

การสร้างเกมสอนร้องเพลงไทยลูกทุ่งด้วยเทคนิคการเทียบระดับเสียง

The development of Thai folk song singing game using voice level calibration techniques

กันตพร เปรมปรีสุข¹, ณัฐโชติ พรหมฤทธิ์^{2*} และ สัจจาภรณ์ ไวจรรยา^{3*}

บทคัดย่อ

บทความนี้นำเสนอเกมสอนร้องเพลงไทยลูกทุ่ง โดยแบ่งโหมดใช้งานออกเป็นสองโหมด ได้แก่ โหมดฝึกร้อง เพื่อให้ผู้ใช้สามารถฝึกออกเสียงตัวโน้ตแต่ละตัวได้อย่างถูกต้อง และโหมดเกมร้องเพลง ที่ให้ผู้ใช้งานฝึกร้องเพลงลูกทุ่งได้จนจบเพลง โดยในโหมดแรกนั้นโปรแกรมรับเสียงผู้ใช้งานเข้าแบบเรียลไทม์ และในโหมดที่สองเป็นการบันทึกไฟล์เสียงของผู้ใช้เอาไว้ โดยนำเสียงที่รับเข้าทั้งสองแบบเข้าสู่โมเดลเพื่อทำนายความถี่และแปลงค่าออกมาเป็นโน้ตดนตรี ในการเปรียบเทียบความถูกต้องของเสียงผู้ใช้งานและเสียงต้นฉบับในโหมดเกมร้องเพลงได้ใช้ BLEU Score ในการเปรียบเทียบและให้คะแนนผู้ใช้งานแบ่งออกเป็นห้าระดับ แทนด้วยดาวจำนวน 1 – 5 ดวง โปรแกรมในโหมดเกมร้องเพลงผู้ใช้งานต้องเล่นเกมเพื่อสะสมจำนวนดาวในแต่ละด่านเพื่อใช้สำหรับปลดล๊อคเพลงในด่านถัดไป

คำสำคัญ: ฝึกร้องเพลง เกมร้องเพลง เพลงไทยลูกทุ่ง การตรวจจับโน้ตดนตรีแบบเรียลไทม์

การเปรียบเทียบข้อมูลไฟล์เท็กซ์สองไฟล์ด้วยบลูสกอร์

^{1, 2, 3} ศูนยเชี่ยวชาญปัญญาประดิษฐ์และภาษาธรรมชาติ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

E-mail: prempresuk_k@silpakorn.edu

*Corresponding author, e-mail: waijanya_s@silpakorn.edu

*Co-Corresponding author, e-mail: promrit_n@silpakorn.edu

Abstract

This article divides Thai folk song singing teaching method as a game into two modes. First, singing practice mode for correctly practicing pronunciations of each note (mode A). Second, singing game mode for users to practice singing Thai folk songs until the end of the song (mode B). In mode A, the program receives users' voices as an input in real time. In mode B, the program receives voices by recording in .wav file. Then, the input voices from both modes were predicted to determine the frequencies and transform into notes. BLEU Score was used for comparing users' voices accuracy and the original voices from the singing game mode.

The comparing and rating of users were divided into five levels, as the number of five stars, from 1 to 5 stars. In the Singing Game mode, the users have to play the game to collect the number of stars in each stage in order to unlock the next song.

Keywords: Singing practice, singing game, Thai folk song, real-time pitch detection, compare different text file with bleu score

1. บทนำ

เพลงเป็นประเภทของความบันเทิงที่ได้รับความนิยมเป็นอย่างมากเพราะมีความหลากหลายและสามารถเข้าถึงได้ง่าย ในแต่ละประเทศต่างมีวัฒนธรรมในการฟังหรือร้องเพลงที่แตกต่างกันออกไป ทั้งในเรื่องของแนวดนตรีและการใช้ภาษา เช่นเดียวกับประเทศไทยซึ่งมีเพลงไทยลูกทุ่งที่มีการใช้ภาษา คำร้อง และทำนองอย่างมีเอกลักษณ์ แต่ในปัจจุบันเพลงไทยลูกทุ่งไม่เป็นที่นิยมสำหรับคนรุ่นใหม่มากนัก เนื่องจากวัฒนธรรมจากต่างชาติที่เข้ามามีอิทธิพลในหลาย ๆ ด้าน รวมไปถึงเรื่องของภาษาและแนวเพลง จากการพูดคุยสอบถามกับผู้คนส่วนใหญ่ที่ไม่เลือกร้องเพลงลูกทุ่ง พบว่าเหตุผลคือเพลงลูกทุ่งนั้นร้องยากเนื่องจากต้องมีการเอื้อนคำ รวมไปถึงการใช้ลูกคอในการร้องต่าง ๆ จึงเลือกร้องเพลงแนวอื่น ๆ หรือเลือกฝึกร้องเพลงแนวอื่น ๆ แทน

ในด้านของการฝึกร้องหรือเรียนร้องเพลงในปัจจุบันมีแอปพลิเคชันสอนร้องเพลงทั้งบนเว็บไซต์และบนสมาร์ตโฟนให้ใช้งานกันได้อย่างแพร่หลาย แต่แนวเพลงที่ผู้ใช้สามารถเลือกในการฝึกร้องได้นั้นส่วนใหญ่มียังมีแค่สตริงและสากลเท่านั้นไม่มีเพลงไทยลูกทุ่ง และแอปพลิเคชันส่วนใหญ่นั้นเป็นภาษาอังกฤษที่มีรูปแบบการใช้งานค่อนข้างซับซ้อนและเข้าใจได้ยาก อย่างเช่น มีเพียงให้ผู้ใช้ออกเสียงให้ตรงกับตัวโน้ตต่าง ๆ แต่ไม่มีตัวอย่าง

เสียง หากผู้ใช้งานไม่มีความรู้ในเรื่องของตัวโน้ตก็จะไม่สามารถออกเสียงให้ถูกต้องได้และไม่มีการสอนร้องเป็นเพลงจนจบเพลง ไม่สามารถตอบโจทย์ในด้านของการฝึกร้องเพลงไทยลูกทุ่งได้โดยตรง

บทความนี้จึงนำเสนอเว็บแอปพลิเคชันสอนร้องเพลงไทยลูกทุ่งที่สามารถเข้าใช้งานได้ง่าย การใช้งานไม่ซับซ้อน และสามารถเป็นตัวเลือกหนึ่งให้กับผู้ที่สนใจฝึกร้องเพลงลูกทุ่ง โดยเนื้อหาในบทความหัวข้อที่ 2 จะกล่าวถึงทฤษฎีและความรู้ที่เกี่ยวข้อง หัวข้อที่ 3 ขั้นตอนและวิธีการดำเนินงาน หัวข้อที่ 4 ผลการดำเนินงาน และหัวข้อที่ 5 สรุปและอภิปราย โดยละเอียดต่อไป

2. ทฤษฎีและความรู้ที่เกี่ยวข้อง

2.1 เสียง

2.1.1 คลื่นเสียง (Sound Wave) [1]

คลื่นเสียงมีองค์ประกอบที่สำคัญในการทำให้เกิดคลื่นเสียงสามอย่าง คือ

2.1.1.1) ความถี่เสียง (Frequency) เสียงสูง เสียงกลาง และเสียงต่ำที่มนุษย์สามารถได้ยินนั้น สามารถวัดได้ในช่วงระยะเวลา 1 วินาทีว่าคลื่นเสียงนั้นมีการสั่นขึ้นลงกี่รอบ ตัวอย่างเช่น คลื่นสั่นขึ้นลง 1000 ครั้งใน 1 วินาที หมายความว่าคลื่นนั้นมีความถี่ 1000 เฮิรตซ์ (Hertz) โดยความถี่เสียงจะเป็นตัวบ่งบอกว่าเสียงนั้นเป็นเสียงสูงหรือเสียงต่ำ โดยคลื่นที่มีความถี่ต่ำ เสียงที่ได้ยินเป็นเสียงทุ้ม ส่วนคลื่นที่มีความถี่สูง เสียงที่ได้ยินเป็นเสียงแหลม

2.1.1.2) ความยาวคลื่น (Wavelength) หรือระยะห่างระหว่างยอดคลื่นลูกหนึ่งไปยังคลื่นลูกหนึ่ง มีหน่วยเป็นเมตร โดยสามารถหาความยาวคลื่นได้จากสมการที่ 1

$$\lambda = v/f \quad (1)$$

โดย λ คือ ความยาวคลื่น

v คือ ความเร็วคลื่น

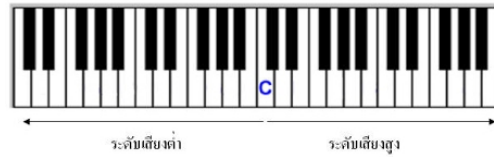
f คือ ความถี่คลื่น

2.1.1.3) ความดัง (Amplitude) บ่งบอกถึงความสูงต่ำของคลื่นเสียง โดยคลื่นเสียงที่มีความสูงมาก เสียงจะดังขึ้นตามความสูง โดยปกติหูของมนุษย์สามารถรับความดังได้ตั้งแต่ 0 dB ไปจนถึง 120 dB ความดังมีหน่วยวัดเป็น Decibel หรือ dB

2.1.2 ระดับเสียง (Pitch) [2]

หมายถึงคลื่นความถี่ที่เกิดจากการสั่นสะเทือนของวัตถุทำให้เกิดเสียงสูงและเสียงต่ำ

เสียงสูงได้มาจากความถี่ที่สั้นกว่าเสียงต่ำ โดยความถี่ที่ได้จากการวัดการสั่นสะเทือนของวัตถุนั้นมีหน่วยวัดเป็นรอบต่อวินาที หรือ เฮิรตซ์ (Hz)



รูปที่ 1 ระดับเสียงสูงและต่ำบนลิ้นเปียโน

โดยมีระยะระหว่างโน้ตเสียงสูงกับโน้ตเสียงต่ำเรียกว่าช่วงเสียง (Pitch range) คำว่าช่วงเสียงนั้นโดยปกติใช้อธิบายถึงระดับการขับร้องของมนุษย์ 4 ระดับ แบ่งเป็น โซปราโน (Soprano) เสียงสูง ผู้หญิง อัลโต (Alto) เสียงต่ำผู้หญิง เทเนอร์ (Tenor) เสียงสูงผู้ชาย เบส (Bass) เสียงต่ำของผู้ชาย [7]

2.2 MIDI numbers [3]

MIDI มาจาก Musical Instrument Digital Interface คือชุดข้อมูลดิจิทัลที่ใช้เก็บรายละเอียดของดนตรี โดย MIDI numbers มีความสัมพันธ์กับความถี่ในการแปลงเสียงออกมาเป็นโน้ตในทางดนตรี

MIDI number	Note number	Keyboard	Frequency (Hz)	Period (ms)
21	22	A0	27.500	36.36
23	24	B0	30.902	32.36
24	25	C1	32.703	30.58
26	27	D1	35.663	27.99
28	29	E1	38.599	25.91
29	30	F1	40.554	24.67
31	32	G1	43.000	23.26
33	34	A1	45.000	22.22
35	36	B1	47.000	21.28
37	38	C2	50.000	20.00
39	40	D2	55.000	18.18
41	42	E2	58.739	17.03
43	44	F2	61.735	16.19
45	46	G2	65.409	15.29
47	48	A2	69.298	14.42
49	50	B2	73.426	13.61
51	52	C3	77.782	12.86
53	54	D3	82.409	12.13
55	56	E3	87.309	11.45
57	58	F3	92.499	10.81
59	60	G3	97.998	10.21
61	62	A3	103.83	9.63
63	64	B3	109.95	9.09
65	66	C4	116.34	8.59
67	68	D4	123.03	8.13
69	70	E4	129.96	7.70
71	72	F4	137.13	7.29
73	74	G4	144.53	6.92
75	76	A4	152.19	6.57
77	78	B4	160.15	6.24
79	80	C5	168.41	5.94
81	82	D5	176.99	5.65
83	84	E5	185.80	5.38
85	86	F5	194.93	5.13
87	88	G5	204.39	4.89
89	90	A5	214.18	4.67
91	92	B5	224.31	4.46
93	94	C6	234.76	4.26
95	96	D6	245.65	4.07
97	98	E6	256.98	3.89
99	100	F6	268.65	3.72
101	102	G6	280.68	3.56
103	104	A6	293.08	3.41
105	106	B6	305.85	3.27
107	108	C7	319.00	3.13
109	110	D7	332.54	3.01
111	112	E7	346.48	2.89
113	114	F7	360.82	2.77
115	116	G7	375.56	2.66
117	118	A7	390.71	2.56
119	120	B7	406.28	2.46
121	122	C8	422.29	2.37
123	124	D8	438.75	2.28
125	126	E8	455.68	2.20
127	128	F8	473.09	2.11
129	130	G8	490.98	2.04
131	132	A8	509.35	1.96
133	134	B8	528.20	1.89
135	136	C9	547.63	1.83
137	138	D9	567.65	1.76
139	140	E9	588.27	1.70
141	142	F9	609.50	1.64
143	144	G9	631.35	1.58
145	146	A9	653.82	1.53
147	148	B9	676.93	1.48
149	150	C10	699.68	1.43
151	152	D10	723.09	1.38
153	154	E10	747.16	1.34
155	156	F10	771.91	1.29
157	158	G10	797.34	1.25
159	160	A10	823.47	1.21
161	162	B10	850.31	1.18
163	164	C11	877.95	1.14
165	166	D11	906.40	1.10
167	168	E11	935.65	1.07
169	170	F11	965.71	1.04
171	172	G11	996.59	1.01
173	174	A11	1028.29	0.97
175	176	B11	1060.81	0.94
177	178	C12	1094.13	0.91
179	180	D12	1128.29	0.88
181	182	E12	1163.29	0.86
183	184	F12	1199.13	0.83
185	186	G12	1235.84	0.81
187	188	A12	1273.43	0.78
189	190	B12	1311.91	0.76
191	192	C13	1351.29	0.73
193	194	D13	1391.58	0.71
195	196	E13	1432.70	0.69
197	198	F13	1474.65	0.67
199	200	G13	1517.44	0.65
201	202	A13	1561.08	0.63
203	204	B13	1605.58	0.61
205	206	C14	1650.94	0.60
207	208	D14	1697.17	0.59
209	210	E14	1744.28	0.57
211	212	F14	1792.28	0.56
213	214	G14	1841.18	0.54
215	216	A14	1890.99	0.53
217	218	B14	1941.72	0.52
219	220	C15	1993.38	0.50
221	222	D15	2045.98	0.49
223	224	E15	2099.53	0.48
225	226	F15	2154.04	0.46
227	228	G15	2209.52	0.45
229	230	A15	2265.98	0.44
231	232	B15	2323.43	0.43
233	234	C16	2381.88	0.42
235	236	D16	2441.34	0.41
237	238	E16	2501.82	0.40
239	240	F16	2563.33	0.39
241	242	G16	2625.88	0.38
243	244	A16	2689.48	0.37
245	246	B16	2754.13	0.36
247	248	C17	2819.84	0.35
249	250	D17	2886.62	0.34
251	252	E17	2954.48	0.33
253	254	F17	3023.43	0.33
255	256	G17	3093.48	0.32
257	258	A17	3164.64	0.31
259	260	B17	3236.92	0.31
261	262	C18	3310.33	0.30
263	264	D18	3384.88	0.29
265	266	E18	3460.58	0.29
267	268	F18	3537.44	0.28
269	270	G18	3615.47	0.28
271	272	A18	3694.68	0.27
273	274	B18	3775.07	0.27
275	276	C19	3856.65	0.26
277	278	D19	3939.43	0.26
279	280	E19	4023.42	0.25
281	282	F19	4108.63	0.24
283	284	G19	4195.07	0.24
285	286	A19	4282.74	0.23
287	288	B19	4371.65	0.23
289	290	C20	4461.81	0.22
291	292	D20	4553.22	0.22
293	294	E20	4645.89	0.21
295	296	F20	4739.83	0.21
297	298	G20	4835.04	0.20
299	300	A20	4931.52	0.20
301	302	B20	5029.28	0.19
303	304	C21	5128.32	0.19
305	306	D21	5228.64	0.18
307	308	E21	5330.25	0.18
309	310	F21	5433.15	0.17
311	312	G21	5537.35	0.17
313	314	A21	5642.85	0.16
315	316	B21	5749.65	0.16
317	318	C22	5857.76	0.15
319	320	D22	5967.18	0.15
321	322	E22	6077.91	0.14
323	324	F22	6189.95	0.14
325	326	G22	6303.30	0.14
327	328	A22	6417.96	0.13
329	330	B22	6533.93	0.13
331	332	C23	6651.22	0.12
333	334	D23	6769.83	0.12
335	336	E23	6889.76	0.12
337	338	F23	7011.01	0.11
339	340	G23	7133.58	0.11
341	342	A23	7257.47	0.11
343	344	B23	7382.68	0.10
345	346	C24	7509.21	0.10
347	348	D24	7637.06	0.10
349	350	E24	7766.23	0.09
351	352	F24	7896.73	0.09
353	354	G24	8028.56	0.09
355	356	A24	8161.72	0.08
357	358	B24	8296.21	0.08
359	360	C25	8432.03	0.08
361	362	D25	8569.18	0.07
363	364	E25	8707.67	0.07
365	366	F25	8847.50	0.07
367	368	G25	8988.68	0.06
369	370	A25	9131.11	0.06
371	372	B25	9274.89	0.06
373	374	C26	9419.93	0.05
375	376	D26	9566.23	0.05
377	378	E26	9713.79	0.05
379	380	F26	9862.61	0.04
381	382	G26	10012.70	0.04
383	384	A26	10164.06	0.04
385	386	B26	10316.69	0.04
387	388	C27	10470.59	0.03
389	390	D27	10625.76	0.03
391	392	E27	10782.20	0.03
393	394	F27	10939.91	0.03
395	396	G27	11098.89	0.03
397	398	A27	11259.13	0.02
399	400	B27	11420.66	0.02
401	402	C28	11583.45	0.02
403	404	D28	11747.52	0.02
405	406	E28	11912.87	0.02
407	408	F28	12079.50	0.01
409	410	G28	12247.42	0.01
411	412	A28	12416.63	0.01
413	414	B28	12587.13	0.01
415	416	C29	12758.93	0.01
417	418	D29	12932.03	0.01
419	420	E29	13106.43	0.00
421	422	F29	13282.13	0.00
423	424	G29	13459.13	0.00
425	426	A29	13637.43	0.00
427	428	B29	13817.03	0.00
429	430	C30	13997.93	0.00
431	432	D30	14179.13	0.00
433	434	E30	14361.53	0.00
435	436	F30	14545.13	0.00
437	438	G30	14729.93	0.00
439	440	A30	14915.93	0.00
441	442	B30	15103.13	0.00
443	444	C31	15291.53	0.00
445	446	D31	15481.13	0.00
447	448	E31	15671.93	0.00
449	450	F31	15863.93	0.00
451	452	G31	16057.13	0.00
453	454	A31	16251.53	0.00
455	456	B31	16447.13	0.00
457	458	C32	16643.93	0.00
459	460	D32	16841.93	0.00
461	462	E32	17041.13	0.00
463	464	F32	17241.53	0.00
465	466	G32	17443.13	0.00
467	468	A32	17645.93	0.00
469	470	B32	17849.93	0.00
471	472	C33	18055.13	0.00
473	474	D33	18261.53	0.00
475	476	E33	18469.13	0.00
477	478	F33	18677.93	0.00
479	480	G33	18887.93	0.00
481	482	A33	19099.13	0.00
483	484	B33	19311.53	0.00
485	486	C34	19525.13	0.00
487	488	D34	19739.93	0.00
489	490	E34	19955.93	0.00
491	492	F34	20173.13	0.00
493	494	G34	20391.53	0.00
495	496	A34	20611.13	0.00
497	498	B34	20831.93	0.00
499	500	C35	21053.93	0.00
501	502	D35	21277.13	0.00
503	504	E35	21501.53	0.00
505	506	F35	2172	0.00

เปรียบเทียบประโยคจะใช้ N-Gram ซึ่งเป็นการประมาณค่าความน่าจะเป็นของคำที่เกิดขึ้น มีค่าเท่ากับผลคูณของความน่าจะเป็นของคำที่พบในประโยคนั้น ๆ โดยใน N-Gram สามารถประกอบไปด้วย Unigram หรือ 1 คำ Bigram หรือ 2 คำ Trigram หรือ 3 คำ โดยจะใช้ในการคำนวณค่าที่พบจากการแปลด้วยเครื่องแปลภาษาเทียบกับเอกสารอ้างอิง

การคำนวณ N-Gram เป็นการวัดแบบ Precision โดยนับจากจำนวน Unigram, Bigram, Trigram, 4-Gram ที่เกิดขึ้นหารด้วยจำนวนคำทั้งหมด ซึ่งมีสูตรการคำนวณที่แปลงมาจาก Precision ดังสมการที่ 2

$$P_n = \frac{\sum_{c \in \{candidates\}} \sum_{n-gram \in c} Count_{clip(n-gram)}}{\sum_{c \in \{candidatea\}} \sum_{n-gram' \in c'} Count(n-gram')} \quad (2)$$

ตัวอย่างการเปรียบเทียบประโยคที่พบแบบ Unigram ซึ่งเป็นการเปรียบเทียบแบบคำต่อคำ ดังนี้

Candidate : Jane may go to school

Reference : Jane went to the airport

Modified Unigram Precision : 2/5

โดย Count (n – gram') คือ จำนวนคำทั้งหมดจากประโยค Candidate ที่ถูกนำมาคำนวณในสมการที่ 3 และ 4

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases} \quad (3)$$

$$BLEU = BP \cdot \exp \left(\sum_{n=1}^6 w_n \log \frac{6}{9} \right) \quad (4)$$

โดย P_n คือ ค่า Modified Precision ของแต่ละ gram

N คือ จำนวนสูงสุดของ N – Gram

w_n คือ น้ำหนักของแต่ละ Modified Precision

โดยค่าเริ่มต้นของ N = 4 และ $w_n = 0.25$

C คือ ความยาวของ Candidate

R คือ ความยาวของ Reference

2.4 PyAudio [5]

PyAudio คือ ไลบรารีในภาษาไพธอนสำหรับการเล่นและบันทึกเสียง โดยรองรับไฟล์ .wav เท่านั้น และนอกเหนือจากการเล่นและบันทึกเสียงแล้ว PyAudio ยังมีโมดูลในการ Stream เสียงได้อีกด้วยโดยใช้ `pyaudio.stream.write()` และสามารถใช้อัปเดตข้อมูลที่ได้รับจากการ stream ด้วย `pyaudio.stream.read()` PyAudio รองรับระบบปฏิบัติการ GNU/Linux, Microsoft Windows และ Apple Mac OSX และรองรับทั้ง Python 2 และ Python 3

ขั้นตอนในการใช้งาน PyAudio

- 1) กำหนดไซส์ในการบันทึก ตั้งแต่ 8 – 32 bits
- 2) กำหนดค่า Sampling Rate ตั้งแต่ 0 – 44100 Hz
- 3) กำหนด Channels ในการรับเสียงเข้าด้วย 1 หรือ 2 โดย 1 คือการรับเสียงเข้าแบบ Monotone และ 2 คือการรับเสียงเข้าแบบ Stereo

2.6 DEMUCS [6]

DEMUCS คือโมเดลที่ใช้ในการแยกส่วนประกอบของไฟล์เพลงในโดเมนรูปคลื่น โดย demucs ใช้สถาปัตยกรรมที่ได้รับแรงบันดาลใจมา WAVE-U-Net และ SING ที่พัฒนาโดย MIT Licence

วิธีการใช้งาน demucs สามารถใช้งานได้ทั้งบนระบบปฏิบัติการ Windows และ MacOS โดยติดตั้งผ่าน Python 3.7 โดยมีวิธีการติดตั้งและวิธีการใช้งานที่ไม่ซับซ้อนตามขั้นตอนในรูปที่ 3

```
cd ~
conda install git
git clone https://github.com/facebookresearch/demucs
cd demucs
conda env update -f environment-cpu.yml
conda activate demucs
python3 -m demucs.separate --dl -n demucs -d cpu PATH_TO_AUDIO_FILE_1
```

รูปที่ 3 ขั้นตอนการติดตั้งและการใช้งาน demucs

ขั้นตอนการใช้งานหลังจากติดตั้งคือนำไฟล์เสียงเพลงที่ต้องการเข้าสู่โมเดล demucs ตามวิธีข้างต้น โดยโมเดลจะคืนค่าผลลัพธ์กลับมาเป็นไฟล์องค์ประกอบของเพลง 4 ไฟล์ได้แก่

- bass.wav
- drums.wav
- others.wav
- vocals.wav

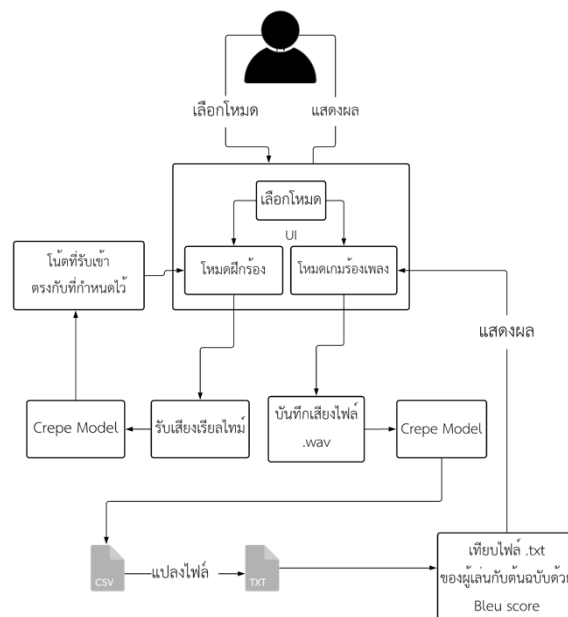
3. ขั้นตอนและวิธีการดำเนินงาน

ขั้นตอนและวิธีการดำเนินงานเพื่อสร้างเกมสอนร้องเพลงไทยลูกทุ่ง ด้วยเทคนิคการเทียบระดับเสียง ผู้วิจัยจะอธิบายถึง 3.1 สถาปัตยกรรมของระบบ 3.2 ขั้นตอนการทำงานของระบบ และ 3.3 การรับเสียงเข้า และแปลงคลื่นความถี่เสียง โดยมีรายละเอียดดังนี้

3.1 สถาปัตยกรรมของระบบ

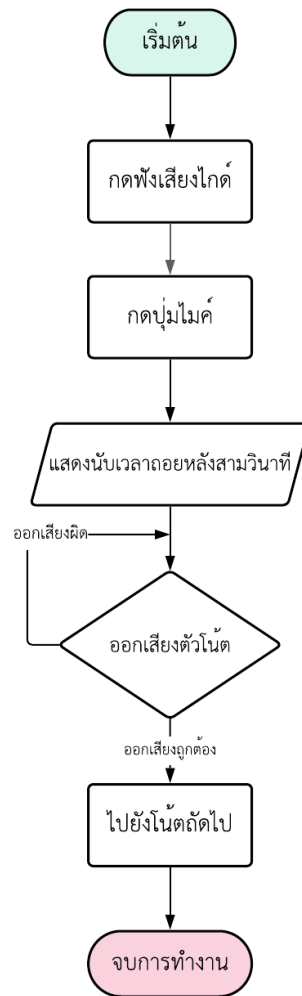
เกมสอนร้องเพลงไทยลูกทุ่งนี้แบ่งออกเป็น 4 ส่วนดังนี้

ส่วนที่ 1 (ผู้ใช้งานเข้าใช้งานเว็บไซต์โดยเลือกโหมดที่จะเข้าใช้งาน ส่วนที่ 2 (ผู้ใช้งานเลือกโหมดฝึกร้องหรือโหมดเกมร้องเพลง ส่วนที่ 3 (ผู้ใช้งานเลือกโหมดฝึกร้องโปรแกรมจะเปิดไมค์เพื่อรับเสียงจากผู้ใช้งานแบบเรียลไทม์จะส่งเสียงที่รับเข้าสู่โมเดลเพื่อนำผลลัพธ์ไปเช็คค่าผู้ใช้งานออกเสียงได้ถูกต้องตามโน้ตที่กำหนดหรือไม่ และส่งกลับไปแสดงผล ส่วนที่ 4 (ผู้ใช้เลือกโหมดเกมร้องเพลง โปรแกรมจะบันทึกเสียงร้องของผู้ใช้ตามเวลาที่กำหนดไว้ในแต่ละด่านเป็นไฟล์ .wav และนำไฟล์ดังกล่าวเข้าสู่โมเดล และนำผลลัพธ์ที่ได้เป็นไฟล์ .csv แปลงเป็นไฟล์ .txt เพื่อใช้ BLEU Score เปรียบเทียบความถูกต้องระหว่างไฟล์ของผู้ใช้งานกับไฟล์ต้นฉบับ ซึ่งแสดงสถาปัตยกรรมของระบบดังรูปที่ 4



รูปที่ 4 สถาปัตยกรรมของระบบ

3.2 ขั้นตอนการทำงานของระบบ

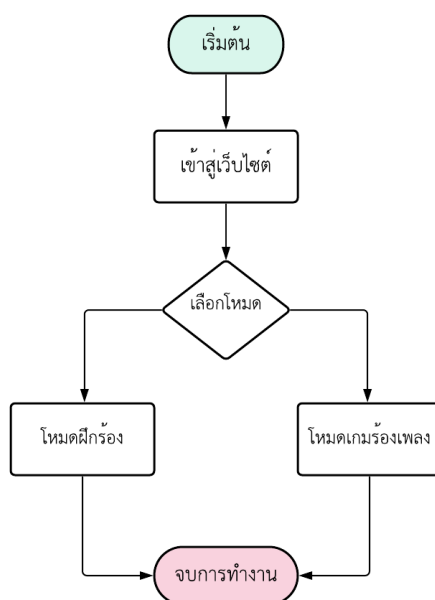


แบ่งการทำงานออกเป็น 5 ส่วนประกอบด้วย

- 1) ส่วนเลือกโหมด
- 2) ส่วนของกระบวนการในโหมดฝึกร้อง
- 3) ส่วนของกระบวนการในโหมดเกมร้องเพลง
- 4) ส่วนของกระบวนการคิดคะแนน
- 5) ส่วนของกระบวนการปลดล๊อคด่านในโหมดเกมร้องเพลง

3.2.1 กระบวนการเลือกโหมด

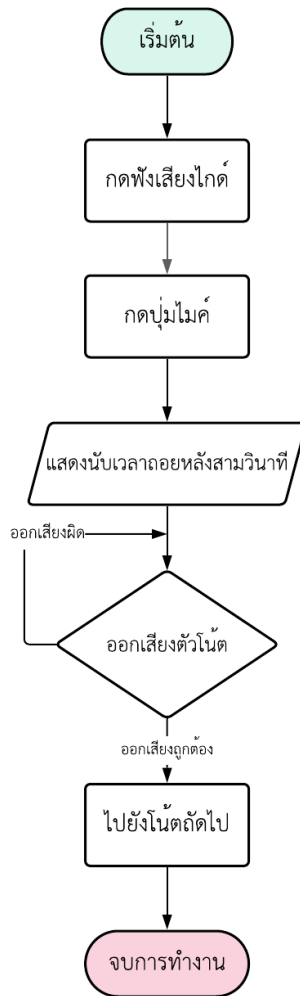
กระบวนการนี้เริ่มต้นจากผู้ใช้งานเข้าสู่เว็บไซต์และเลือกโหมดเพื่อเข้าใช้งาน โดยมีสองโหมดให้เลือก ได้แก่ โหมดฝึกร้อง และ โหมดเกมร้องเพลง ดังรูปที่ 5



รูปที่ 5 กระบวนการเลือกโหมด

3.2.2 กระบวนการทำงานในโหมดฝึกร้อง

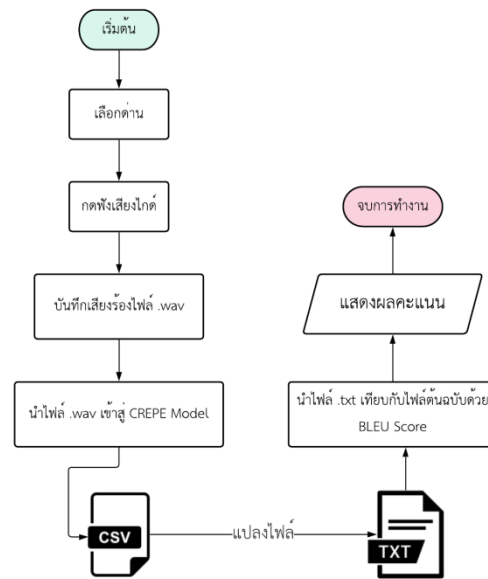
ในกระบวนการนี้ผู้ใช้งานสามารถกดฟังเสียงไกด์ได้ไม่จำกัดจำนวนครั้ง โดยเมื่อพร้อมให้กดปุ่มรูปไม้คีย์เพื่อเข้าสู่หน้าจอเสียง โดยก่อนแสดงหน้าจอเสียงโปรแกรมจะแสดงเวลานับถอยหลัง 3 วินาที หลังจากนั้นให้ผู้ใช้งานออกเสียงตัวโน้ตที่กำหนดไว้ให้ถูกต้อง เมื่อออกเสียงโน้ตตัวแรกถูกต้องแล้ว โปรแกรมจะนำผู้ใช้งานเข้าสู่หน้าโน้ตตัวถัดไป จนถึงโน้ตตัวสุดท้าย โปรแกรมจะนำผู้ใช้งานเข้าสู่โหมดเกมร้องเพลงโดยอัตโนมัติ ดังรูปที่ 6



รูปที่ 6 กระบวนการทำงานในโหมดฝึกร้อง

3.2.3 กระบวนการในโหมดเกมร้องเพลง

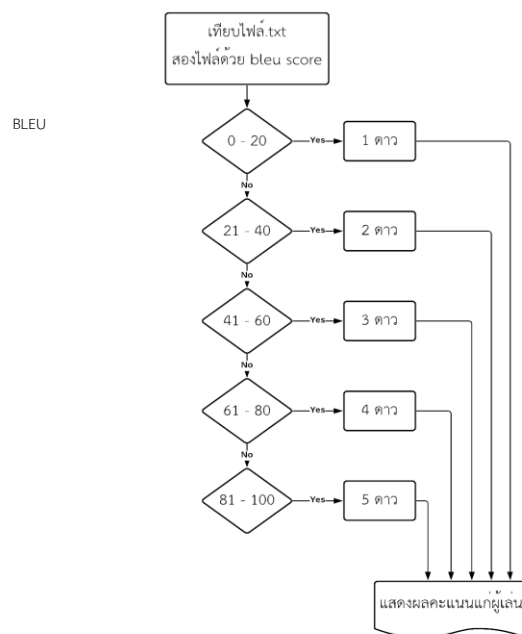
ในกระบวนการนี้เริ่มต้นจากผู้ใช้งานเลือกด้านที่จะร้องเพลง และกดเพื่อฟังเสียงไกด์ จากนั้นกดปุ่มเพื่อร้องเพลง โดยในส่วนนี้โปรแกรมจะบันทึกเสียงของผู้ใช้เอาไว้ตามเวลาที่กำหนดโดยบันทึกเป็นไฟล์ .wav จากนั้นโปรแกรมจะนำไฟล์ดังกล่าวเข้าสู่ CREPE model เพื่อทำนายความถี่ในแต่ละวินาที จะได้ผลลัพธ์เป็นไฟล์ .csv และแปลงไฟล์ .csv ให้อยู่ในรูปแบบไฟล์ .txt เพื่อใช้ BLEU Score ในการเปรียบเทียบกับไฟล์ต้นฉบับ ดังรูปที่ 7



รูปที่ 7 กระบวนการทำงานในโหมดเกมร้องเพลง

3.2.4 กระบวนการคิดคะแนน

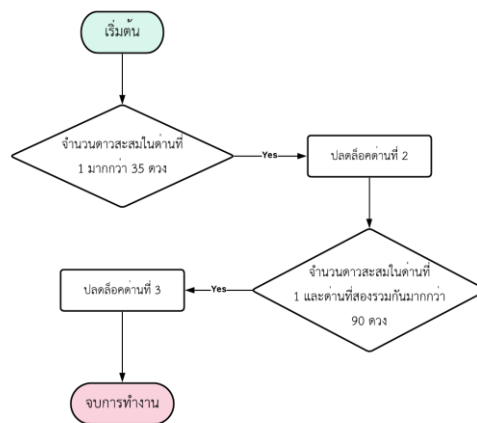
ในส่วนของการคิดคะแนน จะเกิดขึ้นในโหมดเกมร้องเพลงเท่านั้น โดยคิดจากคะแนนที่ได้รับจากการเปรียบเทียบด้วยการใช้ BLEU ซึ่งมีผลคะแนนตั้งแต่ 0 – 1 คะแนน การให้คะแนนในกระบวนการนี้จะแสดงผลเป็นรูปดาวตั้งแต่ 1 – 5 ดวงตามช่วงคะแนนดังรูปที่ 8



รูปที่ 8 กระบวนการคิดคะแนน

3.2.5 กระบวนการปลดล๊อคด่านในโหมดเกมร้องเพลง

ในกระบวนการนี้จะใช้จำนวนดาวสะสมของผู้ใช้งานในด้านย่อยเพื่อปลดล๊อคด่านใหญ่ถัดไป ตามเงื่อนไขดังนี้ ต้อง ผู้ใช้งานมีดาวจำนวน 35 ดวงในด้านเพลงสาวนาสิ่งแพน หน้าเว็บไซต์ถึงจะแสดงปุ่มเพื่อกดไปยังด่านที่ 2 เพลงสยามเมืองยิ้ม และในด่านที่ 2 เพลงสยามเมืองยิ้มผู้ใช้งานต้องมีจำนวนดาวสะสมของเพลงในด้านแรกและด่านที่สองรวมกันไม่ต่ำกว่า 90 ดวงเพื่อปลดล๊อคเพลงโลกของผึ้ง ดังรูปที่ 9



รูปที่ 9 กระบวนการปลดล๊อคด่านในโหมดเกมร้องเพลง

3.3 การรับเสียงเข้าและแปลงเป็นคลื่นความถี่ด้วย CREPE (A Convolutional Representation for Pitch Estimation)

```

stream = pyaudio.PyAudio().open(format=pyaudio.paInt16,
                                channels=1,
                                rate=FSAMP,
                                input=True,
                                frames_per_buffer=FRAME_SIZE)

stream.start_stream()
  
```

รูปที่ 10 โค้ดในการเปิดไมค์รับเสียงแบบเรียลไทม์

โดยใช้ PyAudio ซึ่งเป็นโมดูลในการบันทึกเสียงหรือเล่นเสียงสำหรับภาษาไพธอน ในส่วนนี้จะเป็นส่วนของการบันทึกเสียง โดยเริ่มจากการกำหนดฟอร์แมต paInt16 เนื่องจากเสียงมีรูปแบบในการจัดเก็บเป็นเลขฐานสองและเพื่อให้โปรแกรมรู้ว่าจะเริ่มบันทึกและสิ้นสุดตรงไหนจึงต้องจำกัดไซส์โดยตั้งค่าเป็น 16 bit ในบรรทัดต่อมา channels คือการเลือกการรับเสียงเข้าผ่านไมโครโฟน โดยรับเป็นแบบ mono จึงใส่ค่าเป็น 1 และ rate คืออัตราในการสุ่มคลื่นความถี่ใน Hz โดยโปรแกรมจะกำหนดแยกส่วนไว้ด้านบนในตัวแปร FSAMP

= 22050 โดยปกติมักตั้งค่านี้อาไว้ที่ 44100 เพื่อให้ได้ความละเอียดสูงสุด แต่ในบทความนี้เมื่อตั้งค่าดังกล่าว โปรแกรมจะไม่สามารถรันได้ จึงลดเหลือเพียงแค่ 22050 ซึ่งเป็นความละเอียดที่เพียงพอในการใช้งาน

โดยในการทดลองนี้นำค่าที่ได้จากการรับเสียงเข้าแบบเรียลไทม์ใส่เข้าเป็นข้อมูลในการทำนายผลของ CREPE model เพื่อทำนายความถี่ในแต่ละวินาทีพร้อมกับค่า confidence และค่า activation ดังโค้ดในรูปที่ 11

```
data = np.fromstring(stream.read(CHUNK, exception_on_overflow = False), dtype=np.int16)
time, frequency, confidence, activation = crepe.predict(data, RATE, viterbi=True)
```

รูปที่ 11 การนำข้อมูลเข้าสู่โมเดล

จะได้ผลลัพธ์ดังนี้ คือ วินาที ความถี่ในวินาทีนั้น ๆ และค่า confidence โดยได้เอาต์พุตออกมาเป็นไฟล์ .csv ที่มีข้อมูลดังตารางที่ 1

ตารางที่ 1 เอาต์พุตที่ได้จากการนำไฟล์เสียงเข้า CREPE Model

Time	frequencies	confidence
0.010	551.501	0.296300
0.020	32.685	0.160744
0.030	195.367	0.139968
0.040	208.385	0.112159
:	:	:
6.020	49.508	0.053884
6.030	52.223	0.058571

จากตารางที่ 1 แสดงข้อมูลที่ได้จากการนำไฟล์เสียงเข้าสู่โมเดลเพื่อทำนายคลื่นความถี่ซึ่งได้ค่าออกมาเป็นความถี่และค่าความเชื่อมั่นในแต่ละวินาที โดยในตารางจะแสดงตัวอย่างเพียงวินาทีที่เริ่มต้นและวินาทีสุดท้ายของเพลง ซึ่งไฟล์เอาต์พุตที่ได้จะมีขนาดที่แตกต่างกันออกไปตามไฟล์เสียงที่ใส่เข้าสู่โมเดล

แต่เนื่องจาก Crepe Model ทำนายค่าออกมาเป็นเพียงความถี่เท่านั้น ผู้พัฒนาจึงแปลงค่าความถี่ที่ได้ให้อยู่ในรูปแบบของ midi numbers เพื่อความสะดวกในการเปรียบเทียบ และเนื่องจากการทดลองนี้มีผลลัพธ์ที่ประกอบไปด้วยค่าความเชื่อมั่น ซึ่งสามารถเขียนเงื่อนไขว่าจะพิจารณาความถี่ในวินาทีนั้น ๆ ก็ต่อเมื่อมีค่าความเชื่อมั่นที่สูงกว่า 0.7 ขึ้นไป ดังรูปที่ 12

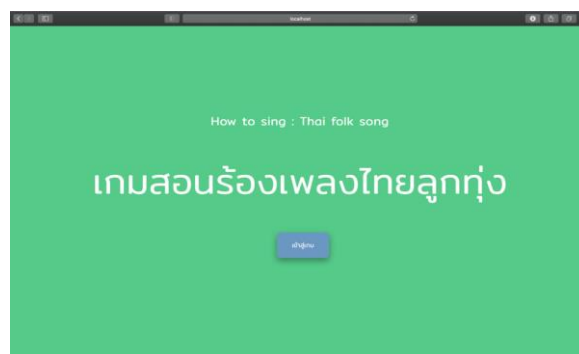
```
with open('sing.csv') as csvfile:
    reader = csv.reader(csvfile)
    for row in reader:
        if float(row[2]) > 0.7:
            ff = 69 + 12*np.log2(float(row[1])/440.0)
            midi = int(ff)
            a.write(str(midi)+ ' ')
        else:
            a.write(' ')
```

รูปที่ 12 โค้ดในการแปลงความถี่เป็น midi numbers และการตรวจสอบค่าความเชื่อมั่น

4 ผลการดำเนินงาน

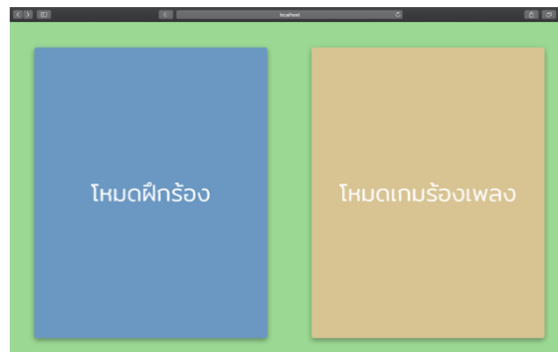
4.1 ผลการทำงานของระบบ

จากการออกแบบและพัฒนาจะได้เว็บไซต์ที่ประกอบไปด้วยหน้าจอการทำงานต่าง ๆ โดยเมื่อเข้าสู่เว็บไซต์จะพบกับหน้าจอหลัก ซึ่งประกอบไปด้วย ชื่อเว็บไซต์ภาษาอังกฤษ ภาษาไทย และปุ่มเข้าสู่เกม ดังรูปที่ 13



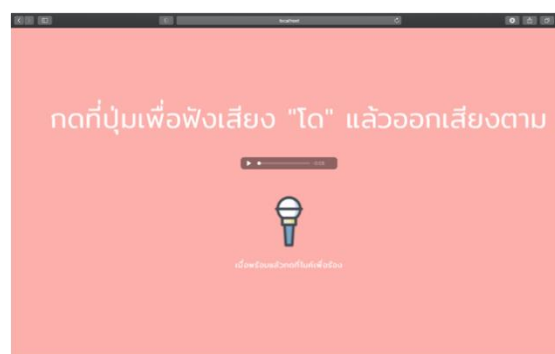
รูปที่ 13 หน้าจอหลักของเว็บไซต์

เมื่อผู้ใช้งานกดปุ่มเข้าสู่เกมในหน้าจอหลักของเว็บไซต์ จะแสดงหน้าจอถัดไปในส่วนของการเลือกโหมด โดยในหน้าจอสำหรับเลือกโหมดจะประกอบไปด้วยปุ่มสองปุ่ม คือ ปุ่ม “โหมดฝึกร้อง” และ ปุ่ม “โหมดเกมร้องเพลง” โดยใช้สีที่แตกต่างกัน ดังรูปที่ 14



รูปที่ 14 หน้าจอสำหรับเลือกโหมดเข้าใช้งาน

จากรูปที่ 14 ผู้ใช้งานกดปุ่มโหมดฝึกร้อง หน้าจอจะแสดงหน้าจอในโหมดฝึกร้อง ซึ่งจะประกอบไปด้วย คำอธิบายการใช้งานในโหมดฝึกร้อง ว่าผู้ใช้งานกำลังจะฟังเสียงไกด์และออกเสียงตามในโน้ตอะไร ซึ่งในแต่ละโน้ตจะมีสีพื้นหลังที่แตกต่างกันออกไป ในหน้าจอจะมีเสียงไกด์ให้ผู้ใช้งานกดฟังอยู่เหนือปุ่มรูปไมค์ ดังรูปที่ 15



เมื่อผู้ใช้งานกดที่ปุ่มรูปไมค์โปรแกรมจะนำผู้ใช้เข้าสู่หน้าจอออกเสียง โดยจะแสดงหน้าเวลานับถอยหลังสามวินาที

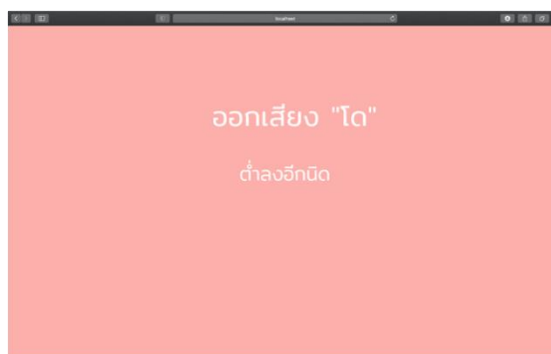
รูปที่ 15 หน้าจอแรกในโหมดฝึกร้อง

ก่อนเข้าสู่หน้าสำหรับออกเสียง ในรูปที่ 16



รูปที่ 16 หน้าจอแรกแสดงเวลานับถอยหลังก่อนเข้าสู่หน้าสำหรับออกเสียง

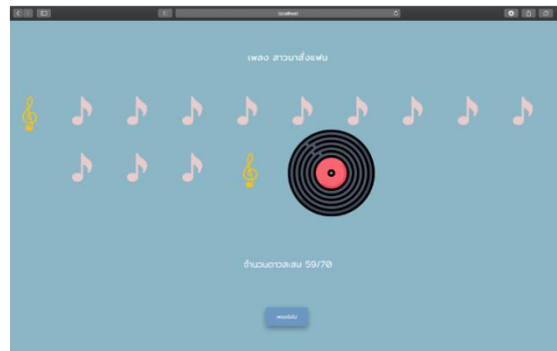
เมื่อสิ้นสุดเวลานับถอยหลังโปรแกรมจะแสดงหน้าจอสำหรับออกเสียงให้ผู้ใช้งานเริ่มออกเสียงตามจากเสียงไกด์ที่ฟัง ในรูปที่ 5.9 และในหน้าจอนี้จะแสดงคำว่า “สูงขึ้นอีกหน่อย” ในกรณีที่ผู้ใช้งานออกเสียงต่ำกว่าโน้ตที่กำหนดไว้ และคำว่า “ต่ำลงอีกนิด” เมื่อผู้ใช้งานออกเสียงสูงเกินไปกว่าโน้ตที่กำหนดไว้ ดังรูปที่ 17



รูปที่ 17 หน้าจอสำหรับออกเสียง

เมื่อผู้ใช้งานออกเสียงได้ถูกต้องตามโน้ตที่กำหนดไว้แล้ว โปรแกรมจะนำผู้ใช้เข้าสู่เสียงถัดไปทันที และผู้ใช้งานจะต้องออกเสียงให้ถูกต้องให้ครบทั้ง 7 ตัวโน้ต เพื่อเข้าสู่โหมดเกมร้องเพลงอัตโนมัติหรือสามารถกดเข้าใช้งานโหมดดังกล่าวได้จากหน้าเลือกโหมด

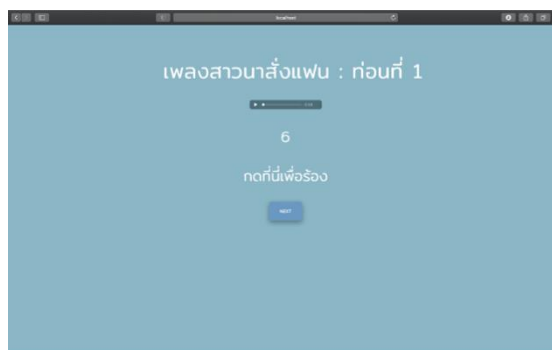
ในโหมดเกมร้องเพลง ผู้ใช้งานสามารถกดเลือกด้านรูปกุญแจซอลและตัวโน้ตเพื่อเข้าเล่นเกมในด่านนั้น ๆ ได้ดังรูปที่ 18



รูปที่ 18 โหมดเกมร้องเพลง

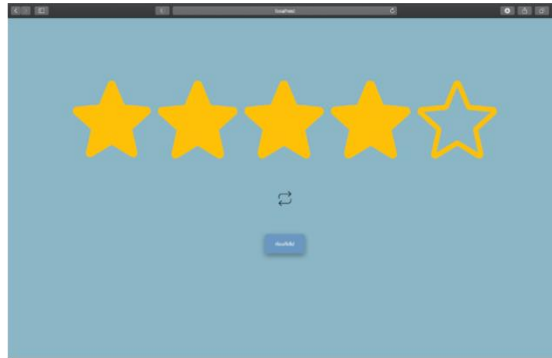
หน้านี้จะประกอบไปด้วยปุ่มกดรูปโน้ตและกุญแจซอลซึ่งหมายถึงค่าน้อย และปุ่มกดรูปแผ่นเสียงที่ผู้ใช้งานสามารถกดเพื่อเข้าดูจำนวนดาวสะสม และจำนวนดาวที่ได้ในแต่ละค่าน้อย

เมื่อผู้ใช้งานกดเลือกค่าน้อยโปรแกรมจะแสดง ค่าน้อยที่ผู้ใช้งานเลือก ซึ่งหน้าเว็บไซต์นั้นจะประกอบไปด้วย ชื่อเพลง ชื่อค่าน้อย เสียงไกด์ เวลาที่จะบันทึกเสียงร้องของผู้ใช้งานและปุ่มกดสำหรับบันทึก ดังรูปที่ 19



รูปที่ 19 ค่าน้อยที่ 1 ในโหมดเกมร้องเพลง

ในหน้าดังกล่าวเมื่อผู้ใช้งานกดปุ่ม “กดที่นี่เพื่อร้อง” ตัวเลขที่อยู่ใต้เสียงไกด์จะนับเวลาถอยหลังเพื่อแสดงว่ากำลังบันทึกเสียงของผู้ใช้ตามเวลาที่แสดงไว้ และเมื่อสิ้นสุดการบันทึกโปรแกรมจะคำนวณคะแนนและแสดงผลทางหน้าจอ ดังรูปที่ 20



รูปที่ 20 หน้าจอแสดงคะแนนด้านย่อย

ในรูปที่ 20 จะประกอบไปด้วย คะแนนที่ผู้ใช้งานได้รับ และปุ่มย้อนกลับเพื่อกลับไปเล่นด่านเดิมซ้ำอีกครั้ง หรือปุ่ม ท่อนัดไป เมื่อผู้ใช้งานเล่นเกมในด่านย่อยจนครบทุกด่านแล้ว ปุ่มท่อนัดไปจะเปลี่ยนเป็นปุ่ม คะแนนรวม เพื่อแสดงจำนวนดาวสะสมที่ทำได้ในด่านใหญ่ และจำนวนดาวที่ทำได้ในแต่ละด่านย่อย ดังรูปที่ 21



รูปที่ 21 หน้าจอแสดงคะแนนรวม

4.2 ผลการให้คะแนนความถูกต้องตรงกันของข้อมูลจากไฟล์สองไฟล์ โดยใช้ BLEU Score

BLEU เป็นอัลกอริทึมที่ใช้ในการเปรียบเทียบประโยคสองประโยคเข้าด้วยกัน โดยในบทความนี้ได้ใช้ BLEU ในการเปรียบเทียบเนื้อหาภายในไฟล์สองไฟล์ว่ามีความถูกต้องตรงกันหรือไม่ และคิดเป็น กี่เปอร์เซ็นต์ ดังตัวอย่างในตารางที่ 2

ตารางที่ 2 คะแนนในการเปรียบเทียบความแตกต่างของไฟล์สองไฟล์

ครั้งที่	Ref.txt	Hyp.txt	คะแนน
1	65 68 69 71 77 99 90 99 65	65 68 69 71 77 99 90 99 65	100.0
2	65 68 69 71 77 99 90 75 95	56 68 69 71 77 99 80 57 59	46.71

ด้วยการใช้ไลบรารี BLEU ในภาษาไพธอน ดังรูปที่ 21

```
@socketio.on('sfmgt1')
def sfmgt1(message):
    print(message)
    hyp_file = 'candidate.txt'
    ref_files = ['reffmgirl/reft1.txt']
    result = file_bleu(ref_files,hyp_file)
    emit('finish',result)
    socketio.sleep()
```

รูปที่ 21 การเทียบไฟล์สองไฟล์ด้วยไลบรารี BLEU

โดยนำไฟล์ .txt ที่ได้จากผู้ใช้งานใส่ลงในตัวแปร hyp_file เทียบกับไฟล์ต้นฉบับในตัวแปร ref_files โดยใช้คำสั่ง

```
result = file_bleu(ref_files,hyp_file)
```

โดยตัวแปร result จะเก็บผลลัพธ์ที่เป็นค่าความตรงกันของข้อมูลตั้งแต่ค่า 0 – 100 คะแนนดังรูปที่ 22

```
@socketio.on('sfmgt1')
def sfmgt1(message):
    print(message)
    hyp_file = 'candidate.txt'
    ref_files = ['reffmgirl/reft1.txt']
    result = file_bleu(ref_files,hyp_file)
    emit('finish',result)
    socketio.sleep()
```

รูปที่ 22 ผลลัพธ์ที่ได้จาก BLEU

4.3 ผลลัพธ์จากการทำ Usability Test

จากการทำแบบสอบถามหลังใช้งานเว็บไซต์ของผู้ใช้จำนวน 5 คน ได้ผลลัพธ์ดังตารางที่ 3

ตารางที่ 3 ผลลัพธ์จากการทำ Usability Test

	มากที่สุด (5)	มาก (4)	ปานกลาง (3)	น้อย (2)	น้อยที่สุด (1)	ค่าเฉลี่ย
1.ความง่าย ในการใช้ งาน	4.00	1.00	0.00	0.00	0.00	4.8
2.ความ สวยงาม ของเว็บไซต์	3.00	2.00	0.00	0.00	0.00	4.6
3.ความ เหมาะสม ของ ตัวหนังสือ บนเว็บไซต์	2.00	3.00	0.00	0.00	0.00	4.4
4.ระดับ ความยาก ง่ายของเกม	1.00	3.00	1.00	0.00	0.00	4.4
5.ระดับ ความพึง พอใจต่อ เว็บไซต์	4.00	1.00	0.00	0.00	0.00	4.8
เฉลี่ย	2.8	2.0	0.2	0.0	0.0	5

5. สรุปและอภิปราย

บทความนี้นำเสนอเว็บแอปพลิเคชันเพื่อการฝึกร้องเพลงไทยลูกทุ่ง ที่ช่วยให้ผู้ที่สนใจฝึกร้องเพลงไทยลูกทุ่งสามารถเรียนรู้ได้ โดยผู้ใช้งานจะได้รับการฝึกออกเสียงตัวโน้ตแต่ละตัวตามเสียงไกด์ ซึ่งสามารถฟังและฝึกออกเสียงตามได้ง่ายกว่าการออกเสียงตามเครื่องดนตรีในกรณีที่ใช้ไม่มีพื้นฐานในการร้องเพลงมาก่อน และมีเกมในโหมดเกมร้องเพลง ซึ่งจะทำให้ผู้ใช้งานสามารถฝึกฝนและร้องเพลงจากระดับง่ายไปยาก ตามด่านที่โปรแกรมระบุไว้จำนวนสามเพลง และสามารถประเมินระดับความถูกต้องของเสียงร้องจากผู้ใช้งานเทียบกับเสียงร้องต้นฉบับในแต่ละด่านได้

แนวทางพัฒนาในอนาคตสามารถพัฒนาให้มีการแสดงตัวโน้ตแบบเรียลไทม์ในขณะที่ผู้ใช้งานร้องเพลงในโหมดเกมร้องเพลง และสามารถเพิ่มเพลงเข้าไปให้หลากหลายมากยิ่งขึ้น

6. เอกสารอ้างอิง

- [1] F. P. L. F. Sound, "คลื่นเสียงประกอบด้วยอะไรบ้าง," [Online]. Available: <https://www.facebook.com/874521629240999/posts/1128702130489613/>. [Accessed 1 มิถุนายน 2563].
- [2] J. Koatsombat, "ทฤษฎีดนตรี (Music Theory)," [Online]. Available: <https://sites.google.com/site/mrjisclassroom/thvsdi-not-khan-phun-than/-note-reading/-pitch>. [Accessed 1 มิถุนายน 2563].
- [3] J. Wolfe, "Note names, MIDI numbers and frequencies," [Online]. Available: <https://newt.phys.unsw.edu.au/jw/notes.html>. [Accessed 1 มิถุนายน 2563].
- [4] R. Khandelwal, "Toward Data Science," [Online]. Available: <https://towardsdatascience.com/bleu-bilingual-evaluation-understudy-2b4eab9bdfd1?fbclid=IwAR3rmPtOdBDiNrHVZT2weOYlmvxviLLZLvG4ChIPCdyjddH8pCE1FrXzTe4>.
- [5] hubert, "PyAudio," [Online]. Available: <http://people.csail.mit.edu/hubert/pyaudio/>.
- [6] F. Research, "Github," [Online]. Available: <https://github.com/facebookresearch/demucs>.
- [7] MC.AI, "มารู้จัก Transformer กันเถอะ(ตอนที่ 1)," [Online]. Available: <https://mc.ai/มารู้จัก-transformer-กันเถอะ-ตอนที่1> . [Accessed 23 ตุลาคม 2563].