International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Study of Text Mining Using Hybrid Agglomerative Clustering With ACO Algorithms

Manpreet Kaur[1], Sukhpreet Kaur[2]

[1]M-Tech Research Scholar, Shri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, India

[2]Assistant Professor, Department Of CSE, Shri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, India

Abstract: ***Textual document clustering technique was introduced in the area of text mining. The two important main goals in document clustering are achieving high performance or efficiency and obtaining highly accurate data clusters that are closed to their natural classes or textual document cluster quality To enhance this work, we are going to propose a new hybrid clustering algorithm using Agglomerative Clustering with ACO (Ant Colony Optimization) algorithm. ACO algorithms are a class of algorithms inspired by the observation of real ants. In this paper single linkage and K-nearest Neighbor are used to achieve the high efficiency and high quality. And also used four parameters recall, precision, time, document are calculated for high efficiency and high quality.***

Keywords: Data Mining, Text Mining in Clustering, Ant Colony Optimization (ACO), Hierarchical Clustering,Single-Linkage Agglomerative Clustering.

1. Introduction

Data mining is the process of handle data from different summarizing it into useful information [13]. Data mining is also known as Knowledge Discovery in Data (KDD) [13]. Text mining usually involves the process of structuring the input text (usually parsing, along with the addition of some derived linguistic features and the removal of others, and subsequent insertion into a database), deriving patterns within the structured data, and finally evaluation and interpretation of the output [14][15]. Typical text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling (*i.e.*, learning relations between named entities) [13]. Swarm intelligence (SI) is an artificial intelligence technique and it is collective behavior of trustworthy, decentralized, self-organized system [13][14]. One of the swarm intelligence techniques is Ant Colony Optimization (ACO) [11]. Ant Colony Optimization is an meta heuristic algorithm inspired in the cooperative foraging behavior of ants to find and exploit the food source that is nearest to nest [11]. ACO is based on supportive search paradigm that can be applicable to the solution of combinatorial optimization problem [11]. Ants communicate with each other by means of an indirect form of communication mediated by pheromone [9][10]. The first ACO algorithm for discovering classification rules was Anti-miner and it was proposed by Parpinelli, Lopes and Freitas [11]. A simple agglomerative clustering algorithm is described in the Single-Linkage Clustering [13]. The minimum distance between elements of each cluster is called single-linkage clustering [13].

Clustering in Text Mining

Clustering and classification are both fundamental tasks in Data Mining [2][3]. Classification is used mostly as a supervised learning method, clustering for unsupervised learning (some clustering models are for both). The goal of clustering is descriptive, that of classification is predictive [2]. Since the goal of clustering is to discover a new set of categories, the new groups are of interest in themselves, and their assessment is intrinsic. In classification tasks, however, an important part of the assessment is extrinsic, since the groups must reflect some reference set of classes [2]. "Understanding our world requires conceptualizing the similarities and differences between theentities that compose it." Clustering group data instances into subsets in such a manner that similar instances are grouped together, while different instances belong to different groups [3]. The instances are thereby organized into an efficient representation that characterizes the population being sampled. Formally, the clustering structure is represented as a set of subsets C = C1; : : : ;Ckof S, such that: S =Ski=1 Ciand Ci \ Cj= ; for i 6= j. Consequently, any instance in S belongs to exactly one and only one subset [2][3].

2. Proposed Work

ACO algorithms are a class of algorithms inspired by the observation of real ants. Ants are capable of exploring and exploiting pheromone information, which have been left on the ground when they traversed [17]. They then can choose routes based on the amount of pheromone. While building the solutions, each artificial ant collects pheromone information on the problem characteristics and uses this information to modify the representation of the problem, as seen by the other artificial ants [17]. The larger amount of pheromone is left on a route, the greater is the probability of selecting the route by artificial ants. In ACO, artificial ants find solutions starting from a start node and moving to feasible neighbor nodes in the process of ants' generation and activity. During the process, information collected by artificial ants is stored in the so-called pheromone trails [18]. In the process, artificial ants can release pheromone while building the solution (online step-by-step) or while the solution is built (online delayed). An ant-decision rule, made up of the pheromone and heuristic information, governs artificial ants to search towards neighbor nodes

stochastically. Pheromone evaporation is a process of decreasing the intensities of pheromone trails over time. This process is used to avoid locally convergence and to explore more search space [18].

The following algorithm presents the frame of ACO.
Step 1. Firstly initialization. Let the initial pheromone trailτ0 = *k* , where *k* is a parameter,

Step 2. Main loop. In the loop, each of the m ants constructs a sequence of n nodes.This loop is executed for Itemax = s iterations and each iteration has two steps.

Step 2.1. Constructing a node sequence by each ant. A set of artificial ants is initially created. Each ant starts with an empty sequence and then successively appends an unscheduled node to the partial sequence until a feasible solution is constructed (i.e., all nodes are scheduled). In choosing the next node j to be appended at the current position i is determined by *kpij* which is probability of ant *k* transforming from node *i* to node *j* at time *t*

$$p_{ij}^{k} = \begin{cases} \dfrac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{u \in UI} \tau_{iu}^{\alpha}(t)\eta_{iu}^{\beta}(t)} & j \in UI \\ o & otherwise \end{cases}$$

Where *U* is the set of unscheduled jobs and τ *ij*(*t*) is the pheromone trail associated with the assignment node *j* to position *i* at time *t*. The parameter η*ij*(*t*) is the heuristic desirability of assigning node *j* to position *i* at time *i*. The parameter α is the relative importance of the trace and β is the parameter which determines the relative importance of the heuristic information.

Step 2.2. Update of pheromone trail. The updating rule is applied after each ant has completed a feasible solution (i.e., an iteration). Following the rule, the pheromone trail is added to the path of the incumbent global best solution, i.e., the best solution found so far. If node *j* is placed at position *i* in the global best solution during iteration t, then

τ*ij*(*t* + 1) = *p*τ*ij*(*t*) + Δτ
whereρ ,(0<ρ <1) is a parameter representing the evaporation of pheromone. The amount 1*ij* *wt*Δτ= , where *wt*is the weight tardiness of the global best solution.
The algorithm forms clusters in a Agglomerative Hierarchical Clustering manner, as follows [11]:

- Firstly, put each article in its own cluster.
- Among all current clusters, pick the two clusters with the minimum distance.
- Replace or Transfer these two clusters with a new cluster, formed by merging the two original ones.
- Repeat the first two steps until there is only one remaining cluster in the pool.

A simple agglomerative clustering algorithm is described in the single-linkage clustering.

3. Related Work

ACO algorithms show similarities with some optimization, learning and simulation approaches like heuristic graph search and evolutionary computation. These similarities are briefly discussed in the following.

Heuristic Graph Search [17], In ACO algorithms each ant erforms an heuristic graph search in the space of the components of a solution: ants take biased probabilistic decisions to choose the next component to move to, where the bias is given by an heuristic evaluation function which favors components which are perceived as more promising. It is interesting to note that this is ifferent from what happens, for example, in stochastic hillclimbers [17] or in simulated annealing [17], where (i) an acceptance criteria is defined and only thosen randomly generated moves which satisfy the criteria are executed, and (ii) the search is usually performed in the space of the solutions.

Implicit solution evaluation [18], One of the interesting aspects of real ants shortest path finding behavior is that they exploit implicit solution evaluation: if an ant takes a shorter path it will arrive to the destination before any other ant that took a longer path. Therefore, shorter paths will receive pheromone earlier and they start to attract new ants before longer paths. This implicit solution evaluation property is exploited by the ACO algorithms applied to routing, and not by those applied to static optimization problems. The reason for this is that implicit solution evaluation is obtained for free whenever the speed with which ants move on the problem representation is inversely proportional to the cost of each state transition during solution construction. While this is the most natural way to implement artificial ants for network applications, it is not an efficient choice for the considered static problems. In fact, in this case it would be necessary to implement an extra algorithm component to manage each ant's speed, which would require extra computation resources without any guarantee of improved performance.

4. Working

Hybrid algorithm of agglomerative clustering and ACO in the following:

Initialisation:
Scatter data vector, ya, randomly on a grid Initialise parameters γ, α, ti& r.

Step 1: Analysing learning data vectors
a-Nlearn **y**
Load Ant-leader-memory (Al-m, m = 1,…….,mem-max, ya)
for Ant-leader-memory, Al-m, (m = 1, …….,Alm-max) do
Compute Closest Matrix () Equation 5
Compute Neighbour Function *f* (ya, yb) Equation 3
if (Pd (ya) > = ti) **then** Equation 2
Drop (ya, yb)
Update Threshold (ti, tnew) **end-for** Equation 6&7

Step 2: Clustering testing data vectors
a-Ntest **y**
While (Iter < Max_Iteration)
Load (Aw-m, m = 1,…, mem-max , ya)
for all ant-Worker Aw, (w = 1,……., Aw-max) **do**
Find Minimum Average Distance (ya, Ci)
Compute Neighbour Function *f* (ya, Ci) Equation 3
if Pd (ya) >= t **then** Equation 2
Drop(ya, Ci);
Update Threshold (ti, tnew) **end-for** Equation 6&7
end-while

Step 3: Merging the most similar and neighboring Clusters
for all Ci, (i = 1,……., C-max) **do**
Compute Clusters Minimum Equation5
Average Distance MAD (Ci, Cj)
Compute Cluster Neighbour Function *f* (Ci, Cj) Equation 3
if f(ya, Cj) ³ t || f(yb, Ci) ³ t **then**
Merge Clusters (Ci, Cj); **end-for**

Step 4: Cluster Refinement: Detecting Small Clusters and Checking Boundaries Rule 1-3
Find Clusters mean points (Cm) R. 1
for all Cmi& Cmj , (i &j = 1,……..,Cm-max) **do**
Compute Neighbour Function Equation 3
for Close Clusters *f* (Cmi , Cmj)
if Pp (Cmi) >= t **then** Equation 1
Merge Clusters (Ci, Cj); **end-for**
Find Intersection Vectors for Close Clusters (Ci, Cj) R. 2
for all ya, (a = 1,……., y-max) **do**
Compute Neighbour Function *f* (ya ,Ci) &*f* (ya, Cj) Equation 3
if Pp (ya , Ci) > Pp (ya , Cj) && R. 3
Pd (ya , Cj) >= t **then**
Add(ya , Cj); **end-for**
Hybrid Agglomerative Clustering and Ant Colony Optimization (ACO) provide high efficiency and high quality data clustering.

5. Results

This method is used to achieve high efficiency and high-quality data clustering. The method is also beneficial to be used in textual document clustering algorithms for many text domain applications. To enhance this work, we are proposing a new hybrid clustering algorithm using agglomerative clustering with ACO algorithm. In this we have done pre-processing from new document by document clustering and using agglomerative & ACO algorithms then after we evaluate the performance. By using these algorithms we can achieve high efficiency and high quality.

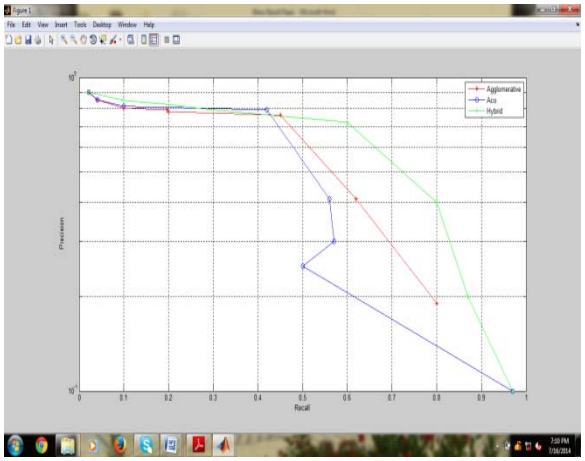

Figure 4.1: Showing Agglomerative Clustering Hybrid with ACO

Our algorithm allow for two input parameters: Precision and Recall. As in Text Mining in Clustering, Hybrid algorithms Agglomerative and ACO for high efficiency and high quality. Recall set for Hybrid is 0.9 and precision 0.2. Hybrid algorithm accurate from another is value around 20%.

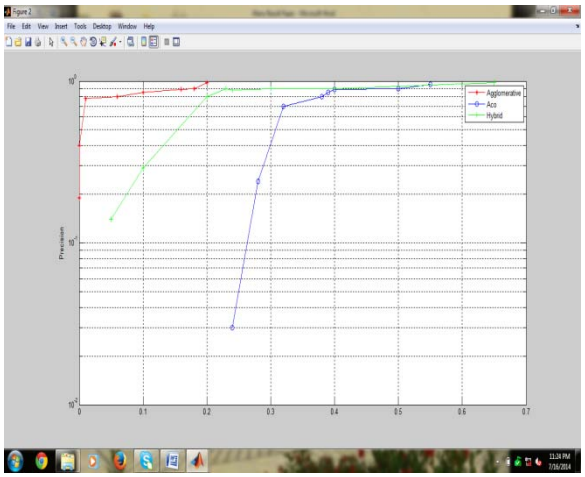

Figure 4.2: Showing Agglomerative Clustering Hybrid with ACO.

Our algorithm allow for two input parameters: Precision and Recall. As in Text Mining in Clustering, Hybrid algorithms Agglomerative and ACO for high efficiency and high quality. Recall set for Hybrid is 0.7 and precision 0.2. Hybrid algorithm accurate from another is value around 25%.

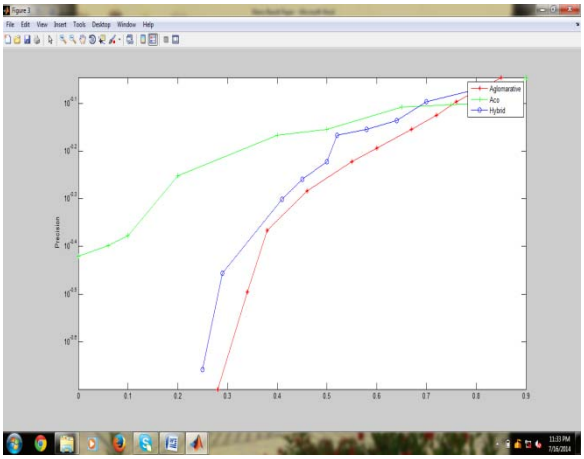

Figure 4.3: Showing Agglomerative Clustering Hybrid with ACO.

Our algorithm allow for two input parameters: Precision and Recall. As in Text Mining in Clustering, Hybrid algorithms Agglomerative and ACO for high efficiency and high quality. Recall set for Hybrid is 0.7 and precision 0.2. Hybrid algorithm accurate from another is value around 32%.

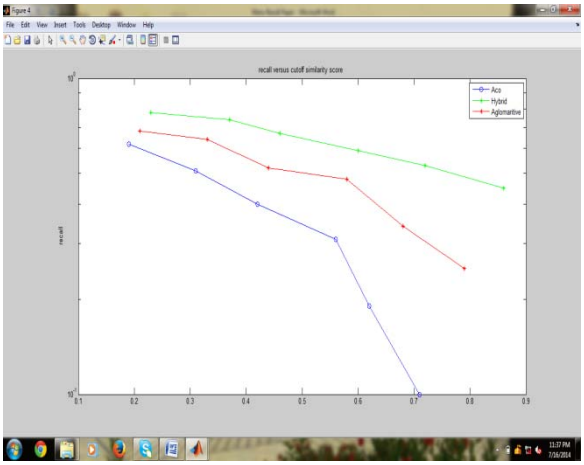

Figure 4.4: Showing Agglomerative Clustering Hybrid with ACO for Recall and Recall cut off Similarity Score.

Our algorithm allow for two input parameters: Recall and Recall cut off Similarity Score. As in Text Mining in Clustering, Hybrid algorithms Agglomerative and ACO for high efficiency and high quality. Recall set for Hybrid is 0.2 and Recall cut off Similarity Score 0.9. Hybrid algorithm accurate from another is value around 20%.

Table 1: Showing Clustering Performance Evaluation of Precision and Recall.

Partition	Cluster	Precision	Recall	F-measure
P1	C1	0.82	0.566	0.649
	C2	0.32	0.5865	0.4049
P2	C3	1	0.4255	0.6086
	C4	0.94	0.2876	0.3533
P3	C5	1	0.5	0.6543
	C6	0.4987	1	0.9765
P4	C7	1	0.3233	0.456
	C8	1	0.3211	0.52345
P5	C9	0.775	0.2356	0.28764
P6	C10	1	0.3567	0.4876
	C11	0.8	0.45	0.396
P7	C12	1	0.3567	0.4676
	C13	0.65	0.0975	0.1834
P8	C14	1	0.3032	0.453
	C15	0.3876	0.0975	0.17543
P9	C16	1	0.06756	0.13654
P10	C17	0.82	0.3	0.35678
P11	C18	0.55	0.07532	0.18765
P12	C19	1	0.27	0.45643
	C20	0.7	0.27	0.37875
P13	C21	1	0.6	0.687643
	C22	0.56	0.3	0.33333
P14	C23	1	0.71	0.16543

We have used the metrics precision, recall and F- measure shown in table 1 for evaluating the performance of the proposed approach. The evaluation metrics used in the proposed approach is given below:

F(i, j)= 2 * Recall (i, j) * Precision (i, j) / Precision (i, j) + Recall (i, j)

For each cluster, we compute the Precision, Recall and F – measure with the help of the above mentioned equation. The obtained results are shown in table 1.

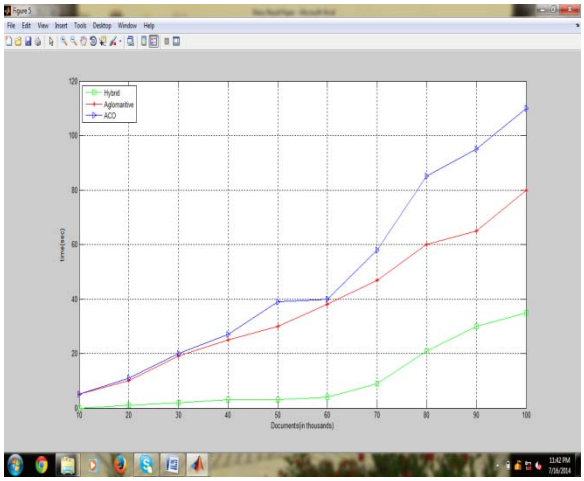

Figure 4.5: Showing Agglomerative Clustering Hybrid with ACO For Time and Document.

Our algorithm allow for two input parameters: Time and Document. As in Text Mining in Clustering, Hybrid algorithms Agglomerative and ACO for high efficiency and high quality. Time set for Hybrid is 30 and Document 100. Hybrid algorithm accurate from another is value around 30%.

6. Conclusion and Future Work

Data mining process is to extract information from a data set and transform it into an understandable structure for further use. In our work, we load a dataset on which preprocessing will be performed. Document clustering will be performed on the dataset using hybrid agglomerative clustering algorithm and ACO (Ant colony optimization) to get higher efficiency and higher quality rate. Performance will be evaluated on the basis of four parameters named as recall, precision, time and document. For future, we can enhance the performance of our work by using better optimization algorithm to get more efficiency in results.

References

[1] Wael M.S. Yafooz, Siti Z.Z. Abidin, Nasiroh Omar, Rosenah A. Halim, " Dynamic Semantic Textual DocumentClustering Using Frequent Terms and Named Entity" , IEEE 3rd International Conference on System Engineering and Technology, Vol.11, Issue.5, pp.1345-1351, Shah Alam , Malaysia, 2013.
[2] S.Murali Krishna, S.Durga Bhavani, "An Efficient Approach for Text Clustering Based on Frequent Itemsets" ,European Journal of Scientific Research, Vol.42, Issue.3, pp.385-396, 2010.
[3] Abdelmalek Amine1,2, Zakaria Elberrichi1, and Michel Simonet, "Evaluation of Text Clustering Methods Using WorldNet" ,The International Arab Journal of Information Technology, Vol.7, Issue.4, pp.590-780, October 2010.
[4] Benjamin C.M. Fung, Ke Wang, Martin Ester, "Hierarchical Document Clustering Using Frequent Itemsets" , Journal of the American Statistical Association, Vol.8, Issue.4, pp.590-780, 2010.
[5] Tru H. Cao, Vuong M. Ngo, Dung T. Hong, Tho T. Quan, "A Named – Entity - Based Multi - Vector Space Model for Semantic Document Clustering" , Faculty of Computer Science and Engineering, Vol.13, Issue.4, pp.68-75, 2012.
[6] Julien Ah-Pine, Guillaume Jacquet, "Clique-Based Clustering for improving Named Entity Recognition systems" , Proceedings of the 3rd NIST TREC Conference, Vol. 10, Issue.4, pp.105-109,2009.
[7] Florian Beil, Martin Ester, Xiaowei Xu, "Frequent Term-Based Text Clustering" , The International Arab Journal of Information Technology, Vol.8, Issue.5, pp.121-145, 2006.
[8] Hiroyuki Toda, Ryoji Kataoka, "A Search Result Clustering Method using Informatively Named Entities" ,Canadian Journal of Information, Vol.5, Issue.4, pp.133-143, 2005.
[9] Ye - Hang Zhu1, Guan-Zhong Dai1, Benjamin C. M. Fung2, De-Jun Mu," Document Clustering Method Based on Frequent Co-occurring Words" , International Journal of Computer Science & Engg., Vol. 7, Issue.5, pp.60-78, 2006.
[10] Bader Aljaber, Nicola Stokes, James Bailey, Jian Pei, "Document Clustering of Scientific Texts Using Citation Contexts" , International Journal of Computer Science & Engg., Vol. 8, Issue.4, pp.110-121, 2006.
[11] M.J.Zaki and C.J.Hsiao, "An Efficient Algorithm For Closed Itemset Mining" ,In Proc.2002 SIAM Int. Conf. Data Mining, Vol.10, Issue.5, pp.457-473, Apr.2002.
[12] F.Rosenblatt, "A Probabilistic Model For Information Storage And Organization in the Brain" Rev." , Journal of Emerging Technologies in Web Intelligence, Vol.4, Issue.2, pp.386-498, 1958.
[13] "Data Mining Knowledge Discovery Handbook" , 2nd ed., Vol.10, Issue.3, pp.190-260, 2010.
[14] Dunham, M.H., " Data Mining: Introductory and Advanced Concepts" , Pearson Education, Vol.6, Issue.5, pp.35-99, 2006.
[15] Han, J. and Kamber, M., "Data Mining:Concept & Technologies" , Morgan Kaufman Publisher, Vol.6, Issue.4, pp.1-1000, 2006.
[16] Han, J. and Kamber, M., "Data Mining:Concept & Technologies" , Morgan Kaufman Publisher, Vol.11, Issue.4, pp.1-1000, 2011.
[17] N. J. Nilsson. Artificial Intelligence: A New Synthesis. Morgan Kauffmann, Vol. 1437, Issue.8, pp. 27-55, 1998.
[18] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. Science, Vol.220, Issue.4598, pp.671–680, 1983.

Author Profile

Manpreet Kaur is Student of M.Tech in the department of Computer Science and engineering at Shri Guru Granth Sahib World University (SGGSWU) , Fatehgarh Sahib,Punjab. She has done B.Tech from Yadvindra College of engineering, Talwandi Sabo under Punjabi University, Patiala.

Sukhpreet Kaur received the B. Tech degree in Computer Science and Engineering from Punjab Technical University, Punjab, India in 2007 and M.Tech Degree in Computer Engineering from Punjabi University, Patiala, India in 2010. She worked as an Assistant Professor at Department of Computer Science and Engineering in Baba Farid College of Engineering and Technology,Punjab, India till 2012. She is working as Assistant Professor at Department of Computer Science and Engineering, Shri Guru Granth Sahib World University Fatehgarh Sahib, Punjab, India since July 2012. She has published more than 30 research papers in different international journals and conferences. Her research areas include Software Engineering and Data mining.