The SPoRC Package

IC²S² 2026 Tutorial · Podcasts as Social Data

David Jurgens and Dallas Card
July 28, 2026

UNIVERSITY OF MICHIGAN

What is actually in SPoRC

SPoRC Statistic	Value
Podcasts	228,099
Episodes	1,124,058
Speaker turns	185M, 65% episodes diarized
Audio	~650,000 hours, transcribed with Whisper base.en
On disk	~57 GB, partitioned by podcast
Terms	dataset gated, research and education only

- **Per episode you get:** title, description, RSS URL, mp3_url, duration, up to 10 iTunes categories, language, num_main_speakers, overlap proportion, predicted host and guest names
- **Per turn you get:** text, speaker label, start/end time, inferred role, word count, words per second, six acoustic means (MFCC 1–4, one F0, one F1)

SPoRC does not contain audio, but you can recrawl the URL mp3

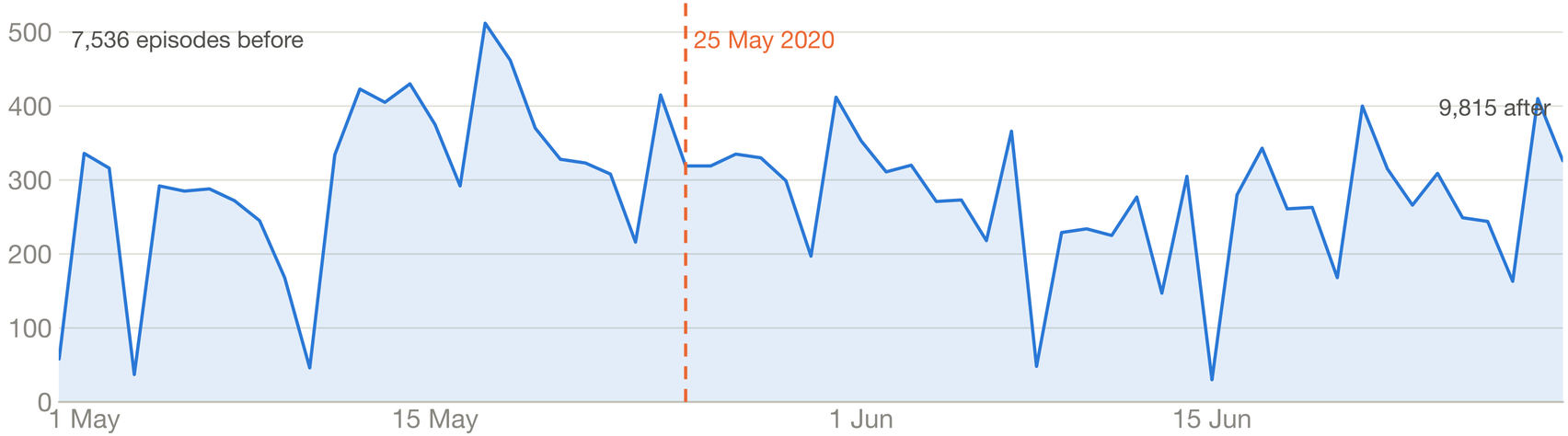
Litterer, Benjamin Roger, David Jurgens, and Dallas Card. "Mapping the Podcast Ecosystem with the Structured Podcast Research Corpus." *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vienna, July 2025, 25132–25154. DOI 10.18653/v1/2025.acl-long.1222. https://aclanthology.org/2025.acl-long.1222/

SPoRC is a two month snapshot of May/June 2020

It was a crazy time, if you remember

- **COVID-19, first wave** — lockdowns across the whole window, and remote recording everywhere
- **25 May 2020 —** George Floyd is murdered in Minneapolis. Protests run through June

The window straddles a sharp, dateable event with roughly four weeks on each side, which makes before/after designs relatively clean.



You don’t need all 57GB to work with SPoRC

```
from sporc import SPORCDataset
sporc = SPORCDataset()                              # catalogs only, ~195 MB
sporc = SPORCDataset(subset=["The NPR Politics Podcast", "Radiolab"])
sporc = SPORCDataset(subset="my_ids.txt", allow_downloads=False)  # pin the slice
sporc = SPORCDataset(parquet_dir="subsets/subset_1")              # fully offline
sporc = SPORCDataset(lazy=False)                # the whole thing, ~31 GB, ~685k files
```

Study size	Data fetched
10 podcasts	~750 KB
200 podcasts	~15 MB
1,000 podcasts	~75 MB
whole corpus	~31 GB

- The corpus is partitioned by podcast, so you fetch podcasts, not the corpus.
- Selection is metadata-only. Search the catalogs first, `prefetch()` second, and you never touch a partition you did not need
- `allow_downloads=False` will throw a `DataNotLocalError` if you end using operations that would pull in 30GB

Search metadata first

```
sporc.search_podcast("The NPR Politics Podcast")
sporc.search_episodes(category="news", min_duration=1800,
                      min_speakers=2, max_speakers=4,
                      language="en", max_episodes=50)
sporc.search_episodes_by_subcategory("Technology",
                                     max_episodes=10)
sporc.filter_episodes_by_metrics(min_word_count=5000,
                                 max_host_proportion=0.6)
```

Filter on	Criterion
Length	`min_duration` / `max_duration` (seconds)
Number of speakers	`min_speakers` / `max_speakers` → `num_main_speakers`
Language	`language="en"`
Topic	`category` / `subcategory`, from up to 10 iTunes categories
People	`host_name` / `guest_name`
Turn structure	`filter_episodes_by_metrics`: `word_count`, `turn_count`, `speaking_rate`, `discourse_marker_rate`,

- **Always pass `max_episodes`.** Matching is metadata-only, but *building* each Episode reads that podcast's partition; e.g., `category="comedy"` matches 62,622 episodes across 14,668 podcasts
- `filter_episodes_by_metrics` implies turn data, so it silently covers only the diarized 65%

Search for words and phrases

```
sporc.search_turns(“social science”)            # BM25-ranked
sporc.search_turns("climate", mode="exact", speaker_role="host")
sporc.search_episodes_by_text("mask mandate", mode="regex")
sporc.concordance("like", context_words=5)              # keyword in context
```

Local data	Scan (pyarrow)	26 GB DuckDB index
250 episodes (5 MB)	0.37 s	— needs the full index
1,000 episodes (17 MB)	**1.1 s**	"
5,000 episodes (83 MB)	5.4 s	"
all turns	impractical	1.7 s open, ~5 s/query (50 s cold)

- Below ~5,000 episodes, scanning is faster. The 26GB index is for whole-corpus work only
- Without the index, mode="fts" ranks by raw term frequency rather than BM25, and covers only what you have locally
  - The package will also warn you and reports how many podcasts it scanned

Host and guest fields in SPoRC

Podcasts are social networks too

Field	What it is
host_predicted_names / guest_predicted_names	a name was detected in the text
host_speaker_labels / guest_speaker_labels	a name was attached to a diarized voice
turn.inferred_speaker_role	this turn is host, guest, or `NO_INFERRED_ROLE`
num_main_speakers	how many voices the diarizer found

- Role labels cover a minority of turns.
- `get_host_turns()` and `get_guest_turns()` therefore return a slice, not a partition
- An unlabelled turn does not mean nobody spoke
- There are still likely issues from variations in name transcription

Let's see `sporc` in action