Supplementary Material

Konstantina Dritsa[1], Kaiti Thoma[2], John Pavlopoulos[3], and Panos Louridas[4]

[1,2,3,4]Athens University of Economics & Business, Greece, [3] Stockholm University, Sweden,
[1]dritsakon@aueb.gr, [2]aikelthoma@gmail.com, [3]annis@aueb.gr, [4]louridas@aueb.gr

1 Dataset Description

The dataset includes 1,280,918 speech fragments of Greek parliament members in debate order exported from 5,355 parliamentary sitting record files, with a total volume of 2.12 GB. The speeches extend chronologically from July 1989 up to July 2020. Table 1 shows the contents of the dataset.

member_name	The name of the person speaking. The names of the speakers are provided in the format "last_name patronym first_name (nickname)". In cases with more than one first or last names, the names that belong to the same category (first or last) are connected with a dash, e.g., "merkouri stamatiou amalia-maria (melina)".
sitting_date	The date the sitting took place in the format DD/MM/YYYY.
parliamentary_period	The name and/or number of the parliamentary period that the speech took place in, e.g., "period 1". A parliamentary period is defined as the time span between one general election and the next. A parliamentary period includes multiple parliamentary sessions.
parliamentary_session	The name and/or number of the parliamentary session when the speech took place, e.g., "session 1". A session is a time span of usually 10 months within a parliamentary period during which the parliament can convene and function as stipulated by the constitution. A parliamentary session includes multiple parliamentary sittings.
parliamentary_sitting	The name and/or number of the parliamentary sitting that the speech took place in, e.g., "sitting 1". A sitting is a meeting of parliament members.
political_party	The political party of the speaker, e.g., "new democracy".
government	The government in power when the speech took place as well as the start and end date of the government, e.g., "tzannetaki tzanni (02/07/1989-12/10/1989)"
member_region	The electoral district the speaker belonged to, e.g., "b piraeus" (2nd Piraeus district)
roles	Information about the speaker's parliamentary and/or government roles in a list format, accompanied by their start and end dates, e.g., ["Deputy Speaker of the Parliament (07/03/1989–21/11/1989)"]
member_gender	The gender of the speaker, e.g., "female"
speech	The speech delivered during the parliamentary sitting

Table 1: Contents of the Parliament Proceedings Dataset

An additional column "speaker_info" is available in the dataset, with information that accompanied the name of some speakers in the sitting records in brackets, usually including the parliamentary role of the person speaking or other information about the speaker that the record keeper considered

36th Conference on Neural Information Processing Systems (NeurIPS 2022) Track on Datasets and Benchmarks.

important. The data of this column is not a consistent or complete source of information but we include it in the dataset for reasons of completeness.

2 Resources

- Record files of parliament sittings: `https://zenodo.org/record/6644941`
  DOI: 10.5281/zenodo.6644941
- Dataset & supplementary files: `https://zenodo.org/record/7005201`
  DOI: 10.5281/zenodo.7005201
- Source code: `https://github.com/Dritsa-Konstantina/greparl`

3 System Requirements

In order to reproduce the results, it is necessary to set up the proper folder hierarchy. First clone the repository of the source code with the command:

```
git clone git@github.com:Dritsa-Konstantina/greparl.git
```

Then, download the supplementary files from Zenodo at `https://zenodo.org/record/7005201` and place them in the folder `out_files`.

Download the record files from Zenodo at `https://zenodo.org/record/6644941` and place the folders `original_data` and `_data` in the same folder level with the `src` folder. Move the files `tell_all.csv` and `tell_all_FILLED.csv` in the `out_files` folder.

Create a Python 3 virtual environment and activate it with the following commands:

```
cd greparl
python3 -m venv .parl_env
. ./.parl_env/bin/activate
```

While in the virtual environment, you will need to install the specific Gensim[1] version needed for the Compass approach. It is important to install the specified Gensim version inside the virtual environment as it might clash with other Gensim installations in the system. Then, install Compass [2]; for using Compass in a reproducible manner, we adjusted the source code of Compass to receive as input a seed parameter in its training functions [3]. Install further dependencies from the `requirements.txt` file. The commands are listed below:

```
pip install wheel
pip install --upgrade setuptools
pip install git+https://github.com/vinid/gensim.git
cd src
git clone git@github.com:Dritsa-Konstantina/cade.git
cd cade
python setup.py install
cd ../..
pip install -r requirements.txt
```

The final folder hierarchy should look like this:

```
greparl
├── _data
├── original_data
├── out_files
├── requirements.txt
└── src
```

[1] `https://radimrehurek.com/gensim/`

[2] `https://github.com/vinid/cade`

[3] `https://github.com/Dritsa-Konstantina/cade`

└── cade

In order to run the Jupyter Notebooks, you will need to run the following commands inside the virtual environment:

```
python -m ipykernel install --user --name=.parl_env
jupyter notebook
```

4 Data Collection and Cleaning

We provide further insight on the Greek parliament sitting records as well as an ordered and detailed list of the scripts used for data collection and cleaning.

4.1 Record Format

In what follows we explain the format of the Greek Parliament sitting records. Each record begins with some introductory information such as the date, period, session, sitting, an introductory text, descriptions of procedures. The record then continues with the debate that took place at that sitting. Typically, each speaker's full name is written in full capital letters at the beginning of a new line and is followed by a colon and the corresponding speech. The speech can extend to multiple paragraphs. For specific roles, such as the speech of the chair of the sitting, the format changes slightly, with the person's role preceding their name. In Fig. 1 you can see a the beginning of a record file for the sitting of April 17, 2017.

On several occasions, the records are poorly formatted and do not follow the recommended structure that would help us distinguish speakers and speeches. Sometimes the colon that delimits each speaker from their speech would be absent, each new speaker speech would not be separated with the previous speaker speech by a new line, or speaker names would not be written in capital letters. An instance of a malformed record is depicted in Fig. 2, where all speeches are poorly formatted and not separated by new lines.

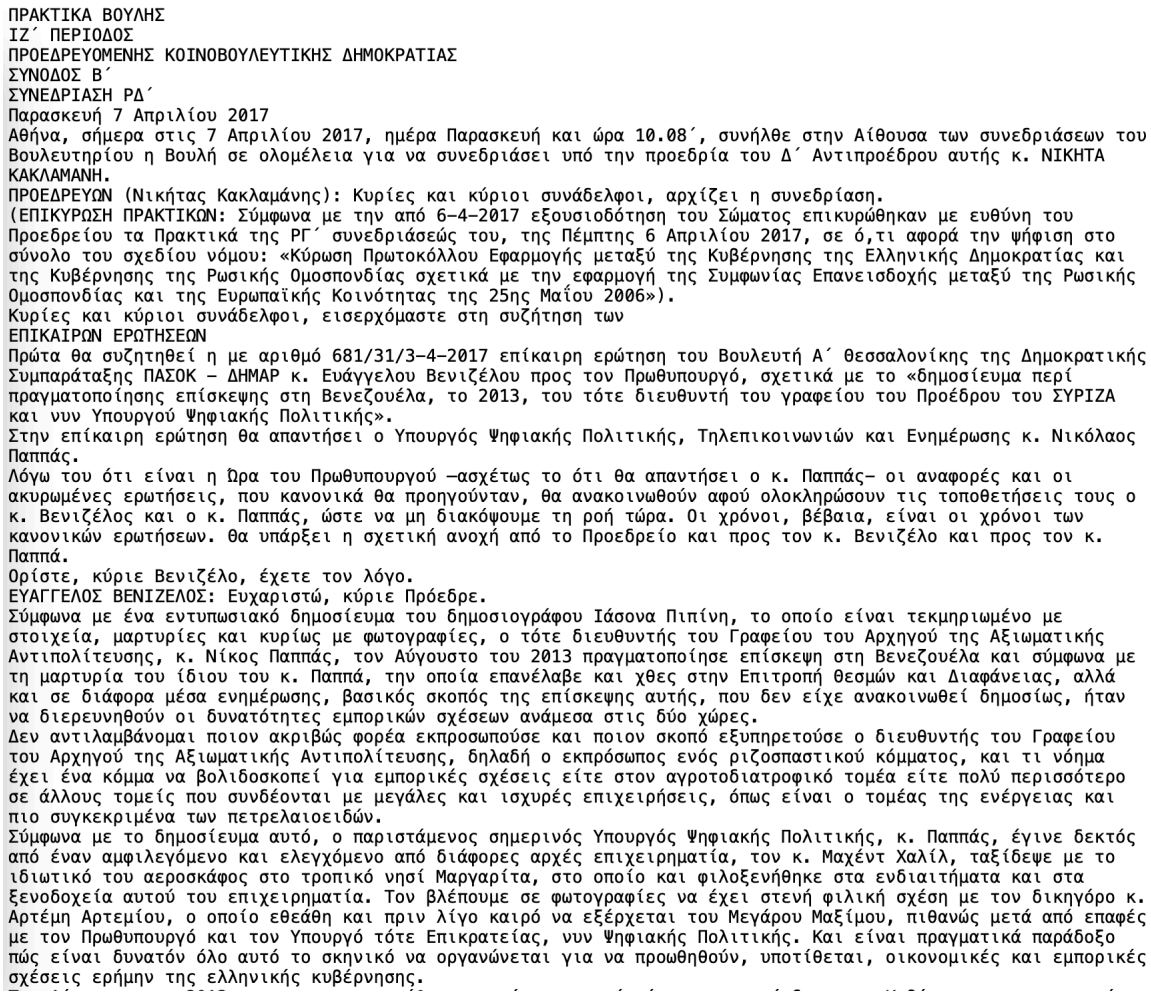
There are several instances with no speaker name but a description, in capital letters, stating that the speaker is "A parliament member from the x political party" or just "A parliament member" or even "Many parliament members", followed by a colon and the speech. In other instances, the beginning of the line that specifies the speaker consists of the role of the parliament member, for example "SPEAKER OF THE PARLIAMENT" (meaning the member of parliament presiding), followed, but not always, by the actual full name of the person in parenthesis. When no name was available but the political party was mentioned before the colon, we kept their speech. When no political party could be detected, we kept the speech with a generic reference.

Finally, while many records do not follow correctly the aforementioned format, there are cases where this format is used for reasons other than indicating a new speech. For instance, capital letters at the beginning of a line followed by a colon and more text can be references to foreign countries or organizations.

4.2 Record Collection and Cleaning

1. **web_crawler_for_proceeding_files.py:** Downloads record files from an HTML table spanning multiple pages available at `https://www.hellenicparliament.gr/Praktika/Synedriaseis-Olomeleias` to the `original_data` folder. It also changes the filenames to match the template `recordDate_id_periodNo_sessionNo_sittingNo.ext`. The `id` is a unique number we assigned to each file. The script logs the rows in the record files table with missing record files in `rows_with_no_files.txt`. It implements the record collection using Selenium[4] and a Chrome driver we include in the `src` folder. In the same folder we include an older version of this script, `web_crawler_for_proceeding_files_old.py`. That script was used to donwload a big part of the proceeding record files of this dataset but since 2020 it stopped working properly due to security software installed in the Greek Parliament website that prevents multiple automated requests.

[4] `https://pypi.org/project/selenium/`



ΠΡΑΚΤΙΚΑ ΒΟΥΛΗΣ
ΙΖ΄ ΠΕΡΙΟΔΟΣ
ΠΡΟΕΔΡΕΥΟΜΕΝΗΣ ΚΟΙΝΟΒΟΥΛΕΥΤΙΚΗΣ ΔΗΜΟΚΡΑΤΙΑΣ
ΣΥΝΟΔΟΣ Β΄
ΣΥΝΕΔΡΙΑΣΗ ΡΔ΄
Παρασκευή 7 Απριλίου 2017
Αθήνα, σήμερα στις 7 Απριλίου 2017, ημέρα Παρασκευή και ώρα 10.08΄, συνήλθε στην Αίθουσα των συνεδριάσεων του Βουλευτηρίου η Βουλή σε ολομέλεια για να συνεδριάσει υπό την προεδρία του Δ΄ Αντιπροέδρου αυτής κ. ΝΙΚΗΤΑ ΚΑΚΛΑΜΑΝΗ.
ΠΡΟΕΔΡΕΥΩΝ (Νικήτας Κακλαμάνης): Κυρίες και κύριοι συνάδελφοι, αρχίζει η συνεδρίαση.
(ΕΠΙΚΥΡΩΣΗ ΠΡΑΚΤΙΚΩΝ: Σύμφωνα με την από 6-4-2017 εξουσιοδότηση του Σώματος επικυρώθηκαν με ευθύνη του Προεδρείου τα Πρακτικά της ΡΓ΄ συνεδριάσεώς του, της Πέμπτης 6 Απριλίου 2017, σε ό,τι αφορά την ψήφιση στο σύνολο του σχεδίου νόμου: «Κύρωση Πρωτοκόλλου Εφαρμογής μεταξύ της Κυβέρνησης της Ελληνικής Δημοκρατίας και της Κυβέρνησης της Ρωσικής Ομοσπονδίας σχετικά με την εφαρμογή της Συμφωνίας Επανεισδοχής μεταξύ της Ρωσικής Ομοσπονδίας και της Ευρωπαϊκής Κοινότητας της 25ης Μαΐου 2006»).
Κυρίες και κύριοι συνάδελφοι, εισερχόμαστε στη συζήτηση των
ΕΠΙΚΑΙΡΩΝ ΕΡΩΤΗΣΕΩΝ
Πρώτα θα συζητηθεί η με αριθμό 681/31/3-4-2017 επίκαιρη ερώτηση του Βουλευτή Α΄ Θεσσαλονίκης της Δημοκρατικής Συμπαράταξης ΠΑΣΟΚ - ΔΗΜΑΡ κ. Ευάγγελου Βενιζέλου προς τον Πρωθυπουργό, σχετικά με το «δημοσίευμα περί πραγματοποίησης επίσκεψης στη Βενεζουέλα, το 2013, του τότε διευθυντή του γραφείου του Προέδρου του ΣΥΡΙΖΑ και νυν Υπουργού Ψηφιακής Πολιτικής».
Στην επίκαιρη ερώτηση θα απαντήσει ο Υπουργός Ψηφιακής Πολιτικής, Τηλεπικοινωνιών και Ενημέρωσης κ. Νικόλαος Παππάς.
Λόγω του ότι είναι η Ώρα του Πρωθυπουργού -ασχέτως το ότι θα απαντήσει ο κ. Παππάς- οι αναφορές και οι ακυρωμένες ερωτήσεις, που κανονικά θα προηγούνταν, θα ανακοινωθούν αφού ολοκληρώσουν τις τοποθετήσεις τους ο κ. Βενιζέλος και ο κ. Παππάς, ώστε να μη διακόψουμε τη ροή τώρα. Οι χρόνοι, βέβαια, είναι οι χρόνοι των κανονικών ερωτήσεων. Θα υπάρξει η σχετική ανοχή από το Προεδρείο και προς τον κ. Βενιζέλο και προς τον κ. Παππά.
Ορίστε, κύριε Βενιζέλο, έχετε τον λόγο.
ΕΥΑΓΓΕΛΟΣ ΒΕΝΙΖΕΛΟΣ: Ευχαριστώ, κύριε Πρόεδρε.
Σύμφωνα με ένα εντυπωσιακό δημοσίευμα του δημοσιογράφου Ιάσονα Πιπίνη, το οποίο είναι τεκμηριωμένο με στοιχεία, μαρτυρίες και κυρίως με φωτογραφίες, ο τότε διευθυντής του Γραφείου του Αρχηγού της Αξιωματικής Αντιπολίτευσης, κ. Νίκος Παππάς, τον Αύγουστο του 2013 πραγματοποίησε επίσκεψη στη Βενεζουέλα και σύμφωνα με τη μαρτυρία του ίδιου του κ. Παππά, την οποία επανέλαβε και χθες στην Επιτροπή θεσμών και Διαφάνειας, αλλά και σε διάφορα μέσα ενημέρωσης, βασικός σκοπός της επίσκεψης αυτής, που δεν είχε ανακοινωθεί δημοσίως, ήταν να διερευνηθούν οι δυνατότητες εμπορικών σχέσεων ανάμεσα στις δύο χώρες.
Δεν αντιλαμβάνομαι ποιον ακριβώς φορέα εκπροσωπούσε και ποιον σκοπό εξυπηρετούσε ο διευθυντής του Γραφείου του Αρχηγού της Αξιωματικής Αντιπολίτευσης, δηλαδή ο εκπρόσωπος ενός ριζοσπαστικού κόμματος, και τι νόημα έχει ένα κόμμα να βολιδοσκοπεί για εμπορικές σχέσεις είτε στον αγροτοδιατροφικό τομέα είτε πολύ περισσότερο σε άλλους τομείς που συνδέονται με μεγάλες και ισχυρές επιχειρήσεις, όπως είναι ο τομέας της ενέργειας και πιο συγκεκριμένα των πετρελαιοειδών.
Σύμφωνα με το δημοσίευμα αυτό, ο παριστάμενος σημερινός Υπουργός Ψηφιακής Πολιτικής, κ. Παππάς, έγινε δεκτός από έναν αμφιλεγόμενο και ελεγχόμενο από διάφορες αρχές επιχειρηματία, τον κ. Μαχέντ Χαλίλ, ταξίδεψε με το ιδιωτικό του αεροσκάφος στο τροπικό νησί Μαργαρίτα, στο οποίο και φιλοξενήθηκε στα ενδιαιτήματα και στα ξενοδοχεία αυτού του επιχειρηματία. Τον βλέπουμε σε φωτογραφίες να έχει στενή φιλική σχέση με τον δικηγόρο κ. Αρτέμη Αρτεμίου, ο οποίο εθεάθη και πριν λίγο καιρό να εξέρχεται του Μεγάρου Μαξίμου, πιθανώς μετά από επαφές με τον Πρωθυπουργό και τον Υπουργό τότε Επικρατείας, νυν Ψηφιακής Πολιτικής. Και είναι πραγματικά παράδοξο πώς είναι δυνατόν όλο αυτό το σκηνικό να οργανώνεται για να προωθηθούν, υποτίθεται, οικονομικές και εμπορικές σχέσεις ερήμην της ελληνικής κυβέρνησης.

Figure 1: A parliamentary record.

2. **convert2txt.py:** Converts all types of downloaded record files (PDF, doc, docx) to text format with the use of tika-app-1.20.jar included in the `src` folder and translates the filenames from Greek to English. It saves the converted files in the folder `_data`. The main challenge of translating the files from Greek to English was the conversion of the Greek alphabetic numerals to indo-arabic numerals. Greek alphabetic numerals, also known as Ionic, Ionian, Milesian, or Alexandrian numerals, are a system of writing numbers using the letters of the Greek alphabet. Greek numerals are used in the Greek Parliament proceedings to enumerate the periods, sittings, and sessions. Within the Greek numerals, we also found archaic letters of the Greek alphabet that represented numbers. Finally, sometimes Latin letters were written by mistake instead of Greek. For example the Latin letter "a" instead of "α" and the Latin letter "p" instead of "ρ". Apart from the conversion of the Greek numerals to numbers, we translated all the Greek words to English, while trying to keep the special meaning of the parliamentary definitions. For example the string "τμήμα διακοπής εργασιών βουλής θέρους" was all together translated to "-summer-recess-section-". We also corrected mistakes in the file names. For example we added a space in the string "γ'τμήμα" so that it became "γ' τμήμα" and we could more easily separate the Greek numeral "γ'" (meaning 3rd) from the word "τμήμα" (meaning section) and produce the best translation.

ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Κύριοι συνάδελφοι, εισερχόμαστε στην ημερήσια διάταξη

ΤΗΣ ΝΟΜΟΘΕΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

θα γίνει προεκφώνηση των νομοσχεδίων, που είναι γραμμένα στην ημερήσια διάταξη, μήπως και περάσουν χωρίς συζήτηση -σύμφωνα με τον Κανονισμό-ορισμένα νομοσχέδια. Υπουργείου Εξωτερικών Μόνη συζήτηση επί της αρχής, των άρθρων και του συνόλου του σχεδίου νόμου: "Κύρωση Συμφωνίας Ελλάδος-Ζαΐρ, για την προώθηση και αμοιβαία προστασία των επενδύσεων". θα συζητηθεί αμέσως μετά, γιατί έχει φέρει αντιρρήσεις ο κ. Πάγκαλος. Υπουργείου Υγείας Πρόνοιας και Κοινωνικών Ασφαλίσεων Συνέχιση της συζήτησης επί των άρθρων και του συνόλου του σχεδίου νόμου: "Εκσυγχρονισμός και οργάνωση συστήματος υγείας". Επίσης, θα συζητηθεί αμέσως μετά το πάρα πάνω νομοσχέδιο του Υπ. Εξωτερικών. Υπουργείου Οικονομικών Μόνη συζήτηση επί της αρχής, των άρθρων και του συνόλου του σχεδίου νόμου: "Αναμόρφωση της άμεσης φορολογίας και άλλες διατάξεις". Ερωτάται το Σώμα, γίνεται Δεκτό,; ΜΕΡΙΚΟΙ ΒΟΥΛΕΥΤΕΣ: Κρατείται. ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Συνεπώς, το σχέδιο νόμου "Αναμόρφωση της άμεσης φορολογίας και άλλες διατάξεις" κρατείται και θα συζητηθεί κατά τον Κανονισμό. Υπουργείου Υγείας Πρόνοιας και Κοινωνικών Ασφαλίσεων Μόνη συζήτηση επί της αρχής, των άρθρων και του συνόλου του σχεδίου νόμου: "Ρύθμιση επαγγέλματος ειδικού τεχνικού προθετικών και ορθωτικών κατασκευών και λοιπών ειδών αποκατάστασης". Ερωτάται το Σώμα, γίνεται Δεκτό,; ΜΕΡΙΚΟΙ ΒΟΥΛΕΥΤΕΣ: Κρατείται. ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Συνεπώς, το σχέδιο νόμου "Ρύθμιση επαγγέλματος ειδικού τεχνικού προθετικών και ορθωτικών κατασκευών και λοιπών ειδών αποκατάστασης" κρατείται και θα συζητηθεί κατά τον Κανονισμό. Υπουργείου Μεταφορών και Επικοινωνιών Μόνη συζήτηση επί της αρχής, των άρθρων και του συνόλου του σχεδίου νόμου: "Κύρωση του Κώδικα Οδικής Κυκλοφορίας". Ερωτάται το Σώμα, γίνεται Δεκτό,; ΣΠΥΡΟΣ ΓΙΑΝΝΟΠΟΥΛΟΣ: Κρατείται. ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Είχε περάσει στην Ειδική Επιτροπή ομόφωνα το νομοσχέδιο αυτό, απλώς ερώτησα. ΣΤΡΑΤΗΣ ΚΟΡΑΚΑΣ: Κρατείται. ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Συνεπώς, το σχέδιο νόμου "Κύρωση του Κώδικα Οδικής Κυκλοφορίας" κρατείται και θα συζητηθεί κατά τον Κανονισμό. Μόνη συζήτηση επί της αρχής, των άρθρων και του συνόλου του σχεδίου νόμου: "Κύρωση της τροποποιημένης Συμφωνίας, για την υιοθέτηση ομοιομόρφων προϋποθέσεων έγκρισης και αμοιβαίας αναγνώρισης εγκρίσεως εξοπλισμών και εξαρτημάτων μηχανοκίνητων οχημάτων". Ερωτάται το Σώμα, γίνεται Δεκτό,; ΟΛΟΙ ΟΙ ΒΟΥΛΕΥΤΕΣ: Δεκτό, Δεκτό. θΕΟΔΩΡΟΣ ΠΑΓΚΑΛΟΣ: Δεκτό. ΣΤΡΑΤΗΣ ΚΟΡΑΚΑΣ: Δεκτό. ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Συνεπώς, το σχέδιο νόμου "Κύρωση της τροποποιημένης Συμφωνίας για την υιοθέτηση ομοιόμορφων προϋποθέσεων έγκρισης και αμοιβαίας αναγνώρισης εγκρίσεως εξοπλισμών και εξαρτημάτων μηχανοκίνητων οχημάτων" έγινε Δεκτό, σε μόνη συζήτηση κατ' αρχήν, κατ' άρθρον και στο σύνολο, ομοφώνως, και έχει ως εξής: (Να καταχωρισθεί το κείμενο του νομοσχεδίου) ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Υπουργείου Υγείας Πρόνοιας και Κοινωνικών Ασφαλίσεων Μόνη συζήτηση επί της αρχής, των άρθρων και του συνόλου του σχεδίου νόμου: "Κύρωση Πρωτοκόλλου Τεχνικής Συνεργασίας στον τομέα της υγείας μεταξύ της Κυβερνήσεως της Ελληνικής Δημοκρατίας και της Κυβερνήσεως των Σεϋχέλλων". Ερωτάται το Σώμα, γίνεται Δεκτό,; ΟΛΟΙ ΟΙ ΒΟΥΛΕΥΤΕΣ: Δεκτό, Δεκτό. ΣΠΥΡΟΣ ΓΙΑΝΝΟΠΟΥΛΟΣ: Δεκτό. ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Συνεπώς, το σχέδιο νόμου: "Κύρωση Πρωτοκόλλου τεχνικής συνεργασίας στον τομέα της υγείας, μεταξύ της Κυβερνήσεως της Ελληνικής Δημοκρατίας και της Κυβερνήσεως των Σεϋχέλλων" έγινε Δεκτό, σε μόνη συζήτηση κατ' αρχήν, κατ' άρθρον και στο σύνολο, ομοφώνως, και έχει ως εξής:

(Να καταχωρισθεί το κείμενο του νομοσχεδίου)

ΠΡΟΕΔΡΕΥΩΝ (Δημήτριος Φράγκος): Υπουργείου Εξωτερικών Μόνη συζήτηση επί της αρχής, των άρθρων και του συνόλου του σχεδίου νόμου: "Κύρωση του Πρόσθετου Πρωτοκόλλου που προσαρτάται στη Συμφωνία μεταξύ των Κρατών-Μελών της Ευρωπαϊκής Κοινότητας Άνθρακα και Χάλυβα και της Ευρωπαϊκής Κοινότητας Άνθρακα και Χάλυβα αφ'ενός και της Δημοκρατίας της Αυστρίας αφ'ετέρου ως και του δεύτερου Πρόσθετου Πρωτοκόλλου μεταξύ των ιδίων Μερών, συνεπεία της προσχωρήσεως του Βασιλείου της Ισπανίας και της Πορτογαλικής Δημοκρατίας στην Κοινότητα". Ερωτάται το Σώμα, γίνεται Δεκτό,; ΟΛΟΙ ΟΙ ΒΟΥΛΕΥΤΕΣ: Δεκτό, Δεκτό.

Figure 2: A malformed parliamentary record.

4.3 Modern Greek Names and Surnames Collection and Cleaning

3. **greek_name_cases_wiki_crawler.py:** Crawls the wiktionary lists of modern Greek female and male names and surnames and additionally collects all the grammatical cases, when available, in tables within each entry page. The output consists of four JSON files located under `out_files/wiki_data`, namely `female_name_cases.json`, `male_name_cases.json`, `female_surname_cases.json`, `male_surname_cases.json`.
4. **produce_cases_from_nominative.py:** Takes as input the JSON files produced from the script `greek_name_cases_wiki_crawler.py` and produces the missing grammatical cases based on the nominative case. The output consists of four JSON files, namely, `female_name_cases_populated.json`, `male_name_cases_populated.json`, `female_surname_cases_populated.json`, and `male_surname_cases_populated.json`.

4.4 Parliament Members Data Collection and Cleaning

5. **web_crawler_for_parliament_members.py:** The Greek Parliament website includes a list[5] of all the members of parliament since the fall of the military junta in Greece, in 1974. The script downloads information of the members of parliament from the list us-

[5] `https://www.hellenicparliament.gr/Vouleftes/Diatelesantes-Vouleftes-Apo-Ti-Metapolitefsi-Os-Simera/`

ing Selenium and the Chrome driver included in the `src` folder. We kept information from 1989 onwards, matching the records that we have at our disposal. The output is written in `original_parl_members_data.csv`.

6. **parl_members_data_cleaner.py:** It cleans and also formats the above file `original_parl_members_data.csv` for further use. For each member of parliament, the list used by the previous script provides a table where each row corresponds to an event in their specific term of office. We categorized the events in the following three groups:
   - Start events: elected, replaced.
     A member starts their parliamentary term with any of the start events such as being elected or by replacing someone. Each member has at least one start event for each parliamentary period.
   - Change events: switch to another political party, become willingly independent (outside any political party), become independent (outside any political party) due to a temporary suspension or become independent (outside any political party) due to permanent expulsion from a political party.
   - End events: resignation from office, death, removal following indictment and conviction, assassination.

   An individual can change multiple political parties during a parliamentary period. In order to extract the exact date range of each individual's activity as a member of a political party or as an independent member, we developed an algorithm that takes into account the order of the start, change and end events per member during a parliamentary period and extracts the correct date ranges. We also correct some data entry mistakes and write the output to `parl_members_activity_1989onwards.csv`.
7. **add_gender_to_members.py:** Adds a column with gender information to the `parl_members_activity_1989onwards.csv` file using the `male_name_cases_populated.json` and `female_name_cases_populated.json` files, and creates `parl_members_activity_1989onwards_with_gender.csv`. The resulting file includes the full names of the members, the date range of their service, the political party in which they belonged, their electoral district, and their gender.

4.5 Government Members Data Collection and Cleaning

8. **web_crawler_for_government_members.py:** Crawls the website `https://gslegal.gov.gr/?page_id=776&sort=time` and collects information about all the governments from 1989 up to 2020 and all the members that were assigned government roles. It creates `governments_1989onwards.csv` that includes the names of governments since 1989, their start and end dates, and a URL that points to the respective official government web page of each past government. It also creates `original_gov_members_data.csv` with the crawled raw data from the website.
9. **gov_members_data_cleaner.py:** Cleans `original_gov_members_data.csv`. Converts names and surnames from genitive to nominative case and adds gender with the use of the files crawled from Wiktionary (these are `female_name_cases_populated.json` and `female_surname_cases_populated.json`, and their corresponding male counterparts `male_name_cases_populated.json` and `male_surname_cases_populated.json`). Furthermore, it automatically checks for inconsistencies in the data, which led to resolving issues such as a ministry's renaming and data entry mistakes. Finally, the script includes corrections in roles, member names, and wrong entries found after manual inspection of the data and outputs `formatted_roles_gov_members_data.csv` that includes the full names of the official individuals, their role, the date range of their service at each specific role, and their gender.

4.6 Speech Extraction

10. **join_members_activity.py:** This script concatenates three files with information about the parliament members and extra members of parliament. The input files are: `parl_members_activity_1989onwards_with_gender.csv` (includes elected members of parliament), `formatted_roles_gov_members_data.csv` (includes all government members

that have been assigned a government role but may not have been necessarily elected in the parliament) and `extra_roles_manually_collected.csv` (includes manually collected additional roles from Wikipedia such as party leaders, opposition leaders etc.). An extra column is added with the name of the government during each member's activity, using `governments_1989onwards.csv`. The final output of this script is the file `all_members_activity.csv` with columns: member_name, member_start_date, member_end_date, political_party, administrative_region, gender, roles, government_name.

11. **member_speech_matcher.py:** Extracts speeches from record files and matches them to the corresponding member of parliament or government. The script takes as arguments from the command line the path of the folder with the record files and the path to the folder where it outputs the speeches and the corresponding speakers. For example, for the current folder hierarchy:

```
python3 member_speech_matcher.py -f '../_data/' \
    -o '../out_files/tell_all.csv'
```

The start of each new speech is signaled by the reference to the speaker that delivers it. So, in order to distinguish a speech, we must first detect a valid speaker name. To do so, we created an extensive list of regular expressions that capture all possible variations of references to speakers as well as cases of poorly formatted speaker names. As mentioned, in many cases the text did not follow the expected format. In moments of commotion, speeches were assigned in the records to "A parliament member from the x political party" or even "Many parliament members" instead of a specific individual. We kept those speeches with a blank speaker name and assigned them to a political party, when mentioned, or used a generic reference. For speeches delivered by holders of specific posts, such as Speakers of Parliament, the individual's post would precede their name. On several occasions the text would be poorly formatted, with new speeches not starting at the beginning of a new line or with missing closing brackets in the speaker's reference. Furthermore, a common pattern was to mention the name of the chair of the sitting once in the beginning of the record and then initiate their subsequent speeches by mentioning only their role. We filled in the missing name of the chair for each sitting, as long as the sitting had only one chair.

Below we give the source code of the regular expressions we used:

```
import re

# Regular expression for detecting a candidate speaker written in capital
# case, ocassionally followed by a parenthesis with extra information
# and finally a colon that indicates the beginning of their speech.
# E.g., AIKATERINI PAPANATSIOU (Deputy Minister of Finance):
speaker_regex = re.compile(r"((\s*[A-ΩΆ-ΏΪΫΐΰ-]+)(\s+[A-ΩΆ-ΏΪΫΐΰ-]+)*\s*(\
↪  (.*?\))?\s*\:)")

# Regular expression for detecting nicknames given in parentheses
# following the speaker's name and written in capital case, e.g., (PANOS)
caps_nickname_in_parenthesis = re.compile(r"(\([A-ΩΆ-ΏΪΫΐΰ]+\))+") #(ΠΑΝΟΣ)

# Regular expression for detecting nicknames given in parentheses
# following the speaker's name and written in lower case, e.g., (Panos)
lower_nickname_in_parenthesis = re.compile(r"(\([α-ω]{2,}\))") #(πανος)

# Regular expression for detecting information about the speaker given
# in parentheses following the speaker's name, e.g.,
# (Deputy Minister of Finance)
text_in_parenthesis = re.compile(r"(\(.*?\)){1}") #(Υπουργός Εσωτερικών)

# Regular expression for detecting speakers that are either assigned
# Chair of the sitting or Speakers of the Parliament
roedr_regex = re.compile(r"(^(((Π+Ρ(Ο|Ό)+(Ε|Έ))|(Ρ(Ο|Ό)+(Ε|Έ)Δ)|(ΠΡ(Ε|Έ)(Ο
↪  |Ό))|(ΠΡ(Ο|Ό)Δ)|(Η ΠΡ(Ο|Ό)(Ε|Έ)ΔΡ)|(ΠΡ(Ε|Έ)Δ))|(ΠΡΟΣΩΡΙΝΗ
↪  ΠΡΟΕΔΡΟΣ)|(ΠΡΟΣΩΡΙΝΟΣ ΠΡΟΕΔΡΟΣ)))")
```

```
# Regular expession for detecting only the Speaker of the Parliament
proedros_regex =
↪  re.compile(r"ΠΡ((Ο|Ό|(ΟΟ))(Ε|Έ)|((ΕΟ)|(ΈΟ)|(ΕΌ)|(ΈΌ)))ΔΡΟΣ")

# Regular expession for detecting only the first speaker of each sitting
# that is always delivered by the Chair of a sitting
proedreuon_first_speaker = re.compile(r"((\s*[A-ΩΆ-ΏΪΫΐΰ-]+)(\s+\(([A-ΩΆ-Ώ
↪  α-ωά-ώϊϋΐΰΪΫΐΰ-]\s*)+\))?\s*\:)$")

# Regular expression for detecting a speaker that is referenced in a
# generic manner, e.g., A PARLIAMENT MEMBER
general_member_regex =
↪  re.compile(r"((Β(Ο|Ό)(Υ|Ύ)(Ε|Έ)Λ)|(Β(Ο|Ό)(Υ|Ύ)Λ(Ε|Έ)(Υ|Ύ)?Τ[^(Α|Ά)]))")

# Regular expression that detects a left parenthesis
left_parenthesis_regex = re.compile(r"\(")

# Regular expression that detects a right parenthesis
right_parenthesis_regex = re.compile(r"\)")

# Regular expression that detects a parenthesis that is left open
# and includes the nickname of a speaker
incomplete_nickname_parenthesis = re.compile(r"\([A-ΩΆ-ΏΪΫΐ]{3,}\s")

# Regular expression that detects a phrase delivered by the Chair of each
# sitting that signals its end
sitting_terminated_regex = re.compile(r"λ(υ|ύ)εται\s+η\s+συνεδρ(ι|ί)αση")
```

After the detection of a speaker in a record file with the use of regular expressions, we search the detected speaker in the `all_members_activity.csv`. However, there are many different name pattern variations in the records. In many cases, the speakers were referenced with their nicknames instead of their official names. In cases were a person had more than one first name and surname, some of them where missing. Finally, the order of the first and last names was not always the same. To resolve this string comparison task, we employed the Jaro-Winkler string similarity metric [6]. It is a variant of the Jaro distance [5], which has been applied mainly to the record linkage problem, and whose goal is to compute string similarity based on the common elements and the number of transpositions between them. The Jaro-Winkler distance extends the Jaro distance by boosting it using a scaling factor p when the first l characters match exactly. Since the Jaro-Winkler metric takes into account the order of letters in a string comparison task, the names "Fotini Gennimata" and "Gennimata Fotini" would have zero similarity. Similarly, the names "Fofi Gennimata", "Fotini Gennimata", "Fotini-Fofi Gennimata", which refer to the same person, do not match. For this reason, for each comparison between a detected speaker and an official individual, we created all possible variations of an official name, alternating the order of the words that make up that name and replacing or combining the name with its diminutives. In Fig. 3, we display an example of the naming variations produced for each member of the official members list. Each variation is compared with the detected speaker from the record. We then calculate the degree of similarity between all possible pairs and choose the one with the highest degree. In order to search efficiently and effectively for the detected speaker in the list of official names, we compare the detected speaker only with the official members that were active at the date of the sitting. For the string comparison between names, `greek_names_alts_only.txt` is used with a list of Greek names that have alternatives. For licensing reasons this is the only support file that cannot be shared publicly.

Due to the fact that the speakers' names had misspellings, or could be missing characters or syllables, we accepted matches with similarity equal to or higher than 0.95. Table 2 presents a manual evaluation of a sample of 150 entity pairs that were compared in order to match a detected candidate speaker from the records with an official member of the parliament. The column max_sim denotes that the automated search of the detected speaker in the members

data we collected yielded the maximum similarity with the official name depicted in the respective column. If the maximum similarity is equal to or higher than our threshold of 0.95, we accept the match. As we can see from the manual evaluation of the sample, our approach yields in total 0 False Positives, 4 False Negatives, 137 True Positives and 9 True Negatives. The aim of our approach was to minimize the False Positive rate, in order to avoid the violation of any ethical guidelines by assigning a speech to a person that did not deliver it. Most cases that present False Negative results include speakers that have more than one first name and one of their first names is replaced in the parliament records with a Greek relevant diminutive.

Table 2: Sample of 150 cases of entity pairs, their Jaro-Winkler similarity and the category the results fall into from the following: False Positive, False Negative, True Positive, True Negative. The results are displayed in ascending similarity order.

Detected speaker	Official name	max_sim	FP	FN	TP	TN
πρν ελληνικη λυση	σουκουλη-βιλιαλη δημητριου μαρια-ελενη (μαριλενα)	0.72	0	0	0	1
επικυρωση πρακτικων	περκα χαραλαμπου θεοπιστη (πετη)	0.73	0	0	0	1
πρνπροεδρευουσα	παπανδρεου ανδρεα γεωργιος	0.73	0	0	0	1
ελληνικη λυση	λιακουλη θεοφανη ευαγγελια	0.73	0	0	0	1
ναι ελληνικη λυση	λιακουλη θεοφανη ευαγγελια	0.74	0	0	0	1
εσυ	σουκουλη-βιλιαλη δημητριου μαρια-ελενη (μαριλενα)	0.74	0	0	0	1
ελγα	δελης ιωαννη ιωαννης	0.77	0	0	0	1
ναι συριζα	μπουρας κωνσταντινου αθανασιος	0.8	0	0	0	1
συριζα	συριγος ματθαιου ευαγγελος (αγγελος)	0.84	0	0	0	1
σπυριδων-παναγιωτης-σπηλιος λιβανος	λιβανος διονυσιου σπυριδωνας-παναγιωτης (σπηλιος)	0.88	0	1	0	0
παναγιωτης-σπυριδων λιβανος	λιβανος διονυσιου σπυριδωνας-παναγιωτης (σπηλιος)	0.9	0	1	0	0
σπυριδων-παναγιωτης λιβανος	λιβανος διονυσιου σπυριδωνας-παναγιωτης (σπηλιος)	0.9	0	1	0	0
μαρια-αλεξανδρα κεφαλα	κεφαλα στεφανου μαρια-αλεξανδρα	0.93	0	1	0	0
κριτων-ηλιας αρσενης	αρσενης δημοκρατη κριτων-ηλιας	0.95	0	0	1	0
μανουσος-κωνσταντινος βολουδακης	βολουδακης γεωργιου μανουσος-κωνσταντινος	0.95	0	0	1	0
δομνα-μαρια μιχαηλιδου	δομνα-μαρια μιχαηλιδου	0.95	0	0	1	0
αναστασια-αικατερινη αλεξοπουλου	αλεξοπουλου κωνσταντινου αναστασια-αικατερινη	0.96	0	0	1	0
ιωαννης-μιχαηλ λοβερδος	λοβερδος πετρου ιωαννης-μιχαηλ (γιαννης)	0.96	0	0	1	0
βασιλειος-νικολαος υψηλαντης	υψηλαντης αναστασιου βασιλειος-νικολαος	0.96	0	0	1	0
αλεξανδρος-χρηστος αυλωνιτης	αυλωνιτης νικολαου αλεξανδρος-χρηστος	0.96	0	0	1	0
διονυσιος-χαραλαμπος καλαματιανος	καλαματιανος χρηστου διονυσιος-χαραλαμπος	0.96	0	0	1	0
χαρα καφανταρη	καφανταρη φωτιου χαρουλα (χαρα)	0.96	0	0	1	0
σοφια-χαιδω ασημακοπουλου	ασημακοπουλου δημητριου σοφια-χαιδω	0.96	0	0	1	0
βασιλειος-πετρος σπανακης	σπανακης νικολαου βασιλειος-πετρος	0.96	0	0	1	0
διονυσια-θεοδωρα αυγερινοπουλου	αυγερινοπουλου ζησιμου διονυσια-θεοδωρα	0.96	0	0	1	0
σπυριδων-αδωνις γεωργιαδης	γεωργιαδης αθανασιου σπυριδων-αδωνις	0.96	0	0	1	0
χριστοφορος-εμμανουηλ μπουτσικακης	μπουτσικακης ιωαννη χριστοφορος-εμμανουηλ	0.96	0	0	1	0
μαρια-ελιζα ξενογιαννακοπουλου	ξενογιαννακοπουλου διονυσιου μαρια-ελιζα (μαριλιζα)	0.97	0	0	1	0
μαρια-αλεξανδρα κεφαλα	κεφαλα στεφανου μαρια-αλεξανδρα	0.97	0	0	1	0
αννα μανη-παπαδημητριου	μανη-παπαδημητριου ευαγγελου αννα	0.98	0	0	1	0
χαραλαμπος αθανασιου	αθανασιου χριστοφα χαραλαμπος	1	0	0	1	0
ιωαννης μελας	μελας παναγιωτη ιωαννης	1	0	0	1	0
γεωργιος καμινης	καμινης βασιλειου γεωργιος	1	0	0	1	0

νικολαος παπαναστασης	παπαναστασης αθανασιου νικολαος	1	0	0	1	0
κωνσταντινος χητας	χητας αχιλλεως κωνσταντινος	1	0	0	1	0
αγγελικη αδαμοπουλου	αδαμοπουλου αθανασιου αγγελικη	1	0	0	1	0
κωνσταντινος μπογδανος	μπογδανος σπυριδωνος κωνσταντινος	1	0	0	1	0
μαριλιζα ξενογιαννακοπουλου	ξενογιαννακοπουλου διονυσιου μαρια-ελιζα (μαριλιζα)	1	0	0	1	0
κυριακος βελοπουλος	βελοπουλος ιωσηφ κυριακος	1	0	0	1	0
ελευθεριος οικονομου	ελευθεριος οικονομου	1	0	0	1	0
γεωργιος παπαηλιου	παπαηλιου ηλια γεωργιος	1	0	0	1	0
νικητας κακλαμανης	κακλαμανης μιχαηλ νικητας	1	0	0	1	0
οδυσσεας κωνσταντινοπουλος	κωνσταντινοπουλος κωνσταντινου οδυσσεας	1	0	0	1	0
σοφια σακοραφα	σακοραφα ηλια σοφια	1	0	0	1	0
αθανασιος δαβακης	δαβακης παναγιωτη αθανασιος	1	0	0	1	0
ευαγγελος συριγος	συριγος ματθαιου ευαγγελος (αγγελος)	1	0	0	1	0
κωνσταντινα γιαννακοπουλου	γιαννακοπουλου ιωαννη κωνσταντινα (ναντια)	1	0	0	1	0
βασιλειος βιλιαρδος	βιλιαρδος διονυσιου βασιλειος	1	0	0	1	0
παναγιωτης λιβανος	λιβανος διονυσιου σπυριδωνας-παναγιωτης (σπηλιος)	1	0	0	1	0
εμμανουηλ συντυχακης	συντυχακης δανιηλ εμμανουηλ	1	0	0	1	0
χαραλαμπος καλαματιανος	καλαματιανος χρηστου διονυσιος-χαραλαμπος	1	0	0	1	0
ανδρεας ξανθος	ξανθος γεωργιου ανδρεας	1	0	0	1	0
δημητριος βαρτζοπουλος	βαρτζοπουλος χρυσοστομου δημητριος	1	0	0	1	0
ανδρεας πουλας	πουλας θεοδοσιου ανδρεας	1	0	0	1	0
γεωργιος λαμπρουλης	λαμπρουλης αριστειδη γεωργιος	1	0	0	1	0
μαρια αθανασιου	αθανασιου ευαγγελου μαρια	1	0	0	1	0
κλεων γρηγοριαδης	γρηγοριαδης γεωργιου κλεων	1	0	0	1	0
παυλος πολακης	πολακης πετρου παυλος	1	0	0	1	0
βασιλειος κικιλιας	κικιλιας παναγιωτη βασιλειος	1	0	0	1	0
βασιλειος γιογιακας	γιογιακας νικολαου βασιλειος	1	0	0	1	0
κωνσταντινος μαρκου	μαρκου βασιλειου κωνσταντινος	1	0	0	1	0
γεωργιος φραγγιδης	φραγγιδης σταυρου γεωργιος	1	0	0	1	0
χαρουλα καφανταρη	καφανταρη φωτιου χαρουλα (χαρα)	1	0	0	1	0
διονυσιος ακτυπης	ακτυπης δημητριου διονυσιος	1	0	0	1	0
αθανασιος παπαδοπουλος	παπαδοπουλος αποστολου αθανασιος (σακης)	1	0	0	1	0
βασιλειος κοντοζαμανης	βασιλειος κοντοζαμανης	1	0	0	1	0
μιχαηλ κατρινης	κατρινης ιωαννη μιχαηλ	1	0	0	1	0
γιανης βαρουφακης	βαρουφακης γεωργιου γιανης	1	0	0	1	0
αθανασιος μπουρας	μπουρας κωνσταντινου αθανασιος	1	0	0	1	0
αθανασιος καββαδας	καββαδας ιωαννη αθανασιος	1	0	0	1	0
ανδρεας μιχαηλιδης	μιχαηλιδης φραγκουλη ανδρεας	1	0	0	1	0
αποστολος βεσυροπουλος	βεσυροπουλος φωτιου αποστολος	1	0	0	1	0
δημητριος μαρκοπουλος	μαρκοπουλος κωνσταντινου δημητριος	1	0	0	1	0
μεροπη τζουφη	τζουφη στεφανου μεροπη	1	0	0	1	0
ελευθεριος αβραμακης	αβραμακης αντωνιου ελευθεριος	1	0	0	1	0
γεωργιος κωτσηρας	κωτσηρας αναστασιου γεωργιος	1	0	0	1	0
ευκλειδης τσακαλωτος	τσακαλωτος στεφανου ευκλειδης	1	0	0	1	0
μπουρχαν μπαραν	μπαραν νετζαντη μπουρχαν	1	0	0	1	0
νικολαος συρμαλενιος	συρμαλενιος ευαγγελου νικολαος	1	0	0	1	0
κωνσταντινος μπαρκας	μπαρκας θεοφανη κωνσταντινος	1	0	0	1	0
αποστολος αβδελας	αβδελας κωνσταντινου αποστολος	1	0	0	1	0
κωνσταντινος μαραβεγιας	μαραβεγιας αριστοτελη κωνσταντινος	1	0	0	1	0
σουλτανα ελευθεριαδου	ελευθεριαδου παυλου σουλτανα	1	0	0	1	0
νεοκλης κρητικος	κρητικος δημοσθενη νεοκλης	1	0	0	1	0
αθανασιος λιουπης	λιουπης κωνσταντινου αθανασιος	1	0	0	1	0
ανδρεας νικολακοπουλος	νικολακοπουλος θεοδωρου ανδρεας	1	0	0	1	0
γεωργιος μουλκιωτης	μουλκιωτης βασιλειου γεωργιος	1	0	0	1	0
κωνσταντινος ζαχαριαδης	ζαχαριαδης εμμανουηλ κωνσταντινος	1	0	0	1	0

ζησης τζηκαλαγιας	τζηκαλαγιας γεωργιου ζησης	1	0	0	1	0
ζωη ραπτη	ραπτη γεωργιου ζωη	1	0	0	1	0
χριστινα αλεξοπουλου	αλεξοπουλου μιχαηλ χριστινα	1	0	0	1	0
φωτεινη μπακαδημα	μπακαδημα αγαθοκλη φωτεινη	1	0	0	1	0
εμμανουηλ θραψανιωτης	θραψανιωτης μιχαηλ εμμανουηλ	1	0	0	1	0
νικολαος καραθανασοπουλος	καραθανασοπουλος πετρου νικολαος	1	0	0	1	0
αθανασια αναγνωστοπουλου	αναγνωστοπουλου πετρου αθανασια (σια)	1	0	0	1	0
κωνσταντινος βλασης	βλασης γεωργιου κωνσταντινος	1	0	0	1	0
ιωαννης σαρακιωτης	σαρακιωτης αθανασιου ιωαννης	1	0	0	1	0
ιωαννης πλακιωτακης	πλακιωτακης ιωσηφ ιωαννης	1	0	0	1	0
μαρια κομνηνακα	κομνηνακα αποστολου μαρια	1	0	0	1	0
ιωαννης κεφαλογιαννης	κεφαλογιαννης αχιλλεα ιωαννης	1	0	0	1	0
βασιλειος κεγκερογλου	κεγκερογλου αλεξανδρου βασιλειος	1	0	0	1	0
δημητριος οικονομου	δημητριος οικονομου	1	0	0	1	0
γερασιμος θωμας	γερασιμος θωμας	1	0	0	1	0
σωκρατης φαμελλος	φαμελλος πετρου σωκρατης	1	0	0	1	0
νικολαος παππας	παππας στυλιανου νικολαος	1	0	0	1	0
ιωαννης δραγασακης	δραγασακης ανδρεα ιωαννης	1	0	0	1	0
θεοπιστη περκα	περκα χαραλαμπου θεοπιστη (πετη)	1	0	0	1	0
κωνσταντινος χατζηδακης	χατζηδακης ιωαννη κωνσταντινος (κωστης)	1	0	0	1	0
γεωργιος αρβανιτιδης	αρβανιτιδης πετρου γεωργιος	1	0	0	1	0
χριστοφορος βερναρδακης	βερναρδακης δημοσθενη χριστοφορος	1	0	0	1	0
κωνσταντινος τσιαρας	τσιαρας αλεξανδρου κωνσταντινος	1	0	0	1	0
ιωαννης βρουτσης	βρουτσης βασιλειου ιωαννης	1	0	0	1	0
θεανω φωτιου	φωτιου βασιλειου θεανω	1	0	0	1	0
θεοδωρος λιβανιος	θεοδωρος λιβανιος	1	0	0	1	0
λεωνιδας στολτιδης	στολτιδης δημητριου λεωνιδας	1	0	0	1	0
ιωαννης δελης	δελης ιωαννη ιωαννης	1	0	0	1	0
παναγιωτης σκουρλετης	σκουρλετης βασιλειου παναγιωτης (πανος)	1	0	0	1	0
νικολαος παπαθανασης	νικολαος παπαθανασης	1	0	0	1	0
χρηστος σπιρτζης	σπιρτζης παναγιωτη χρηστος	1	0	0	1	0
σαββας χιονιδης	χιονιδης γεωργιου σαββας	1	0	0	1	0
ευαγγελια λιακουλη	λιακουλη θεοφανη ευαγγελια	1	0	0	1	0
διονυσιος χατζηδακης	χατζηδακης διονυσιου διονυσιος	1	0	0	1	0
κωνσταντινος κυρανακης	κυρανακης ιωαννη-παναγιωτη κωνσταντινος	1	0	0	1	0
νικολαος βουτσης	βουτσης γεωργιου νικολαος	1	0	0	1	0
παναγιωτης θεοδωρικακος	θεοδωρικακος δημητριου παναγιωτης (τακης)	1	0	0	1	0
γεωργιος κωτσος	κωτσος κωνσταντινου γεωργιος	1	0	0	1	0
σταυρος κελετσης	κελετσης δημητριου σταυρος	1	0	0	1	0
αλεξανδρος τριανταφυλλιδης	τριανταφυλλιδης γεωργιου αλεξανδρος (αλεκος)	1	0	0	1	0
κωνσταντινος μπουμπας	μπουμπας ιωαννη κωνσταντινος	1	0	0	1	0
βασιλης κεγκερογλου	κεγκερογλου αλεξανδρου βασιλειος	1	0	0	1	0
γεωργιος βαρεμενος	βαρεμενος βασιλειου γεωργιος	1	0	0	1	0
κωνσταντινος σκανδαλιδης	σκανδαλιδης γεωργιου κωνσταντινος	1	0	0	1	0
στυλιανη μενδωνη	στυλιανη μενδωνη	1	0	0	1	0
χαραλαμπος μαμουλακης	μαμουλακης αντωνιου χαραλαμπος (χαρης)	1	0	0	1	0
στεφανος γκικας	γκικας σωτηριου στεφανος	1	0	0	1	0
αθανασιος ζεμπιλης	ζεμπιλης γεωργιου αθανασιος	1	0	0	1	0
στυλιανος πετσας	στυλιανος πετσας	1	0	0	1	0
δημητριος κουβελας	κουβελας σωτηριου δημητριος	1	0	0	1	0
νοτης μηταρακης	μηταρακης αντωνιου παναγιωτης (νοτης)	1	0	0	1	0
γεωργιος αμανατιδης	αμανατιδης ισαακ γεωργιος	1	0	0	1	0
γεωργιος κοτρωνιας	κοτρωνιας νικολαου γεωργιος	1	0	0	1	0
αθανασιος λιουτας	λιουτας αχιλλεως αθανασιος	1	0	0	1	0
μαριος κατσης	κατσης σπυριδωνος μαριος	1	0	0	1	0
γεωργιος γεωργαντας	γεωργαντας παυλου γεωργιος	1	0	0	1	0
φιλιππος φορτωμας	φορτωμας αριστοτελη φιλιππος	1	0	0	1	0

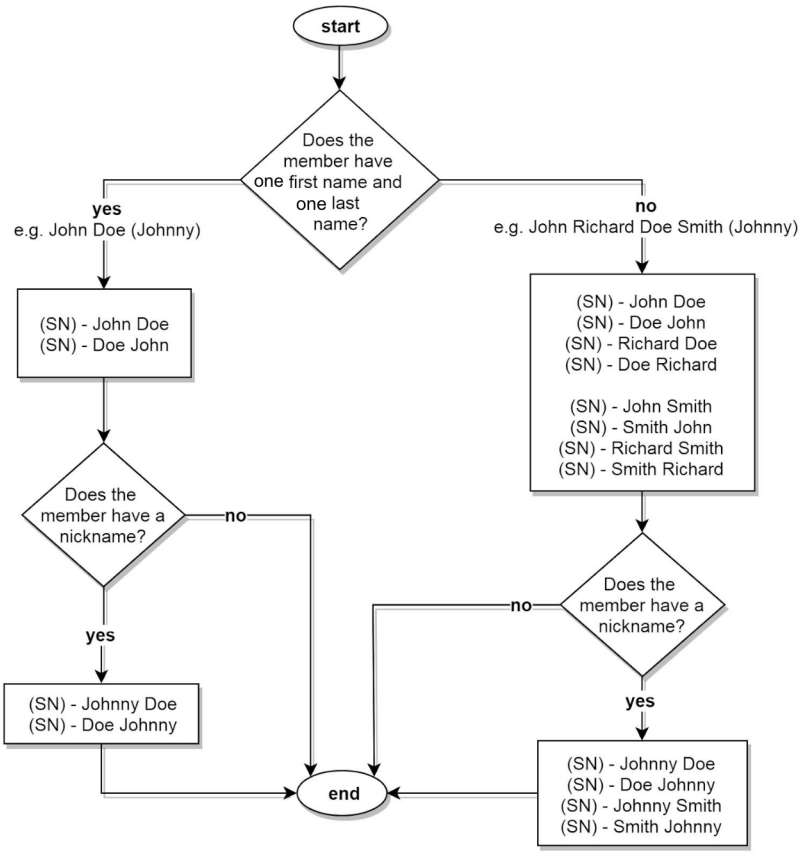


Figure 3: Representation of the name variations produced during the string comparison process between the detected speaker in the record file and the name of the corresponding member parliament; (SN) stands for generated speaker name.

γεωργιος στυλιος	στυλιος δημοσθενη γεωργιος	1	0	0	1	0
μιλτιαδης χατζηγιαννακης	χατζηγιαννακης ευαγγελου μιλτιαδης	1	0	0	1	0
γεωργιος ψυχογιος	ψυχογιος κωνσταντινου γεωργιος	1	0	0	1	0
παναγιου πουλου	πουλου λεωνιδα παναγιου (γιωτα)	1	0	0	1	0
χρηστος κελλας	κελλας αχιλλεα χρηστος	1	0	0	1	0

The output file is `tell_all.csv` with columns: member_name, sitting_date, parliamentary_period, parliamentary_session, parliamentary_sitting political_party, government, member_region, roles, member_gender, speaker_info, speech. This script also creates the file `files_with_content_problems.txt`, where record files that are skipped due to encoding issues are logged.

12. **fill_proedr_names.py:** Sometimes the name of the speaker of a sitting is mentioned in their first speech but only their role as speaker is mentioned thereafter in the same sitting. In order to assign their speeches to themselves, and not just their role as speakers, this script fills in the names of speakers in the same sittings when they are not mentioned in a specific line of the record file, as long as the sitting has only one Speaker. The output is `tell_all_FILLED.csv`.

13. **csv_concat.py:** Additional script that concatenates all `tell_all.csv` files created in case the script `member_speech_matcher.py` is run in parallel for time-optimization and created different csv files for different batches of the data. The output of this script is `tell_all_final.csv`.

4.7 Preprocessing

14. **corpus_preprocessing.ipynb**: This script takes as input the file `tell_all_FILLED.csv` and preprocesses it. The preprocessing includes the replacement of all references to political parties with the symbol "@" followed by an abbreviation of the party name, using regular expressions that capture different grammatical cases and variations. It also includes the removal of accents, strings with length less than two characters, all punctuation except full stops, and the replacement of stopwords with "@sw". The output is the file `tell_all_cleaned.csv`.

15. **common_vocab_per_period_viz.ipynb**: This script produces Fig. 2 in Section 3.5 of the manuscript. This figure shows the common vocabulary in terms of tokens between pairs of consecutive parliamentary periods in three different steps: before preprocessing, after preprocessing, and after merging smaller parliamentary periods with their following large parliamentary periods.

4.8 Gender Distribution

16. **female_members_per_period_per_party.ipynb**: This script takes as input the file `tell_all_cleaned.csv` from which it computes the percentage of the characters delivered in the parliament speeches by female individuals. It also takes as input `all_members_activity.csv`, with which it computes the percentage of female members of the parliament. The computations are calculated per parliamentary period and per political party for a selection of political parties. The scipt outputs the results to the `gender_distribution_per_period_per_party_with_chars.png`, which corresponds to Fig. 1 of the manuscript.

5 Stability Comparison of Word Usage Change Detection Approaches

This section describes the implementation details of the comparison stability described in Section 4 of the manuscript. For the stability comparison, we facilitate the metric intersection@k, proposed by Gonen et al. [2], which computes the k most changed words for a number of restarts, each time changing the random seed. We ran each approach 10 times, each time introducing a different random seed in the range 0–9, and collected the top-k most changed words, where $k \in [10, 20, 50, 100, 200, 500, 1000]$. The seed is used as the NumPy random seed, the Python random seed, and the seed parameter of the word2vec models. We also set the `PYTHONHASHSEED` environment variable to 0. In addition, we limit the word2vec model training to a single worker thread. Then, for each of the $\binom{10}{2} = 45$ pairs of different runs and for each of the values of k, we measured the percentage of shared words in the most changed words predictions. A value of zero between a pair of runs means that there are no shared words in their predictions, indicating high variability in the results, while value of one indicates high stability.

5.1 Implementation Details per Approach

5.1.1 Compass

We train a Compass [1] model on the concatenation of the two decades' corpora. We then train a model for each decade, based on the Compass model. We proceed with gathering the common vocabulary between the two models. For each word of the common vocabulary we compute the cosine similarity of its vectors from the two decades. We use word2vec vectors with 300 dimensions and the default settings of the Compass tool, which include a 5-words context window and a minimum word frequency count of five.

To run the Compass stability computations, use the following command from inside the `src` folder:

```
PYTHONHASHSEED=0 python compass_stability_1990s_2010s.py --run=0 \
    --iterations=10 --diter=5 --siter=5 --size=300 --rows='all'
```

5.1.2 Compass Variation with Frequency Cut-offs

We use Compass [1] but also adopt the frequency cut-offs introduced in the NN [2] approach. This means that we remove from the vocabulary of each model the stopwords, the 200 most frequent words and words that appear less than 200 times.

In order to do that, we first compute the frequencies of words in the corpora of the decades 1990–1999 and 2010–2019 with the script `freq_counter_for_semantic_shift_per_decade.py`. This script outputs the files `freqs_for_semantic_shift_cleaned_data_decade1990.csv` and `freqs_for_semantic_shift_cleaned_data_decade2010.csv` that are then used by the script `compass_stability_1990s_2010s_freq_cutoffs.py`. The result is a filtered list with candidates for semantic shift for each model. We collect the intersection of the filtered lists for the two models. Finally, for each word of the intersection we compute the cosine similarity of its vectors from the two decades.

To run the stability computations for this approach, use the following command from inside the `src` folder:

```
PYTHONHASHSEED=0 python compass_stability_1990s_2010s_freq_cutoffs .py \
    --run=0 --iterations=10 --diter=5 --siter=5 --size=300 --rows='all'
```

5.1.3 Orthogonal Procrustes

To implement Orthogonal Procrustes [4], we train a word2vec model for each decade using word2vec vectors with 300 dimensions, a 5-words context window and a minimum word frequency count of 20. We then align the two vector spaces based on the common vocabulary of the two models. Finally, we compute the cosine similarity between the vectors of each word in the decades.

To run the stability computations for this approach, use the following command from inside the `src` folder:

```
PYTHONHASHSEED=0 python procrustes_stability_1990s_2010s.py --run=0 \
    --iterations=10 --size=300 --rows='all'
```

5.1.4 NN

To implement the NN approach [2], we train a word2vec model for each decade using word2vec vectors with 300 dimensions, a 4-words context window and a minimum word frequency count of five, as per the implementation of Gonen et al. [2]. We remove from the vocabulary of each model the stopwords, the 200 most frequent words and words that appear less than 200 times. In our case, the frequency distribution for each corpus is long-tailed, with only 5% of the vocabulary of each decade having 200 or more occurrences. The result is a filtered list with candidates for semantic shift for each model. We collect the intersection of the filtered lists for the two models. Then, for each word of the filtered list we collect two sets of the top-1000 most similar words, one for each decade. In this process we only take into consideration neighbors that exist in the vocabularies of both models and appear at least 100 times in each training corpus. Finally, we compute the word usage change by measuring the intersection of the two sets of neighbors.

To run the stability computations for this approach, use the following command from inside the `src` folder:

```
PYTHONHASHSEED=0 python goldberg_stability_1990s_2010s.py --run=0 \
  --iterations=10 --size=300 --rows='all'
```

5.1.5 Second-Order Similarity

To implement the Second-Order Similarity approach [3], we train a word2vec model for each decade using word2vec vectors with 300 dimensions, a 5-words context window and a minimum word frequency count of 20. As candidates for usage change, we collect words that appear in both model vocabularies. Then, for each candidate word we collect the word union of the top-25 neighbors of a word w from two different corpora. We remove from this list any words that are not found in the vocabularies of both corpora. Then, for each corpus, we compute the cosine similarity between each

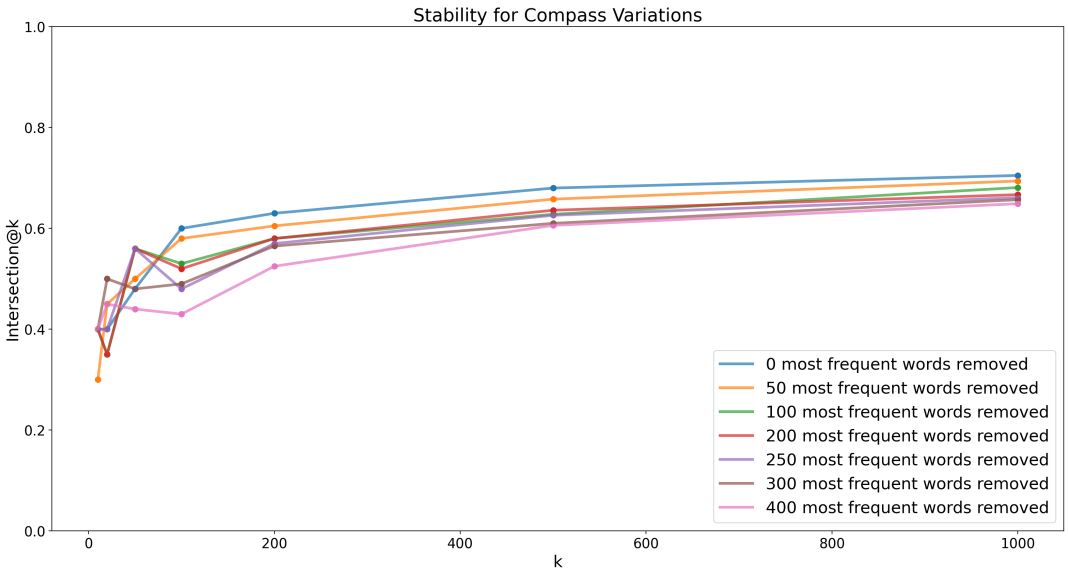


Figure 4: Stability study of Compass variations with different frequency cut-offs of the most frequent words.

word vector from the union list and the vector of the word w. This results in a second-order vector s for each corpus, that contains the cosine similarity of the word w with its neighbors at each vector space. Finally, we compute the cosine distance of these two second-order vectors in order to find the usage change of the word w.

To run the stability computations for this approach, use the following command from inside the `src` folder:

```
PYTHONHASHSEED=0 python hamilton_stability_1990s_2010s.py --run=0 \
    --iterations=10 --size=300 --rows='all'
```

5.1.6 Stability results

To produce Fig. 3 and Table 3 of the manuscript, we used `stability_comparison_all.ipynb`. This notebook takes as inputs the semantic computations of the aforementioned approaches, computes their stability with the intersection@k metric and creates Fig. 3 of the manuscript. It also creates one file per approach with the top 100 most changed words of the iteration with seed parameter set to 0. The output files are located in the `/out_files` folder and include: `compass_top100.csv`, `compass_fc_top100.csv`, `procrustes_top100 .csv`, `nn_top100.csv`, `second_order_top100.csv`. The files are also merged in file `top100_minfreq50.xls` for convenience.

In addition, we studied the effect that different frequency cut-offs have on the Compass variation. We implemented the Compass variation with different values of frequency cut-offs on the most frequent words for the task of detecting word usage changes between the decades 1990–1999 and 2010–2019. Specifically for each different random seed in the range 0–9, and for each frequency cut-off c, where $c \in [0, 50, 100, 200, 250, 300, 400]$, we collected the top-$k$ most changed words, where $k \in [10, 20, 50, 100, 200, 500, 1000]$. For all Compass variations in this study, we removed words that appear less than 200 times. Fig 4 presents the results of the stability study. For $k > 100$, the results remain consistent and show that as the number of most frequent words rises, the stability of the approach decreases. For $k < 100$, the results do not show a consistent behavior. For $c < 50$ and $c > 250$, the intersection@k measure is lower, indicating lower stability. However, the implementations with $c \in [100, 200, 250]$ present higher stability. These results imply that there is room for future studies on the effect of different frequency cut-offs on the stability of various semantic change detection approaches. For example, it would be interesting to investigate how the frequency cut-offs of the least frequent words might also affect the stability of an approach, either separately or when combined with different frequency cut-offs of the most frequent words in a corpus.

6 Word Usage Change Detection with Compass

For the reproducibility of the computations of word usage change in Section 5 of the manuscript, we use Compass and set the seed parameter to 5. We use word2vec vectors with 300 dimensions, we limit the word2vec model training to a single worker thread and we use the default settings of the Compass tool. Again, we set `PYTHONHASHSEED=0` for the scripts that facilitate Compass to compute word usage changes.

6.1 Top Changed Words before and after the Greek Economic Crisis

We detect word usage changes between the decade before the economic crisis ($t1$: 1997–2007) and the decade during ($t2$: 2008–2018). We compute the cosine similarity of the vectors of each word between the two decades, for those words that are common in both vocabularies and have a frequency of at least 50 occurences on any of the two time periods. We compute the frequencies of words with the script `freq_counter_for_crisis.py` that takes as input the file `tell_all_cleaned.csv` and creates as output files of word frequencies for $t1$ and $t2$, namely `freqs_for_semantic_shift_cleaned_data_period1997_2007.csv` and `freqs_for_semantic_shift_cleaned_data_period2008_2018.csv`.

Following that, the `script compass_crisis.py` uses these two CSV files and produces the file `semantic_shifts_dichotomy_crisis_compass_1997_2007_2008_2018_atleast50.csv` that includes the top 100 most changed words that occur at least 50 times in any of the two time periods.

The command for the Compass computations of the most changed words during the Greek economic crisis is the following:

```
PYTHONHASHSEED=0 python compass_crisis.py
```

6.2 Usage Change per Period

In order to compute the usage change of words between pairs of consecutive parliamentary periods, we create the respective training texts for Compass with the notebook `create_training_texts_per_period_for_compass.ipynb`. This notebook takes as input `tell_all_cleaned.csv`, preprocessed the speeches, groups them by parliamentary period, writes the grouped speeches to `PERperiod_df.csv` and finally writes the corpus of each period in a separate file in the directory `/src/training_texts/PERperiod/`.

6.2.1 Usage Change of Popular Topics

We estimate the usage change of selected topics that were debated across parliamentary periods. For the selection of topics, we consulted the website of Vouliwatch [6], a non-partisan parliamentary monitoring organization that provides an extensive comparison of party positions on topics of significant political interest. We found these topics to be popular, generic and of significant political interest. We populated the list of topics with different grammatical cases and singular or plural when needed, in order to capture different references of the topics in the records.

The initial list consists of 69 topics and is shown in Table 3:

We extended this list with 22 topics shown in Table 4, selected for their popularity:

We compute the usage change of the aforementioned topics with the script `selected_topics_shift_per_period_with_compass_multiple_iterations.py`. The script takes as an input parameter the random seeds that will be used for the computations. This format allows the user to run the script in parallel, each time defining different random seeds. The script outputs the results of its run in csv files in the `out_files` folder, named `/selected_topics_shift_per_period_compass_50iterations_seeds_***.csv` where `***` is the range of seeds used. We ran the script for seeds in range 0–49 in order to calculate the mean values and confidence intervals. An example of how to run the script is shown below:

[6] `https://vouliwatch.gr`

agricultural	national	preschool	traffic code
farmers	defence	asylum	police
growth	armed	university	refugee
investements	foreign	secondary	migratory
insurance	international	primary	refugees
labor	shipping	higher	immigrants
rights	islands	public	tourism
contract	water supply	private	health
collective	fishing	schools	welfare
salary	economy	environment	subsidies
minimum	tax-exempt	energy	subsidy
part-time	decentralization	system	infrastructure
brain	vat	culture	transportation
drain	businesses	sports	remodeling
oaed	tax	culture	public transport
justice	taxation	sports	
transparency	education	protection	
adoption	research	police	

Table 3: List of 69 topics collected from the website of Vouliwatch.

reduction	ose	homosexuals
raise	transportation	eoppy
retirement	bill	turkey
macedonia	religion	church
macedonian	religious	crisis
hirings	woman	arbitrariness
redundancies	man	
ekas	same-sex	

Table 4: List of 22 additional popular political topics.

```
PYTHONHASHSEED=0 python \
    selected_topics_shift_per_period_with_compass_multiple_iterations.py -s 0 1 2 3 4
```

We concatenate the results of different random seeds with the following command, using the script `concat_csvs_of_different_seeds.py`.

```
ls ../out_files/selected_topics_shift_per_period_compass_50iterations_seeds_* | python
concat_csvs_of_different_seeds.py -o
'../out_files/selected_topics_shift_per_period_compass_50iterations.csv'
```

The script outputs the file `selected_topics_shift_per_period_compass_50iterations.csv` with the semantic shifts of the aforementioned topics between consecutive pairs of periods.

We visualize the results with the script `usage_change_of_selected_topics_through_time_with_errorbars` which outputs the image `usage_change_of_selected_topics_through_time_with_errorbars.png`, correspondsing to Fig. 4 of the manuscript.

6.2.2 Usage Change of Political Party Name Embeddings

We gauge the usage change of the names of political parties that have played an important role in recent political history. Concerning the speech representation of each political party, there is significant deviation in size. We study usage changes on the most represented political parties, that have large enough corpora to produce valid results. New Democracy (hereafter ND), the Panhellenic Socialist Movement (hereafter PASOK) have the greatest shares and consist the most popular political parties in Greece, with each one having around 350M characters of speech in total. They are followed by the Coalition of the Radical Left—Progressive Alliance (hereafter SYRIZA) and the Communist Party of Greece (hereafter KKE), that span between 80M–120M characters of speech.

Next in size is the Coalition of the Left, of Movements and Ecology (hereafter SYN) with around 35M characters of speech. Finally, for historical purposes, we also include Golden Dawn (hereafter GD) with around 20M characters of speech. GD is a banned far-right ultranationalist political party that rose to prominence during Greece's financial crisis.

We train Compass between consecutive pairs of parliamentary periods and compute the cosine distance between the vectors of political party names with `compass_party_embeddings_per_period_multiple_iterations.py`. The script takes as an input parameter the random seeds that will be used for the computations. This format allows the user to run the script in parallel, each time defining different random seeds. The script outputs the results of its run in csv files in the `out_files` folder, named `semantic_shifts_party_embeddings_per_period_merged_compass_50iterations_seeds***.csv` where *** is the range of seeds used. We ran the script for s in range 0–49 in order to produce error bars and confidence intervals. An example of how to run the script is shown below:

```
PYTHONHASHSEED=0 python compass_party_embeddings_per_period_multiple_iterations.py -s 0
1 2 3 4
```

We concatenate the results of different random seeds with the following command, using the script `concat_csvs_of_different_seeds.py`. The parameter *-o* takes as input the name of the output file.

```
ls \
    ../out_files/semantic_shifts_party_embeddings_per_period_merged_compass_50iterations_seeds_* \
  | python concat_csvs_of_different_seeds.py -o \
  '../out_files/semantic_shifts_party_embeddings_per_period_merged_compass_50iterations.csv'
```

The script outputs the file `../out_files/selected_topics_shift_per_period_compass_50iterations.csv` with the semantic shifts of the political party name embeddings between consecutive pairs of periods.

We visualize the results with the script `compass_party_embeddings_per_period_errorbars.ipynb` which outputs the image `compass_party_embeddings_per_period_errorbars.png`, corresponding to Fig. 5 of the manuscript.

7 Dataset License & Maintenance

The dataset, support files, original parliament sitting records as well as the source code for the collection, cleaning and analysis of the dataset are and will remain publicly available on Zenodo and GitHub, under the Creative Commons Attribution 4.0 International license.

The primary data for our dataset are in the public record, as the official parliamentary proceedings of the country. That said, we are of course committed to consider any reasonable and lawful request, if and when it comes, to amend or remove entries.

References

[1] Valerio Di Carlo, Federico Bianchi, and Matteo Palmonari. Training Temporal Word Embeddings with a Compass. In *Proceedings of the Thirty–Third AAAI Conference on Artificial Intelligence*, AAAI'19, pages 6326–6334, 2019. doi: 10.1609/aaai.v33i01.33016326.

[2] Hila Gonen, Ganesh Jawahar, Djamé Seddah, and Yoav Goldberg. Simple, Interpretable and Stable Method for Detecting Words with Usage Change across Corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, ACL 2020, pages 538–555, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.51. URL `https://aclanthology.org/2020.acl-main.51`.

[3] William L. Hamilton, Jure Leskovec, and Dan Jurafsky. Cultural Shift or Linguistic Drift? Comparing Two Computational Measures of Semantic Change. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2016, pages 2116–2121, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1229. URL `https://www.aclweb.org/anthology/D16-1229`.

[4] William L. Hamilton, Jure Leskovec, and Dan Jurafsky. Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL 2016, pages 1489–1501, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1141. URL https://www.aclweb.org/anthology/P16-1141.

[5] Matthew A. Jaro. Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida. *Journal of the American Statistical Association*, 84(406):414–420, 1989. doi: 10.1080/01621459.1989.10478785.

[6] William E. Winkler. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. In *Proceedings of the Section on Survey Research Methods*, 1990.