

File Classifier (By Date)

Table of contents

The file structure expected by the algorithm	1
The way the algorithm works	2

The file structure expected by the algorithm

The algorithm provides a simple and relatively fast way of classifying unsorted files into groups that works well under certain assumptions:

1. all filenames embed a date (e.g., The New York Times_12-01-1949_2_1);
2. some of the files are already sorted into groups and into distinctive sub folders with a folder;
3. topics in each subfolder Z are temporally spaced out (i.e., the dates of the filenames in each of the m Z s are unlikely to be close together;
4. n unsorted files are stored in a separate directory.

In sum, files are stored in the following way:

1. a folder Y containing m subfolders (Z), each subfolder containing a variable set of documents that talk about topic Z ;
2. a folder X containing a list of n unsorted documents each one of which needs to be placed in one of the m Z subfolders.

Graphically, this is what the file structure would look like:

Name	Date modified	Type	Size
3	2/18/2020 10:14 PM	File folder	
4	2/18/2020 10:15 PM	File folder	
5	2/18/2020 10:15 PM	File folder	
6	2/18/2020 10:15 PM	File folder	
7	2/18/2020 10:15 PM	File folder	

Name	Date modified	Type	Size
3_Jim Cobb	2/18/2020 10:14 PM	File folder	
4_Frank Hardeman	2/18/2020 10:15 PM	File folder	
5_Palseo	2/18/2020 10:15 PM	File folder	
6_Owen Jones	2/18/2020 10:15 PM	File folder	
7_Jet Hicks	2/18/2020 10:15 PM	File folder	

Figure 1 – Different representations of folder Y containing m subfolders (Z)

But whatever naming criteria one adopts, each subfolder contains the articles that describe a specific event. Thus, event number 3 (Jim Cobb, 1918) contains the following articles.
















Name	Date modified	Type	Size
 Charlotte Daily Observer_09-23-1906_1_1.txt	2/8/2020 2:42 PM	TXT File	6 KB
 Chicago Daily Tribune_09-23-1906_1_5.txt	2/8/2020 2:42 PM	TXT File	7 KB
 Chicago Daily Tribune_09-24-1906_8_4.txt	2/8/2020 2:42 PM	TXT File	3 KB
 Daily Press_09-23-1906_1_1.txt	2/8/2020 2:42 PM	TXT File	5 KB
 Dallas Morning News_09-23-1906_2_1.txt	2/8/2020 2:42 PM	TXT File	1 KB
 Dallas Morning News_09-23-1906_2_1_2.txt	2/8/2020 2:42 PM	TXT File	1 KB
 Dallas Morning News_09-23-1906_2_1_3.txt	2/8/2020 2:42 PM	TXT File	1 KB
 Dallas Morning News_09-23-1906_2_1_4.txt	2/8/2020 2:42 PM	TXT File	1 KB
 Dallas Morning News_09-23-1906_2_1_5.txt	2/8/2020 2:42 PM	TXT File	1 KB
 Dallas Morning News_09-23-1906_2_1_6.txt	2/8/2020 2:42 PM	TXT File	1 KB
 Los Angeles Time_09-23-1906_2_4.txt	2/8/2020 2:42 PM	TXT File	4 KB
 Los Angeles Times_09-24-1906_1_2.txt	2/8/2020 2:42 PM	TXT File	8 KB
 Los Angeles Times_09-24-1906_6_1.txt	2/8/2020 2:42 PM	TXT File	1 KB
 New York Times_09-23-1906_1_1.txt	2/8/2020 2:42 PM	TXT File	9 KB
 New York Times_9-24-1906_2_5.txt	2/8/2020 2:42 PM	TXT File	11 KB

Figure 2 –Representation of folder X with n unsorted files

The way the algorithm works

For each of the n files in X we extract the embedded date (let's refer to this as the *source date*) and we then traverse each of the m subfolders Z in Y, extract the date from each of the filenames contained in each subfolder (let's refer to this as the *target date*), and check that source and target dates fall within a user-specified rage (e.g., 5 days, 7 months). If the date comparison fails, we move to the next subfolder until the date comparison passes and we move the file in X containing the source date to the subfolder Z with files that embed the target date.

Computing-intensive solution. The solution is similar to the “Find the intruder” tool based on social actors list combined with NER values for Location, Date, Person, Organization obtained using Stanford CoreNLP. From the variable-number of documents in Z (m of them) we construct a summary index of group values (based on social actors and NER values). For each of the n documents in folder X, we construct a similar index based on social actors and NER values and then compare this value with the index of each Z subfolder to see if the currently processed document in X belongs to group Z.