Research Article

Literati Journals
Empowering Researchers

Application of Parallel Glowworm Swarm Optimization Algorithm for Data Clustering on Large Datasets

Hajira Tabasum M,[1*] Akram Pasha[1]

Abstract: Data Analyzing is the primary task with unstructured large data sets which is the major concern in many application areas. Many data analyzing algorithms need to be modified in order to handle the large data sizes efficiently. In this work, Glowworm Swarm Optimization for data clustering algorithm is designed and implemented to handle unstructured large data sets. The algorithm uses the optimized glowworm swarm to evaluate the data analyzing problem. The algorithm for data analyzing is used as it is very advantageous in resolving problems with multimodality, which in terms of clustering means finding the number of centroids. Glowworm Swarm Optimization for data clustering algorithm uses the Map Reduce methodology for the parallelization since it provides balancing the load in a parallel fashion, localizing the data and tolerant towards fault. The experiments conducted on various data sets shows that scalability with large unstructured data sets and achieves a better performance with data clustering, thus proving the Glowworm Swarm Optimization for data clustering algorithm is efficient compared to the traditional algorithms on clustering of large unstructured data sets.

Asian Journal of Engineering and Technology Innovation
Volume 4, Issue 7
Published on: 7/05/2016

Cite this article as: Hajira Tabasum M, Akram Pasha. Application of Parallel Glowworm Swarm Optimization Algorithm for Data Clustering on Large Datasets. Asian Journal of Engineering and Technology Innovation, Vol 4(7): 34-39, 2016.

INTRODUCTION

Clustering of large unstructured data sets is the main task in analyzing the data. The main objective of analyzing data is to divide the unstructured data sets into different divisions and each division is named as Clusters.Each cluster will be having common specfications between the cluster members. An efficient clustering algorithm is one that yields clusters with very high quality, that is, the measures to check the similarity between the data objects in a particular cluster has to be maximized, the measures to check the similarity between the data objects from different clusters has to be minimal. Hence,clustering of large unstructured data sets can be defined as a optimization problem with multiple objectives.Many algorithms have been introduced to resolve the problems with optimized data clustering such as glowworm swarm intelligence. Glowworm swarm intelligence simulates the natural behavior of the swarms such as birds flock, colonies of ants,large number of fish together and growth of bacteria.

Considering the natural behavior of the swarm, the members of the swarm sense the interaction of each member in the swarm and share the information with each other, the information such as helping each other to reach the food resources. All the swarm members participation is required to achieve the goal instead being a central member in the swarm to co-ordinate all the swarm members. Some examples of swarm intelligence algorithms are Glowworm Swarm Optimization, Particle Swarm Optimization and Ant Colony Optimization.Clustering large amount of data is one of the challenging tasks in many application areas like social networking, bioinformatics and many others. Many Traditional clustering algorithms needs modification to handle the large data sizes. In this work, the algorithm on Optimization of Glowworm Swarm for clustering of datais designed and implemented tohandle large unstructured data sets. The proposed algorithm uses optimized glowworm swarm to evaluate the clustering algorithm. Glowworm Swarm Optimization for data clustering algorithm is used as it is very advantageous in resolving multimodal problems, which in terms of clustering means finding multiple centroids. The algorithmuses the Map and Reduce methodology for the parallelization since it provides balancing of load in a parallel fashion, localizing the unstructured data set and tolerant towards faults. The experimental results shows that Glowworm Swarm Optimization for data clustering algorithm scales better with increase in data set sizes and achieves a very close to linear speed with maintenance of the clustering.

The paper is organized with the following sections: Section II briefly describesthe related works carried out in the domain of analyzing the large datasets. Section III describes the problem definition, the existing system and the proposed system

[1]Reva Institute of Technology and Management, Rukmini Knowledge Park, Kattigenahalli, Yelahanka, Near Border Security Bustop, Bengaluru, Karnataka-560064, India.
E-mail: hajiramansoor72@gmail.com
*Corresponding author

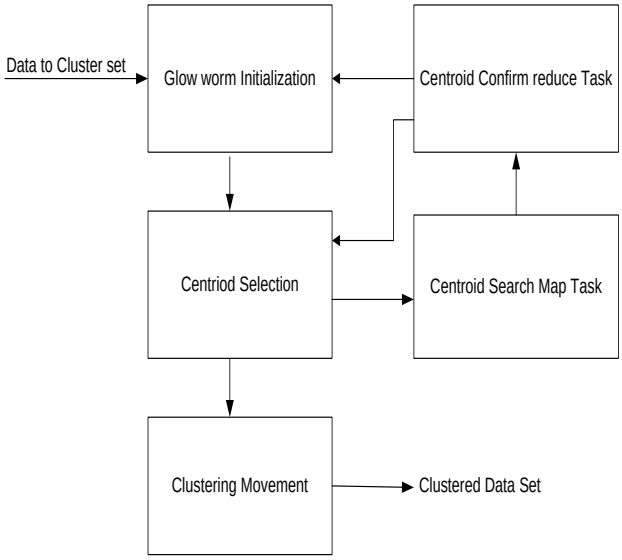

Figure 1: Architecture of proposed system

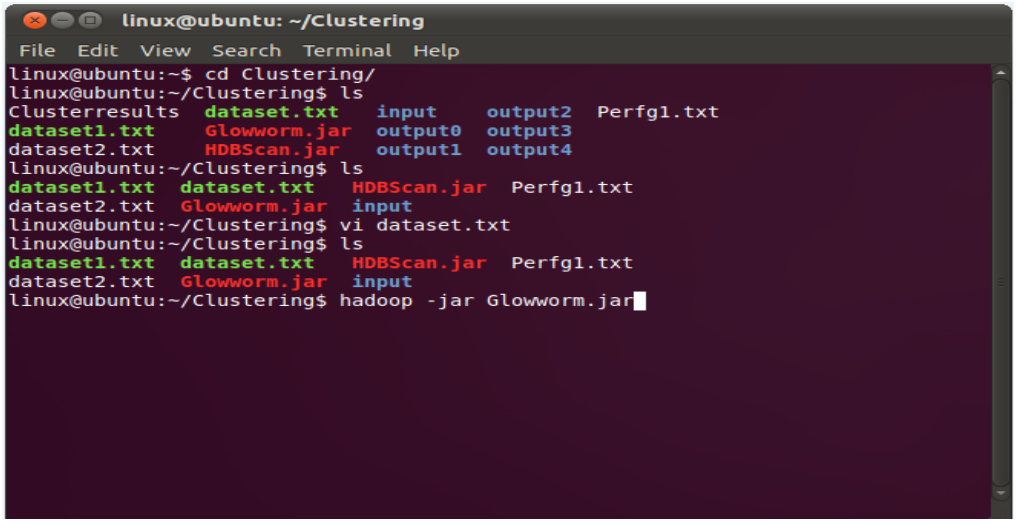

Figure 2: Execution of glowworm clustering algorithm

explained using glowworm clustering algorithm process and the Map Reduce programming framework.Section IV explains the experiments conducted on various data sets and the results obtained. The work proposed is concluded in Section V.

LITERATURE SURVEY

This section reviews the related work carried out in the domain of analyzing the large datasets, considering the data clustering to be a prime problem.

The work proposed by R. Eberhart and J. Kennedy in [1] introduced a new method for optimizing the continuous nonlinear functions. This method was introduced through simulation of a very simple social model; thus the social metaphor is considered, and the algorithm stands without metaphorical support. This work specifically describes the particle swarm optimization concept in terms of its precursors, taking into consideration the stages of its development from social simulation to optimizer. Particle swarm optimization has roots in two main component methodologies. Perhaps more obvious are its ties to artificial life in general, and to fish schooling, bird flocking and particularly the swarming theory. It is also related to evolutionary computation, and is tied to both evolutionary programming and genetic algorithms. These relationships are briefly described in this paper.

Particle swarm methodology was introduced with the concept for optimization of nonlinear functions. Several paradigms were evolved and are outlined, and the implementation of Benchmark testing paradigm is discussed. Applications including trainings on neural network, optimization of nonlinear functions and benchmark testing of the paradigm is described, and are proposed. The relationships between optimization of particle swarm methodology and both genetic algorithms and artificial life are described.

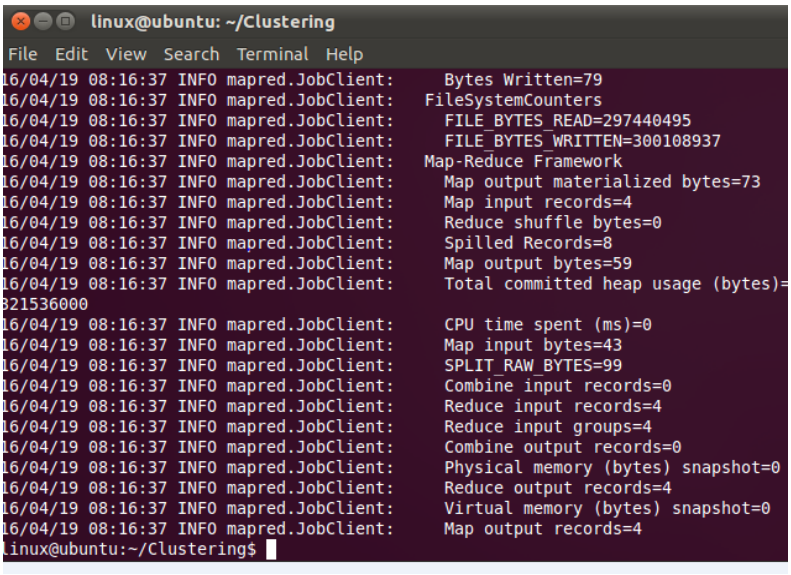

Figure 3: Output of glowworm clustering algorithm

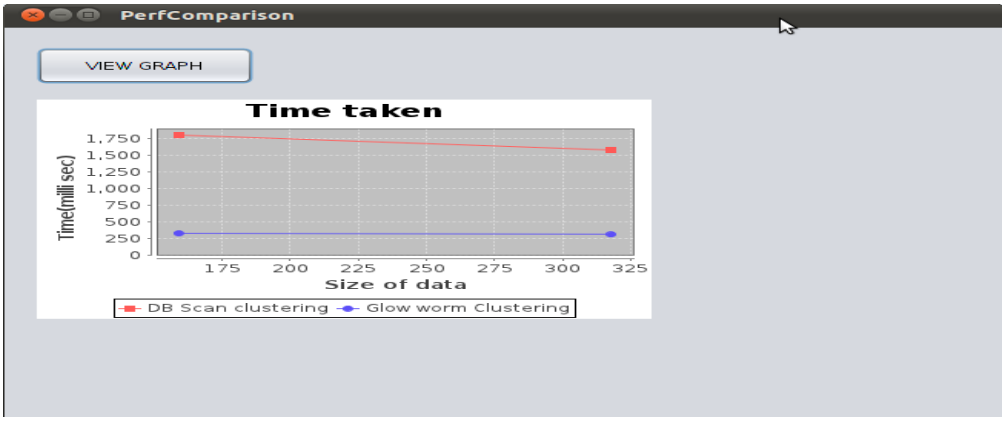

Figure 4: Performance comparison of DB Scan clustering and glowworm clusterin

In the work proposed by Z. Weizhong, M. Huifang, and H. Qing in [2], a k-means parallel clustering algorithm using a programming model Map Reduce was introduced, which is a simple yet powerful parallel programming technique. The results through different experiments demonstrate that the algorithm proposed can efficiently process large data sets and scale well on commodity hardware. Clustering of data has been received considerable attention in many applications, such as document retrieval, data mining,pattern classification and image segmentation. The large volumes of information evolved by the progress of technology, makes clustering of very large scale of data a challenging task. Many renowned researchers tried their best to design very efficient parallel clustering algorithms In order to deal with these data clustering problems.

Map and Reduce functions are part of a programming model on a Hadoop platform is effectively used for the associated implementation of processing and generation of large datasets that is relatively clustered to cater all the real-world tasks. The computation is specified in terms of the functions map and reduce, and the underlying runtime systemparallelizes automatically the computation across large-scale data clusters of machines.

In this work, k-means algorithm is adapted in Map and Reduce framework which in turn is implemented by Hadoop to make the data clustering method applicable to large scale of data sets. By applying proper pairs, the proposed algorithm can be parallel executed effectively. Comprehensive experiments were conducted to evaluate the proposed algorithm. The experimental results demonstrate that the algorithm introduced can effectively deal with large scale unlabeled datasets.

In the work proposed by I. Aljarah and Ludwig S. in [3], algorithm on glowworm swarms introduced as an optimization algorithm inspired by nature simulating the natural behavior of the lighting worms. Glowworm algorithm has many applications in the problems requiring multiple solutions in huge search space, having equal or different objective function values. Therefore, in this work, a data clustering algorithm

based on glowworm is proposed, in which the glowworm algorithm is modified to solve all the data clustering problem and to effectively locate multiple optimal centroids based on the multimode search capabilities of the glowworm algorithm. The optimized algorithm on glowworm for data clustering algorithm ensures that the similarity between the members within a cluster is maximum and the similarity among members from different clusters is minimal.

Fitness functions are proposed to evaluate the goodness of the members of glowworm algorithm in achieving better quality data clusters. The proposed algorithm is tested by real-world and artificial large unstructured data sets. The better performance of the proposed algorithm over popular clustering algorithms is demonstrated using many data sets. The results shows that Glowworm Swarm Optimization for data clustering algorithm can efficiently be used for clustering of large data sets.

This work furtherextends the functionality of Glowworm swarm optimization to solve the clustering problem, taking into account the advantages of the glowworm swarm optimization algorithm multimode search capability to locate and optimize centroids. Subsequently, the proposed algorithm is capable of identifying the number of clusters to be generated without specifying the number explicitly in advance. Moreover, three different fitness functions are introduced to add robustness and flexibility to the proposed algorithm.

In the work proposed by He, H. Tan, Luo, S. Feng, and J. Fan in [4], DBSCAN (density-based spatial clustering of noise based applications), a very important spatial data clustering method was introduced, which is widely adopted in a number of applications. The size of datasets is extremely large in these days the processing of very complex data sets analysis in parallel such as density based algorithm becomes obsolete. However, there are three major disadvantages in the existing approach of parallel processing of density based spatial algorithms. First, density based spatial algorithms are not effective in properly balancing the load among tasks in parallel, specifically when data are heavily skewed. Secondly, the algorithms are limited in scalability because all the critical sub-procedures are not parallelized. Third, the algorithms are notdesigned primarily for shared-nothing environments, which limit the portability to emerging parallel processing paradigms.

However, in this work, MR-density based algorithm is presentedas a scalable density based spatial algorithm by unleashing the inherent parallel distributed computing framework, such as Map and Reduce programming model. In this algorithm, all the critical sub-procedures are parallelized completely. As such, there is no performance bottleneck caused by sequential processing. Mainly, a novel data partitioning technique based on estimation and computation cost is proposed. The primary objective is to achieve desired load balancing even in the context of heavily skewed data.

PROBLEM DEFINITION

This section describes the problem associated with analyzing associated with clustering of large unstructured data. Clustering large unstructured data is one of the primary tasks recently known which is used in many application areas such as data analysis associated with banking transactions, data analysis associated with social networking sites, and many more application areas.

Clustering of unstructured data is the important data analyzing tasks with the main objective of dividing a set of unstructured data objects into different groups named as clusters; each cluster is having some specifications in common between the cluster members. Clustering very large data sets that contain large numbers of unlabeled records with very high dimensions is very difficult and computationally expensive.

EXISTING SYSTEM

The parallelization of the algorithm is the solution to enable existing approaches to work feasibly on big data, which can be carried out using different methodologies such as Message Passing Interface, or Map Reduce, and many others. Map Reduce is a prominent parallel processing of data framework, which has been gaining significant interest from both academia and industry.

Map Reduce data framework enables users to develop large-scale distributed applications efficiently by supporting load balancing, fault tolerance and data locality.

PROPOSED SYSTEM

We propose a scalable design and implementation of glowworm swarm optimization clustering using the Map Reduce methodology called Glowworm Swarm Optimization for data clustering algorithm with Map Reduce. Glowworm Swarm Optimization for data clustering algorithm with Map Reduce is different from Glowworm Swarm Optimization for data clustering algorithm as the algorithm implements the Map and Reduce functions in order to achieve the main goal of solving big data clustering problems and enhancement in itsscalability.

The Modules in this Project as Shown Below

Glowworm Initialization: The unlabeled data is taken for clustering. In this technique, initially a swarm of agents in a search space are distributed in a random fashion. The agents are considered as glowworms which carry a good quantity of luminescence called luciferin with them.

The glowworms emit a light whose intensity is proportional to the associated luciferin and interacts with other agents available within a variable neighborhood. The glowworm identifies its neighbors and calculates its movements by exploiting an adaptive neighborhood, which is bounded above by its sensor range.

Centroid Selection: The swarm agents are selected as centroid to reduce the tasks. Here map and reduce functions are executed. Clustering Movement: In this module, the various agents are clustered and moved.

EXPERIMENTS AND RESULTS

Map Reduce programming data framework is a highly scalable and robust technique and can be used across many computer nodes in parallel, and is mostly applicable for intense data applications when there are some limitations on multiple processing and with large shared-memory machines. Map Reduce programming data modelling framework utilizes two main functions on Hadoop: Map and Reduce. Both Map function and Reduce function takes inputs and produce outputs in the form of <key, value>. The Map function moves over a large number of records and extracts interesting information from each record, and then all values with the same key is provided as input to the same Reduce function. Moreover, the Reduce function aggregates intermediate results, generated from the Map function that has the same key, and then generates the final results.

This section illustrates the algorithm used for the implementation in achieving the desired outcome.

Table 1: Algorithm Glowworm

Input : Data Set
Output : Clustered Data
Steps
1) Load the Data Set
2)Initialize the swarms to a random position
3)Centroid Selection based on sub coverage distance
4)Perform Map and reduce tasks for each iteration
5) Find the fitness value
6)Cluster the data based on sub coverage-distance and fitness value

The following snapshots (Given after conclusion) define the results or outputs that is produced after step by step execution of all the modules of the system.

The above graph illustrates the Time versus Data size taken for DB Scan Clustering and Glowworm Clustering algorithms

CONCLUSION

In this work, efficient data clustering algorithm is introduced which is highly scalable. The algorithm is designed and implemented on Hadoop with the use of Map and Reduce programming model. The algorithm is optimized for clustering of large unstructured data sets, furthermore the processing time required to analyze large unstructured data sets is significantly higher. Therefore, the algorithmis implemented on Hadoop with Map and Reduce programming framework to overcome the inefficiency with processing time on large unstructured data sets.The optimized algorithm on glowworm for data clustering illustrates that Glowworm Swarm Optimization for data clustering algorithm can efficiently be parallelized with Map and Reduce programming model to process very large data sets.The experimental results proved that the proposed algorithm yields high accuracy and scalability used with very large data sets.

Snapshots

$k \leftarrow 1$ {initialization}
$f(x(k)^*) \leftarrow \infty$
for $m = 1$ to M **do**
  Generate_Solution($x_m(k)$)
  $f(x_m(k)) \leftarrow$ Evaluate_quality($x_m(k)$)
  $\iota_m(0) \leftarrow \iota_0$
  $r_m(0) \leftarrow r_0$
end for
{main loop}
repeat
  {update luciferin quantity}
  for $m = 1$ to M **do**
    $\iota_m(k) \leftarrow (1-\rho)\iota_m(k-1) + \gamma f(x_m(k))^{-1}$
  end for
  {move glowworms}
  for $m = 1$ to M **do**
    $N_m(k) \leftarrow Find\_Neighborhood(x_m(k))$
    {sum selection probalities for all p neighbors in $N_m(k)$}
    $P_{sum} \leftarrow sum(\iota_p(k) - \iota_m(k))$
    for all $x_j(k)$ in $(N_m(k))$ **do**
      $p_{mj}(k) \leftarrow (\iota_j(k) - \iota_m(k))/P_{sum}$
    end for
    $q \leftarrow Select\_neighbor\_index(p_m(k))$
    {move selected}
    $x_m(k+1) \leftarrow x_m(k) + s(x_q(k) - x_m(k)) / (\|x_q(k) - x_m(k)\|)$
    $r_m(k+1) \leftarrow \min\{ r_s, \max\{0, r_m(k) + \beta(N_{set} - |N_m(k)|)\}\}$
  end for
  for $m = 1$ to M **do**
    if $Was\_Moved(x_m(k)) =$ **false then**
      $x_m(k+1) \leftarrow x_m(k)$
    end if
    $f(x_m(k)) \leftarrow$ Evaluate_quality($x_m(k)$)
    if $f(x_m(k)) < f(x(k)^*)$ **then**
      $x(k)^* \leftarrow x_m(k)$
    else
      $x(k)^* \leftarrow x(k-1)^*$
    end if
  end for
  $stop\_condition \leftarrow Check\_stop\_condition()$
  $k \leftarrow k+1$
until $stop\_condition =$ **false**
return $f(x(k)^*)$, $x(k)^*$, k

REFERENCES AND NOTES

1. Impetus white paper, March, 2011, “Planning Hadoop/NoSQL Projects for 2011” by Technologies,

Available:http://www.techrepublic.com/whitepapers/Planninghadoopnosql-projects-for-2011/2923717, March,2011.

2. Andrea C. Arpaci-Dusseau, Remzi H. Arpaci-Dusseau, David E. Culler, Joseph M. Hellerstein, and David A. Patterson. High-performance sorting on networks of workstations. In Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data, Tucson, Arizona, May 1997.
3. Jeffrey Dean and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters", Google, Inc.
4. Zibin Zheng, Jieming Zhu, and Michael R. Lyu , "Service-generated Big Data and Big Data-as-a-Service: An Overview" , 978-0-7695-5006-0/13 2013 IEEE.
5. McKinsey Global Institute, 2011, Big Data: The next frontier for innovation, competition, and productivity, Available:www.mckinsey.com/~/media/McKinsey/dotcom/Insights%20and%20pubs/MGI/Research/Technology%20and%20Innovation/Big%20Data/MGI_big_data_full_report.ashx, Aug, 2012.
6. Aditya B. Patel, Manashvi Birla, Ushma Nair, "Addressing Big Data Problem Using Hadoop and Map Reduce", NUiCONE-2012, 06- 08DECEMBER, 2012.
7. Apache Software Foundation. Official apache hadoop website, http://hadoop.apache.org/, Aug, 2012.
8. Hung-Chih Yang, Ali Dasdan, Ruey-Lung Hsiao, and D.StottParker from Yahoo and UCLA, "Map-Reduce-Merge: Simplified Data Processing on Large Clusters", paper published in Proc. of ACM SIGMOD, pp. 1029–1040, 2007.
9. H. Mi, H.Wang, Y. Zhou, M. R. Lyu, and H. Cai, "Towards fine-grained, unsupervised, scalable performance diagnosis forproduction cloud computing systems,"IEEE Transaction on Parallel and DistributedSystems, no.PrePrints, 2013.
10. M. Chen, E. Kiciman, E. Fratkin, A. Fox, and E. Brewer, "Pinpoint: problem determination in large, dynamic internet services", in Proceedingof the International Conference on Dependable Systems andNetworks (DSN'02), pp. 595–604.
11. B. H. Sigelman, L. A. Barroso, M. Burrows, P. Stephenson, M. Plakal, D. Beaver, S. Jaspan, and C. Shanbhag, "Dapper, a large-scaledistributed systems tracing infrastructure," Google, Inc., Tech. Rep., 2010.
12. S. Lohr, "The age of big data," New York Times, vol. 11, 2012.
13. "Challenges and opportunities with big data," leading Researchers across the United States, Tech. Rep., 2011.
14. E. Slack, "Storage infrastructures for big data workflows," Storage Switchland, LLC, Tech. Rep., 2012.
15. "Big data-as-a-service: A market and technology perspective," EMC Solution Group, Tech. Rep., 2012.
16. J. Horey, E. Begoli, R. Gunasekaran, S.-H. Lim, and J. Nutaro, "Big data platforms as a service: challenges and approach," in Proceedings of the 4th USENIX conference on Hot Topics in Cloud Ccomputing, ser. HotCloud'12, 2012, pp. 16–16.
17. "Why big data analytics as a service?" "http://www.analyticsasaservice.org/why-big-data-analytics-as- aservice/", August 2012.
18. P. O'Brien, "The future: Big data apps or web services?" "http://blog.fliptop.com/blog/2012/05/12/the-future-big-data-appsor- web-services/", 2013.
19. Apache Cassandra, http://cassandra.apache.org.
20. N. Lynch and S. Gilbert, "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services," SIGACT, 2002.
21. RevolutionAnalyltics,https://github.com/RevolutionAnalytics/RHadoop/ wiki.