Implementierung

Herunterladen der Protokolle

Konvertierung der XML-Dateien in tibbles

Reparieren von Fehlern

Analyse

Herunterladen der Protokolle

Funktion: `fetch_all(download_dir)`

- Protokolle als XML-Dateien von `bundestag.de` herunterladen und in `download_dir` speichern.

Herunterladen der Protokolle

Funktion: `fetch_all(download_dir)`

- Protokolle als XML-Dateien von `bundestag.de` herunterladen und in `download_dir` speichern.
- Problem: Maschinenunfreundliche Webseite

Herunterladen der Protokolle

Funktion: `fetch_all(download_dir)`

- Protokolle als XML-Dateien von `bundestag.de` herunterladen und in `download_dir` speichern.
- Problem: Maschinenunfreundliche Webseite
- Lösung: Source Code von `bundestag.de` nach Schnittstelle durchsuchen

Konvertierung der XML-Dateien in tibbles

Funktion: `read_all(filepath)`

- Liest jede XML-Datei in angegebenem Dateipfad einzeln

Konvertierung der XML-Dateien in tibbles

Funktion: `read_all(filepath)`

- Liest jede XML-Datei in angegebenem Dateipfad einzeln
- Extrahiert Sitzungsdatum, Rednerliste und Sitzungsverlauf

Konvertierung der XML-Dateien in tibbles

Funktion: `read_all(filepath)`

- Liest jede XML-Datei in angegebenem Dateipfad einzeln
- Extrahiert Sitzungsdatum, Rednerliste und Sitzungsverlauf
- Konvertiert Rednerliste in eine R Liste.

Konvertierung der XML-Dateien in tibbles

Funktion: `read_all(filepath)`

- Liest jede XML-Datei in angegebenem Dateipfad einzeln
- Extrahiert Sitzungsdatum, Rednerliste und Sitzungsverlauf
- Konvertiert Rednerliste in eine R Liste.
- Iteriert durch den Sitzungsverlauf, extrahiert Reden, Redebeiträge, Kommentare und Beifall

Konvertierung der XML-Dateien in tibbles

Funktion: `read_all(filepath)`

- Liest jede XML-Datei in angegebenem Dateipfad einzeln
- Extrahiert Sitzungsdatum, Rednerliste und Sitzungsverlauf
- Konvertiert Rednerliste in eine R Liste.
- Iteriert durch den Sitzungsverlauf, extrahiert Reden, Redebeiträge, Kommentare und Beifall
- Kombiniert alle Redner, Reden, Redebeiträge, Kommentare und Beifall zu 5 tibbles und gibt benannte Liste zurück.

Tabellen

Ergebnis der Konvertierung ist eine benannte Liste `res` mit tibbles:

Tabellen

Ergebnis der Konvertierung ist eine benannte Liste `res` mit tibbles:

```
> res$speaker
# A tibble: 1,025 x 7
  id      prename              lastname    fraction title role_short    role_long
  <chr>   <chr>                <chr>       <chr>    <chr> <chr>         <chr>
1 110021  Alterspraesident D   Otto Solms  NA       NA    Alterspraesi  Alterspraesi
2 110032  Carsten              Schneider   SPD      NA    NA            NA
# with 1,023 more rows
```

Tabellen

Ergebnis der Konvertierung ist eine benannte Liste `res` mit tibbles:

```
> res$speaker
# A tibble: 1,025 x 7
  id      prename              lastname   fraction title role_short   role_long
  <chr>   <chr>                <chr>      <chr>    <chr> <chr>         <chr>
1 110021  Alterspraesident D   Otto Solms NA       NA    Alterspraesi Alterspraesi
2 110032  Carsten              Schneider  SPD      NA    NA           NA
# with 1,023 more rows
```

```
> res$speeches
# A tibble: 25,068 x 3
  id          speaker  date
  <chr>       <chr>    <date>
1 ID19100100 11002190 2017-10-24
2 ID19100200 11002190 2017-10-24
# with 25,066 more rows
```

Tabellen

Ergebnis der Konvertierung ist eine benannte Liste `res` mit tibbles:

```
> res$speaker
# A tibble: 1,025 x 7
  id      prename              lastname   fraction title role_short   role_long
  <chr>   <chr>                <chr>      <chr>    <chr> <chr>        <chr>
1 110021  Alterspraesident D   Otto Solms NA       NA    Alterspraesi Alterspraesi
2 110032  Carsten              Schneider  SPD      NA    NA           NA
# with 1,023 more rows
```

```
> res$speeches
# A tibble: 25,068 x 3
  id          speaker  date
  <chr>       <chr>    <date>
1 ID19100100 11002190 2017-10-24
2 ID19100200 11002190 2017-10-24
# with 25,066 more rows
```

```
> res$talks
# A tibble: 63,663 x 3
  speech_id  speaker  content
  <chr>      <chr>    <chr>
1 ID19100100 11002190 "Guten Morgen, liebe Kolleginnen und Kollegen! Nehmen Sie
2 ID19100300 11003218 "Sehr geehrter Herr Praesident! Sehr geehrte Kolleginnen u
# with 63,661 more rows
```

```
> res$comments
# A tibble: 83,649 x 5
  speech_id  on_speaker fraction      commenter        content
  <chr>      <chr>      <chr>         <chr>            <chr>
1 ID19100300 11003218   BUENDNIS 90/D Katrin Goering   Was?
2 ID19100300 11003218   CDU/CSU       Volker Kauder    Warum habt ihr das bei Ge
# with 83,647 more rows
```

```
> res$comments
# A tibble: 83,649 x 5
   speech_id  on_speaker fraction      commenter        content
   <chr>      <chr>      <chr>         <chr>            <chr>
 1 ID19100300 11003218   BUENDNIS 90/D Katrin Goering   Was?
 2 ID19100300 11003218   CDU/CSU       Volker Kauder    Warum habt ihr das bei Ge
# with 83,647 more rows
```

```
> res$applause
# A tibble: 89,586 x 8
   speech_id  on_speaker CDU_CSU SPD   FDP   DIE_LINKE BUENDNIS_90_DIE_GRU  AfD
   <chr>      <chr>      <lgl>   <lgl> <lgl> <lgl>     <lgl>                <lgl>
 1 ID19100300 11003218   FALSE   TRUE  FALSE TRUE      TRUE                 FALSE
 2 ID19100300 11003218   FALSE   TRUE  TRUE  TRUE      FALSE                FALSE
# with 89,584 more rows
```

Reparieren von Fehlern

Problem: Uneinheitliche Schreibweisen / Fehler in den Rednerlisten.

Reparieren von Fehlern

Problem: Uneinheitliche Schreibweisen / Fehler in den Rednerlisten.
Lösung: Funktion: `repair_speaker(speakers)`

Reparieren von Fehlern

Problem: Uneinheitliche Schreibweisen / Fehler in den Rednerlisten.
Lösung: Funktion: `repair_speaker(speakers)`

- Erhält `tibble` von Rednern

Reparieren von Fehlern

Problem: Uneinheitliche Schreibweisen / Fehler in den Rednerlisten.
Lösung: Funktion: `repair_speaker(speakers)`

- Erhält `tibble` von Rednern
- Entfernt Redner mit ungültigen, doppelt vergebenen IDs

Reparieren von Fehlern

Problem: Uneinheitliche Schreibweisen / Fehler in den Rednerlisten.
Lösung: Funktion: `repair_speaker(speakers)`

- Erhält `tibble` von Rednern
- Entfernt Redner mit ungültigen, doppelt vergebenen IDs
- Vereinheitlicht Schreibweisen der Fraktionen, Namen und Titel der Redner

Reparieren von Fehlern

Problem: Namen in Kommentaren Rednern aus Rednertabelle zuordnen

Reparieren von Fehlern

Problem: Namen in Kommentaren Rednern aus Rednertabelle zuordnen
Lösung: Funktion `repair_comments(comments, speakers)`

- Erstellt für jeden Redner einen Regulären Ausdruck aus dem Namen
- Sucht für jeden Kommentar nach dem entsprechenden Eintrag in der Rednertabelle

Reparieren von Fehlern

Beide Reparaturschritte werden in der Funktion `repair` zusammengefasst.

Analyse

Stelle Hilfsfunktionen zur Analyse der Daten zur Verfügung:

- `bar_plot_fractions`: Erstellt ein Balkendiagramm aus einer Tabelle mit Fraktionsdaten

Analyse

Stelle Hilfsfunktionen zur Analyse der Daten zur Verfügung:

- `bar_plot_fractions`: Erstellt ein Balkendiagramm aus einer Tabelle mit Fraktionsdaten
- `find_word`: Fügt in der Redebeiträgetabelle zu jedem Redebeitrag die Häufigkeit eines Regulären Ausdrucks hinzu.

Analyse

Stelle Hilfsfunktionen zur Analyse der Daten zur Verfügung:

- `bar_plot_fractions`: Erstellt ein Balkendiagramm aus einer Tabelle mit Fraktionsdaten
- `find_word`: Fügt in der Redebeiträgetabelle zu jedem Redebeitrag die Häufigkeit eines Regulären Ausdrucks hinzu.
- `word_usage_by_date`: Zählt an welchen Daten (Tagen) ein regulärer Ausdruck wie oft verwendet wird.

Analyse

Stelle Hilfsfunktionen zur Analyse der Daten zur Verfügung:

- `bar_plot_fractions`: Erstellt ein Balkendiagramm aus einer Tabelle mit Fraktionsdaten
- `find_word`: Fügt in der Redebeiträgetabelle zu jedem Redebeitrag die Häufigkeit eines Regulären Ausdrucks hinzu.
- `word_usage_by_date`: Zählt an welchen Daten (Tagen) ein regulärer Ausdruck wie oft verwendet wird.
- `join_speaker`: Fügt einer Tabelle mit Spalte `speaker` die enstprechenden Informationen aus der Rednertabelle hinzu.