Replication Data for: Proksch, Wratil and Wäckerle, 2018, Testing the Validity of Automatic Speech Recognition for Political Text Analysis. Political Analysis, forthcoming.

Preface

The sections below describe what the R code provided in the replication files will do. Some parts of the replication might take a long time to run, specifically the simulations. You can skip most dataset parts of the analysis and, provided you downloaded the replication files in full, the code should run. In one instance, you need to download external data. These instances are indicated below. Additionally, the replication file already comes with R datasets that let you skip the files generating these datasets. This will save you a lot of time, specifically for the wersim simulations (see details below). Essentially, you can either reproduce all steps of the analysis, or you can skip some steps like corpus construction and simulations and go right to the models and graphs. In the R code, most of these things should be clearly explained as well.

R Packages

For the replication, you will need the following packages:

- quanteda
- readtext
- extrafont
- tidyverse
- wersim (download from github with code in R file 1.2)
- austin (download from github with code in R file 1.3)
- stm
- RecordLinkage
- stargazer
- stringr
- Hmisc
- plm
- MCMCpack
- rjags
- coda
- googleLanguageR
- xtable

The extrafont package requires running the function “font_import()” once on each machine. This will take several minutes and might create some error messages. The code will then run normaly, so you can disregard these error messages. If you want to run the code without using the extrafont package you simply have to erase all font changes from the figures in the code (family=”Verdana”). The figures will then be produced in the default font and will look slightly different from those in the paper.

For the rjags package to work, you will need a version of JAGS. From the rjags documentation:

"JAGS is a clone of BUGS (Bayesian analysis Using Gibbs Sampling). See Lunn et al (2009) for a history of the BUGS project. Note that the rjags package does not include a copy of the JAGS library: you must install this separately. For instructions on downloading JAGS, see the home page at http://mcmc-jags.sourceforge.net."

Lunn D, Spiegelhalter D, Thomas A, Best N. (2009) The BUGS project: Evolution, critique and future directions. Statistics in Medicine, 28:3049-67.

We provide the exact versions we used for running the code at the top of each R file and test whether you have them installed. Having a different package version does not necessarily mean that the code won't work, but sometimes functions change over time and we want to make sure that this is not the reason for the code failing. You can install old versions of packages via the "install_version" command in R.

Running time

Here is the approximate running time for all R files.
Please note that the running time will differ depending on the system you use.

- 1.1: 1 minute
- 1.2: 35 minutes
- 1.3: 25 minutes
- 1.4: 4 hours
- 2.1: 1 minute
- 2.2: 2 minutes
- 2.3: 4 minutes
- 2.4: 25 hours
- 3.1: 1 minute
- 3.2: 15 minutes
- 3.3: 15 hours

These times are based on the following system:
- iMac (Retina 5K, 27-inch, 2017)
- Processor: 3,4 GHz Intel Core i5
- Memory: 8 GB 2400 MHz DDR4

R files

1.1_create_soteu_corpora.R: This script creates corpora from the raw transcripts obtained from YouTube and the Google API. Those transcripts are stored in the folder „data"

1.2_soteu_word_error_rate.R: This script calculates the word error rate for all State of the European Union corpora comparing them to human transcriptions. It uses the wersim package that can be downloaded from github (code is in the script).

1.3_soteu_wordfish_sentiment.R: This script calculates the Wordfish and sentiment measures for Youtube and API corpora in English, German and French. It does so for all the

preprocessing specifications laid out in Figure 2. It also runs several additional descriptions and robustness tests that are contained in the Appendix.

1.4_soteu_simulations.R: This script applies the WERSIM procedure to the SOTEU corpora, producing Table 1. It uses the wersim package. The simulations in this script take several hours to run. They can be skipped by loading the two files "soteusimulations_twenty_wf.RData" and "soteusimulations_twenty_senti.RData" as described in the code.

2.1_austria_corpus.R: This script combines the raw transcripts from the Austria debates into two datasets: the main one called "textcorpus.austria" and one including minor speakers for the appendix called "textcorpus.austria.appendix". It also calculates the Word Error Rate for a part of the corpus.

2.2_austria_datasets.R: This script runs Wordfish models on the corpora created in 2.1.

This script adds in positional estimates for the parties and combines all countries. It uses the 2014 Chapel Hill Expert Survey (CHES), available at (https://www.chesdata.eu/2014-chapel-hill-expert-survey). The dataset is provided in the "data" folder.

Polk, Jonathan, Jan Rovny, Ryan Bakker, Erica Edwards, Liesbet Hooghe, Seth Jolly, Jelle Koedam, Filip Kostelka, Gary Marks, Gijs Schumacher, Marco Steenbergen, Milada Vachudova and Marko Zilovic. 2017. "Explaining the salience of anti-elitism and reducing political corruption for political parties in Europe with the 2014 Chapel Hill Expert Survey data," Research & Politics (January-March): 1-9.

It then adds in CHES data for the model in different ways (means, medians). Then it does the same for the appendix version.

2.3_austria_models.R: This script runs the models for the Austrian TV debates. It produces Figure 3 in the paper. It also produces the numbers for Table 2 in the paper and tables A3, A4, A5 in the Appendix.

2.4_austria_simulations.R: This script runs the WERSIM procedure for the Austria analysis. It produces Figure 4 in the paper. It uses the wersim package. The simulations in this script take several hours to run. They can be skipped by loading file "austriasimulations.RData" as described in the code.

3.1_mff_corpus.R: This script creates a corpus for the MFF debates, then it does the same for the API version

3.2_mff_models.R: This script runs the models on the MFF data. It produces Figures 5 and 6 and Table 3 in the paper. It also produces Figures A12, A13, A14, A15, A16, A17 and A18 in the Appendix

3.3_mff_simulation.R: This script applies the WERSIM method to the MFF data and produces Figure A19. The simulations in this script take several hours to run. They can be skipped by loading the file "mff_simulations.RData" as described in the code.

4.1_demo_apicalls.R: This script provides a demonstration of how we used the API calls. Please read up on the googleLanguageR package here: https://code.markedmondson.me/googleLanguageR/articles/speech.html You will need own credentials for making API calls and an account for storing the video files that should be transcribed. New accounts come with ample free credit that should be sufficient for most analysis.

Data files

2011_soteu_xxx_xx.txt: There are 10 text files in the data folder that start with this way. They are either the transcriptions provided by YouTube or the transcriptions obtained from the Google API each in English, German and French. The files containing "human" are the human transcriptions we created of the debates. You can find details for how we made these transcriptions in the Appendix. Additionally, there is one file that is the protocol version obtained from the parliament homepage here: http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-%2f%2fEP%2f%2fTEXT%2bCRE%2b20110928%2bITEM-003%2bDOC%2bXML%2bV0%2f%2fEN&language=EN

austria_transcripts_speakers: This folder contains the transcripts for each TV debate in the Austria corpus, split by speaker. The full transcriptions of the debates were downloaded from YouTube and then cut into speakers.

elefantenrunde_24092017_verbatim.txt: This is the human transcription of one debate on September 24, 2017 that we obtained in order to calculate an estimate of the Word Error Rate for the Austria corpus. You can find details for how we transcribed the videos in the Appendix.

elefantenrunde_24092017.txt: This is the full YouTube transcript of the debate on September 24, 2017. We use this to calculate an estimate of the Word Error Rate.

2014_CHES_dataset_expert-level.csv: 2014 Chapel Hill Expert Survey (CHES), available at (https://www.chesdata.eu/2014-chapel-hill-expert-survey).

Polk, Jonathan, Jan Rovny, Ryan Bakker, Erica Edwards, Liesbet Hooghe, Seth Jolly, Jelle Koedam, Filip Kostelka, Gary Marks, Gijs Schumacher, Marco Steenbergen, Milada Vachudova and Marko Zilovic. 2017. "Explaining the salience of anti-elitism and reducing political corruption for political parties in Europe with the 2014 Chapel Hill Expert Survey data," Research & Politics (January-March): 1-9.

Lsde_frenche_germane.RData: This is an RData file containing the extended sentiment dictionaries as used in:

Proksch, S. , Lowe, W. , Wäckerle, J. and Soroka, S. (2018), Multilingual Sentiment Analysis: A New Approach to Measuring Conflict in Legislative Speeches. *Legislative Studies Quarterly*. doi:10.1111/lsq.12218

mff/API: This subfolder contains the API transcripts for the MFF deabtes, cut by speakers.

mff/Youtube: This subfolder contains the YouTube transcripts for the MFF deabtes, cut by speakers.

mff/MFF_APISample: This file is a sample of speeches obtained from the API (at least two per country). It is used to calculate an estimate of the Word Error Rate for the MFF corpus.

mff/MFF_YouTubeSample: This file is a sample of speeches obtained from YouTube (at least two per country). It is used to calculate an estimate of the Word Error Rate for the MFF corpus.

mff/MFF_VerbatimSample: This file is the same speeches as in the two samples before, but transcribed by humans. You can find details for how we made these transcriptions in the Appendix. It is used to calculate an estimate of the Word Error Rate for the MFF corpus.

eu_budgetcontr: This is an overview of receipts and contributions as a percentage of gdp, as downloaded from www.money-go-round.eu.

Metadata files

The files in this folder provide information about speaker names, group affiliation and gender for the speakers in the 2011 State of the European Union debate.

Tables

The folder “tables” contains the tables used in the paper and the Appendix. Please note that the tables in the paper were not created by sourcing those tables, but rather after making some cosmetic adjustments. The information in the tables folder is equivalent to the tables in the paper, but they look slightly different. The folder also contains a .tex file that compiles all tables.