MUSIC MOOD CLASSIFICATION

NHÓM 24

Nguyễn Hoàng Tân

Nguyễn Minh Trí

Đỗ Trí Nhân

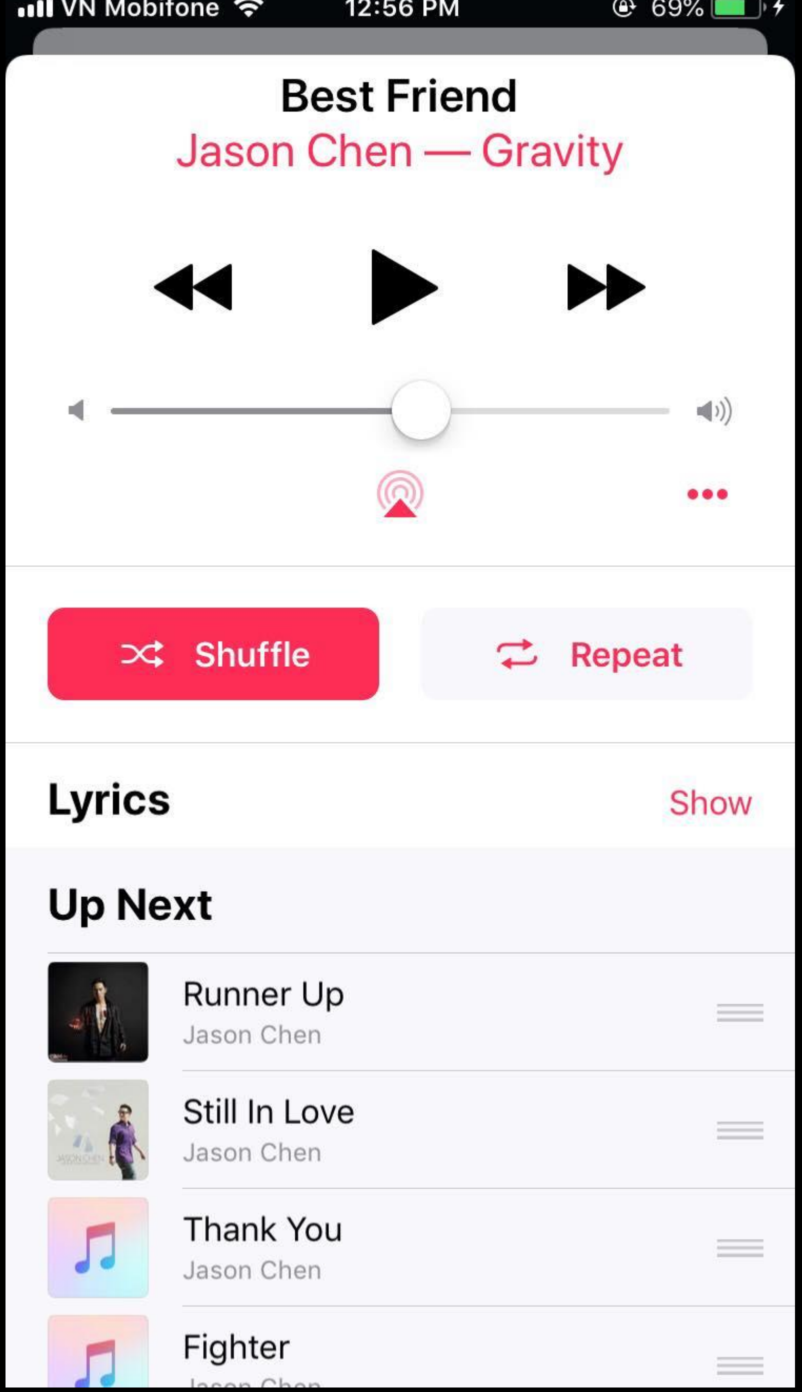
Huỳnh Thanh Vy

Motivation

- Since the shuffle function doesn’t suggest her the similar song for our mood.
- A good chance for all of us to study about machine learning.
- Similar projects is available.

Objective

- Making an AI function that can suggest a similar song base on the mood of the first song user selected.
- Apply this to shuffle function, which can be used offline.
- Break the objective into small objectives



Break the objective into small objectives

Define mood of each song

Classify song base on its mood

Recommendation function base on mood.

Heuristic

- **Step 1:** Find the similar project that have a dataset.
- **Step 2:** Reimplement the project to get the result as the first result
- **Step 3:** Classify music by our own method and get the second result
- **Step 4:** Compare the first result to the second result

Dataset

- Feature analysis and metadata from **Million Song Dataset** (The Echo Nest)
- 1200 songs, 2 Mood, removed all non-English songs
  labeled as **happy and sad** (by Sebastian Raschka University of Wisconsin-Madison)
- **Lyric:** Script to download http://lyrics.wikia.com
- **Audio:** Not include any audio (only the derived features).

Train: 1000 song

Train: 200 song

Methodology

Author's method

- Built upon on a Naive Bayes classifier

Our method

- Use both audio features and lyrics to predict
- Try manual
- Use Deep learning

Compare Accuracy

Lyric

Author Method

-Couter Vector
-TF-IDF
-Ngram (BoW)

→

-Naïve Bayes
-Linear Classifier
-SVM
-Bagging
-Boosting Model

```
->
NB, Count Vectors:  0.725
NB, WordLevel TF-IDF:  0.54
NB, N-Gram Vectors:  0.54
NB, CharLevel Vectors: 0.54
->
LR, Count Vectors:  0.645
LR, WordLevel TF-IDF:  0.655
LR, N-Gram Vectors:  0.615
LR, CharLevel Vectors:  0.67
->
SVM, N-Gram Vectors:  0.475
SVM, CharLevel Vectors:  0.475
->
RF, Count Vectors:  0.545
RF, WordLevel TF-IDF:  0.535
->
Xgb, Count Vectors:  0.645
Xgb, WordLevel TF-IDF:  0.61
Xgb, CharLevel Vectors:  0.695
```

PRE 88.89%
ACC 72.5%
Reimplement base on paper

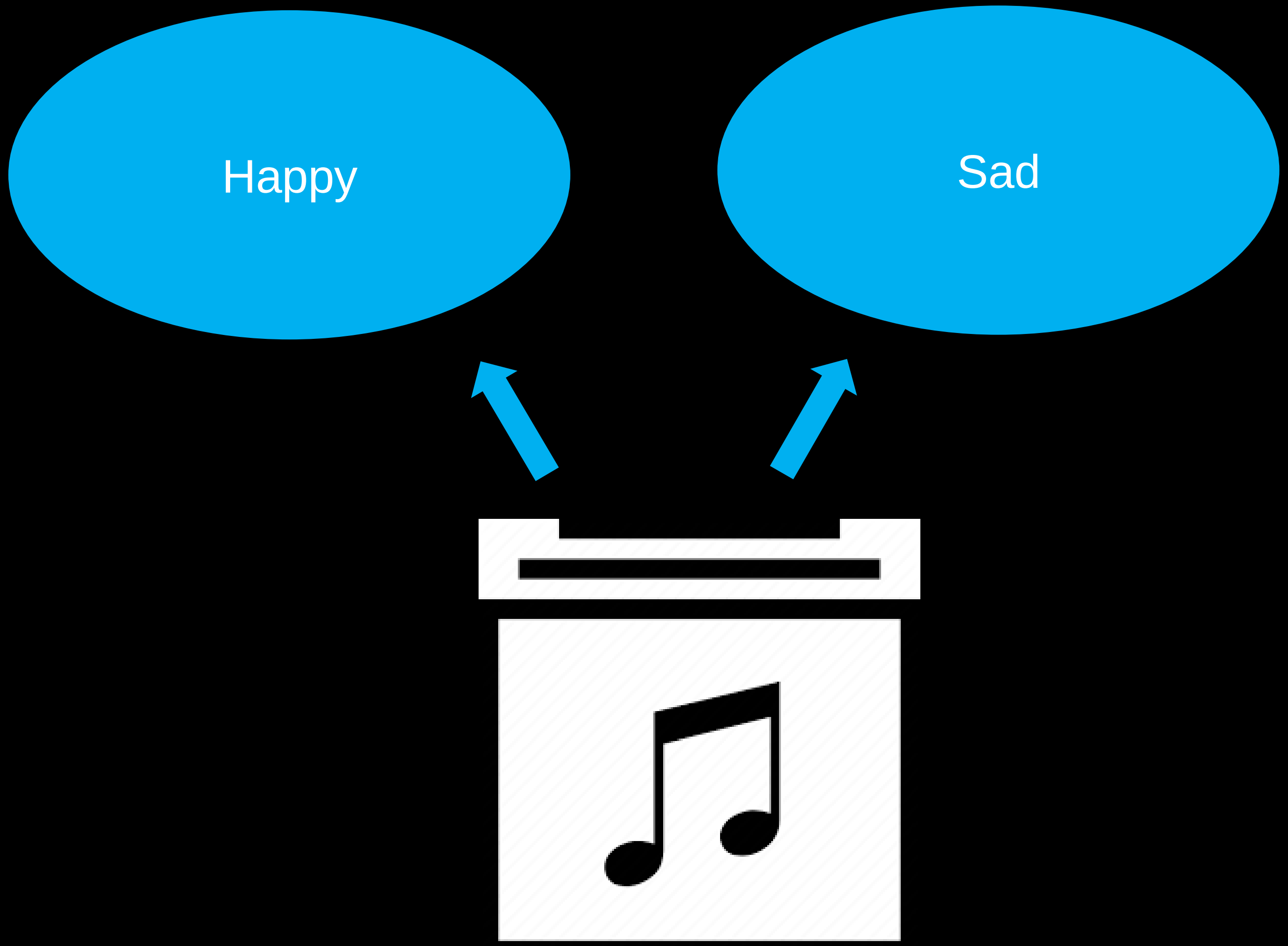
Manual: Lexicon-based

-Build Trie
-Add Song to Trie
-Check Test Song

=>48%

-Stopword
-Balance Dataset

=>55%



Deep Learning:

Preprocessing:

Remove Stopword punctuations

Lemmatization

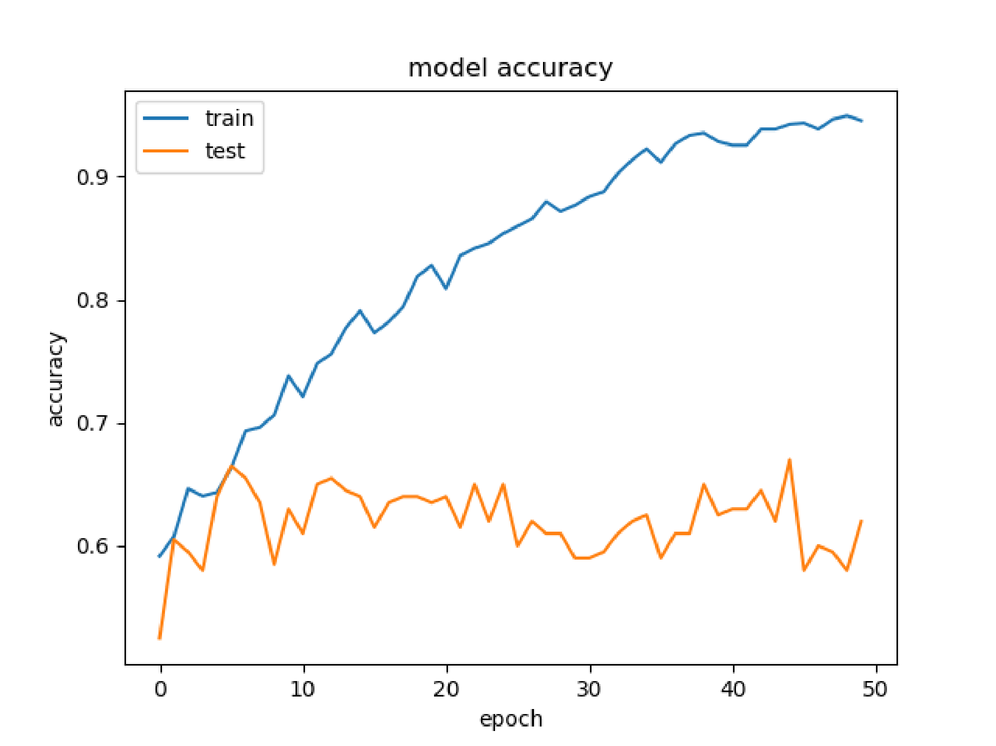
Stemming !!!

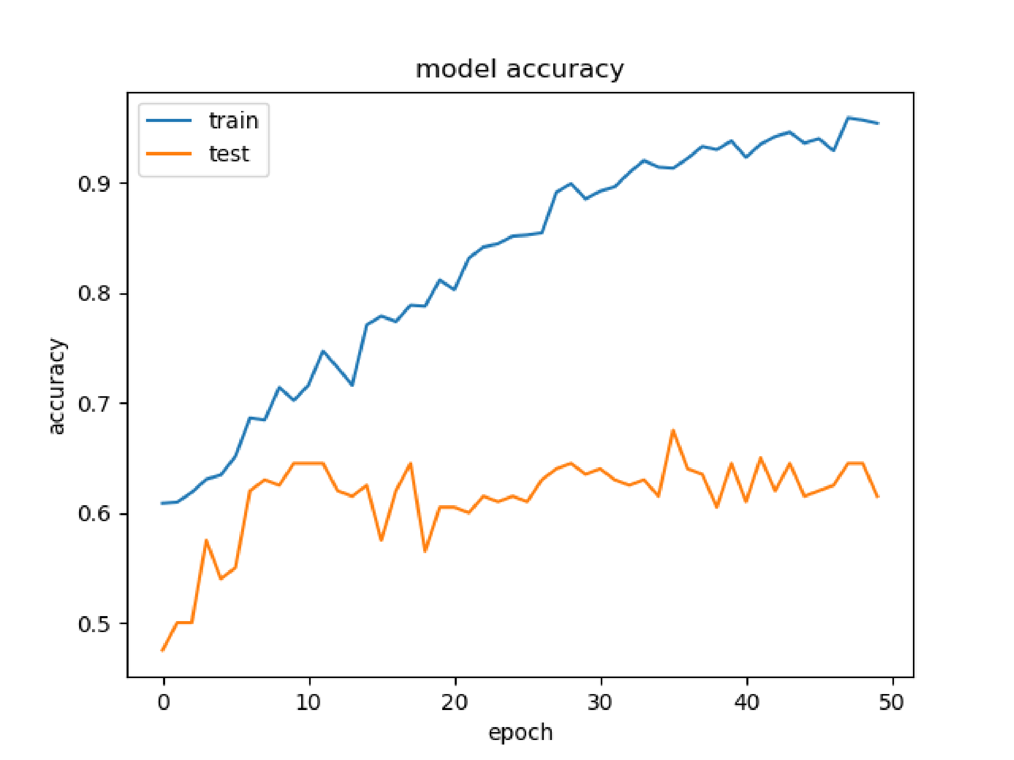
Exaggerated word shortening

Feature

Word Embedding

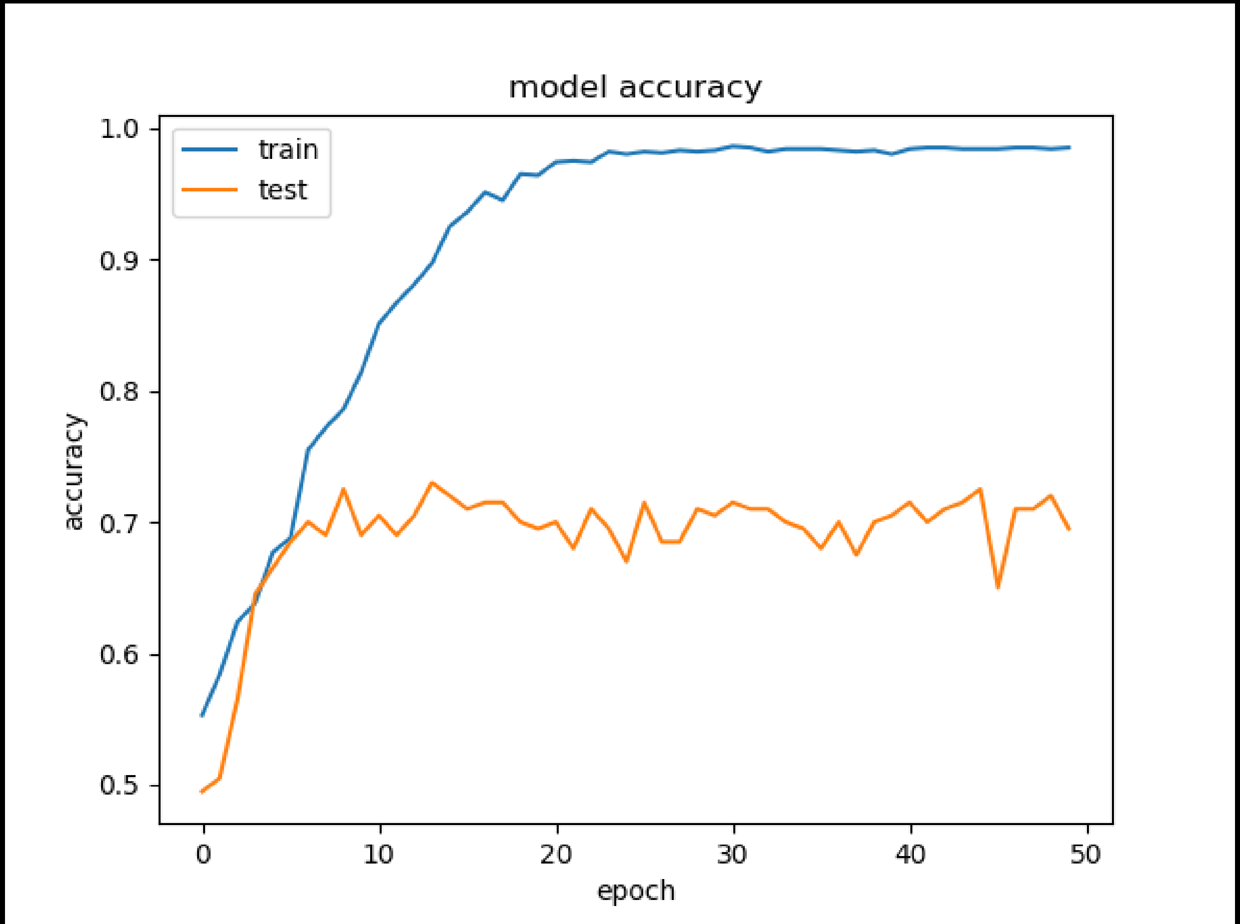
Model: **RNN**





LSTM	**GRU**
64%	**68%**

Model: **CNN**



71.5% Stopword Lema **74.5%** **(best case)**

Audio

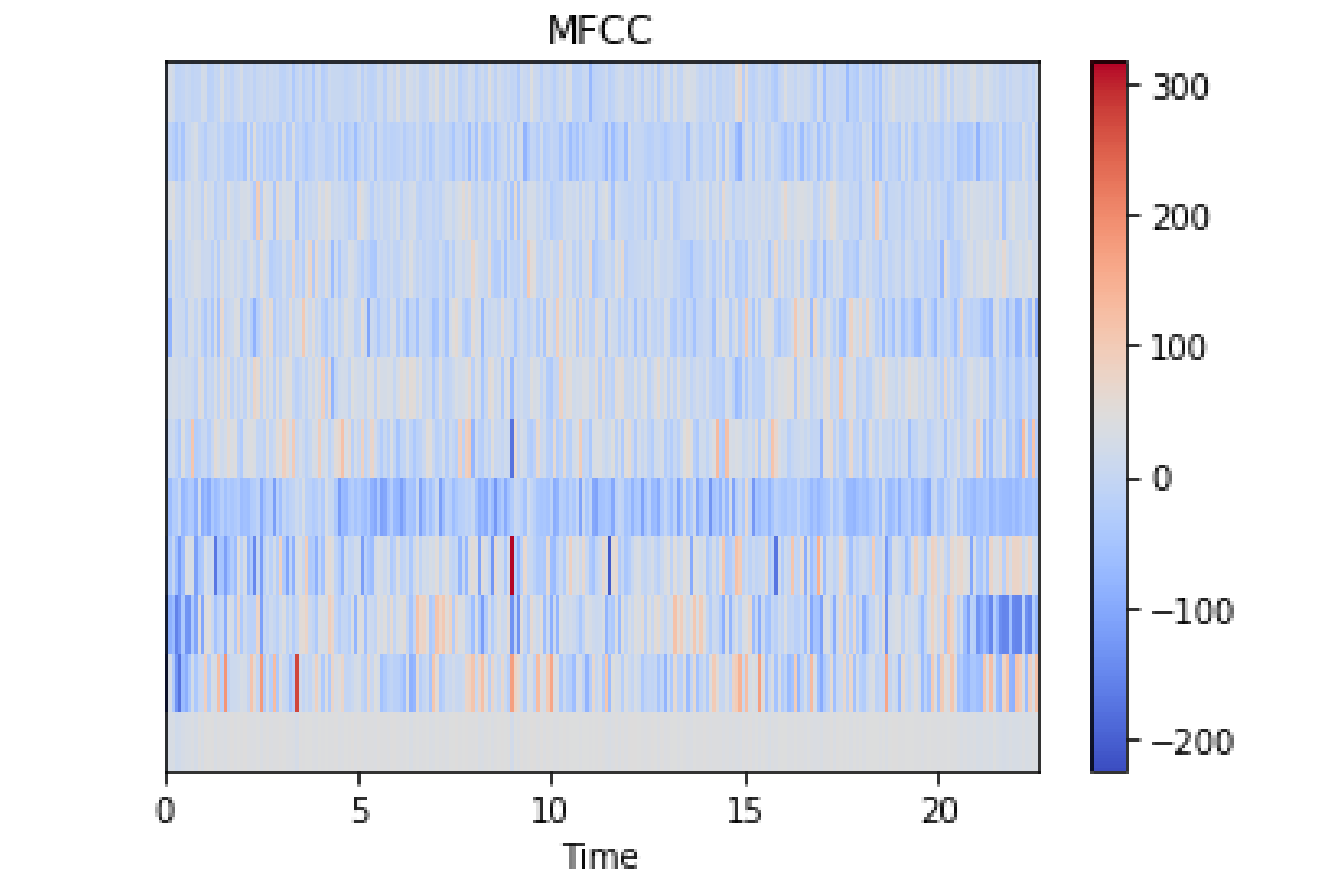
MFCC features

- The shape of the vocal tract including tongue, teeth etc determines what sound comes out
- The shape of the vocal tract manifests itself in the envelope of the short time power spectrum, and the job of MFCCs is to accurately represent this envelope

MFCC features retrive from dataset

segments_timbre: shape = (935, 12)

MFCC-like features for each segment



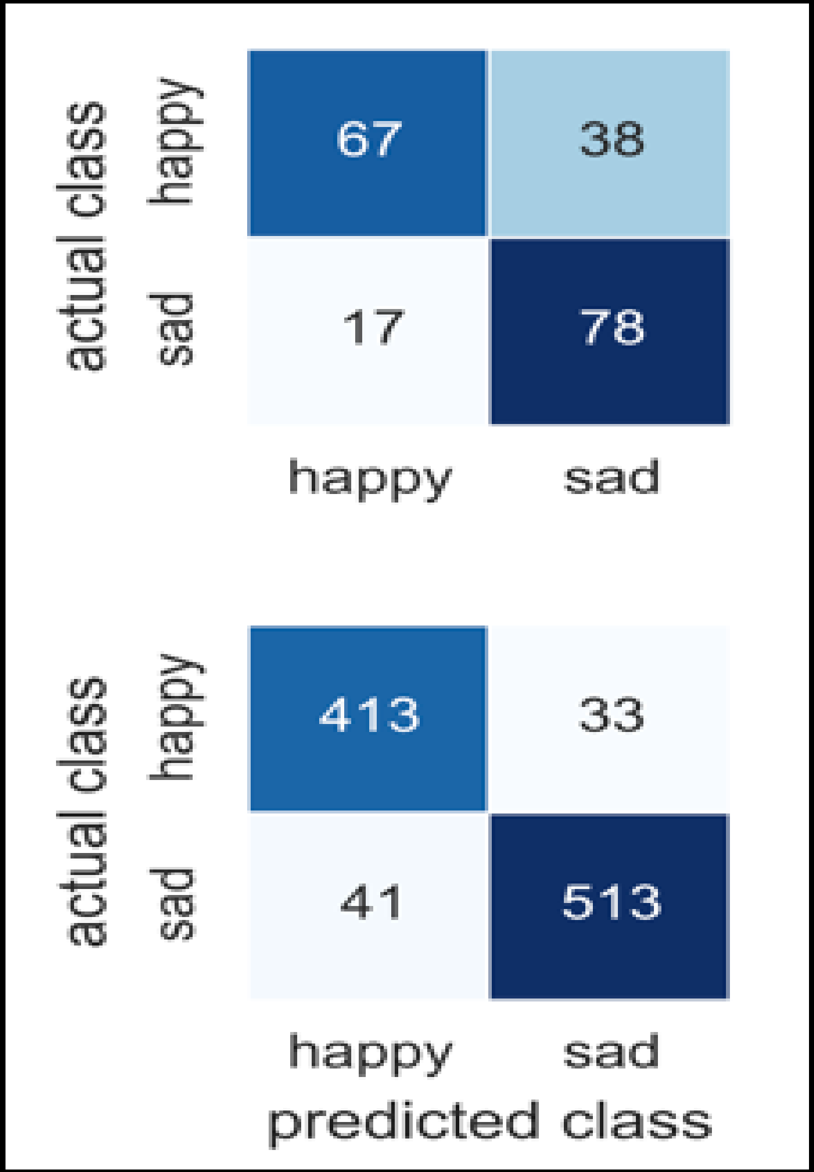
Training model

Input_1	Input	(None, 814, 12)
	Output	(None, 814, 12)
Time_distributed_1: Dense_1	Input	(None, 814, 12)
	Output	(None, 814, 64)
Time_distributed_2: Dense_2	Input	(None, 814, 64)
	Output	(None, 814, 64)
LSTM	Input	(None, 814, 64)
	Output	(None, 256)
Dense_3	Input	(None, 256)
	Output	(None, 1)

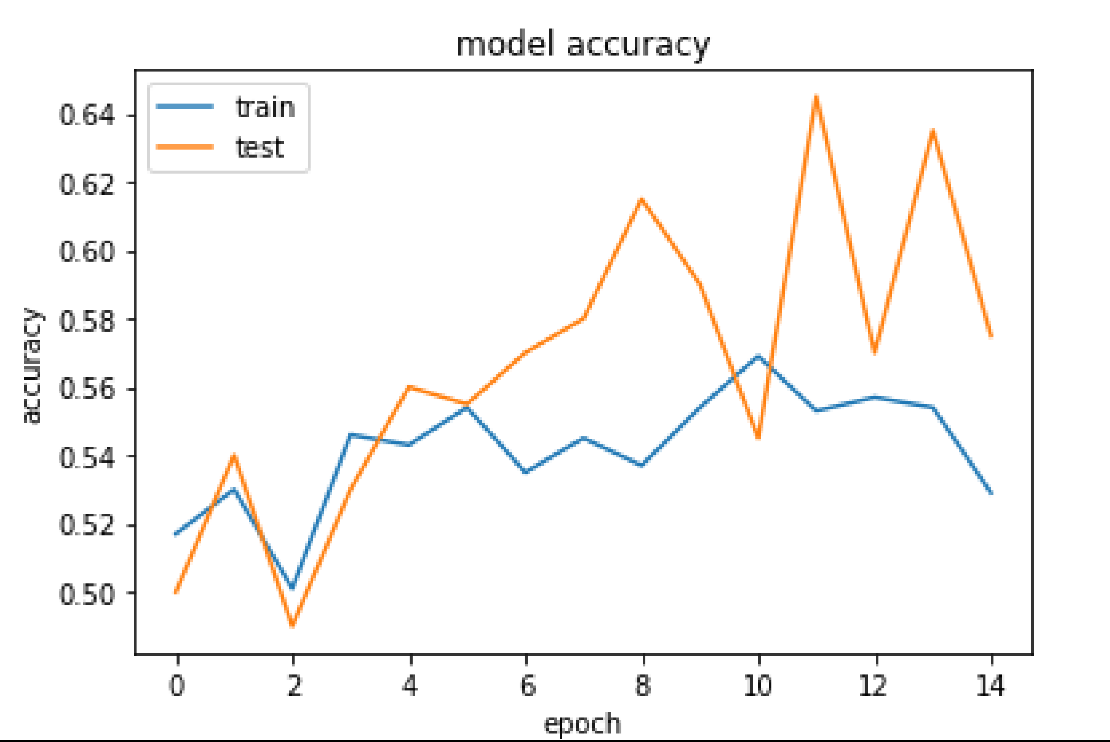
Compare the results

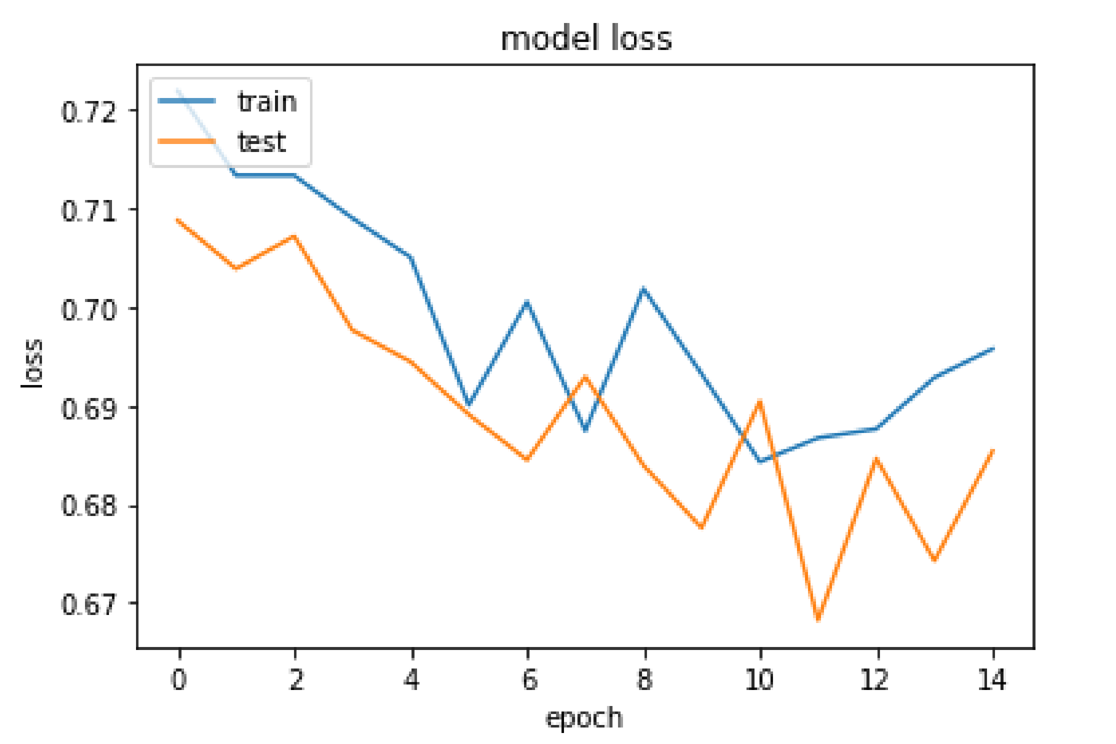
The existed method

- Hand label: 67+38 happy songs
- Model label: 67 songs, 38 songs were labeled wrongly



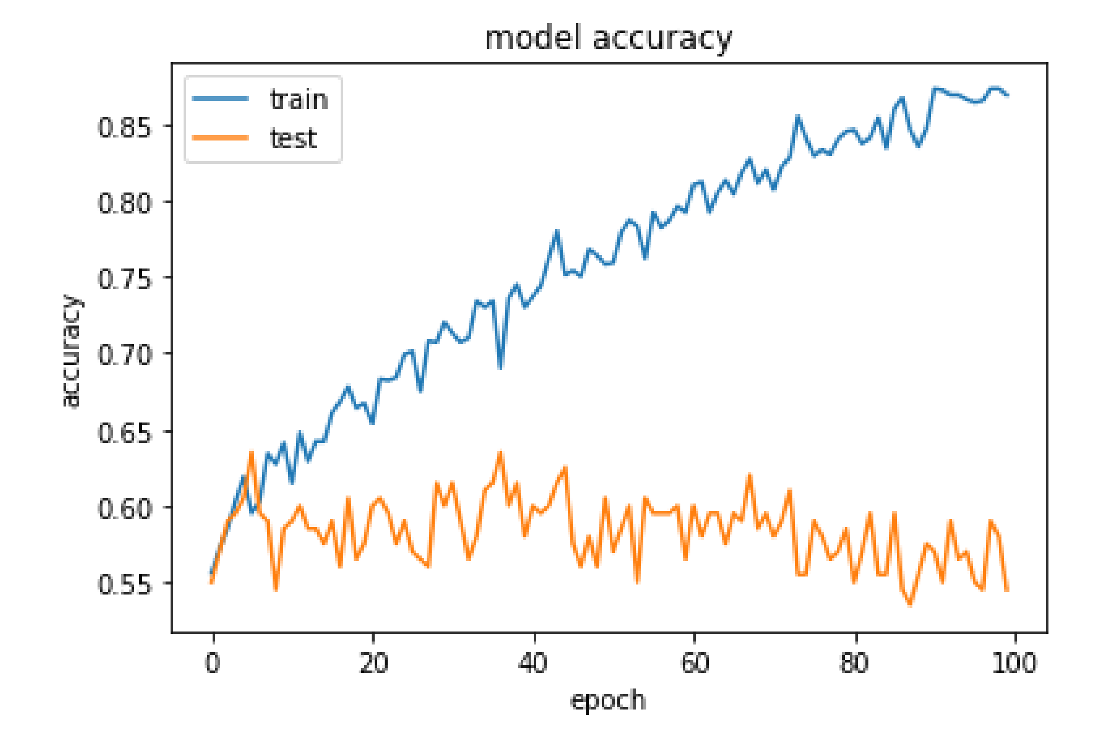
Our method
Initial result





LSTM networks

Our method
Initial result



LSTM networks

Let’s try it in real life

2 song to let the model predict

Oasis – Morning Glory 49% sad

Xi Guan Liang Ge Ren 80% sad

Future work

- Combine both audio features and lyrics
- Apply Treebank to RNTN (Recursive Neural Tensor Network) in Deep learning)
- Apply adjective annotation dataset
- Use different song’s mood grid (spotify and gracenote)
- Testing on large and more precise, mood dataset

References

- Dataset: https://github.com/rasbt/musicmood
- Paper: https://arxiv.org/abs/1611.00138
- Method deeplearning: https://www.analyticsvidhya.com

Thank You

NHÓM 4

Nguyễn Hoàng Tân

Nguyễn Minh Trí

Đỗ Trí Nhân

Huỳnh Thanh Vy