Assignment 5: Emoji space 2.0

In this assignment you will preprocess the words included in the streamed tweets and use them to improve the Emoji space.

1 Load/acquire data

1.1 Tweets

Load your data.

Again, to work well, latent semantic analysis requires a substantial amount of data. I recommend, in principle, using a dataset rich in Emojis containing 100,000 tweets or more. However, including words in the analysis will result substantially slower code execution. Thus, while testing your code better use a smaller amount of tweets (1,000 - 10,000).

1.2 Emojis

Load the new Emoji list. Remember to remove Emoji 2283, if it creates errors.

2 Preprocess words

Extract words and apply a series of preprocessing steps.

2.1 Extract

Split tweets in individual words using `stri_split()` and `" "` as the regex pattern. Put all words in a single vector called `words`.

2.2 Convert `tolower`

Convert all words to lower case using `stri_trans_tolower()`.

2.3 Remove Special words

Remove all words that contain one of `http`, `www`, `#`, `0123456789`, `rt`, `&amp` using `stri_detect_regex()`.

2.4 Remove Stopwords

Load stopword list from **here** and remove all stopwords from `words` using `%in%`.

2.4 Remove punctuation

Remove all punction from words using `stri_replace_all_regex()` and `[:punct:]`.

2.5 Remove Emojis

Remove all Emojis - we want to include them using the Emoji list - from words using `stri_detect_regex()`.

2.6 Remove short words

Remove all words with less than 3 characters using `nchar`.

2.7 Stemming

Stem words using `wordStem` from the `SnowballC` package.

3 Rerun last assignment

Select a subset of words and rerun the last assignment using words and Emojis.

3.1 Select words

Select a relatively small number of words, e.g., the 500 most frequent words.

3.2 Create term list

Combine Emojis and the selected words into a `term` vector.

3.3 Rerun analysis

Rerun the last assignment with the `term` vector.

4 Plot result

4.1 Plot term space

Redo the plot of last the last assignment including both Emojis and Words.

4.2 Plot closest associates

Plot the `n` most frequent Emojis and their `k` closest associates (in terms of the cosine similarity). The recomennded layout places the Emojis PNGs in the `n` rows of the left-most column and then fills the columns of each row from left to right with the 'k' closest associates. Choose `k` and `n` somewhere between 10 and 30. Add Emojis using the `add_emoji()` and the associates using `text()`.

4.3 Post your results

Post your results on twitter

END