only needs one parameter to be adjusted in the calculation. The sigmoid function has a natural exponential form:

$$sigmoid(x) = \frac{1}{1 + e^{-\beta x}} \tag{6}$$

where β is a slope constant and can speed up the algorithm convergence if it is increased. Thus we choose larger parameter β to help ants to drop faster the outliers (with high dissimilarity to all other neighborhood) at the later stage of algorithm.

Now, we can give the description of the improved ant-based document clustering algorithm in Figure 1.

The time complexity of the improved ant-based clustering algorithm is approximately $O(Mn \times ant\_number \times (s^2 + near\_num))$, where Mn is the maximum number of iteration, ant_number is the number of ants, s is the side length of local region, and $near\_num$ is the average number of document vectors within the local region [3].

4. VISUALIZATION

The role of clustering visualization is to assist users in insight of the clustered document data. Usually, simplified graphical and pictorial forms are preferred representations of clusters over numbers. Users can find it more intuitive to identify patterns that exist in data by using visual cues. There are many research efforts such as dotmap, distance mapping, and U-matrix for visualization [6], [10], [11]. Some of them focus on document visualization by SOM [5], [11], [12].

The ant-based clustering algorithm is capable of mapping high-dimensional data onto 2-dimensional visualization space. We investigate it as an alternative from SOM. Figure 2 illustrates an example of 200

1. Initialize the number of ants: *ant_number*, maximum number of iteration: *Mn*, side length of local region: *s*, maximum speed of ants moving: v_{max}, and other parameters: α, β.
2. Project the document vectors on a plane, i.e. give a pair of coordinate (x, y) to each vector randomly.
3. Each ant that is currently unloaded chooses a document vector at random.
   for $i = 1, 2, \ldots, Mn$
   for j = 1, 2, ..., *ant_number*
   4.1 compute the similarity of a document vector within a local region by formula (3), where v is chosen as random value between 1 and v_{max};
   4.2 If the ant is unloaded, compute picking-up probability P_p by formula (4). If P_p is greater than a random probability, and this vector is not picked up by the other ants simultaneously, then the ant picks up the vector, labels itself as loaded, and moves the vector to a new position; else the ant does not pick up this vector, and reselect another vector randomly;
   4.3 If the ant is loaded, compute dropping probability P_d by formula (5). If P_d is greater than a random probability, then the ant drops the vector, labels itself as unloaded, and reselects a new vector randomly; else the ant continues moving the vector to a new position.
5. for $i = 1, 2, \ldots, N$ // for all document vectors
   5.1 If a document vector is isolated, or the number of its neighbor is less than a given constant, then label it as an outlier;
   5.2 Else give this vector a cluster sequent number, and recursively label the same sequent number to those vectors who is the neighbors of this vector within local region.

Figure 1. The improved ant-based document clustering algorithm

documents with 8 clusters, each document having single topic from the Reuters-21578 collection [13]. The x-axis and the y-axis of the U-matrix indicate a document vector's position on the plane, and the z-axis is the average similarity of document vector from its adjacent ones. We can visually observe 8 clusters almost separated, whereas the similar documents appear in close proximity.

In order to represent better visualization of cluster document populations, Figure 3 gives a landscape format. Documents are grouped into clusters around the similarity represented by mountain peaks on the landscape. The height of the surface above the plane is proportional to the numbers of documents at that position on the plane. Furthermore, a topic corresponding to its cluster can be labeled as landmarks easily.

5. CONCLUSION

In this paper we introduced an improved ant-based clustering and

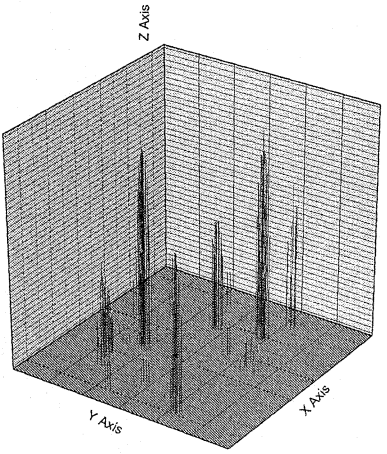

Figure 2. U-matrix of 200 documents clusterings algorithm, and a clusters map is generated to visualize the document populations.

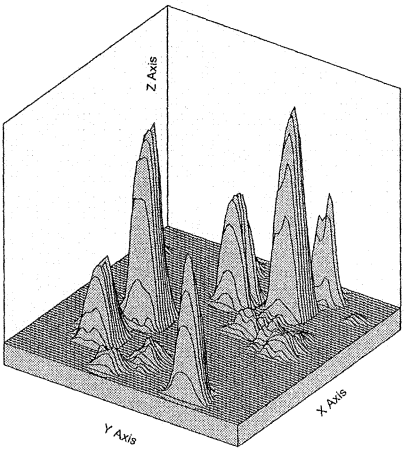

Figure 3. Landscape map of 200 documents clusterings

visualization simply. For future work, our approach can operate as a core mechanism of search engine. When users specify a query, matching documents returned by search engine are clustered based on a document index analysis and ant-based clustering algorithm, and a clusters map is generated to visualize the document populations.

ACKNOWLEDGEMENTS

This work was partially funded by the Key Basic Application Founding of Sichuan Province (04JY029-001-4) and the Science Development Founding of Southwest Jiaotong University (2004A15).

REFERENCES

1. Deneubourg, J. L., Goss, S., Franks, N., Sendova-Franks, A., Detrain, C., Chretien, L.: The Dynamics of Collective Sorting: Robot-like Ant and Ant-like Robot. In Meyer, J. A., Wilson, S. W. (eds.): Proc. First Conference on Simulation of Adaptive Behavior: From Animals to Animats. Cambridge, MA: MIT Press (1991) 356-365
2. Lumer, E., Faieta, B.: Diversity and Adaptation in Populations of Clustering Ants. Proc. Third International Conference on Simulation of Adaptive Behavior: From Animals to Animats 3. Cambridge, MA: MIT Press (1994) 499-508
3. Wu, B., Zheng, Y., Liu, S., Shi, Z.: CSIM: a Document Clustering Algorithm Based on Swarm Intelligence. IEEE World Congress on Computational Intelligence (2002) 477-482
4. Yang, Y., Kamel, M.: Clustering Ensemble Using Swarm Intelligence. IEEE Swarm Intelligence Symposium (2003) 65-71
5. Himberg, J.: A SOM Based Cluster Visualization and Its Application for False Coloring. Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN 2000), Vol. 3 (2000) 587-592
6. Liao, G., Chen, Y., Shi, T.: Research on Visualization of SOM Network. Computer Engineering and Application (2003) 35-37
7. Handl, J., Meyer, B.: Improved Ant-based Clustering and Sorting in A Document Retrieval Interface. 7th International Conference on Parallel Problem Solving from Nature (2002) 221-230
8. Salton, G., Wong, A., Yang, C.: A Vector Space Model for Automatic Indexing. Communications of the ACM, vol. 18(11) (1975) 613-620
9. Porter, M. F.: An Algorithm for Suffix Stripping. Program, vol. 14(3) (1980) 130-137
10. Morris, S., Yong, C. D., Wu, Z., Salman, S., Yemenu, D.: DIVA: A Visualization System for Exploring Document Databases for Technology Forecasting. Computers & Industrial Engineering, Vol. 43 (2002) 841-862
11. Jin, H., Shum, W. H., Leung, K. S.: Expanding Self-organizing Map for Data Visualization and Cluster Analysis. Information Sciences, Vol.163 (2004) 157-173
12. Sangole, A., Knopf, G. K.: Visualization of Randomly Ordered Numeric Data Sets Using Spherical Self-organizing Feature Maps. Computers & Graphics, Vol.27 (2003) 963-976
13. http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html