## **📄 Key Insights from *The Spotify Million Playlist Dataset Challenge***

Below are some of the important details and insights from the papers and articles about Spotify’s Million Playlist Dataset (MPD) and the RecSys Challenge. These are what we might write down in our notes.

### **1. Dataset Details**

* **Size & Scope**:  
  + 1,000,000 playlists. [Kaggle+3Spotify Research+3arXiv+3](https://research.atspotify.com/2020/09/the-million-playlist-dataset-remastered?utm_source=chatgpt.com)
  + Over **2 million unique tracks**. [Spotify Research+1](https://research.atspotify.com/2020/09/the-million-playlist-dataset-remastered?utm_source=chatgpt.com)
  + ~ 300,000 unique artists. [arXiv+1](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
  + Many playlists include metadata: playlist name/title, number of tracks, number of albums, number of followers, etc. [AIcrowd+2Spotify Research+2](https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge?utm_source=chatgpt.com)
* **Time period**: Playlists were created between ~January 2010 and November 2017. [Spotify Research+2arXiv+2](https://research.atspotify.com/2020/09/the-million-playlist-dataset-remastered?utm_source=chatgpt.com)
* **Fields / Features**:  
  + Playlist title (name) [Spotify Engineering+2Spotify Research+2](https://engineering.atspotify.com/introducing-the-million-playlist-dataset-and-recsys-challenge-2018?utm_source=chatgpt.com)
  + Tracks list with track metadata (artist, album, track name) [arXiv+2Spotify Research+2](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
  + Other metadata: number of tracks, number of albums, number of edits, modified timestamp. [arXiv+2Spotify Research+2](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
* **Challenges in the data**:  
  + Many playlists have no seed tracks (i.e., empty) in test scenarios. [arXiv+1](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
  + Playlist titles are very sparse and noisy; using title alone is hard. [arXiv+1](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
  + The dataset does **not** include acoustic/audio features in the base dataset — you need to fetch them via Spotify API or external sources if needed. [arXiv+1](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)

### 

### **2. Task Definition & Evaluation**

* **Task**: *Automatic Playlist Continuation (APC)* — given some seed tracks (maybe also a playlist title), predict or recommend what tracks should be added to continue the playlist. [AIcrowd+1](https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge?utm_source=chatgpt.com)
* **Different scenarios**: In the test set, there are playlists with different levels of “seed” information:  
   e.g.
  + Only title, no tracks
  + Title + first 5 tracks
  + First 5 tracks
  + Title + first 10 tracks
  + First 10 tracks
  + Title + first 25 tracks
  + Random 25 tracks + title
  + etc. [arXiv](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
* **Prediction Output**: Usually recommend up to 500 tracks per playlist in ordered list of relevance. [arXiv+1](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
* **Metrics**:  
  + R-precision (how many relevant tracks you retrieved among all relevant ones) [arXiv+1](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
  + NDCG (Normalized Discounted Cumulative Gain) — cares about order of recommendations. [arXiv+1](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
  + “Recommended Songs Clicks” — number of refreshes before encountering a relevant track. Lower is better. [AIcrowd+1](https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge?utm_source=chatgpt.com)

### **3. Methods & What Worked / Didn’t Work**

* Many teams used **ensemble methods**: combining multiple approaches (collaborative filtering, neighborhood-based, content-based, learning-to-rank). [arXiv](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
* Collaborative filtering / matrix factorization models were strong contenders. Neighborhood methods are also used. [arXiv](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
* Using playlist **titles** didn’t help much in many cases. The titles are sparse, noisy, and often don’t give enough signal. [arXiv](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
* When there are **more seed tracks** (more context), models perform much better than when only titles or very few tracks are given. [arXiv](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
* The creative track (where external features like audio descriptors are allowed) did not drastically outperform the main track; i.e. extra features helped but not always by a huge margin. [arXiv+1](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)

### **4. Limitations & Future Work**

* Lack of audio content / acoustic features in base dataset → some teams fetched extra features via Spotify API. This is costly (API rate limits etc.). [Spotify Research+1](https://research.atspotify.com/2020/09/the-million-playlist-dataset-remastered?utm_source=chatgpt.com)
* Sparse/ambiguous playlist titles. Some titles are too generic or don’t reflect mood/theme clearly. [arXiv](https://arxiv.org/pdf/1810.01520?utm_source=chatgpt.com)
* Many tracks/artists might be rare, i.e. long-tail problems. Some tracks have very few examples, making it hard to recommend them well. (Cold start for tracks)
* Scalability: processing large dataset, speed, memory constraints.

### **5. What you can take / use for *your* project**

Here are some ideas how the above insights can inform your own project:

* Because titles are noisy, maybe use **title + lyrics + social posts** together; not rely on title alone.
* Audio features (danceability, valence, energy, etc.) could help enrich predictions (you’ll use them maybe via Spotify API).
* Hybrid approaches (combining collaborative filtering + content + mood detection) tend to do well in these tasks.
* Be careful about scenarios: your system should perform well even when seed tracks are few or none (title-only). So test for those cases. For evaluation, use R-precision, NDCG, perhaps also click-based metric or some proxy, so people can compare or understand.