KITAB Textreuse Data, ver. 2023.1.8
—Release Notes—

This dataset is the text reuse datafiles of the KITAB project as well as the corresponding statistical data, generated based on the OpenITI corpus, version 2023.1.8, using passim.

The text reuse data file (called 'KITAB-TextReuse-pairewise_2023-1-8.zip') contains 1,641,766 CSV files of pairwise (book-to-book) text reuse alignments with a total of 51,491,084 alignments between 8,600 books, authored by 3,306 authors.

The alignments in the CSV files are defined by a set of fields (described below) to show the exact position as well as character-to-character alignments of the aligned piece of text in each book. Other than the fields that passim generated, we also add a few fields to the alignments in the post-processing step.

- id1: human-readable identifier of the milestone[1] in which the alignment was found in Book 1 (versionID (without author and title components) + file extension + milestone number; 'id' field in the input data)
- seq1: milestone number of the alignment in Book 1
- uid1: numeric unique ID of the milestone in Book 1
- ser1: human-readable identifier for Book 1 ('series' field in the input data)
- gid1: 64-bit long hash of the human-readable identifier for Book 1
- s1: the alignment string of the aligned passage Book 1; contains a dash for each character that is added in the passage in Book 2
- bw1: word[2] index of the start of the alignment in the milestone in Book 1
- ew1: word index of the end of the alignment in the milestone in Book 1
- b1: character index of the start of the alignment in the milestone in Book 1
- e1: character index of the end of the alignment in the milestone in Book 1
- len1: number of characters in the alignment in Book 1
- tok1: number of words in the alignment in Book 1
- id2: human-readable identifier of the milestone in which the alignment was found in Book 2 (versionID (without author and title components) + file extension + milestone number; 'id' field in the input data)

[1] Passim, was built for newspaper articles; although it has evolved since, we still find it useful to cut our (sometimes very long) texts into shorter chunks of 300 Arabic-script words (about a page of printed text). In order to be able to easily link these chunks back to the original text, we include a "milestone" tag (ms001, ms002, ...) in the text after every 300 Arabic-script words.

[2] We use non-Arabic characters, including white spaces, to split the texts into words. This may or may not give us the same list of tokens.

- seq2: milestone number of the alignment in Book 2
- uid2: numeric unique ID of the milestone in Book 2
- ser2: human-readable identifier for Book 2 ('series' field in the input data)
- gid2: 64-bit long hash of the human-readable identifier for Book 2
- s2: the text string of the aligned passage Book 2; contains a dash for each character that is added in the passage in Book 1
- bw2: word index of the start of the alignment in the milestone in Book 2
- ew2: word index of the end of the alignment in the milestone in Book 2
- b2: character index of the start of the alignment in the milestone in Book 2
- e2: character index of the end of the alignment in the milestone in Book 2
- len2: number of characters in the alignment in Book 2
- tok2: number of words in the alignment in Book 2
- first1: gid1 < gid2 (true/false)
- first2: gid2 < gid1 (true/false)
- score: score the algorithm gave to the alignment
- matches: number of matching characters in the alignment
- ch_match: number of matching characters at the same position in both s1 and s2 excluding white space
- align_len: length of the alignment strings in characters, which includes the dashes and white spaces
- Matches_percent: matches/align_len * 100
- w_match: number of words matched between the aligned text

Book-to-book statistics of the alignments between each book pair, for which passim has found at least one reuse case, are collected in the 'KITAB-TextReuse-stats_2023-1-8.csv.gz' file. For each pair, there is various information, such as the number of alignments, word and character counts, etc., which is described below.

- _T1: short version ID of T1 (without language component and extension)
- _T2: short version ID of T2 (without language component and extension)
- instances: number of text reuse alignments between T1 and T2
- WM2_Total: total number of words in T2 that are fully matched in any alignment with T1
- WM1_Total: total number of words in T1 that are fully matched in any alignment with T2
- CM2_Total: total number of characters in T2 that are matched in any alignment with T1
- CM1_Total: total number of characters in T1 that are matched in any alignment with T2

- ch_MATCHES_Min: minimum number of matching characters in alignments between T1 and T2
- ch_MATCHES_Max: maximum number of matching characters in alignments between T1 and T2
- ch_MATCHES_Mean: mean number of matching characters in alignments between T1 and T2
- date_B1: death date (hijri era) of the author of T1
- date_B2: death date (hijri era) of the author of T2
- author_B1: Latin-script names of the author of T1 (separated by " :: ")
- author_B2: Latin-script names of the author of T2 (separated by " :: ")
- book1: book URI of T1 (<date><author>.<title>)
- book2:| book URI of T2 (<date><author>.<title>)
- tok_length_B1: number of Arabic-script words in T1
- tok_length_B2: number of Arabic-script words in T2
- ch_length_B1: number of Arabic-script characters in T1
- ch_length_B2: number of Arabic-script characters in T2
- WM_B2inB1: percentage of words in T2 that are fully matched in T1 (= WM2_Total / tok_length_B2)
- WM_B1inB2: percentage of words in T1 that are fully matched in T2 (= WM1_Total / tok_length_B1)
- CM_B2inB1: percentage of Arabic-script characters in T2 that are matched in T1 (= CM2_Total / ch_length_B2)
- CM_B1inB2: percentage of Arabic-script characters in T1 that are matched in T2 (= CM1_Total / ch_length_B1)
- chrono_B1B2: chronological order of the alignment of the pair (book1, book2)
- chrono_B2B1: chronological order of the alignment of the pair (book2, book1)