==

Programming Assignment - 1

==

Instructions

1. This is a programming assignment that needs to be completed individually.
2. You should write all the code in python.
3. You need to upload the assignment in a zip file with the following naming convention.

roll_no.zip

--indexing

--all codes from part 1 here with a brief readme

--compression

--all codes from part 2 here with a brief readme

--result

--part1.txt

--part2.txt

--arith.txt

Getting the Dataset (obtained from https://www.corpusdata.org/iweb_samples.asp)

1. There are 10 datasets in the drive link. You should choose one that ends with your (RollNo.%10). For example: suppose your roll number is 203050059, you should choose dataset data_9.zip.
2. Download and extract chosen data from cs635_2021_pa_datasets. Use your LDAP ID during login. (make sure there are 50000 documents)

The first part of this assignment is to build an inverted index of this corpus. **70 marks**

1. Sort the filenames in lexicographic order and assign doc-IDs sequentially (0 to N) i.e. N+1 docs in total.
2. Use spacy to tokenize and compute document ID gaps (dgaps) for each token (posting lists). (Note - you should discard non-alphanumeric tokens)
3. For calculating dgap distribution, frame the list as {first docid, dgap1, dgap2, ...}. To be clear, the first docid is also treated as a dgap for calculating the distribution.

4. Submit a text file having N+1 lines. Each line ‘k’ must contain a single integer number, which is the count of the number of times the dgap = k, appeared. Indexing starts from 0, which means the first line of the file represents the number of times dgap=0 appeared.
5. Use filename as: part1.txt

The second part involves index compression methods. **30 marks**

1. Compress the above postings using gamma, Golomb and arithmetics codes. (compulsory to write encoding code yourself)
2. Report the compressed sizes and show comparisons with the original and each other.
3. Submit a text file having 4 lines, such that each line has the original and compressed sizes of the postings using the 3 above codes, in the exact same order as follows:
    1. Line 1: Original posting lists sizes.
    2. Line 2: Gamma code compressed sizes of the posting lists.
    3. Line 3: Golomb code compressed sizes of the posting lists.
    4. Line 4: Arithmetic code compressed sizes of the posting lists.
4. Use filename as: part2.txt

******CLARIFICATIONS*** (26th August)**

- Have all tokens as lowercase. ('TOKENS' and 'tokens' is the same token).
- Even though the posting list is tokens -> {doc_ids}, it is {first_docid, dgap1, ...} which we usually encode (for compression). Follow this convention i.e. for the original size report the size of the 'posting list'. For the compressed sizes, report the size of the 'compressed dgap list'.
- For all purposes, you can assume the maximum size of dgap to be 50,000 (your possible symbols are then the numbers 0-50000). The size is just the number of bits you need to encode the dgap/posting list. For example, the original size is just (32*size of the list) bits. Sort the tokens in the lexicographic order and have the sizes sequentially on 4 lines (as in the assignment statement).
  sample (assuming you have 3 lists for 3 tokens):
  <l1-size> <l2-size> <l3-size>
  <l1-size-gamma> <l2-size-gamma> <l3-size-gamma>
  <l1-size-golomb> <l2-size-golomb> <l3-size-golomb>
  <l1-size-arithmetic> <l2-size-arithmetic> <l3-arithmetic>
- For arithmetic encoding, the probability distribution for dgaps should be obtained from part-1.

*****NO PLAGIARISM*****

Your code will be through a MOSS plagiarism checker.

*****CLARIFICATIONS (2 Septemeber)*****

- We do not require arithmetic encoding in part2.txt. So it will now look like:

<l1-size> <l2-size> <l3-size>
<l1-size-gamma> <l2-size-gamma> <l3-size-gamma>
<l1-size-golomb> <l2-size-golomb> <l3-size-golomb>

- Instead, we require a new file arith.txt which contains **encodings (not the sizes)** of the dgaps in the **first posting list only** (ie the list for the lexicographically smallest token) in the following space-separated format:
  <first_docid> <encoding>
  <dgap1> <encoding>
  <dgap2> <encoding>

- Any details (hyperparameters) you wish to report must be mentioned in the README clearly.

- The deadline is extended till 3rd September 23:59. You can contact the TAs if you want an extension further than that.