Richard Hayes Crowley

03/15/2021

CSC\_157\_Lab\_07\_QA

**(1)** Review the article below that discusses the theory and construction of a bag of words. What are some practical or business uses for a bag of words?

[**https://www.excelr.com/blog/data-science/natural-language-processing/implementation-of-bag-of-words-using-python**](https://www.excelr.com/blog/data-science/natural-language-processing/implementation-of-bag-of-words-using-python)

**Constructing a “bag of words” wherein the number of words in a text are counted, and then constructing certain “vectors” wherein the words reside (e.g., positive vectors like “great”, “fun”, “cool” and negative vectors like “bad”, “terrible”, etc.) can be useful for businesses who, for example, want to scrape words from a large number of reviews and target certain reviews to respond to, contact reviewers for testimonials, be alerted to buzz about a topic on twitter, etc. This is a rudimentary form of “sentiment analysis”, a more advanced example of which would be a machine learning algorithm (e.g., Google Cloud’s natural language processing library).**

**(2)** How is a bag similar to an electronic shopping cart that you use to order items from your favorite Books and Music store?

**A bag ADT is similar to an electronic shopping cart in that they are both collections, allow you to view the items within, display the count of all items as well as individual items within, and allow for adding and removing items.**

**(3)** How are bags similar to or different from bins?

**I do not believe we’ve covered bins, nor have I read anything about bins in chapters 1-7 in Lambert. Googling “bin data structure” yields some entries on computational geometry… However, if the question is referring to “bins” in the “merge sort” algorithm from Chatpter 3, then one could say that the “bins” in merge sort (which are essentially partitioned collections of a larger collection) differ from “bags” in that bags are singular collections, while bins represent a part of a larger collection.**

**(4)** For the Bag ADT , which do you prefer and why? A hard - coded

maximum bag size or a user - entered size.

**In the case of an array-based implementation of a bag, I would prefer a hard-coded default size with a *dynamic* limit that expands or contracts the bag based on load factor (some set of rules that establish a threshold for expanding or contracting the size of the bag based on the difference between the logical size and the physical size of the array bag). The only limit in this situation would be the amount of RAM available to the program.**

**For a linked list implementation of a bag, there would be no need to hardcode or ask for a default size, since new items in a linked list are nodes which contain data and a pointer to the next node in the list, and do not need a contiguous physical array of memory. The only limit to a linked list bag would be the space available in the object heap.**

**From a UX (user-experience) perspective, as well as a DX (developer-experience) perspective, it would detract from the usefulness of the implementation if the user, who only wants to think as little as possible, is asked to think about how big their bag would be and to supply a given size or length.**

**However, there ought to be *some* limit to the number of items in a bag to avoid an “out-of-memory” error, and to to keep RAM available for other programs. This would be hard-coded in any circumstance.**

**A quick google tells me that Python has no built-in upper memory limit, and makes all memory in a computer available to your programs, unlike Node.js (a popular library and runtime environment for running JavaScript outside the browser, and which has an explicit upper memory limit that it allocates and which sometimes must be tweaked in order to get expensive programs to run).**

**(5)** It is time for a Grab Bag party. How is a Grab Bag similar or different to a Counting Bag?

**A “grab bag” according to Google / Oxford languages, is a ”container from which a person chooses a wrapped item at random, without knowing the contents,” (Web Search). A “counting bag”, as far as I know, is a bag wherein the contents are known, and furthermore, has “count” accessors that enable users of the counting bag to know how many items of each type are inside.**