Nishant Velagapudi

W251 – Big Data

Homework 4 – GPFS/The Mumbler

**Execution Information:**

Please login to the “gpfs” node (quorum node of the system) using an external IP of 50.23.102.141, with a root password of “Rude98BL” (no quotes). Change to the /gpfs/gpfsfpo/mumbler directory. You can run the mumbler from there as follows: “python mumbler.py <max words> <starting word>.”

**Files:**

* Mumbler.py – runs the mumbler
* Download\_file\_loop\*.sh – scripts to download files, split out so I could download in parallel
* Inputprocessing.sh – takes the individual zipped files, unzips and aggregates one by one to reduce size
* SearchNode\*.sh – uses grep and awk to find records relevant to the word being searched
* mumblerNode.sh – adds node related file endings to each csv for separation of tasks by node

**Explanation:**

The 99 original files are divided into sets of 33. Each set of 33 is hosted on a different node. The inputprocessing.sh script parsed each zip file, aggregated the counts across years (since we aren’t using the year data anyways), and deletes the original zipped and unzipped files. Each node is set to parsing its own set of 33 inputs (SearchNode\*.sh scripts). Each node returns the narrowed down rows to the quorum node, where the sets are combined, probabilities calculated, and a word drawn. This reduces network traffic. Each SSH command times out after 7 seconds: if one node times out, the results from the other two nodes can still be used.

I observe the worst performance for the most common words (“in”, “the”, “and”, etc.). Presumably, these words have the most ngram records and matches and processing is most expensive. In future work, I would try to cache the probabilities for these words specifically, as they are both common and problematic. The common words can take up to a minute to calculate probabilities in the current code – apologies for the delays.