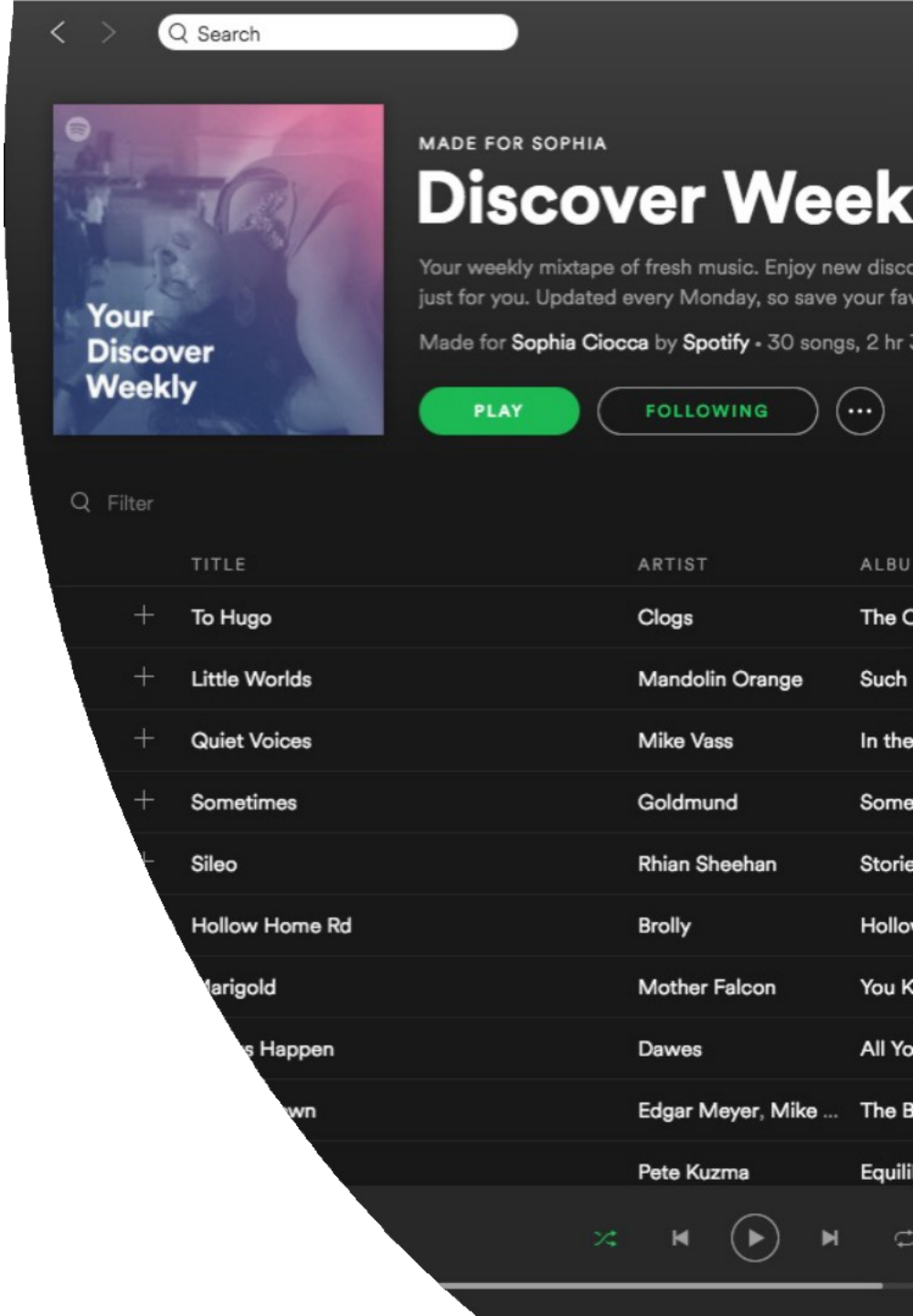
From Spotify to All of You

Machine Learning for Business

Prof. SEE-TO Wing Kuen, Lingnan University

From Spotify to all of you

- Spotify: one of the most popular online music stream platform.
- A feature of this platform is to offer a custom mixtape of 30 songs they've never listened to before but will probably love



From Spotify to all of you

But have you ever asked

How Does Spotify Know You So Well?

Many users show very high satisfactory of this recommendation system

From Spotify to all of you

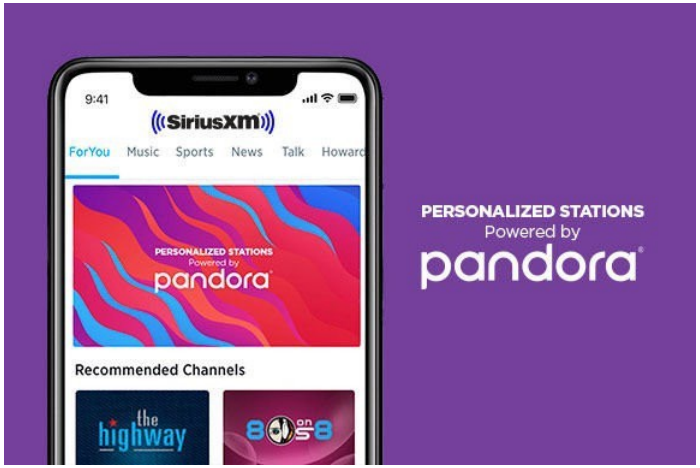
So how does Spotify do such an amazing job of choosing those 30 songs for each person each week?

Let's zoom out for a second to look at how **other music** services have tackled music recommendations, and how Spotify's doing it better.

From Spotify to all of you

Songza: *manual curation*
Couldn't take into account each listener's individual music taste

Pandora: *manually tagging attributes*
A group of people listened to music, chose a bunch of descriptive words for each track, and tagged the tracks accordingly.



From Spotify to all of you

The Echo Nest : *used* ***algorithms*** *to analyze the* ***audio and textual content*** *of music* Allowing it to perform music identification, personalized recommendation, playlist creation, and analysis.

Last.fm: collaborative filtering
To identify music its users might like, but more on that in a moment.

From Spotify to all of you

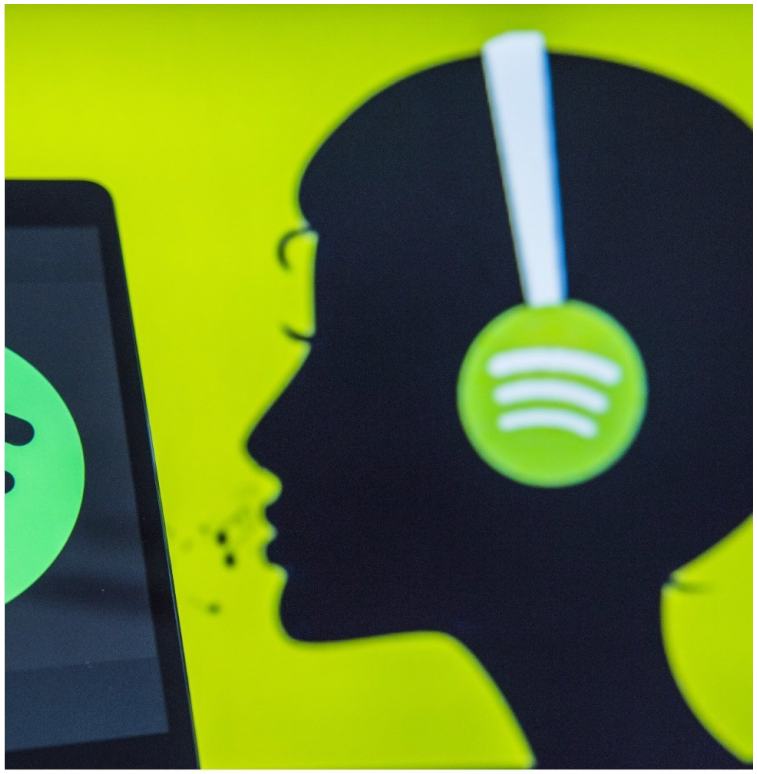


Photo by studioEAST/Getty Images

How does Spotify's magic engine run?

Spotify's Three Types of Recommendation Models

Recommendation Model #1: Collaborative Filtering

Collaborative Filtering

Netflix

- It was one of the first companies to use this method to power a recommendation model.
- Taking users' star-based movie ratings to inform its understanding of which movies to recommend to other similar users.

From Spotify to all of you

But Spotify doesn't have a star-based system with which users rate their music.

Spotify's data is implicit feedback

The **stream counts** of the tracks and additional streaming data, such as -

- whether a user saved the track to their own playlist
- visited the artist's page after listening to a song.

From Spotify to all of you

How does Collaborative Filtering works?

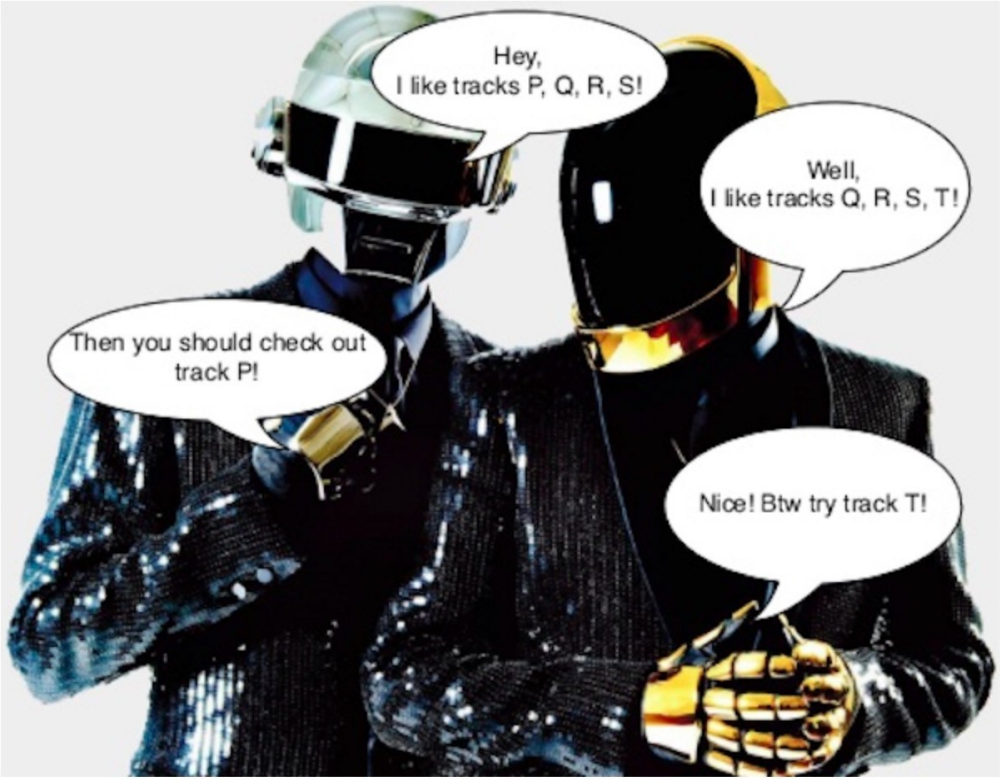


Image source: Collaborative Filtering at Spotify, by Erik Bernhardsson, ex-Spotify.

A likes tracks P,Q,R,S

B likes tracks Q,R,S,T Then…

They may be similar user as they both like Q, R and S

A may like T (B liked)

B may like P (A liked)

From Spotify to all of you

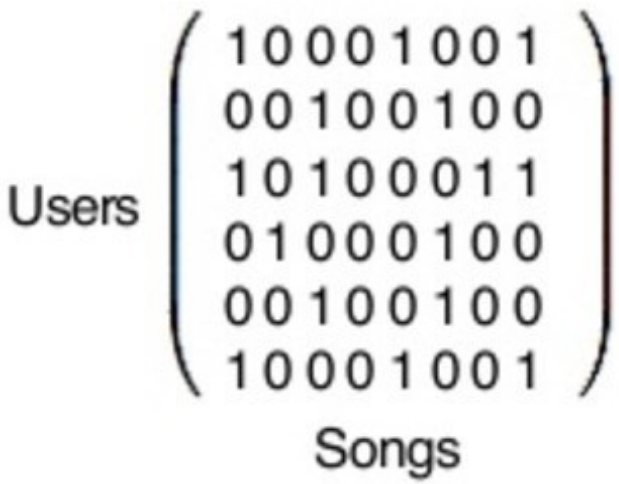
This is the situation of two users.

But how does Spotify actually use that concept in practice to calculate **millions** of users' suggested tracks based on millions of other users' preferences?

$$M = \underbrace{\begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & & & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mn} \end{pmatrix}}_{10^7 \text{ items}} \Bigg\} 10^7 \text{ users}$$

With matrix math, done with Python libraries!

From Spotify to all of you



That is a very big matrix…

In actuality, this matrix you see here is gigantic.

Each row represents one of Spotify's 140 million users — if you use Spotify, you yourself are a row in this matrix

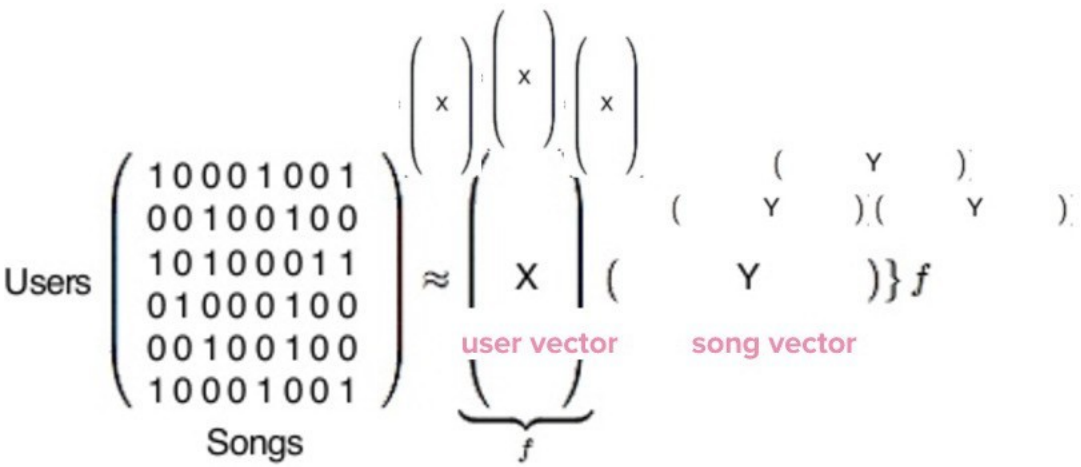
Each column represents one of the 30 million songs in Spotify's database.

From Spotify to all of you

Then, the Python library runs this long, complicated matrix factorization formula:

$$\min_{x,y} \sum_{u,i} c_{ui}(p_{ui} - x_u^T y_i - \beta_u - \beta_i)^2 + \lambda(\sum_u \|x_u\|^2 + \sum_i \|y_i\|^2)$$

Output X and Y vector

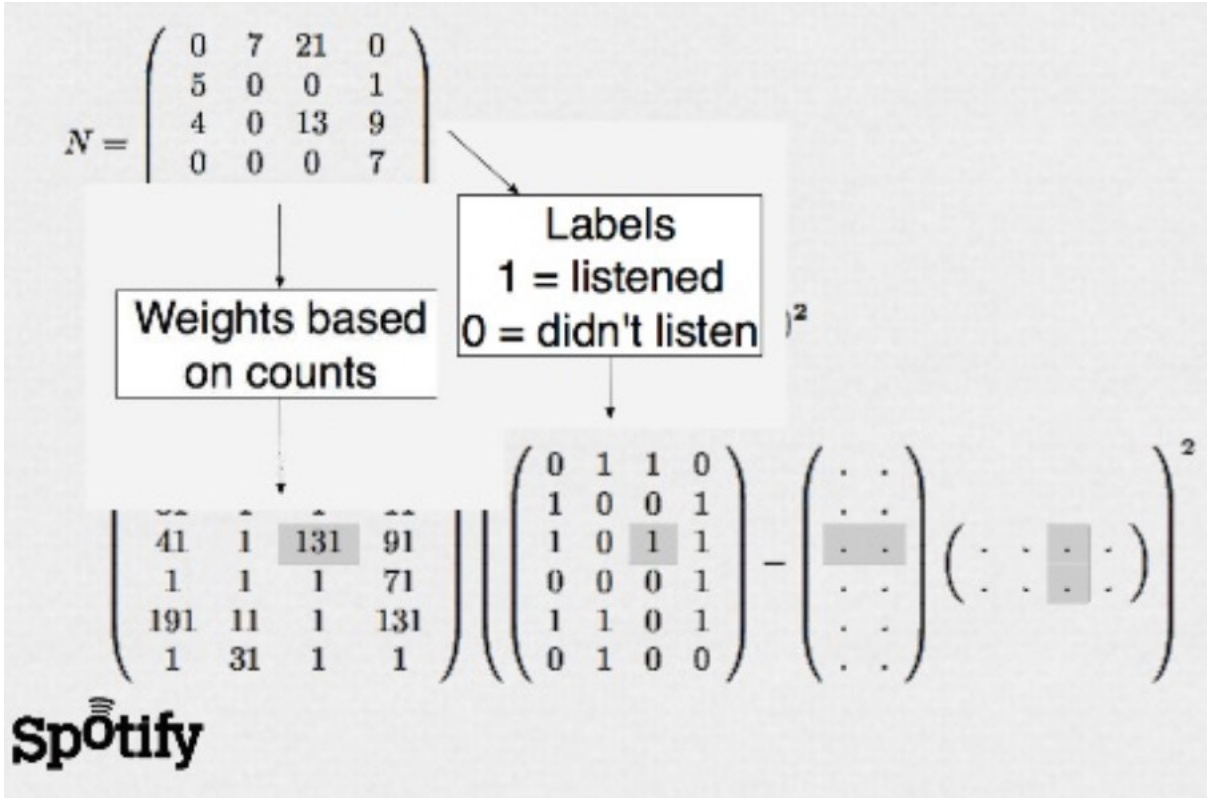


X is a *user vector*, representing one single user's taste

Y is a *song vector*, representing one single song's profile.

From Spotify to all of you

If we want more factors?



Add the labels into the vector

From Spotify to all of you

How does Collaborative Filtering works?

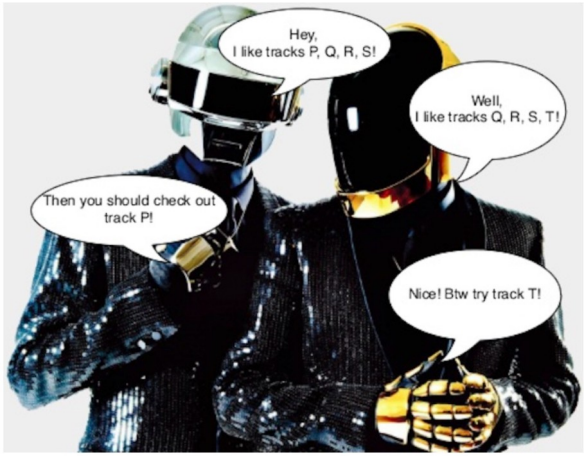
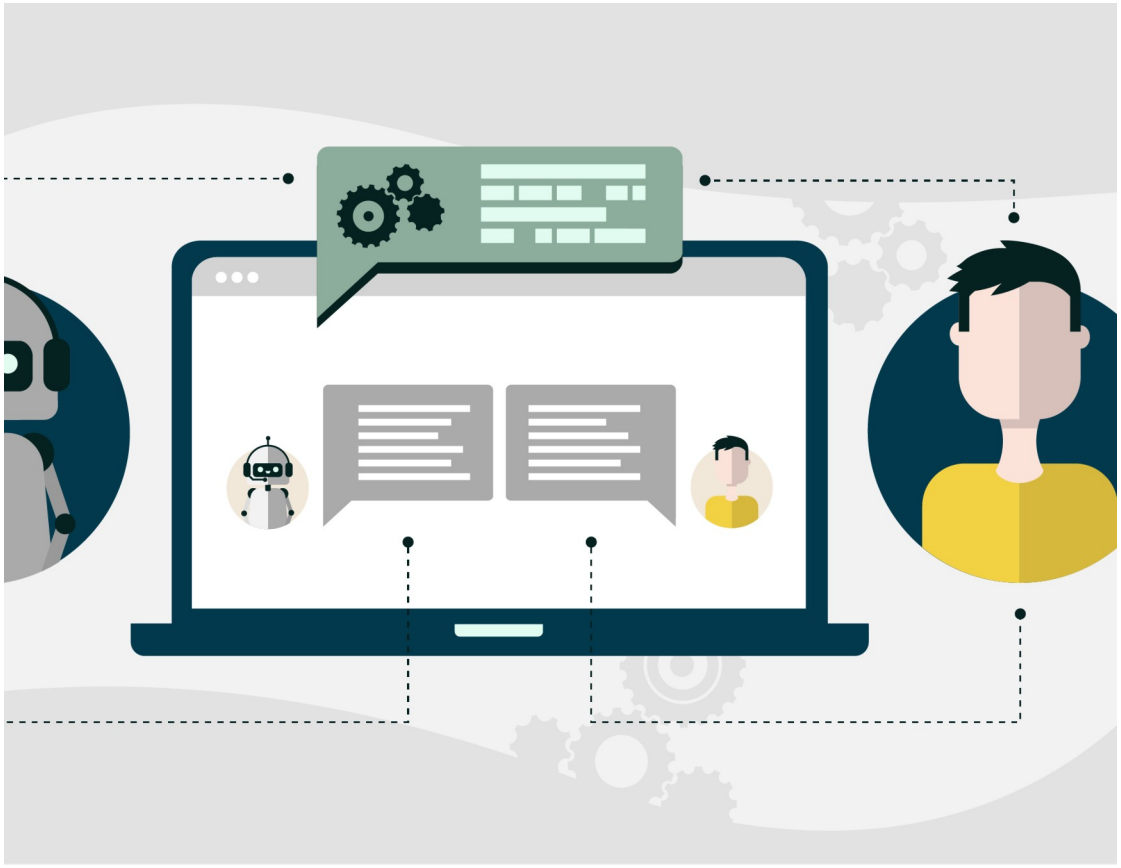


Image source: Collaborative Filtering at Spotify, by Erik Bernhardsson, ex-Spotify.

To find out which users' musical tastes are most similar to mine, collaborative filtering **compares my vector with all of the other users' vectors**, ultimately spitting out which users are the closest matches.

The same goes for the **Y vector, *songs***: you can compare a single song's vector with all the others, and find out which songs are most similar to the one in question.



Recommendation Model #2: Natural Language Processing (NLP)

The source data for these models, as the name suggests, are regular ol' words:

- Track metadata
- News articles
- Blogs
- Other text around the internet

From Spotify to all of you

Natural Language Processing, which is the ability of a computer to understand human speech as it is spoken.

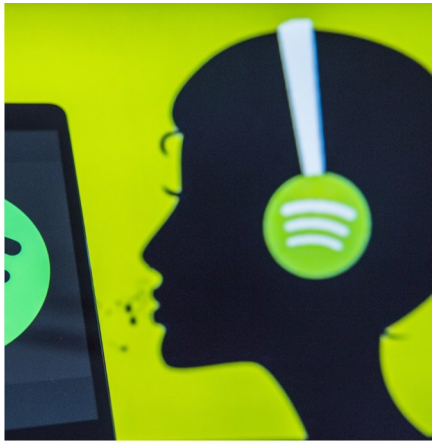


Photo by studioEAST/Getty Images

Spotify **crawls the web** constantly looking for blog posts and other written text about music to figure out what **people are saying** about specific artists and songs, like

- which adjectives
- what particular language is frequently used in reference to those artists and songs, and which other artists and songs are also being discussed alongside them.

From Spotify to all of you

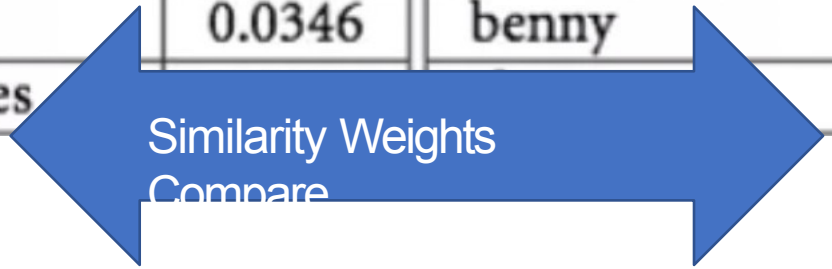
n2 Term	Score	np Term	Score	adj Term	Score
dancing queen	0.0707	dancing queen	0.0875	perky	0.8157
mamma mia	0.0622	mamma mia	0.0553	nonviolent	0.7178
disco era	0.0346	benny	0.0399	swedish	0.2991
winner takes	0.0307	chess	0.0390	international	0.2010
chance on	0.0297	its chorus	0.0389	inner	0.1776
swedish pop	0.0296	vous	0.0382	consistent	0.1508
my my	0.0290	the invitations	0.0377	bitter	0.0871
s enduring	0.0287	voulez	0.0377	classified	0.0735
and gimme	0.0280	something's	0.0374	junior	0.0664
enduring appeal	0.0280	priscilla	0.0369	produced	0.0616

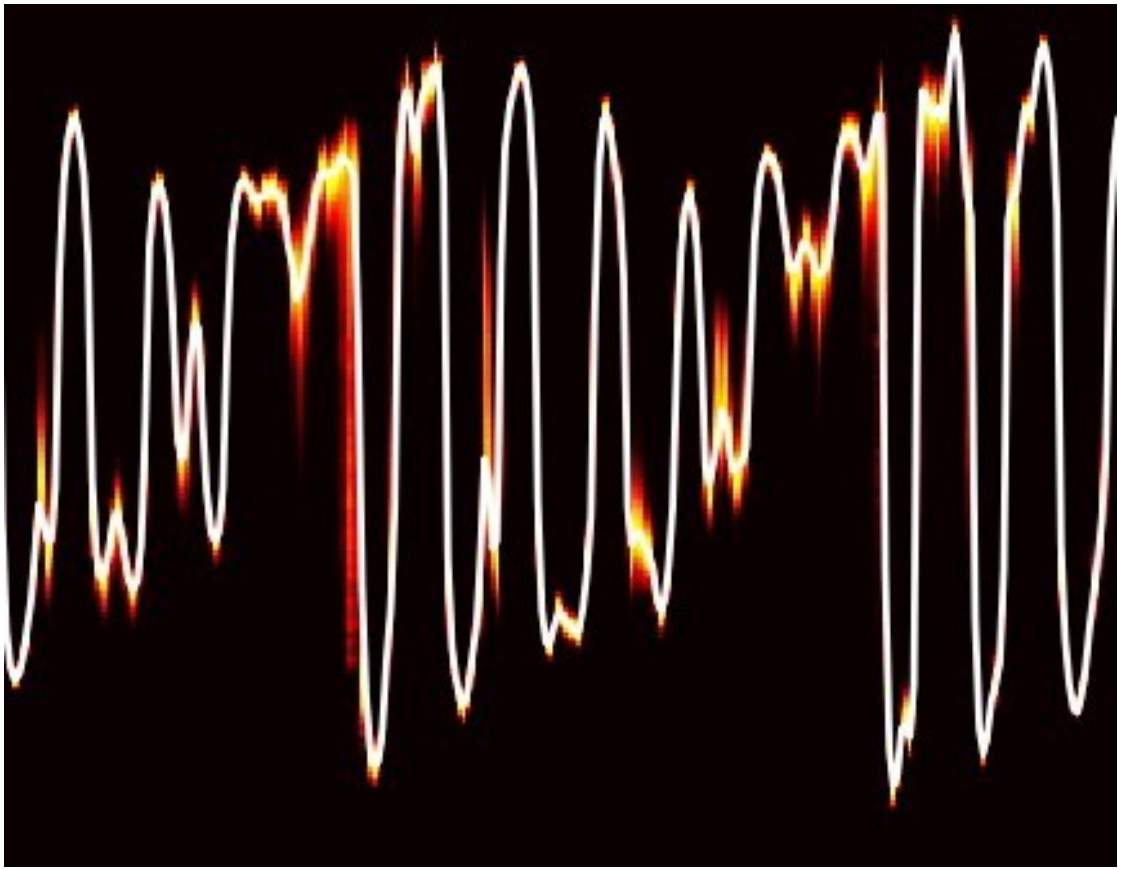
"Cultural vectors" or "top terms," as used by the Echo Nest. Image source: How music recommendation works — and doesn't work, by Brian Whitman, co-founder of The Echo Nest.

From Spotify to all of you

Then, much like in collaborative filtering, the NLP model uses these terms and **weights** to create a vector representation of the song that can be used to determine if two pieces of music are similar.

n2 Term	Score	np Term	Score
dancing queen	0.0707	dancing queen	0.0875
mamma mia	0.0622	mamma mia	0.0553
disco era	0.0346	benny	0.0399
winner takes			0.0390





Recommendation Model #3: Raw Audio Models

We already have so much data from the first two models.

Why do we need to analyze the audio itself, too?

From Spotify to all of you

We have already considering the analytics of popular songs, but how about new songs?

- A song your singer-songwriter friend has put up on Spotify.
- Maybe it only has 50 listens, so there are **few other** listeners to collaboratively filter it against.
- It also **isn't mentioned anywhere** on the internet yet, so NLP models won't pick it up.

How to help users find these good songs out?

From Spotify to all of you

Luckily, raw audio models don't discriminate between new tracks and popular tracks.

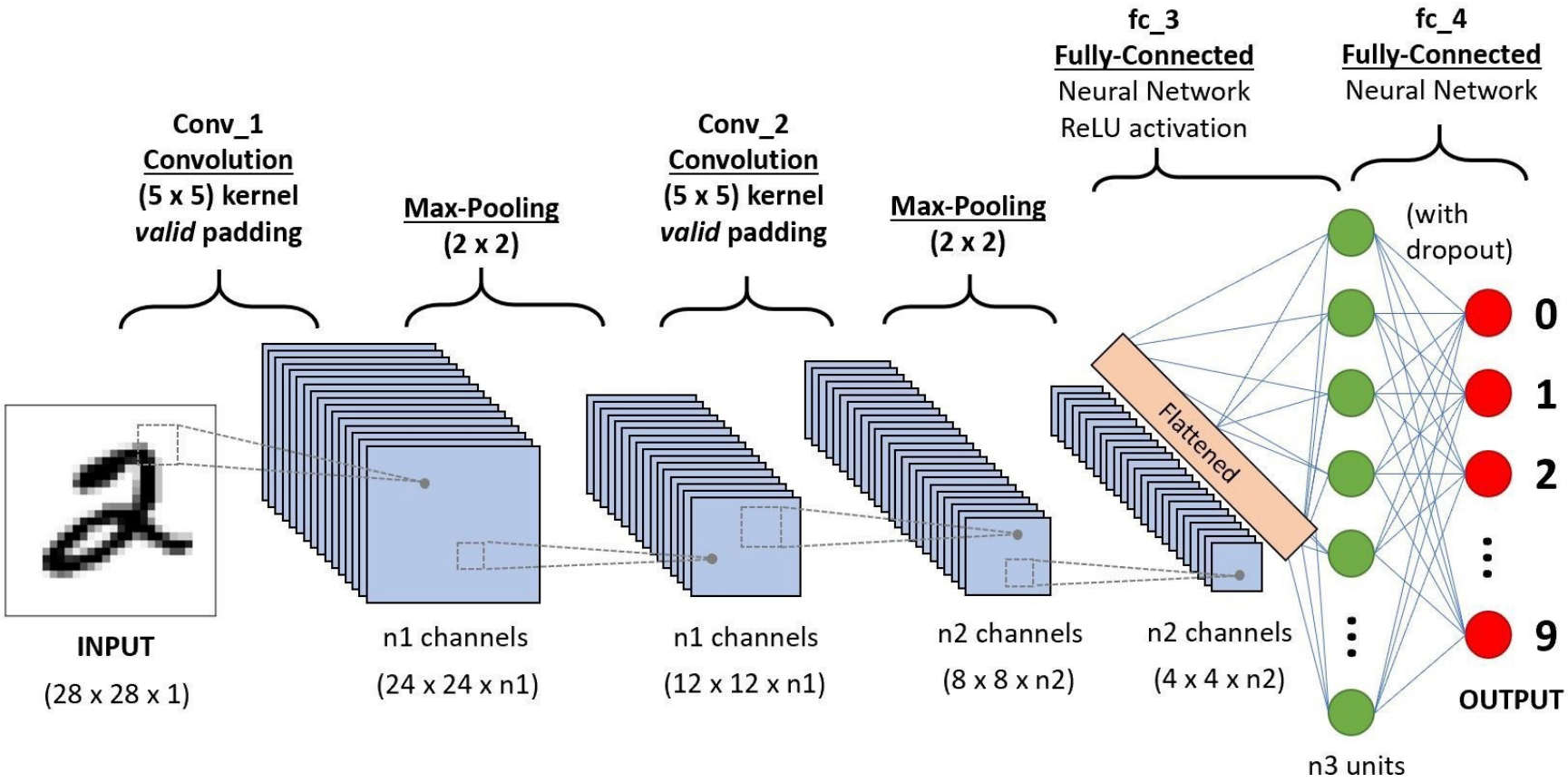
How to analysis raw audio data?

CNN

Convolutional neural networks

From Spotify to all of you

Convolutional neural networks are the same technology used in image recognition.



From Spotify to all of

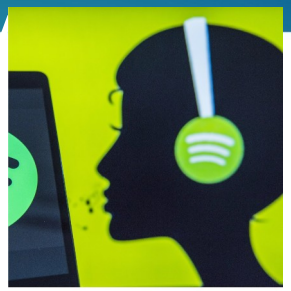


Photo by studioEAST/Getty Images

In Spotify's case, they've been modified for use on audio data instead of pixels.
Here's an example of a neural network architecture:

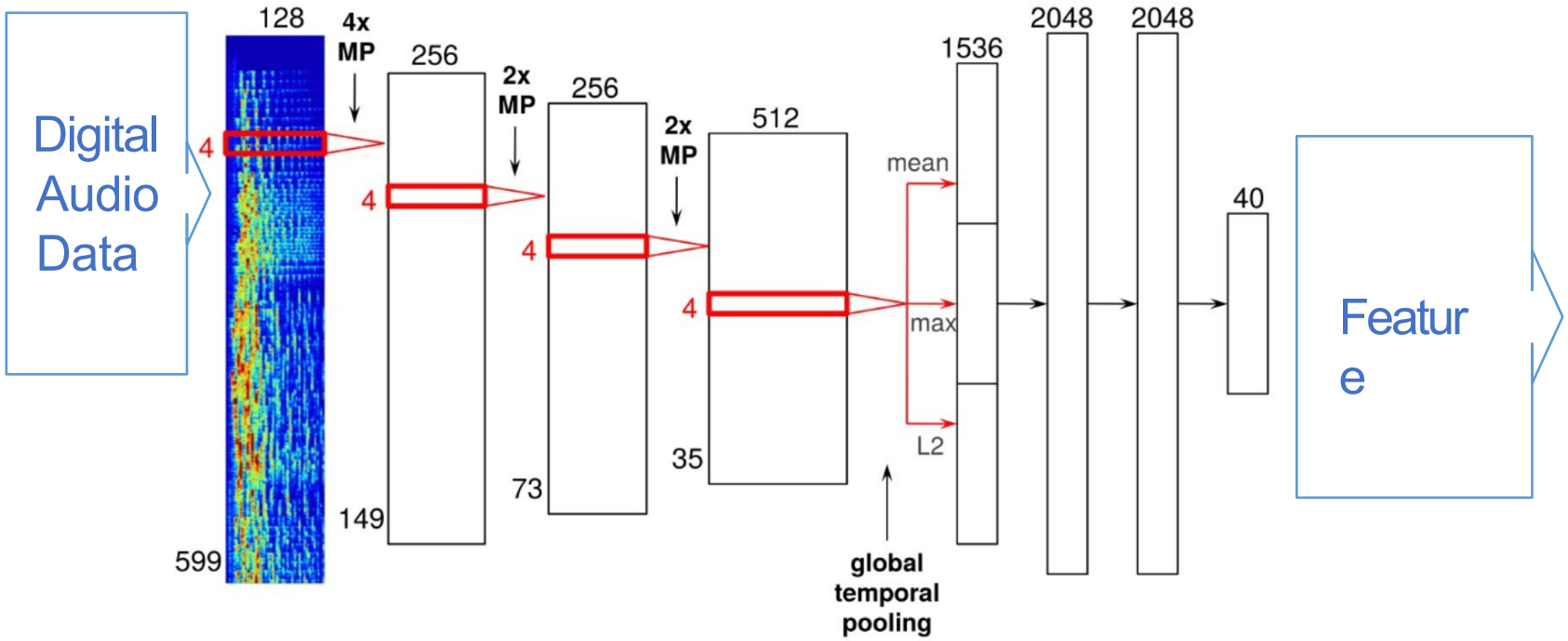
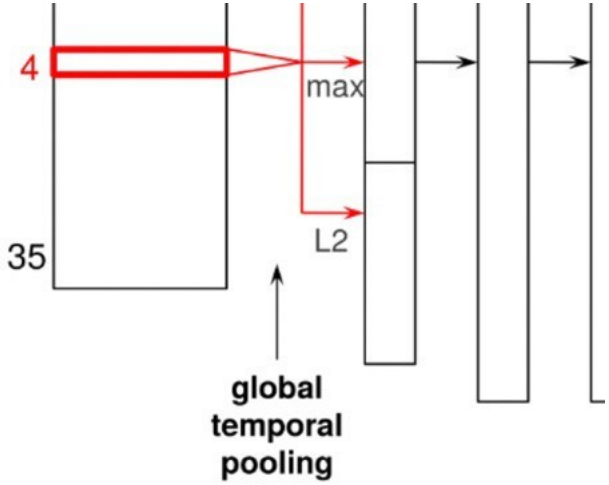


Image source: Recommending music on Spotify with deep learning, Sander Dieleman.

From Spotify to all of you



“global temporal pooling” layer

It pools across the **entire time axis**, effectively computing statistics of the learned features across the time of the song.

From Spotify to all of you

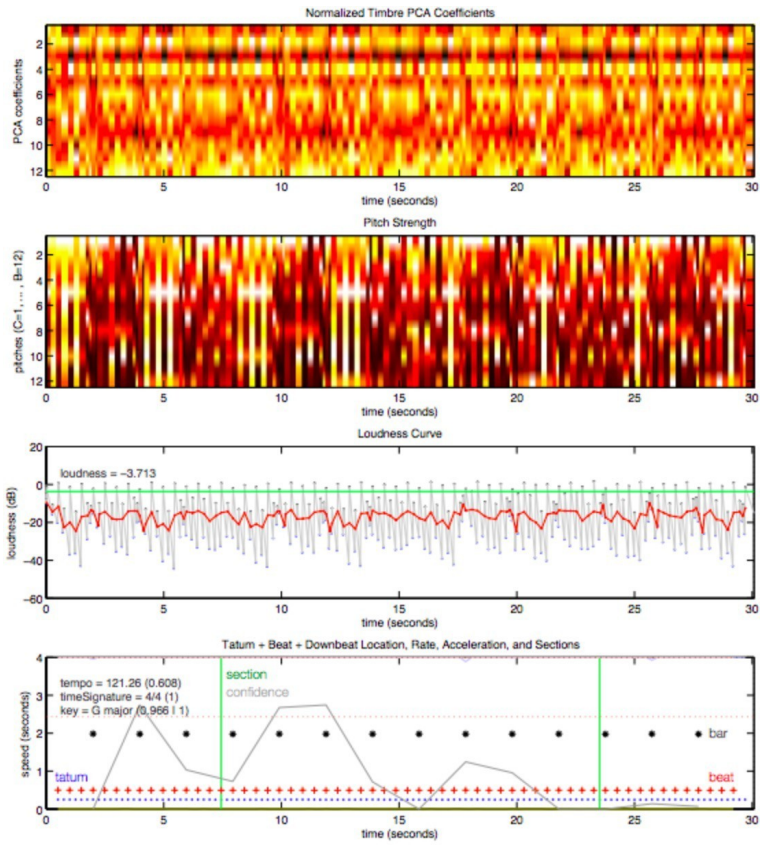
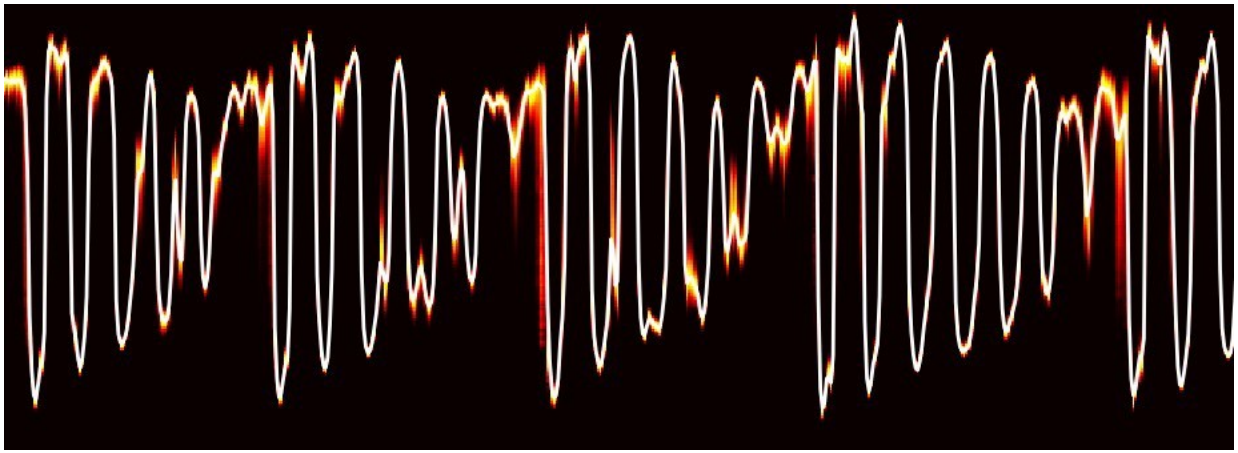


Image source: Tristan Jehan & David DesRoches, via The Echo Nest.

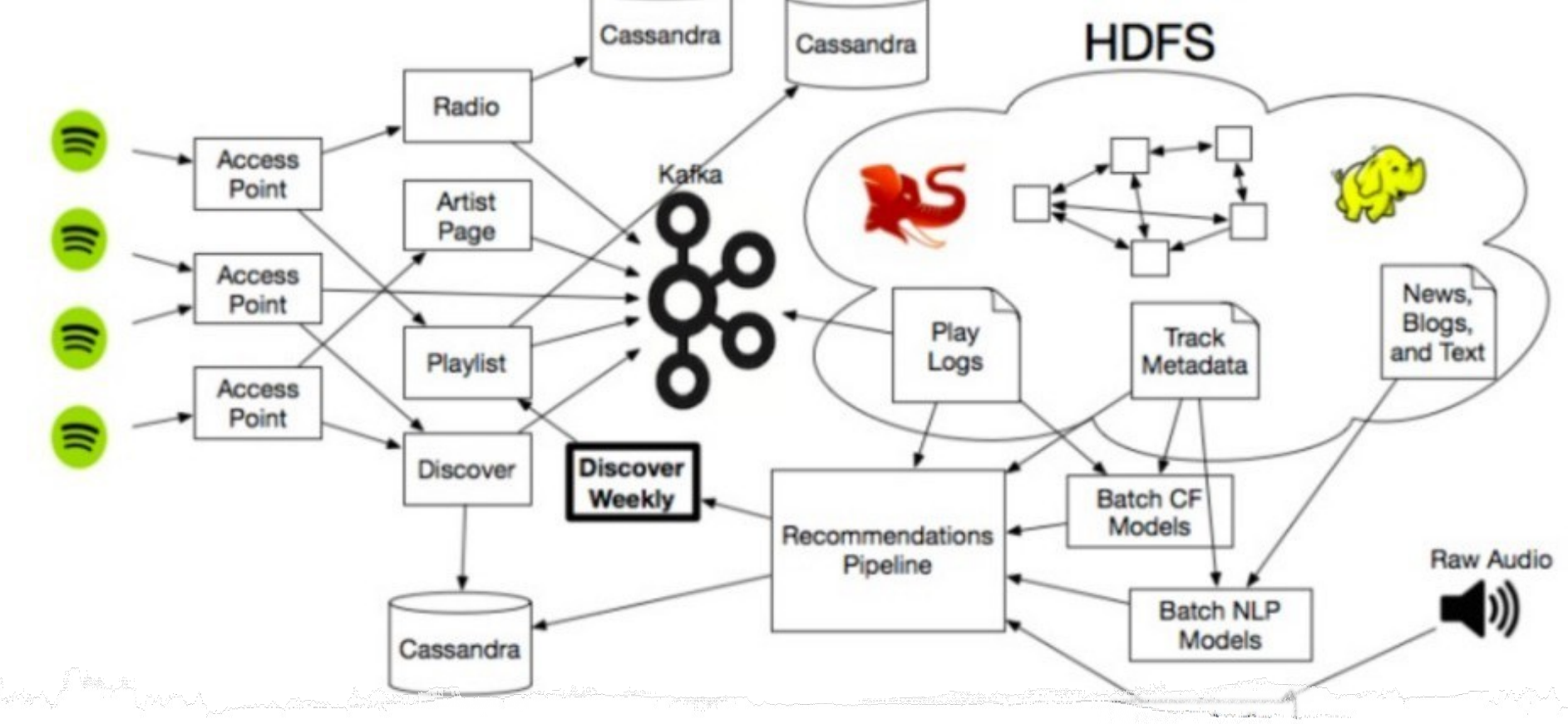
After processing, the neural network spits out an understanding of the song, including characteristics like –

- **Estimated time signature**
- **Key**
- **Mode**
- **Tempo**
- **Loudness**

From Spotify to all of you



Ultimately, this reading of the song's key characteristics allows Spotify to understand fundamental similarities between songs and therefore which users might enjoy them, based on their own listening history.



Conclusion

The recommendation models are all connected to Spotify's larger ecosystem, which includes –

- Giant amounts of data storage
- Lots of Hadoop clusters to scale recommendations
- Make these engines work on enormous matrices
- Endless online music articles
- Huge numbers of audio files

Q & A