Nostalgia in European Party Politics: A Text-Based Measurement Approach

Stefan Müller and Sven-Oliver Proksch

British Journal of Political Science

This README file contains information on all R and Python scripts used to produce tables and plots reported in the main paper and Supporting Information.

Note: If you are interested in the PolNos datasets, we recommend accessing the datasets, corpora, and codebook from the following Harvard Dataverse repository:

> Müller, S. and S.-O. Proksch (2023). PolNos: Political Nostalgia in Party Manifestos. Harvard Dataverse, V1. URL: https://doi.org/10.7910/DVN/L198GI.

Below, we describe each file, the required input data, and outputs (temporary data, figures, tables). The R package versions required to reproduce all results are listed at the start of each script. If the code does not run, package updates might have resulted in these errors. The authors ran all code scripts successfully on 4 September 2023. To reproduce the code, you can install the R version listed in the R script or simply use the *groundhog* package.[1] In order to run the R scripts, we recommend creating RProject[2] in the folder containing all data and scripts. Alternatively, you can use the *here*[3] package or set the working directory with the `setwd()` function. We strongly recommend downloading files in the original format (select "Download" -> "Original File Format").

01a_apply_classifiers_and_text_descriptives.R

Apply the dictionaries and classifiers to the sentence-level machine-translated corpus. It also loads and merges the DistilBERT classification which is conducted in `01c_predict_nostalgia_distilbert.py`. The file also produces tables and plots involving quantitative text analysis (frequencies, keyness, comparison of nostalgia across economic, cultural, and other areas)

Input:

- `data_corpus_translated_full.rds`: machine-translated corpus
- `dictionary_full.rds`: nostalgia dictionary in quanteda format
- `data_coded_all.csv`: all coded sentences used for training and validating the classifiers
- `data_classified_bert1.csv – data_classified_bert6.csv`: DistilBERT predictions

[1] Change `library(name_of_package)` to `groundhog::groundhog.library(name_of_package, date = "2023-09-04")` after installing groundhog: `install.packages("groundhog")`. More details are provided in the R scripts and at https://groundhogr.com/using/

[2] https://support.posit.co/hc/en-us/articles/200526207-Using-RStudio-Projects

[3] https://here.r-lib.org

- data_cmp_codes.csv: hand-coding of policy areas into cultural, economic, other
- CPDS_1960-2018_Update_2020.dta: CPDS dataset
- populist-version-2-20200626.xlsx: PopuList dataset
- data_cmp_main.rds: Manifesto Project manifesto-level dataset

Output:
- Data: data_nostalgia_manifestolevel.rds (used in subsequent scripts)
- Figures: fig_a12.pdf, fig_a13.pdf
- Tables: tab_a05.tex, tab_a06.tex, tab_a07.tex, tab_a15.tex

01b_fine-tune_distilbert.py

Script to fine-tune and validate DistilBERT nostalgia models. data_coded_train.csv and data_coded_test.csv contain the sentences used to validate the SVM model in the R script.

Input:
- data_coded_train.csv: spreadsheet with sentences used for training
- data_coded_test.csv: spreadsheet with sentences for out-of-sample predictions

Output:
- Data: data_classified_bert_test.csv: data with DistilBERT predictions for test set
- Objects: Files in folder distilbert_nostalgic, containing the fine-tuned DistilBERT classifier. To preserve the folder structure, download all files from the Dataverse at once. Alternatively, move config.json, pytorch_model.bin, and training_args.bin into a folder named distilbert_nostalgic.

01c_predict_nostalgia_distilbert.py

Predict nostalgia in full dataset. Note: if you would like to try this classifier with your own data, we recommend following the tutorial in tutorial_classify_nostalgia_distilbert.ipynb.

Input:
- data_coded_train.csv: spreadsheet with sentences used for training
- dat_full_bert1.csv – dat_full_bert6.csv (CSV files containing sentences, created in 01a_apply_classifiers_and_text_descriptives.R)

Output:
- Data: data_classified_bert1.csv–data_classified_bert6.csv: six spreadsheets containing DistilBERT predictions

02_analysis_main_paper.R

R script resulting in plots and tables reported in the main paper.

Input:
- `data_nostalgia_manifestolevel.rds`: manifesto-level dataset
- `data_coded_gpt-3.5_clean.csv`: script containing hand-coding, classification, and GPT-3.5 evaluation of 50 manifestos selected for analysis

Output:
- Figures: `fig_02.pdf/png, fig_03.pdf/png` (Figure 1 is a flowchart that does not require any code)
- Tables: `tab_01.tex/html, tab_02.tex/html`

03_analysis_si_section_a.R

R script resulting in plots and tables reported in SI Section A.

Input:
- `data_nostalgia_manifestolevel.rds`

Output:
- Figures: `fig_a01.pdf, fig_a02.pdf`
- Tables: `tab_a01.tex`

Note: SI Section B does not include any code. Therefore, a replication file is not required

04_analysis_si_section_c.R

R script resulting in plots and tables reported in SI Section C.

Input:
- `data_coded_round_3.csv`: hand-coded sentences for final coding round
- `data_coded_train_clean.csv`: training data for classifiers
- `data_coded_test_clean.csv`: data for out-of-sample predictions
- `data_classified_bert_test.csv`: DistilBERT predictions from `01b_fine-tune_distilbert.py`
- `data_nostalgia_manifestolevel.rds`: manifesto-level dataset
- `nostalgia_survey_pew.csv`: aggregated, country-level data from PEW survey on nostalgia
- `parties_parlgov_2023.csv`: ParlGov dataset of parties (2023 version)
- `1999-2019_CHES_dataset_means(v2).csv`: harmonised CHES dataset, 1999-2019

Output:
- Figures: `fig_a03.pdf, fig_a04.pdf, fig_a05.pdf, fig_a06.pdf`
- Tables: `tab_a02.tex, tab_a03.tex, tab_a04.tex`

05_analysis_si_section_d.R

R script resulting in plots and tables reported in SI Section D.

Input:
- `data_nostalgia_manifestolevel.rds`: manifesto-level dataset
- `data_coded_gpt-3.5_clean.csv`: script containing hand-coding, classification, and GPT-3.5 evaluation of 50 manifestos selected for analysis

Output:
- Tables: `tab_a08.text, tab_a09.tex`

06_analysis_si_section_e.R

R script resulting in plots and tables reported in SI Section E.

Input:
- `data_nostalgia_manifestolevel.rds`: manifesto-level dataset
- `parties_parlgov_2023.csv`: ParlGov dataset of parties (2023 version)
- `1999-2019_CHES_dataset_means(v2).csv`: harmonised CHES dataset, 1999-2019

Output:
- Figures: `fig_a07.pdf, fix_a08.pdf, fix_a09.pdf, fig_a10.pdf, fig_a11.pdf`
- Tables: `tab_a10.tex, tab_a11.tex, tab_a12.tex, tab_a13.tex, tab_a14.tex`

07_tutorial_classify_nostalgia_distilbert.ipynb

Python tutorial on how to import fine-tuned DistilBERT models and how to apply model to sentences stored in a spreadsheet.

Input:
- data stored in folder `distilbert_nostalgic`
- You own sentences, stored (for example) as a csv file with one sentence per row

Output:
- Sentence predictions in the script or – alternatively – stored as a csv file, along with the variables included in the input dataset

function_theme_base.R: custom ggplot2 scheme