**The Effect of Different Clustering Algorithms on Clustering Speed**

**Introduction**

Machine Learning, a buzzword in the field of computer science, business, and medicine, is simply just a method of data analytics that focuses on automating analytic model building (Koza et al, 1996). It is a subfield of Artificial Intelligence that focuses primarily on the idea that machines­ can learn from data, as well as identify patterns and make decisions with very little human interference (Friedman, 1998). Artificial Intelligence comes in many forms, such as natural language processing, which is concerned with the way humans and computers interact, specifically how to program computers to understand and process large amounts of natural language data. Natural language processing (NLP) started out as one of the earliest fields of computer science, when in the 1950s, Alan Turning published an article that would later be known to the world as proposing the Turing test for Intelligence as a criterion to see if a program was truly “intelligent” (Turing, 1988). During that time, NLP did not include machine learning.

Machine Learning and NLP started to merge in the 1980s when there became fewer complex sets of hand-written rules. This was because there was more computational power, in accordance with Moore’s law, and there was less of the idea of an “inborn” sense of grammar and that it is often difficult to learn. This led to Literature Based Discovery (LBD).

LBD is a subfield of natural language processing, that focuses on finding implicit relationships from explicit relationships found in text. For example, if the relationship “penguin” implies “Antarctica” and “Antarctica” implies “cold” is found in a text, LBD would conclude that “penguin” and “cold” are related (Swanson, 1986). This idea was pioneered by Dr. Don Swanson in the 1980s and has been used since. It is important to highlight the fact that LBD does not generate new knowledge through laboratory experiments, but rather connects existing knowledge from results by bringing to light relationships that are implicit and often are ignored (Chen et al, 2011). However, Antarctica could also imply “icy” or “chilly” or “cool” or “freezing”, all of which have similar meanings to “cold.” This would often times skew the results and the ranking of the term which was trying to be found. Thus, from this issue, the idea of clustering alike terms was conceived.

Clustering is a relatively new idea to LBD that focuses on grouping together similar terms. By grouping similar terms together, the ranking of the target term decreases (a lower numerical ranking indicates a more solid relationship with the initial term). This would greatly impact the new relationships that are found between texts. However, there are many different types of clustering algorithms that can be used. Because different algorithms are used, the time taken to run them might differ. In the future, once LBD is implemented into healthcare systems, the time it takes for the code to run will make a difference. Because not all the laptops that doctors or other medical professionals use are supercomputers, the time difference between each algorithm is important, especially when dealing with over one million words, not including the number of phrases that also just mean one thing. This is when the different clustering algorithms come into use. This research of how long it would take for each algorithm to compile a set of words can be used to help speed up the results given to doctors.

The most common software for clustering is a software known as CLUTO, which clusters small and large datasets and analyzes the characteristics of the various clusters. CLUTO comes with five different clustering algorithms for matrices, which is the way the data is presented. CLUTO was developed by Dr. Gary Karypis at the University of Minnesota, Twin Cities.

The problem in this project is trying to see which clustering algorithm is the quickest in clustering the terms. It is hypothesized that if the agglomerative method for clustering is used, then it will run the fastest. In this project, the independent variable is the clustering algorithm that is used. There are five different clustering algorithms in this project. These include the repeated bisections approach, (RB). In this approach, a matrix is clustered into two groups, and then one of these groups is selected and bisected further. This is continued until the specified number of clusters is found. In each step, the cluster is bisected in such a way that the resulting 2-way clustering optimizes a particular standard set by the user. The one used in this research was the default by CLUTO but is not the control because it is still being compared. The RBR method is similar to the RB method. However, in the end, the solution is globally optimized, which means some mathematical operations are down to the results to make sure that the solutions are the most accurate. Direct is a method where the clustering solution is calculated by finding all of the clusters, simultaneously. Agglomerative (agglo), works in the opposite direction as RB, starting out with each word in its own cluster, and then continues with pairs of clusters being merged until everything is one cluster. Graph computes the k-way solution by making every object become a vertex, and then each object is connected to its most similar other object(s). The graph is then split using a graph partitioning algorithm, which is done by the program. The dependent variable in this project is the speed of the clustering algorithm. There is no control group in this project because there is no standard clustering algorithm used.

**Methods and Materials**

A dataset was obtained from the CLUTO website and was formatted in the way the algorithms could understand. The dataset was formatted with the first line containing the number of rows and the number of columns, along with the total number of data points. It should be noted that because this dataset is sparse, which means that not all the points of the dataset are there, there are less total points multiplying the columns and the rows together. This dataset was then opened in a text editor to confirm that all the points where there.

The clustering algorithm was downloaded from the CLUTO website, which was downloaded as a “.tar-zip” file. The file was then unzipped and then the dataset and the clustering algorithm (vcluster) were then put into a separate folder, called “clustering” on the desktop. This folder was then moved onto a virtual machine, which ran Linux (Ubuntu).

Then, a script to run the command (./vcluster [method] t8.8k.dat 10) to run the clustering software, as well as get the timings was created in Python. The script also ran through each clustering algorithm and produced a document called “Clustering Timings” in the “clustering” folder.

For this project, the same computer was used to run the different clustering algorithms. This computer had a battery percentage of 100% while running the experiment, as well as the same internet connection. The computer had the same RAM storage and the same processor. The same data set was used, as well as the same software to run clustering algorithms.

The independent variable was the different clustering algorithm that was used. There were five levels of the IV, agglomerative clustering, repeated bisection clustering, repeated bisection global clustering, graph clustering, and direct 10 clustering, as explained in the Introduction portion of this paper. There were 10 repeated trials for each clustering algorithm. There was no control group in this project, due to the fact that all of the clustering algorithms were being compared against each other and there is no standard clustering technique.

The specialized technique that was used in this experiment was the code. There was one piece of specialized equipment and that was an Asus computer found at the house of the student. There are no safety concerns for this project since it is just a computer simulation.

**Results**

After the code for clustering the dataset was written and run, the output displayed the runtimes for all 10 of the trials of each clustering algorithm. The agglomerative method (agglo) performed the fastest with a mean of 0.00021 seconds. The direct clustering method took an average of 0.00399 seconds. The RB method and RBR method had very similar run times. The RBR method took 0.004043 seconds on average while the RB method took 0.004049 seconds on average to cluster the data. The graph clustering method took the most amount of time with an average time of 0.041708 seconds. See figure 2 for all the exact timings of each trial.

**Discussions and Conclusions**

The purpose of this experiment was to determine which clustering algorithm was quickest when clustering a dataset. Some major findings of this project include the agglomerative clustering algorithm having the lowest mean for clustering the dataset. The average time taken to cluster the dataset was 0.00021 seconds. The graph clustering method had the highest mean time, which was 0.041708 seconds. Because there are no outliers, the experiment utilized means instead of medians as the comparison between algorithms.

The data collected from this experiment supported the hypothesis. The hypothesis stated that if the agglomerative clustering is used to cluster data, then the output will be returned the fastest. When the experiment was run, the agglomerative clustering method returned data the fastest.

For the data analysis, the primary method was through standard deviation. Through standard deviation, it was possible to conclude that the standard deviations for each method were not too high, indicating that each method performed about the same each time and that it was acceptable to use means as the central tendency. Also, because of the lack of high standard deviations, it is possible to conclude that the clustering algorithms did not vary because of the definition of the standard deviation itself.

The results obtained through this experiment can be explained through the fact that agglomerative clustering decreases the required amount of items that need to be clustered by a factor of 2. This essentially means that each piece of data is paired with another similar piece of data. This process continues until the whole dataset is one cluster. This method was very effective in the dataset obtained from the CLUTO website because the dataset was fairly small and completely random. The graph method took the longest amount of time to cluster the dataset because each piece of data is converted into a point on a graph. Each point is then compared to every other point on the graph. This repeats for every single point on the graph, or piece of data. However, the graph clustering algorithm is more effective and thorough when the dataset being clustered is large. This is because the graph clustering algorithm relies the most on machine learning. It depends on there being a large amount of data it can learn from, which allows it to pick out which ones are the most similar without having to go and check against every one. It would be easier for it to draw the boundaries that create the clusters.

For future study and research, the project could be improved by running a larger dataset and a smaller dataset with the clustering algorithms. This would help determine which clustering algorithm is the best applied to data sets where the size is unknown, as well as data sets where the size is known, as the best algorithm could be picked. Also, the idea of working with a dataset where the size is unknown to the algorithm may also be explored in further research. This would help determine how each clustering algorithm deals with unknown data set sizes. Furthermore, another way the project could be improved and changed would be running the clustering algorithms on a different computer. The computer that ran the clustering algorithms was not the quickest nor powerful by any means, so changing the computer that the algorithms were run on could change the results. The clustering algorithms could be run on different bases and the bases could be measured.

The idea of comparing clustering algorithms is still relatively new for the field of computer science, thus very little research has been done on that exact topic. However, other similar papers, such as “A Comparison of Document Clustering Techniques” talk about clustering similar papers together. These types of papers talk about the effect of clustering documents using clustering algorithms (Steinbach et al, 1). This is similar to this project in the sense that these clustering algorithms are being used to classify documents. In these papers, however, because the data set is much larger and consists of whole papers, it ends with graph being the quickest clustering algorithm.

**Works Cited**

Chen, Ran; Hongfei Lin & Zhihao Yang (2011). "Passage retrieval based hidden knowledge discovery

from biomedical literature." *Expert Systems with Applications: An International Journal* (August, 2011), vol. 38, no. 8, pp. 9958–9964.

Friedman, Jerome H. (1998). "Data Mining and Statistics: What's the connection?". Computing Science

and Statistics. 29 (1): 3–9.

Koza, John R.; Bennett, Forrest H.; Andre, David; Keane, Martin A. (1996). Automated Design of Both

the Topology and Sizing of Analog Electrical Circuits Using Genetic Programming. Artificial Intelligence in Design '96. Springer, Dordrecht. pp. 151–170. doi:10.1007/978-94-009-0279-4\_9.

Steinbach, Michael & Karypis, George & Kumar, Vipin. (2000). A Comparison of Document Clustering

Techniques. Proceedings of the International KDD Workshop on Text Mining.

Swanson, D. R. (1986). Fish Oil, Raynauds Syndrome, and Undiscovered Public Knowledge.

*Perspectives in Biology and Medicine,30*(1), 7-18. doi:10.1353/pbm.1986.0087

Turing, A.m. “Computing Machinery And Intelligence.” Readings in Cognitive Science, 1988, pp. 6

19., doi:10.1016/b978-1-4832-1446-7.50006-6.

**Appendix**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **RB** | **RBR** | **GRAPH** | **AGGLO** | **DIRECT** |
| MEAN | 0.004049 | 0.004043 | 0.041708 | 0.00021 | 0.00399 |
| MEDIAN | 0.004015 | 0.004025 | 0.040275 | 0.00025 | 0.004 |
| MODE | 0.00401 | #N/A | #N/A | 0.0003 | 0.004 |
| STD DEV | 7.62168E-05 | 0.000176014 | 0.002590887 | 0.000144568 | 0.000262488 |
| ERROR | 0.00392 - 0.0040 | 0.0039 - 0.0041 | 0.04001 - 0.0434 | 0.0001 - 0.00030 | 0.0038 - 0.0041 |

**Figure 1:** This is a table with the mean, median, mode, standard deviation, and the standard error for each clustering algorithm. All the numbers in the table below are measured in seconds.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RB | RBR | GRAPH | AGGLO | DIRECT |
| TRIAL 1 | 0.00401  sec | 0.00403  Sec | 0.0401  Sec | 0.0001  sec | 0.0044  sec |
| TRIAL 2 | 0.00403  sec | 0.0044  Sec | 0.04  Sec | 0.0003  sec | 0.0039  sec |
| TRIAL 3 | 0.00399  sec | 0.00422  Sec | 0.048  Sec | 0  sec | 0.004  sec |
| TRIAL 4 | 0.00401  sec | 0.00389  Sec | 0.0403  Sec | 0.0003  sec | 0.0035  sec |
| TRIAL 5 | 0.00402  sec | 0.00402  Sec | 0.0441  Sec | 0  sec | 0.0038  sec |
| TRIAL 6 | 0.00398  sec | 0.004  Sec | 0.04001  Sec | 0.0002  sec | 0.0042  sec |
| TRIAL 7 | 0.004  sec | 0.00399  Sec | 0.0439  Sec | 0.0001  sec | 0.004  sec |
| TRIAL 8 | 0.00421  sec | 0.00408  Sec | 0.04009  Sec | 0.0003  sec | 0.0037  sec |
| TRIAL 9 | 0.00406  sec | 0.0037  Sec | 0.04025  Sec | 0.0004  sec | 0.0041  sec |
| TRIAL 10 | 0.00418  sec | 0.0041  Sec | 0.04033  Sec | 0.0004  sec | 0.0043  sec |
|  |  |  |  |  |  |

**Figure 2**: This is a chart of the runtimes for each of the 10 trials that were run for each clustering algorithm.

**Figure 3**: This bar graph shows the average amount of seconds taken to cluster the same data set, but with different clustering algorithms. There were 10 trials for each level of the IV, but only the average time was shown for each clustering algorithm.

'''The program compares the runtimes for each clustering algorithm to cluster a dataset. Each clustering algorithm is run 10 times, for a total of 50 different runtimes (There are 5 different clustering algorithms used.) The data is imported from the CLUTO website. CLUTO is a clustering software. '''

data\_file = 'genes1.mat'

timings = open("Project\_timings.txt", 'w')

import subprocess

'''This list called clustering\_types contains all 5 of the clustering algorithms used in this project.'''

clustering\_types = ['rb', 'rbr', 'graph', 'agglo', 'direct']

'''A 'for loop' is used to run each clustering algorithm 10 times. Inside of this main loop, there is a smaller 'while loop' containing a smaller 'for loop.''''

'''This for loop goes through each clustering method and creates a part of the prompt that is to be placed in the commandline (the type of clustering) '''

for index in clustering\_types:

clustering\_algorithm = index

clustering\_prompt = '-clmethod='+clustering\_algorithm

i = 0

print(type(index))

timings.write('\n')

timings.write(index)

'''this while loop actually allows the code to run 10 times and allows for repeated trials'''

while (i < 10):

return\_string = subprocess.check\_output(["./vcluster",clustering\_prompt, 'genes1.mat', ' 10'])

new\_line = return\_string.split('\n')

#this for loop goes through each line and gets the timings

for line in new\_line:

if line.startswith(' Clustering:'):

timings.write('\n')

print(type(line))

timings.write(line)

i = i + 1

#this closes the file, timings, so that the changes can be permanent

timings.close()