KY 1801

Music Reduction - Chord Identification

Li Shuhe(1155062051)

ZHANG Zhaochen(1155062190)

Content

- Overview
- System Architecture
- Experiment and Analysis
- Conclusion
- Future work

Overview - objective

- Objective
  - A software to recognize the common chords of score.
- Main tasks
  - Collect scores for training and testing
  - For each score
    - Split score into pieces
    - Predict the chord for every piece

Overview – what we have done last semester?

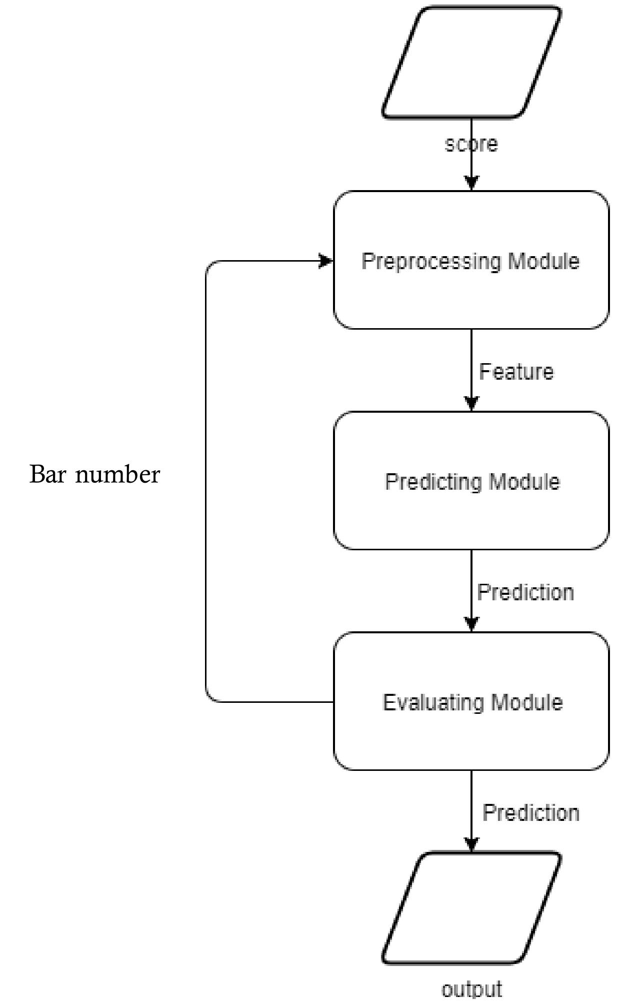
- Exercises to get familiar with musical knowledge
- Redesigned and implemented the Hidden Markov Model with key information as prior knowledge
- Applied Baum-Welch algorithm for HMM training

Overview – what we have done this semester?

- Collected data
- Modified Hidden Markov Model
- Designed and implemented Long Short-term memory model
- Implemented key identification algorithms
- Built chord identification system

System Architecture

- Preprocessing module
- Predicting module
- Evaluating module



Pre-Processing Module

- Key Identification
- Multiple Chords division
- Data Extraction

Key Identification

- Krumhansl-Schmuckler key-finding algorithm
- Implementation

Key Identification-Kumhansl-Schmuckler key-finding algorithm

- Key-finding algorithm
  - Calculate the total durations for each notes
  - Pair durations with profiles of major key and minor key
  - Calculate correlation coefficient for each key
  - The key with highest correlation coefficient is the most possible key

Key Identification-Kumhansl-Schmuckler key-finding algorithm

C	C#	D	D#	E	F	F#	G	G#	A	A#	B
432	231	0	405	12	316	4	126	612	0	191	1

total durations in "Toroko's Theme"

do	do#	re	re#	mi	fa	fa#	so	so#	la	la#	ti
6.35	2.23	3.48	2.33	4.38	4.09	2.52	5.19	2.39	3.66	2.29	2.88

Major profile

la	la#	ti	do	do#	re	re#	mi	fa	fa#	so	so#
6.33	2.68	3.52	5.38	2.60	3.53	2.54	4.75	3.98	2.69	3.34	3.17

Minor profile

Key Identification-Kumhansl-Schmuckler key-finding algorithm

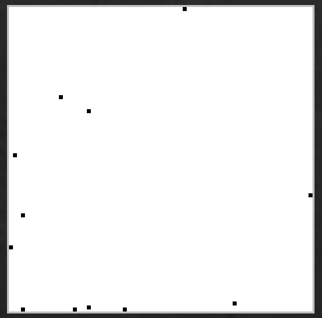
(do,C)	**(6.35,432)**
(do#,C#)	(2.23,231)
(re,D)	(3.48,0)
(re#,D#)	(2.33,405)
(mi, E)	(4.38,12)
(fa,F)	(4.09,316)
(fa#,F#)	(2.52,4)
(so,G)	(5.19,126)
(so#,G#)	(2.39,612)
(la,A)	(3.66,0)
(la#,A#)	(2.29,191)
(ti,B)	(2.88,1)

Pair data for C Major

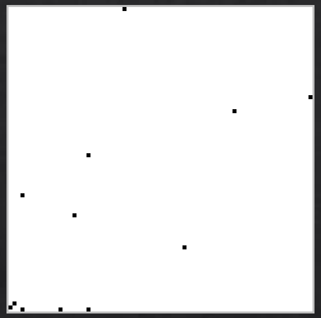
(do,C#)	**(6.35,231)**
(do#,D)	(2.23,0)
(re,D#)	(3.48,405)
(re#,E)	(2.33,12)
(mi, F)	(4.38,316)
(fa,F#)	(4.09,4)
(fa#,G)	(2.52,126)
(so,G#)	(5.19,612)
(so#,A)	(2.39,0)
(la,A#)	(3.66,191)
(la#,B)	(2.29,1)
(ti,C)	(2.88,432)

Pair data for C# Major

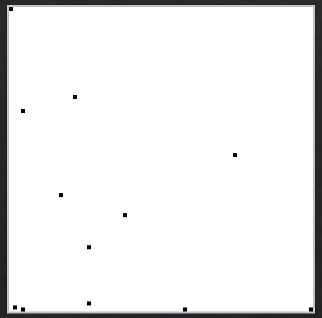
Key Identification-Kumhansl-Schmuckler key-finding algorithm



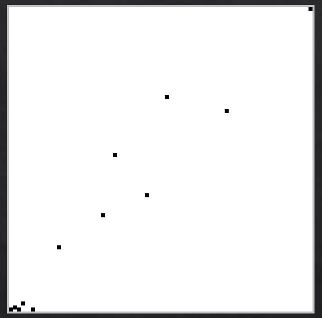
C# minor R=0.09



Cminor R=0.62



Dminor R=-0.31



G# major R=0.97

Key Identification - Implementation

- Divide the score into measures
- For measure occurs key change, combine the current measure and the next two measures as a key check unit
- Using key-finding algorithm to find the key of the key check unit

Key Identification - Result

Testing Score	Accuracy(three measure)
Von fremden Ländern und Menschen	70

Multiple Chords division

- Divide the measure into small time unit according to beat type.
- Check the possible chord for each unit.
- iterate until no possible chords or the unit cannot be divided.

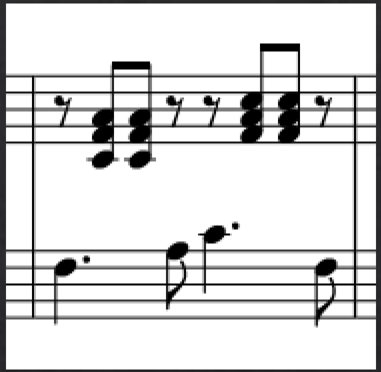
Data Extraction – data Format

- Chromatic scale
- 12 Dimension
- Duration or appearance information

Data Extraction – Implementation

- Key Normalization
  - Chromatic scale for C major:[C,C#,D,D#,E,F,F#,G,G#,A,A#,B,C]
  - Chromatic scale for G major:[G,Ab,A,Bb,B,C,C#,D,Eb,E,F,F#]
- Note Duration calculation

Data Extraction – Example



- Data for duration

C	C#	D	D#	E	F	F#	G	G#	A	A#	B
0.62	0	0	0	0.12	0	0	0.25	0	0	0	0
0.12	0	0	0	0.38	0	0	0.5	0	0	0	0

- Data for appearance

C	C#	D	D#	E	F	F#	G	G#	A	A#	B
1	0	0	0	1	0	0	1	0	0	0	0
1	0	0	0	1	0	0	1	0	0	0	0

System Architecture – Predicting Module

- Goal: predict chord for every piece in data sequence
- Methods: Hidden Markov Model, Long Short-Term Memory

System Architecture – Predicting Module

- Hidden Markov Model

Last term: Hidden Markov model

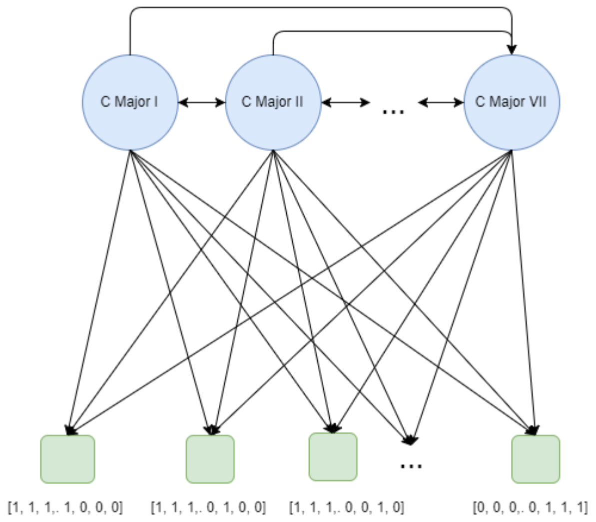
- Limited number of hidden states and observations
- Initialized using chord transition probability comes from description of professional musician – estimate the number

  e.g. "Rare" -> 0.3

This term: Hidden Markov model

- Increase number of hidden states and observations
- Transition probability comes from other research group

  e.g. Probability(Chord1, Chord2) = 0.87
- More data
- Predict all chords



System Architecture – Predicting Module

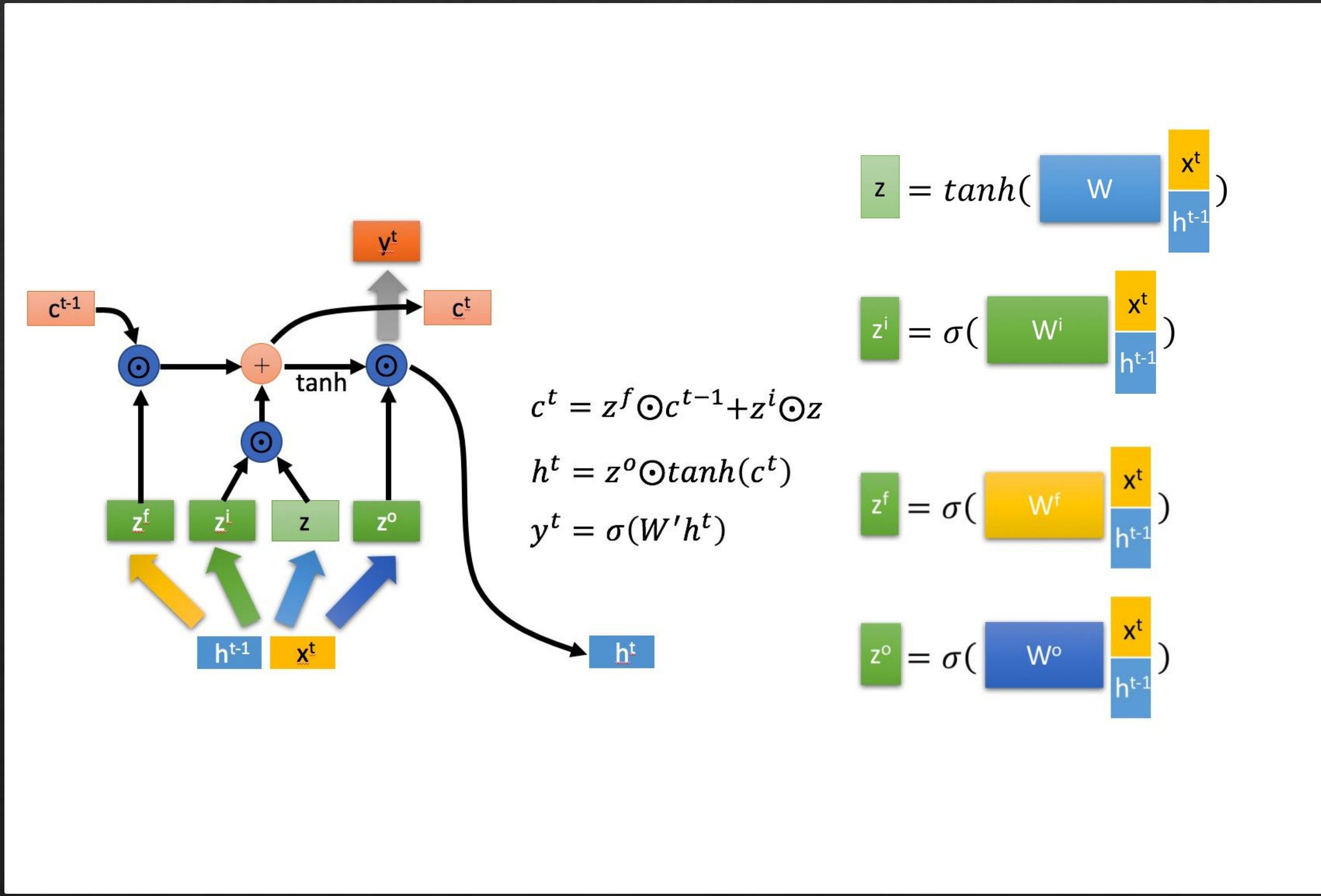
- Long Short-term Memory Model
  - Long Short-term Memory unit

Zf: forget gate

Zi: input gate

Zo: output gate

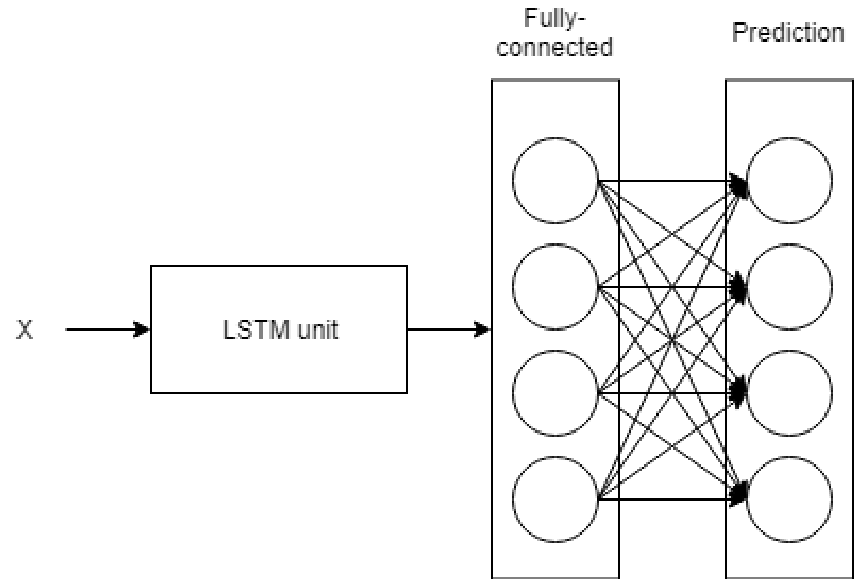
Make use of both data short time before and long time ago!



System Architecture – Predicting Module

- Long Short-term Memory Model
  - Long Short-term Memory Model

Predict = sigmoid(output)



System Architecture – Predicting Module

- Long Short-term Memory Model
- Pretrain

Goal: make use of our musical knowledge

Musical knowledge:

- Transition probability->the probability of transition from one chord to another
- Emission table->possible notes for every chord

Fake data:

X1	X2	Y1	Y2
[0.0, 0.0, 0.33, 0.0, 0.0, 0.33, 0.0, 0.0, 0.0, 0.0, 0.0, 0.33]	[0.25, 0.0, 0.0, 0.0, 0.25, 0.0, 0.0, 0.25, 0.0, 0.25, 0.0, 0.0]	VII	VI

- [[X1, Y1], [X2, Y2]]
- Y1 is chord chosen randomly->uniformly distributed
- Y2 is chord chosen according to transition probability -> $P(Y2|Y1)$ = Transition probability from Y1 to Y2
- X1, X2 are chosen according to emission table -> all possible notes combinations are given same probability to be picked
- The fake data is expected to contain the information of transition probability and emission probability

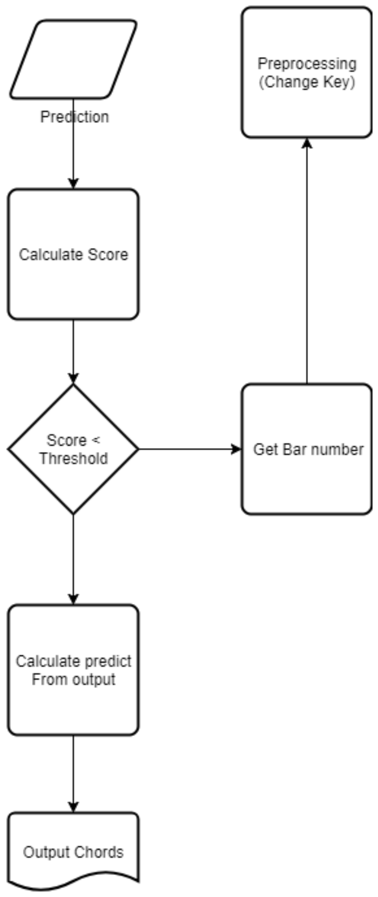
System Architecture – Evaluating Module

Definition:

- Score = mean(likelihood)
- Threshold: predefined value

Duty:

- Evaluate the prediction from predicting module.
- Find the point that maybe the reason for low score.
- Pass the point to preprocessing module for key change.



Experiments and Analysis

- Terms and data explanation
- Model comparison
- Input format comparison

Experiments and Analysis

- Terms and data explanation
- Accuracy = #(Correct prediction) / #(Chords in score)
- For each testing case, the model is trained using other scores
- By default, all the experiments use duration input format

Score name
Sonata_No._16_1st_Movement_K._545 -by Wolfgang Amadeus Mozart
The happy farmer -by Robert Schumann
Von fremden Ländern und Menschen -by Robert Schumann
Piano Sonata No.11 K.331 3rd Movement Rondo alla Turca -by Wolfgang Amadeus Mozart

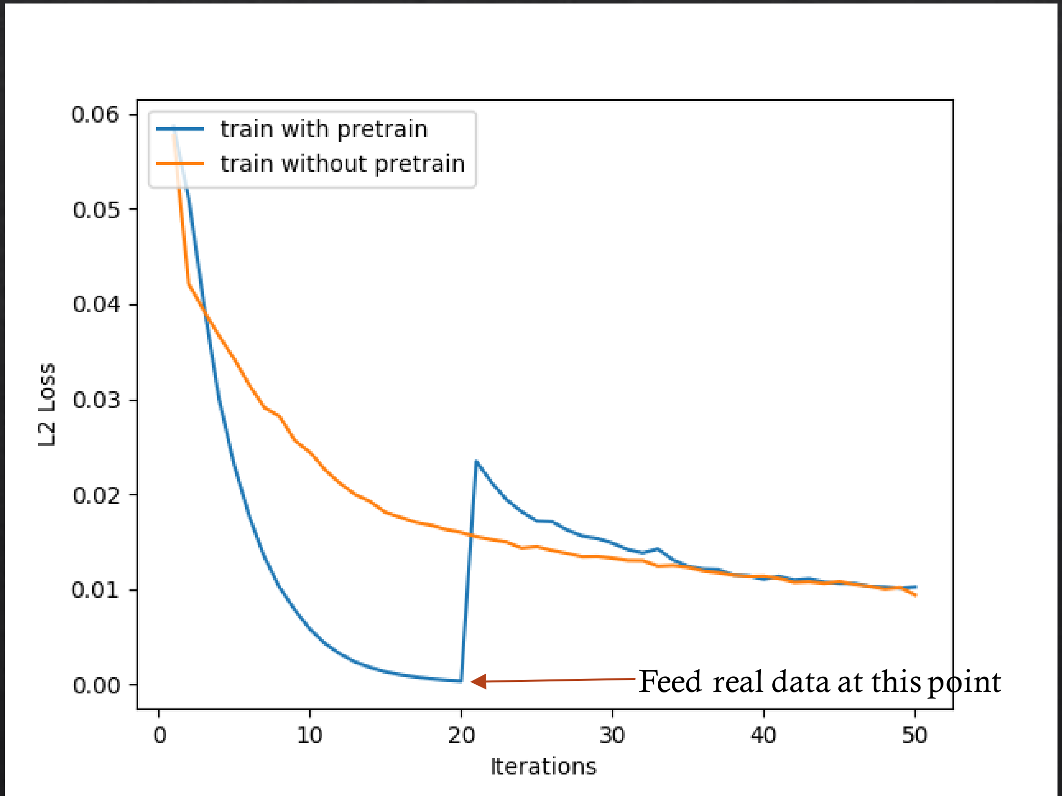
Experiments and Analysis

- Model comparison - accuracy

Testing Score	HMM	LSTM	LSTM with pretrain
The happy farmer	77.5%	80%	87.5%
Von fremden Ländern und Menschen	72.5%	77.5%	85%
Piano Sonata No.11 K.331 3rd Movement Rondo alla Turca	66.7%	75%	83.3%
Eine Kleine Nachtmus ik -by Wolfgang Amadeus Mozart	61.5%	72.9%	78.6%

Experiments and Analysis

- Model comparison – converge curve



Testing Score	HMM	LSTM	LSTM with pretrain
The happy farmer	77.5%	80%	87.5%
Von fremden Ländern und Menschen	72.5%	77.5%	85%
Piano Sonata No.11 K.331 3rd Movement Rondo alla Turca	66.7%	75%	83.3%
Eine Kleine Nach tmusik -by Wolfgang Amadeus Mozart	61.5%	72.9%	78.6%

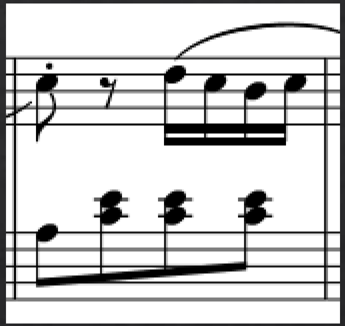
The pretrained-LSTM model takes fake data for pretraining. After 20 iterations, the model is fed with real data for training.
The model can achieve higher accuracy, which verify our assumption that the pretrain strategy indeed offer some additional information.

Experiments and Analysis

- Duration Input vs Appearance Input

Appearance Input:
[1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0]

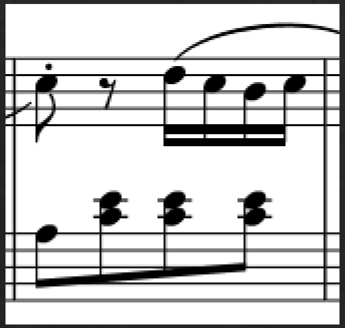
Duration Input:
[0.1, 0.0, 0.05, 0.5, 0.0, 0.05, 0.0, 0.3, 0.0, 0.0, 0.0, 0.0]



Testing Score	Appearance Input	Duration Input
The happy farmer	90%	87.5%
Von fremden Ländern und Menschen	77.5%	85%
Piano Sonata No.11 K.331 3rd Movement Rondo alla Turca	75%	83.3%
Eine Kleine Nachtmusik -by Wolfgang Amadeus Mozart	71.5%	78.6%

Experiments and Analysis

- Duration Input vs Appearance Input



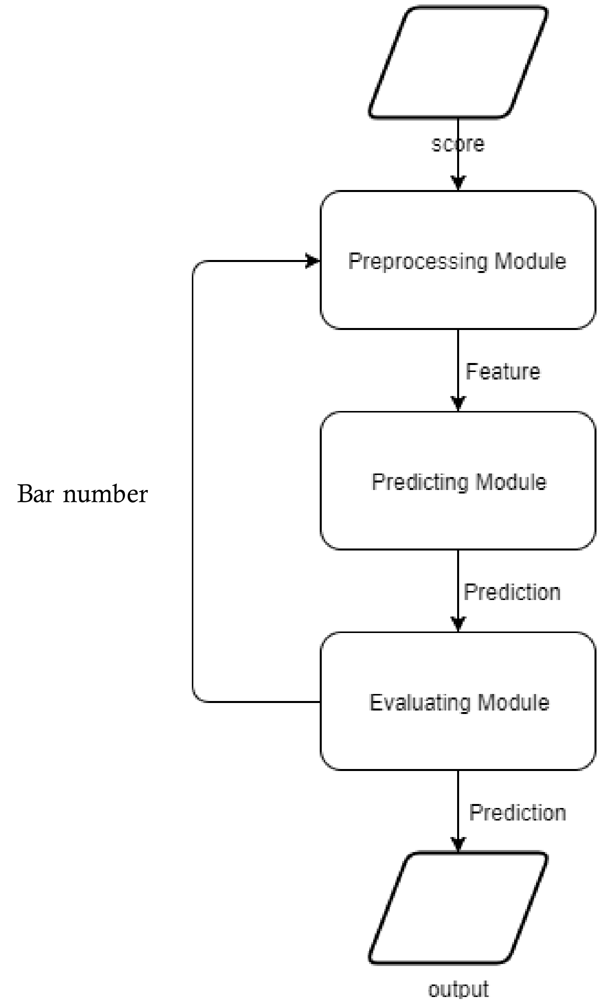
Appearance Input:
[1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0]

Duration Input:
[0.1, 0.0, 0.05, 0.5, 0.0, 0.05, 0.0, 0.3, 0.0, 0.0, 0.0, 0.0]

Testing Score	Appearance Input	Duration Input
The happy farmer	90%	87.5%
Von fremden Ländern und Menschen	77.5%	85%
Piano Sonata No.11 K.331 3rd Movement Rondo alla Turca	75%	83.3%
Eine Kleine Nacht musik -by Wolfgang Amadeus Mozart	71.5%	78.6%

The appearance will capture all the notes appeared in certain time-unit. It is highly unstable since both useful notes and useless notes are treated as the same thing. Thus, the training will be influenced by the useless notes.

System Demonstration



Limitation

- Pre-Processing Module
  - Key Identification
  - Multiple Chord Division
- Evaluating Module
- Data

Limitation - Pre-Processing Module

Measure	Standard	Predict
1	G major	G major
2	G major	G major
3	G major	G major
4	G major	G major
5	G major	C major
6	G major	G major
7	G major	G major
8	G major	e minor
9	G major	e minor
10	G major	e minor
11	e minor	e minor

Key identification result of score Von fremden Länder n und Menschen

Chord Divide result	2
Chord standard result	3

Chord Division result

Limitation – Evaluating Module

- The threshold number used in this module is predefined by our experience. This number could be extracted from the dataset by advanced methods.

Limitation – Data

- Shortage of training data

Assumption and Conclusion

- Assumption
  - LSTM Model have better result than HMM Model
  - Pre-Train method will improve the result
  - Data with duration will provides more useful information
- Conclusion
  - LSTM Model is more suitable for Chord identification than HMM Model
  - Pre-Train with musical knowledge improves the accuracy
  - Duration data provides more information

Future Plan

- Better Key Identification to handle global and local issue.
- Better Multiple chord division to handle missing notes
- Separate key recognition and chord recognition
- Find more data for training and testing

Thanks

Q&A