MapReuse and MapRecycle : Two More Frameworks for Eco-Friendly Data Processing

Mary McGlohon
Carnegie Mellon University
Machine Forgetting Department
5000 Forbes Ave.
Pittsburgh, Penn. USA
mmcgloho@cs.cmu.edu

ABSTRACT

MapReduce has revolutionized data processing for the more environmentally-minded. This work presents an additional two frameworks for eco-friendly data processing: MapReuse and MapRecycle. In both frameworks, like MapReduce, users specify a *map* function that processes a key/value pair to generate a set of intermediate key/value pairs. Then, users specify either a *reuse* or *recycle* function, depending on how much government funding they receive.

1. INTRODUCTION

As discussed in [5], cycle depletion has reached crisis levels. Therefore, a more environmentally-friendly approach to data processing is critical to the continuance of large-scale data applications such as web search, multimedia streaming, and TheFacebook.

While MapReduce has begun to simplify this process, we present two more frameworks, MapReuse and MapRecycle, to give users more ways to process data while respecting Mother Nature. We present these frameworks to be used in favor of less environmentally-friendly data processing processes, such as overfitting, genetic program engineering, strip data mining (which is not only unenvironmental but often visually offensive, see Figure 1), and clustering baby seals (see Figure 2).

A general overview for the execution of MapReduce, MapReuse, and MapRecycle may be found in Figure 3.

2. MAPREUSE

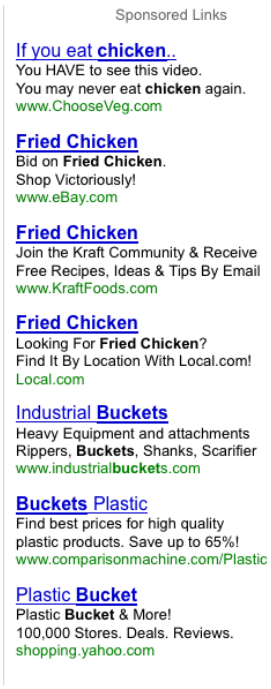
While *reduce* functions reduce the data into more digestible parts, *reuse* functions do not require making new data at all, but rather allow the user to reuse data and/or research results. Some examples of using *MapReuse* include:

- Share your sensitive data with the general public of researchers. Being eco-friendly is more imporant than user privacy [7].
- Bootstrapping [6].
- Plagiarism [1].
- Survey papers and journal submissions.

3. MAPRECYCLE

Sometimes your data suck and MapReduce and MapReuse fail to produce useful results. For these cases, MapRecycle can help make new data or research projects out of it. Thus, an alternate form of cleaning up data processing is to use *recycle* functions. Some examples of *recycle* functions include:

- Compacting and composting of garbage collection [2].



(redacted for improved user experience)

Figure 1: A researcher engages in the frowned-upon practice of strip data mining.

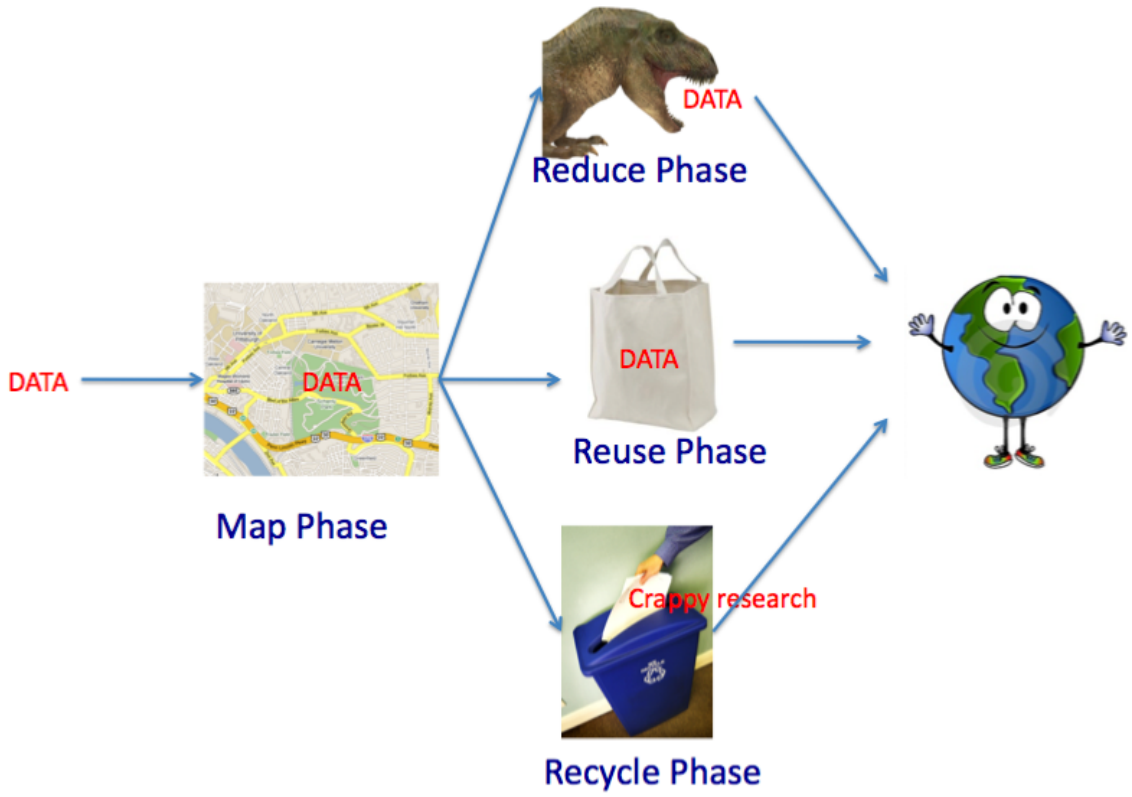


Figure 3: The exeuction overview for the three frameworks of eco-friendly data processing.

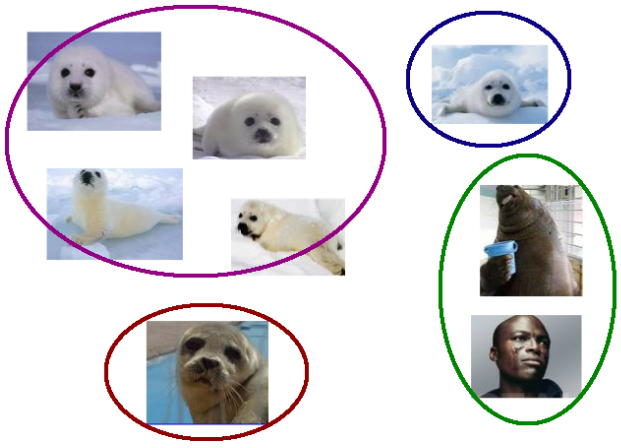


Figure 2: A paint program user engages in the frowned-upon practice of seal clustering.

- Incinerating data that disagrees with intended results.
- Switching research projects. Salvageable portions can often be used as tech reports and "experience". Research advisors may sometimes be recycled.
- Dropping out of grad school to join a startup. This typically successfully produces a free Master's degree and cushy salary, with a by-product of resentment from fellow grad students.

The *recycle* portion has several steps: collection, sorting, processing, and re-constructing.

Collection involves deciding whether or not to do away with different parts of your data or research and kick them to the curb. One important factor in the collection step is the *recycle bin*. Specifying the recycle bin requires one to determine an appropriate bin width. The suggested bin width is:

$$h = 42picas$$

Sorting organizes the crappy data into different types of crap. A detailed review of sorting algorithms may be found in [3].

Processing is the center of the *recycle* function. Here different types of lousy data are consolidated into reuseable components. The exact processing method is determined by the user. For example, the user may use simulated annealing to change the properties of the data. Finally, re-construction of data into a more useful format or project is done with the processed components.

Figure 4: Database management researchers object to environmental soundness in data processing via MapReduce, MapReuse, and MapRecycle. That's because they want all the data to themselves. Jerks.

4. OPEN SOURCE IMPLEMENTATION

MapFreecycle™is made a wide network of users who contribute data and code implementing MapReduce/Reuse/Recycle functions. [1] It is implemented in Free-Trade Java and is based on GreenFS. Clusters for MapFreeCycle have been donated to several universities and third-world villages by Yahoo!.

5. DISCUSSION AND CONCLUSIONS

Other notable information about data mining safety and environmental friendliness can be found in [4]. Database management researchers have objected to many data cleaning methods. We call these people *hoarders* (See Figure 5).

We propose these frameworks to solve problems that were formally non-eco-solvable, such as the Air-Traveling Salesman Problem and the Cloud Covering Problem.

6. REFERENCES

[1] C. Dinwoodie. Plagiarism. In *The 8th Biarennial Workshop about Symposium on Robot Dance Party of Conference in Celebration of Harry Q. Bovik's 0x40th Birthday*, Apr. 2009.

[2] J. Donham. Compacting, composting garbage collection. In *The 6th Biarennial Workshop about Symposium on Robot Dance Party of Conference in Celebration of Harry Q. Bovik's 0x40th Birthday*, Apr. 2007.

[3] D. E. Knuth. *Sorting and Searching*, volume 3 of *The Art of Computer Programming*, section 1.2, pages 10–119. Addison-Wesley, Reading, Massachusetts, second edition, 10 Jan. 1973. This is a full INBOOK entry.

[4] M. McGlohon. Data mining disasters: A report. In *The 7th Biarennial Workshop about Symposium on Robot Dance Party of Conference in Celebration of Harry Q. Bovik's 0x40th Birthday*, Apr. 2008.

[5] J. M. Newcomer and C. B. Weinstock. Cycle depletion–a worldwide crisis. In *The 6th Biarennial Workshop about Symposium on Robot Dance Party of Conference in Celebration of Harry Q. Bovik's 0x40th Birthday*, Apr. 2007.

[6] L. Wasserman. *All of Statistics*. Pink Book Publishing, 2004.

[7] Wikipedia. Aol search data scandal. `http://en.wikipedia.org/wiki/AOL_search_data_scandal`, 2009 (accessed).

[1] Due to the anarchist leanings of many members of the open source community, this was originally implemented as *MapDumpsterDiving*. However, it was unsettling to the wider audience.