Congressional Hearing Database:

A Shiny App for Viewing and Searching Political Dialogue

Taylor Crockett

EPPS 6354: Information Management

6 May 2024

# Introduction

The process of legislation is simultaneously public and private. Elected officials have acclimated to using social media to curate their self-images, stay in touch with their constituents, and advocate for issues (Hoffmann, Suphan and Meckel 2016)—such is the rise of the “celebrity politician” of the 21st century, or one who has adopted communication advances to keep an ever-present hold in citizens’ lives and create competitive advantages. This advent in turn informs a more public manner of legislation that can be followed by those who previously would have lacked outlets to keep up: more and more individuals turn to social media to get their news, past 60% in 2017 (Fletcher and Nielsen 2017).

Because of the nature of public speech on online forums, historically, social media has been a boon for researchers to scrape data on policymakers and perform content analysis: for instance, one study attempted to categorize types of political posts on Instagram,[[1]](#footnote-1) and another study used Facebook text to create political profiles using a network model.[[2]](#footnote-2) However, this type of communication from legislators is intentionally and carefully curated, usually by large teams, to convey certain messages. The likelihood of public stances correlating with actual legislative dialogue is relatively unestablished, which is one of the foremost reasons I began to develop a method to collect and assess data on the content of legislative dialogue in a less-public setting.

In the United States, congressional committee hearings serve as a place for legislators to gather information and facilitate their own decision-making processes. Each committee is composed of a smaller body of legislators who often have experience in the committee’s specific area and who are distributed across parties uniformly according to the current Congress’s party distribution. In a given hearing, by calling witnesses and directing the flow of questioning, each legislator sheds large amounts of light on their own thought processes and intentions. For instance, as I found in a working paper that I’m developing contemporaneously to this project, Republican legislators were significantly more likely to hold hearings on China and to call witnesses who spoke on China than Democrats, especially when Republicans held that hearing’s respective chamber.

The possibilities of using committee hearing data are enormous, firstly because the data will satisfy a compromise between the public-private dimensionality of the legislative process: while committee hearings are most often public with the exception of closed-door hearings entailing classified information, the content of hearings is relatively difficult to access. For instance, to find transcripts of a hearing, one must visit the Government Publishing Office or its website and search for the desired transcript. Even among those who know how to access a hearing by attending, watching or reading, hearing content is relatively unpopular when opposed to floor speeches or interviews because it is heavily procedural and investigative, while other political content is intentionally explosive and flashy. I spoke more on the differences between hearing content and other political dialogue in a previous paper about Diversity, Equity and Inclusion (DEI) policy in the context of congressional committee hearings:

Why is dialogue in committee hearings hypothesized to differ from other political dialogue like press statements and social media posts? First, the dialogue is *actionable*. While political dialogue in a democracy that is meant to be public-facing is meant to be an advertisement, showcasing a policymaker's electability to their potential voters, committee hearing dialogue focuses on the policymaking itself. Hearings are a vital opportunity for legislators to get information that will lead to later decision-making on the floor (Diermeier and Feddersen 2000). Communication to voters is curated, but dialogue in hearings is less so. More importantly, the dialogue is relative to other policymakers rather than directly to the voters: certain legislators grandstand in order to gain leverage within or outside of their party (Park 2021).

Secondly, the dialogue is relatively *insulated*. Though there are some prominent cases of congressional committee hearings attracting public attention,[[3]](#footnote-3) most hearings pass by largely unwatched and unnoticed. In 2021 alone, there were 1,118 published hearing transcripts. To expect the average voter to know the goings-on in committee hearings is to expect the impossible. Therefore, in theory, legislators will engage in committee dialogue more focused on achieving their real policy goals rather than appealing to voters, because chances to make a statement are slim.

Due to these reasons, I see congressional committee hearings as an untapped resource for content analysis research. Furthermore, and more importantly, I see hearings as a vital part of the democratic process that are largely left secretive to the average voter, not because of an inherent secrecy by legislators, but simply because of a difficulty of access. The nature of the data is unwieldy: as accessed, transcripts are uncleaned, unorganized, and untagged. Namely, the transformation of hearing content into an accessible database is intense.

Therefore, a tool to compile all hearing content into an accessible and useful database will first require intense amounts of preprocessing and organization into a schema. Moreover, the data exist in large transcript chunks, but the data is most rich and useful at the person-entity level. Searching for a single hearing and accessing its entire uncleaned transcript is relatively easy, but the overhead of finding every instance of speech in that raw transcript by a single legislator, for instance, is probably enough to turn away anyone who wants to casually access the data as basic information. By parsing each transcript and organizing it at the person-entity level rather than at the hearing-entity level, the accessibility and usefulness of the tool will be multiplied.

The end goal of this project is to create an organized database and a clean, simple, and searchable interface tool to access it.

# Data Methodology

## Collection

Data was retrieved from the U.S. Government Publishing Office (GPO), which digitally provides federal documents at its GovInfo website service, a succession of the previous FDsys service. To pull and compile transcripts was a multistep process that involved the GovInfo developer API.

First, I used the API to pull a list of all published packages in the congressional committee hearing library of content, selecting anything published 2009 and after due to older hearings not having OCR-processed transcripts. As defined by the GPO, a package is a single entity of content that corresponds with a publication event. In the context of hearings, a single package corresponds with a single hearing. Pulling the list of published packages allowed me to get metadata, including unique identifier ‘packageId’ and public URLs to the transcript. Using a Python script, I found the metadata then inserted them into an SQL table in a quick fashion by using a thread pool executer and pooling features in the ‘psycopg2’ package.

With transcript URLs in hand, I iteratively requested each transcript in its ‘.htm’ format, which was unorganized but able to be inserted into a separate SQL table as a text feature. I vertically partitioned the two tables, even though they were organized around hearings as entities and used unique identifier ‘packageId’ as the primary key, because of the massive size of each transcript, so that indexed hearing metadata could be accessed more quickly.

I also relied on the Library of Congress (LoC) BioGuide as a reference for legislators. Data on legislators could be useful to identify individual speakers in the speaker attribution process, and the BioGuide identifier (‘bioguideId’) as provided by the LoC corresponded with metadata provided in XML format from the GPO. Furthermore, using the ‘bioguideId’ I can access government-provided portraits as web assets in the database tool. I pulled a .csv file of all legislators who served in 2009 or after, unnested a json-format column that listed each legislator’s unique terms of service, and created two new tables in SQL.

## Preprocessing

The first task at hand was to divide each transcript into speech instances, which I will refer to as “dialogue turns.” A dialogue turn is a segment of speech attributed to a single speaker before another person takes over speaking. For instance:

|  |  |  |
| --- | --- | --- |
| *Turn 1* | Ms. Lofgren | This joint hearing of the Subcommittee on Immigration, Citizenship, Refugees, Border Security, and International Law, as well as the Subcommittee on the Constitution, Civil Rights, and Civil Liberties, will come to order. |
| *Turn 2* | Mr. King | Thank you, Madam Chair. I appreciate this hearing. And I appreciate the witnesses coming forward to testify. It is never easy to sit down before this Congress (…) |

Separating into dialogue turns breaks down the transcript-level entity table into a person-level entity table. At the person-level, a searchable table of legislators rather than transcripts yields much more information. For instance, looking up every transcript that contains the word “China” may yield half of the available transcripts, but the more granular person-level allows for the analysis of individual contributions, identifying specific legislators who frequently discuss China, thus facilitating targeted searches and more nuanced insights into patterns of discourse among different members.

To create the person-level table, I used regular expressions. While transcript format is not entirely uniform, all transcripts rely on the following format to convey a dialogue turn:

|  |
| --- |
| Title Speaker1. Content (…).  Title Speaker2. Response (…). |

By creating a regular expression query that found every four-space indent followed by any title in a self-defined title dictionary that includes terms like “Mr.,” “Senator,” and “Director,” I am able to extract surnames from the transcript, as well as the content that follows, thereby attaining information at the person-level.

Furthermore, I was able to use more metadata from the GPO to ascertain which legislators were present at each committee hearing. Unfortunately, published records of past committee membership are nearly impossible to find because it is ultimately a less-than-formal process defined by the controlling party—using transcript hearings to trace committee membership may prove to be an effective solution for that data problem.

With that metadata, I joined the ‘bioguideId’ and associated information from the BioGuide onto the person-level table. One step that I wished to implement but did not was a more granular assignment of BioGuide information matched onto that specific Congress: with Congress as a unit of time, I can join that speaker’s party, position, and state into the dialogue table.

## SQL

An SQL database was created and hosted on a serverless platform using CockroachDB. I chose CockroachDB because it provides a large amount of free storage that is later scalable into higher use if I decide to formalize and publish this database tool. Cockroach operates on PostgreSQL infrastructure, so most of my querying, infrastructure and other SQL system approach was in a format for Postgres.

The database design entailed 6 unique tables tied throughout by referring to primary keys ‘packageId’ and ‘bioguideId’. Each are government-provided unique identifiers for hearing content and legislators, respectively. The ‘attendance’ table is a junction table intended to index the intersection of hearings and legislators: common surnames like Smith cannot be discerned without supplemental metadata or else there will be duplicative attribution.

A screenshot of a computer

Description automatically generated

Figure 1: SQL Schema (dbdiagram.io)

# Database Interface

The database interface was designed with simplicity in mind. It follows a two-panel sidebar structure: the sidebar panel is for querying, and the main panel is to display results.

For a simple and deployable system, I used R Shiny, which provides server-side operations that will reflect user input. The Shiny webapp queries as a PostgreSQL user based on configuration arguments hidden in a .yml file. It leverages the smooth operations of SQL to make unique queries to my CockroachDB database by joining the ‘member’ and ‘hearing’ tables for their respective metadata on the ‘dialogue’ table, which is the person-level entity table of dialogue turns. Queries are refined using user input leveraging operators such as LIKE for text matching and BETWEEN for date comparison.

The main panel output was created using the ‘reactable’ package, which is a flexible wrapper of a powerful JavaScript table tool. Reactable allows for sortable and filterable tables with flexible formatting—I was able to insert legislators’ LoC portraits as HTML into the table output.

The keyword user input argument is the most powerful search query because it calls on a pie chart from the ‘highcharter’ package, another JavaScript wrapper, to display the frequency of that term in all filtered content. For instance, if the search is “Smith” for speaker name and “111” for the Congress, the keyword “China” calls on a chart with frequency dialogues turns from legislators “Smith” in the 111th Congress that refer to “China.”

# Future Steps

I have some ideas in mind for future iterations of this tool.

* I want to refine and perfect the backend data processes and also gather metadata on non-legislator witnesses. The regular expression still results in errata and refinement is important to guarantee the legitimacy of the tool.
* More metadata should be provided (and filterable), such as the congressional body, the committee name, the party of the speaker, their position (Senator/Representative) and the state they serve.
* Further information on speakers should exist when a dialogue turn is clicked. For instance, clicking on a representative should call a popup that shows the term they served, their district, and their party.
* The pie chart is informative, but a frequency chart over time of certain keywords would be even more informative.
* Because certain dialogue turns are extremely lengthy, truncating entities with long dialogues will preserve the visual integrity of the output table.

# References

Diermeier, Daniel, and Timothy J. Feddersen. 2000. "Information and Congressional Hearings." *American Journal of Political Science* 44 (1). doi:doi.org/10.2307/2669292.

Fletcher, R., and R.K. Nielsen. 2017. "Are news audiences increasingly fragmented? A cross-national comparative analysis of cross-platform news audience fragmentation and duplication." *Journal of Communication* 476-498. doi:https://doi.org/10.1111/jcom.12315.

Grčić, Klara, Marina Bagić Babac, and Vedran Podobnik. 2017. "Generating Politician Profiles based on Content Analysis." *Journal of Universal Computer Science* 23 (3): 236-255.

Hoffmann, Christian Pieter, Anne Suphan, and Miriam Meckel. 2016. "The impact of use motives on politicians’ social media adoption." *Journal of Information Technology & Politics* 13 (3): 239-256. doi:https://doi.org/10.1080/19331681.2016.1169241.

Koblin, John. 2022. "At Least 20 Million Watched Jan. 6 Hearing." *New York Times.* https://www.nytimes.com/2022/06/10/business/media/jan-6-hearing-ratings.html.

Park, Ju Yeon. 2021. "When Do Politicians Grandstand? Measuring Message Politics in Committee Hearings." *The Journal of Politics* 83 (1).

Peng, Yilang. 2021. "What Makes Politicians’ Instagram Posts Popular? Analyzing Social Media Strategies of Candidates and Office Holders with Computer Vision." *The International Journal of Press/Politics* 26 (1): 143-166. doi:https://doi.org/10.1177/1940161220964769.

1. Peng 2021 [↑](#footnote-ref-1)
2. Grčić, Bagić Babac and Podobnik 2017 [↑](#footnote-ref-2)
3. Koblin 2022 [↑](#footnote-ref-3)