Documentation for article level MaML-datasets

Most applications of the article level data will likely involve some form of aggregation, both according to the time dimension and according to actors of interest. It is important that you read this note carefully, as you otherwise run the risk of compiling a dataset with substantial errors.

The structure of the datasets

The five article_level_*COUNTRY*.dta datasets contain data from articles WITH and WITHOUT political actors, for each country.

Strictly speaking, it is only the latter type of articles – WITHOUT actors – which holds article level information. In the former case – articles WITH actors – rows of the dataset contain actor-article information: Each actor in each article has its own row. The data in these rows thus represent aggregations of information from the full text of news articles, by article and actor. So that each row represents the presence of individual actors (eg "A_NO_1003" in the *ids* column) or party actors (eg "P_1435_mfs" or "P_1435_mfsa" in the *ids* column) within the given article.

"P_*_mfs" and "P_*_mfsa" represent party presence for full party name and party abbreviation (mfs) or full party name, party abbreviation and any individual actors that are members of the party (mfsa). In practical terms, if you are interested in working with data on the party level you would usually filter/subset rows that contain the text "mfsa" in the *ids* column. This will let you keep one row per party from each article (NB – no row if a party does not appear in a given article, see below).

Note that actors and parties that did not appear in an article WILL HAVE NO ROWS for that given article id. This means that when you work with party presence in the news, you should consider running procedures that will create observations of “no presence” when this is theoretically meaningful.

In the case of articles WITHOUT actors, rows of the dataset contain true article level information: Each article without actors is thus represented by only one row, with the value "all" in the *ids* column and the value "0" in the *actor_occ* column (since there are 0 sentences with actors in the article). Note that these rows only have data in columns that relates to the full text of the article (*text_**), and not those columns that are based on actor presence (*actor_**, apart from the already mentioned 0s in the *actor_occ* column).

For articles with actors, we have created an equivalent row with "all" in the *ids* column. Please note that this represents the aggregation of all actors in the article. THIS MEANS THAT IF YOU WANT TO WORK WITH DATA ONLY ON THE ARTICLE LEVEL, YOU CAN SUBSET/FILTER ROWS WITH THE TEXT "all" IN THE *ids* COLUMN.

The *actor_** variables are computed over all sentences with actors in an article, while *text_** variables are computed over all sentences in an article, and *noactor_** variables are calculated over all sentences without any actors in an article (see more details below). Note that for articles WITHOUT political actors, the *noactor_** variables would equal the *text_** variables since these articles have no sentences with actors. Therefore *text_** variables contain data, while *noactor_** variables are set to missing in articles WITHOUT actors.

In addition, all rows (regardless of with / without actors) have meta data on newspaper source, country and some string/character variables for dates/time periods. Furthermore, actor metadata (from the /Actors/maml_actors.xlsx file have already been added for simplicity (eg. *function*, *gender*, *party_name*, *party_family_name*, *first_name*, *minister_name* etc).

Variables (most important ones)

id	Article id
country	Country abbreviation
doctype	Source of article (newspaper)
publication	Publication date
ids	Actor id. Note that this could be the id of an individual actor or the id of a party aggregation (_mfs or mfsa). If it's a party, the clean party id (corresponding to ParlGov party ids) is provided in the party_id column.
party_id	Party id in the format: P_*ParlGov party id*. Eg. P_1435 for the Norwegian Conservatives, Høyre.
actor_id	Id for individual actors (also provided in the ids column) in the format used in the actors list (see /Actors/ maml_actors.xlsx): A_*Country abbreviation*_*4 digit number*. Eg. A_NO_1116 for the Norwegian Conservative party leader Erna Solberg.
year*	Columns with publication dates in the described format (yearly, monthly, weekly, daily), for easy aggregations
function	MP, JunMin, Minister, PM, PartyLeader (if it's a party actor, value is set to Party). Note that it is possible for an actor to fulfill multiple functions at the same time (e.g. PartyLeader and MP or Minister). This actor will then have several actor X article rows corresponding to the number of functions he/she fulfilled at that point in time.
major_topic	The CAP-like issue code
non_domestic	1=articles with no relevant domestic content
actor_occ	Number of actor sentences
text_sentences / noactor_sentences	Number of sentences in whole article text / Number of sentences without actors.
actor_prom	Number of actor sentences divided by total number of sentences in article
actor_first / noactor_first	Position of first actor sentence (e.g. in sentence 3 of the article) / Position of first sentence without actors.
actor_rel_first	1-(position of first actor sentence divided by total number of sentences in article)
gender	1 = female (for individual actors, not for parties)
ministry_id	id of the ministry the actor belonged to at that time (for ministers only)
actor_sent / text_sent / noactor_sent	Sentiment score (for actor sentences, whole article text or sentences without actors), based on the trinary score of individual sentences. Weighted by length of sentences in number of words

actor_sent_words / text_sent_words / noactor_sent_words

Number of words with sentiment (in sentences containing actors, whole article text or sentences without actors).

actor_words / text_words / noactor_words

Number of words (in all sentences containing actors, whole article text or sentences without actors.

actor_arousal / text_arousal / noactor_arousal

actor(text/noactor)_sent_words divided by actor(text/noactor)_words.

Notes on functions and aggregations

As noted in the variable list above, some actors fulfill multiple functions at the same time (e.g. party leader, MP or Minister). These actors will then have the same number of actor X article rows as the number of functions they fulfilled at a specific point in time. However, these duplicates are usually not problematic since most applications of the data will be based on one of the functions. If several functions are studied at the same time (eg. MPs and ministers/PM), please be aware that you have to make decisions about how to treat the different functions, and then of course implement a procedure to get the observations you need – without duplicates.

Do not aggregate **occ*, **prom*, **first*, **rel_first*, **sent*, **sent_words*, **words* and **arousal* across parties AND individuals belonging to the parties. Use the **_mfsa* aggregations instead.

When aggregating across parties or across individuals, the meaning of the resulting measures changes (with the exception of *actor_first* and *actor_rel_first*). Because multiple actors/parties can occur in the same sentence, the theoretical maximum of the different measures becomes undetermined. For scores ranging from 0 to 1 this means they can theoretically go beyond 1 (or -1, for sentiment). For word counts, the count of words can exceed the total number of words in the document.