An Ensemble Approach of Recurrent Neural Networks using Pre-Trained Embeddings for Playlist Completion

Diego Monti[1] Enrico Palumbo[2,3,1] Giuseppe Rizzo[2]
Pasquale Lisena[3] Raphaël Troncy[3] Michael Fell[4]
Elena Cabrio[4] Maurizio Morisio[1]

[1]Politecnico di Torino

[2]ISMB

[3]EURECOM

[4]Université Côte d'Azur

RecSys Challenge 2018

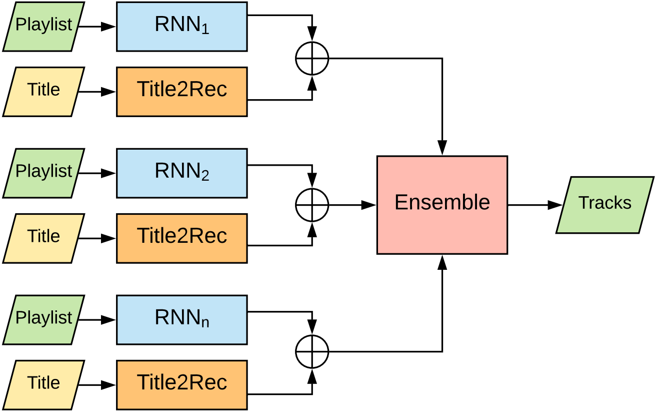
Introduction

We designed an ensemble strategy that combines:

- Playlist sequence embeddings (word2vec);
- Playlist title word embeddings (fastText);
- Features extracted from lyrics (creative track).

The implementation of our approach is publicly available at `https://github.com/D2KLab/recsys18_challenge`.

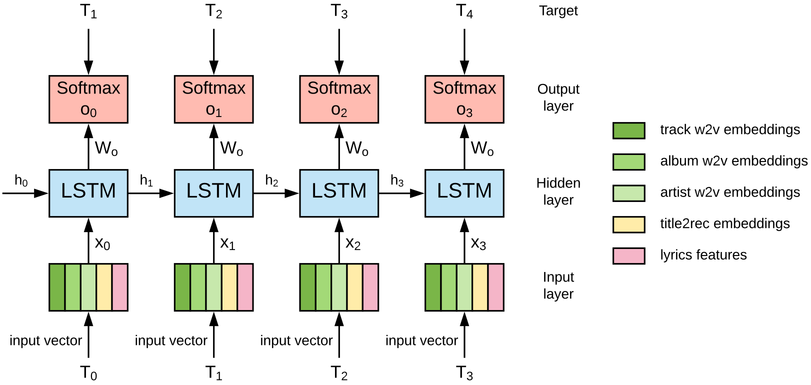
Ensemble


Playlist
RNN$_1$
Title
Title2Rec
Playlist
RNN$_2$
Title
Title2Rec
Playlist
RNN$_n$
Title
Title2Rec
Ensemble
Tracks

Recurrent Neural Networks

- RNNs can effectively model sequential data.
- For example, a typical application is language modeling.
- We trained the network to predict the next track in a playlist.
- We use as input a representation of the track, the album, the artist, the playlist title and, possibly, the lyrics features.

Input Vectors



Track, Album and Artist Embeddings

- We created three different word2vec models.
- They represent the co-occurrence patterns of tracks, albums and artists in the training playlists.

Imagine (track)

- Yesterday
- Let It Be
- Blackbird
- Stand By Me
- Eleanor Rigby

Imagine (album)

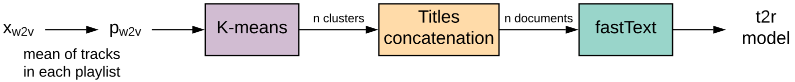
- Let It Be
- Help!
- Rock 'N' Roll
- The Beatles
- Rubber Soul

John Lennon (artist)

- Paul McCartney
- Jeff Buckley
- Elton John
- George Harrison
- The Beatles

Titles Embeddings

- The title of a playlist may contain interesting information.
- Playlists with similar titles may include similar tracks.



Example of a Cluster

yy :) christmas litmas guardians christmas christmas holiday christmas christmas the good stuff xmas himym christmas pop xmas country happy holidays holidays christmas christmas hits 25 just cause stay christmas tis the season 🎄 christmas 🎄 christmas oldbutgold christmas christmas vibes christmas strong christmas winter wonderland christmas time december 15 oldies work in progress christmas christmas playlist christmas music josh 🎄 christmas blah christmas & chill depression secret christmas christmas & chill christmas love :) christmas elite :) christmas special songs christmas good vibes christmas christmas songs christmas christmas christmas favorites christmas christmas 2016 🎄 christmas last christmas christmas all my friends christmas christmas !! chirstmas the weeknd christmas 2015 christmas christmas lyrical party music wake up happy vibes 🎄 christmas calm country winter christmas pop christmas af ❄️ christmas feel good :)) christmas christmas af christmas jams moana christmas merry christmas! christmas playlist christmas christmas silly love songs christmas 🎄 christmas christmas music christmas christmas music 🎄 christmas x-mas christmas bops christmas beachin' dance jamz christmas new wave its christmas christmas 🎄 christmas indie 2 christmas 1980 christmas jams christmas 2015 sunrise christmas playlist christmas jams 🎅 december '15 christmas!!

Lyrics Features

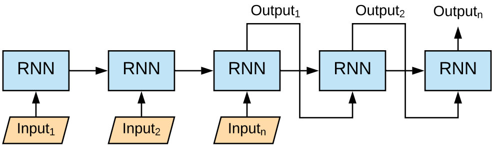
- We obtained and analyzed the lyrics of 416k tracks from the WASABI corpus (`https://wasabi.i3s.unice.fr`).
- The linked ones are about the 20% of the tracks in the MPD, but they cover the 53% of all tracks occurrences.
- We decided to compute lyrics features only on English texts, resulting in a final set of 367k tracks.

Lyrics Features

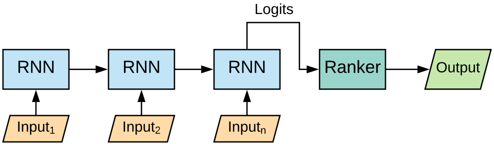
We extracted the following features:

- Vocabulary (type-token ratio)
- Style (line lengths, POS tags, rhymes, echoisms)
- Semantics (LDA model of 60 topics)
- Orientation (fraction of past tense verb forms)
- Emotion (subjectivity, polarity, arousal, valence)
- Structure (line and paragraph lengths)
- Deterministic (structure, vocabulary, style)
- Fuzzy (orientation, emotions, POS tags)

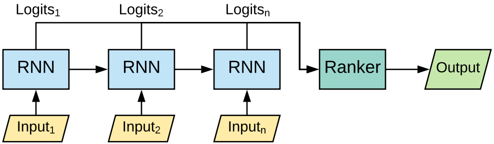
Generating Predictions



(a) do_sample



(b) do_rank



(c) do_summed_rank

Dataset

- We splitted the MPD in a training, a validation, and a test set, that contain the 98%, 1%, and 1% of the playlists.
- Our validation and test set include playlists with the same characteristics of the official challenge set.
- We also created a smaller dataset with 100k playlists in the training set and 1k playlists in the validation and test set.

RNN Optimization

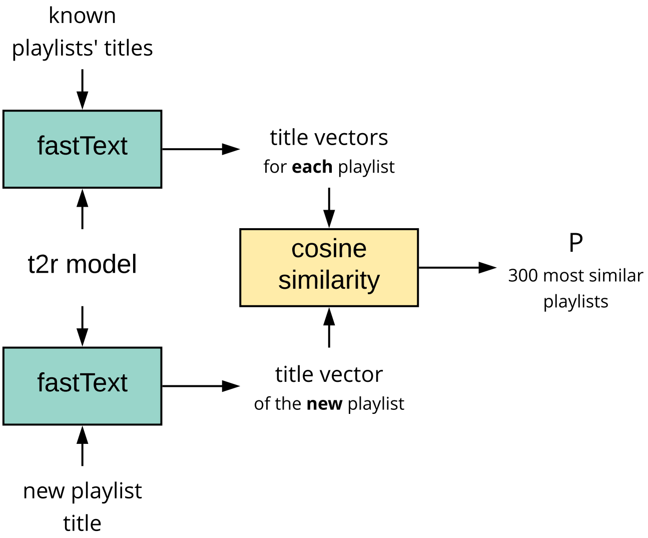
We executed a grid search on the down-sampled version of the dataset considering the following hyper-parameters:

- Optimizer: $\{Gradient, \ RMSProp, \ ADAM\}$
- Learning rate: $\{1, \ 0.5, \ 0.1, \ 0.01\}$
- Number of steps: $\{10, \ 20\}$
- Hidden layer size: $\{50, \ 100\}$

RNN Optimization

Optimizer	L.R.	Steps	Hidden	Perplexity	Hours	R-Prec.
ADAM	1	20	100	1357.04	3:29	0.1739
ADAM	1	10	100	1482.86	3:39	0.1742
Gradient	1	10	100	1693.96	3:32	0.1566
ADAM	1	10	50	1716.92	2:30	0.1745
Gradient	1	10	50	2005.54	2:25	0.1543

Title2Rec



known
playlists' titles
fastText
title vectors
for **each** playlist
t2r model
cosine
similarity
P
300 most similar
playlists
fastText
title vector
of the **new** playlist
new playlist
title

Experimental Results

Approach	Optimizer	Epoch	R-Prec.	NDCG	Click
Most Popular	-	-	0.0373	0.0959	18.529
Title2Rec	-	-	0.0837	0.1260	12.007
Word2Rec	-	-	0.0963	0.1444	8.4322
RNN 300	Gradient	1	0.1417	0.1621	4.1902
RNN 300	Gradient	2	0.1500	0.1656	3.9433
RNN 300	ADAM	1	0.1557	0.1702	3.9213
RNN 300	ADAM	2	0.1457	0.1672	4.4224
RNN 400	ADAM	1	0.1572	0.1708	3.9340
RNN 400	ADAM	2	0.1520	0.1694	4.1307
RNN Emotion	ADAM	1	0.1556	0.1702	4.0101
RNN Emotion	ADAM	2	0.1500	0.1680	4.3594
RNN Fuzzy	ADAM	1	0.1555	0.1698	3.9950
RNN Fuzzy	ADAM	2	0.1503	0.1683	4.3456

Experimental Results

- **Main track:** RNN 300 (Gradient; Epoch 1 and 2), RNN 300 (ADAM; Epoch 1 and 2), and RNN 400 (Epoch 1 and 2).
- **Creative track:** RNN 300 (Gradient; Epoch 1 and 2), RNN 300 (ADAM; Epoch 1), RNN 400 (Epoch 1 and 2), RNN Emotion (Epoch 1 and 2), and RNN Fuzzy (Epoch 1 and 2).

Track	R-Precision		NDCG		Click	
	MPD	Official	MPD	Official	MPD	Official
Main	0.1611	0.1808	0.1710	0.3252	3.6349	3.0861
Creative	0.1634	0.1852	0.1717	0.3334	3.5964	3.0258

Conclusion

- The generation strategy has a great impact on the results.
- The ensemble method has granted a sensible increment in performance because of the complementarity of the runs.
- The computing time has been a crucial experimental factor because the training time of each epoch was about three days.

Thank you!

POLITECNICO DI TORINO 1859 1906

ISMB
Istituto Superiore Mario Boella

EURECOM
Sophia Antipolis

UNIVERSITÉ CÔTE D'AZUR