ARS: Attention-based Recommender Systems

Mohammad Iman Mousaei
Shahid Beheshti University
98222099
m.mousaei@sbu.ac.ir

Amirhossein Asgharivahed
Shahid Beheshti University
98222005
a.asgharivahed@sbu.ac.ir

***Abstract*—Most of recommender systems are based on clustering data through various methods.We took a new approach based on attention, which only uses the playlist as input. Then we compared it to some of classic clustering methods.**

***Index Terms*—recommender system, attention**

I. Introduction

Since Transformers [**?**] were introduced, it has been replacing some of the best models in AI. So we used Attention to design a new architecture for unsupervised learning.

II. Cleaning Data

The first step of every data science research is data cleaning and the first step of every data cleaning is to get rid of(or fill out) the missing values of columns. So we start with dropping the unnecessary columns. Now that we know we don't have any missing values we can check to see if the data is normally distributed and lets be honest if the data was clean and normally distributed, it wouldn't be much of a challenge now would it?
For categorical columns that are important to the model(e.g. genre) we replaced them with category code.
Then we should scale the data to range [0,1] and then make the data standard. For achieving standard normal we can do the following formula for every column:

$$Col' = \frac{Col - Avg(Col)}{\sqrt{variance(Col)}}$$

After all these steps we plot our data so we can visualize it better. And after the cleaning is done we start with the clustering.
You can see the effect of normalizing and standardizing in Figure 1&2.

A. Why?

Research shows that almost every AI model performs better on Gaussian distribution dataset. Also null columns have negative effect on model.
But why normalize? Not only models has better performance on them, but also ARS's output has passed through a Sigmoid and is in range [0,1] ergo we want to match them.

III. Classic Clustering Methods

Describing classic methods we used for reference.

A. Clustering Methods

K-Means, Hierarchical-Clustering and DB-scan.

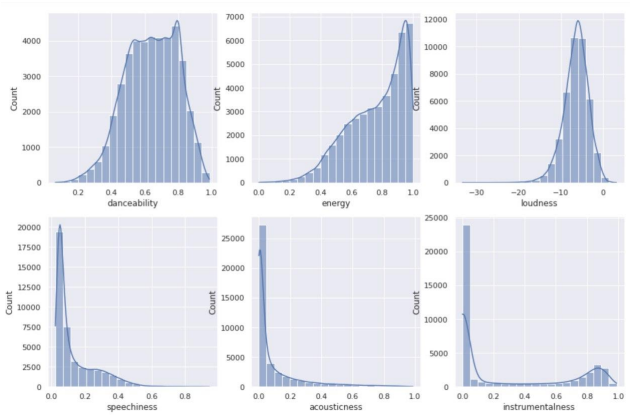

Fig. 1. Before Standardize

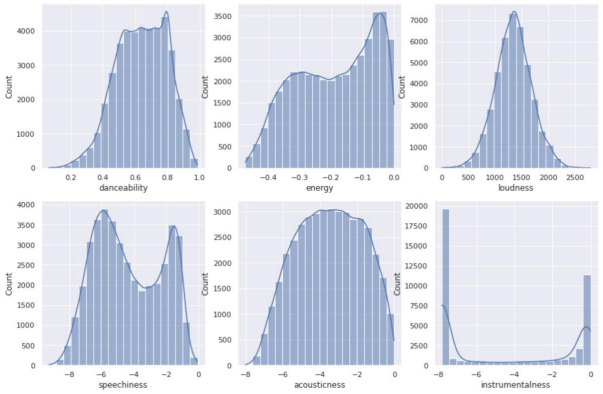

Fig. 2. After Standardize

B. Distance Measure

Suppose list L of size S consists of songs in cluster C that were is the input playlist, So for a point P in a cluster C:

$$DistanceMeasure = \frac{\sum_i^S EuclideanDistance(P, L_i)}{S}$$

This is clearly better than randomly selecting. And if you want my opinion, it's one of the best. To understand it better, suppose every cluster shows a band/genre. If someone likes a song from that particular band/genre, chances are he likes a song just next to it. So practically his taste in that band/genre is more around that.

C. Recommendation

In clustering methods, how you choose to recommend is as important as the method itself. One way is to randomly select cluster-mates. But here we use k minimum values of our DistanceMeasure(explained above).

IV. Model Architecture

The approach of unsupervised learning for a recommender system calls for a new architecture. As we talked about it before, we are going to use attention for the design of our model. The encoder part gets vectors as input and it starts the process of learning with the attention mechanism .

A. Attention

The attention function or block was originally designed to keep track of long source sentence in neural machine translation but here we use it as a method for learning and representing meaningful vectors for our song recommender model. So basically attention uses this formula :

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

B. ARS design

In ARS structure we used the encoder from transformer, but the decoder had to be supervised. So we changed the decoder a little to achieve Unsupervied training.

C. How does it work?

As we can see in Fig.3, first we send out playlist as input. Then the encoder DotProducts input with arbitraty/random weights(i.e.

$$W_k, W_v, W_q$$

) to get Key, Value and Query and then inputs them in Attention function. Then it adds the output with the single input. Then uses it to train an unsupervised simple neural network.
For better performance we can use multiple identical encoders and decoders(with different weights) and get their average.
At last we used Linear Optimization on the last output and sent it through a Sigmoid function(to normalize, like input dataset). The output of the Sigmoid is what we call Centroid of the hypothetical cluster for the input playlist. So with our distance measure we recommend k nearest songs to this point to the user.

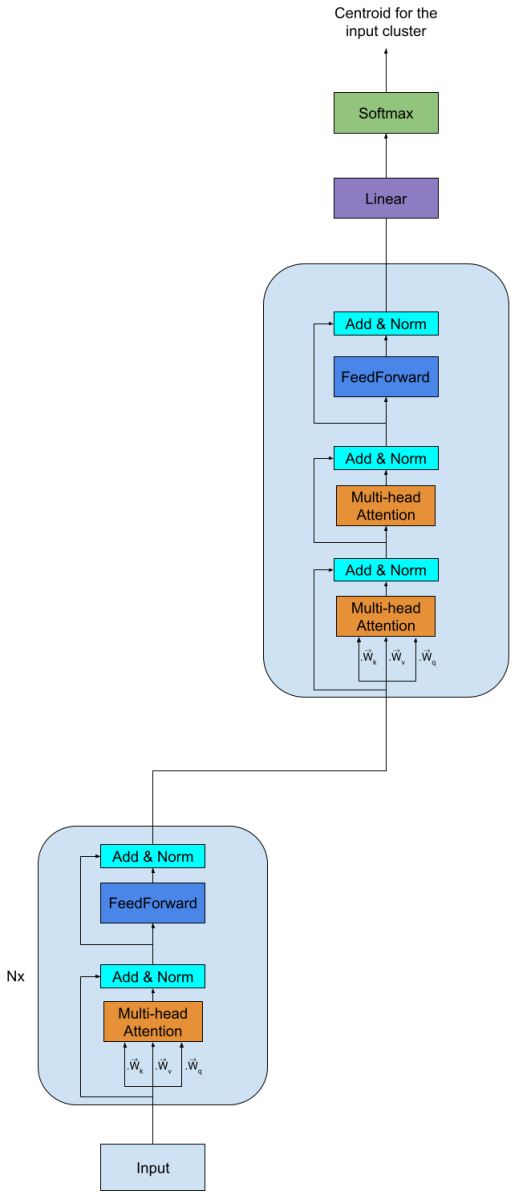

Fig. 3. ARS Architecture

Acknowledgment

Special Thanks to Mohammadreza Mousaei, Hesam Damghani, Saeedreza Kheradpisheh and Hanie Jalili for helping us through the process of writing this paper. Note: Because lack of time, implementaion of ASR wasnt possible for us, but other models were fully implemented.

References

[1] Ashish Vaswani, "Attention Is All You Need".
[2] Zhigang Dai, "UP-DETR".
[3] T. Soni Madhulatha, "An Overview On Clustering Methods".
[4] Mathilde Caron, "Deep Clustering for Unsupervised Learning of Visual Features".
[5] Google Brain, "Tensor2Tensor for Neural Machine Translation".
[6] https://github.com/datamadness/Automatic-skewness-transformation-for-Pandas-DataFrame