For storing each of the text files a hash was used. In any pair of files that were not identically named (ie. comparing f1 to f1), the latter file would be stored in a hash, and the former file would be compared against that hash without itself being hashed. Once done, that hash would be cleared and the next two files would start the process.

Doing the process this way saves on memory since only one whole file is stored in memory at a time; however, this process is dreadfully slow: each file is being tokenized and hashed several times. It would be more efficient with regards to time to just tokenize and hash every file once, and store those hashes in an array of hashes. Then, whenever two files are to be compared, the correct hashes could be immediately retrieved; however, if a plethora of large files were to be used then the system could potentially run out of memory. So if a plethora of large files were to be used where memory might be concern, then the way I have written it is suitable.