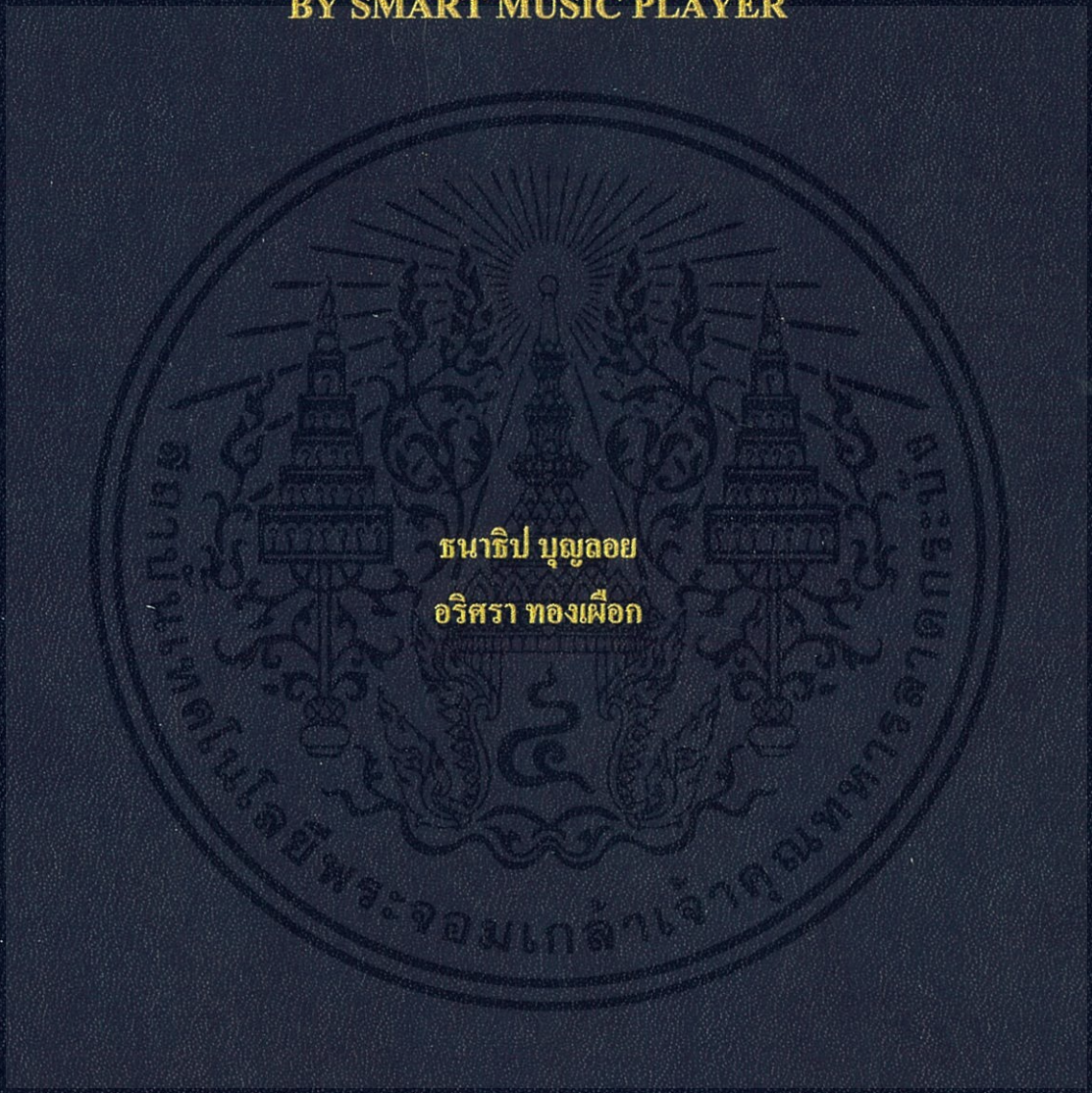
การทำนายแนวเพลงที่ชื่นชอบของผู้ใช้งาน โดยโปรแกรมเล่นเพลงอัจฉริยะ

PREDICTING USER'S FAVORITE MUSICAL STYLE BY SMART MUSIC PLAYER



ธนาธิป บุญลอย

อริศรา ทองเผือก

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

การทำนายแนวเพลงที่ชื่นชอบของผู้ใช้งาน
โดยโปรแกรมเล่นเพลงอันชาญฉลาด

PREDICTING USER'S FAVORITE MUSICAL STYLE
BY SMART MUSIC PLAYER

ธนาธิป บุญลอย

อริศรา ทองเผือก

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2559

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การทำนายแนวเพลงที่ชื่นชอบของผู้ใช้งานโดยโปรแกรมเล่นเพลงอันชาญฉลาด

PREDICTING USER'S FAVORITE MUSICAL STYLE BY SMART MUSIC PLAYER

ผู้จัดทำ

1. นายธนาธิป บุญลอย รหัสนักศึกษา 56010572

2. นางสาวอริศรา ทองเผือก รหัสนักศึกษา 56010654

_____ อาจารย์ที่ปรึกษา

(อาจารย์เกียรติกูล เจียรนัยธนะกิจ)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำนายแนวเพลงที่ชื่นชอบของผู้ใช้งาน โดยโปรแกรมเล่นเพลงอันชาญฉลาด

นาย ธนาธิป บุญลอย 56010572
นางสาว อริศรา ทองเผือก 56010654
รศ.ดร. เกียรติกูล เจียรนัยธนะกิจ อาจารย์ที่ปรึกษา
ปีการศึกษา 2559

บทคัดย่อ

ปัจจุบันระบบปัญญาประดิษฐ์คืออีกส่วนหนึ่งของนวัตกรรมที่มีการพัฒนากันมาอย่างต่อเนื่อง และมีการนำไปประยุกต์ใช้กับเทคโนโลยีต่างๆมากมาย ซึ่งการใช้ระบบปัญญาประดิษฐ์ในการเรียนรู้พฤติกรรมของผู้ใช้งาน เพื่อนำข้อมูลที่ได้ไปประยุกต์ คัดสรร และเลือกสรรข้อมูลต่างๆ ให้ตรงกับความต้องการของผู้ใช้งาน ก็เป็นอีกส่วนหนึ่งที่มีการใช้งานกันมากในปัจจุบัน ปริญญานิพนธ์ฉบับนี้ได้นำระบบปัญญาประดิษฐ์มาประยุกต์ใช้กับแอพพลิเคชันที่ใช้สำหรับเล่นเพลง เพื่อเรียนรู้พฤติกรรมการฟังเพลงและแนวความชอบของผู้ใช้งาน แล้วนำเสนอเพลงใหม่ๆ ที่ตรงตามความต้องการให้กับผู้ใช้งาน โดยในระบบนี้นั้น เป็นแอพพลิเคชันสำหรับเล่นเพลงที่อยู่บนระบบปฏิบัติการแอนดรอยด์ มีการใช้ระบบปัญญาประดิษฐ์ที่จะเรียนรู้พฤติกรรมของผู้ใช้งานแต่ละคนซึ่งแตกต่างกันไปด้วยโครงข่ายประสาทเทียม (Artificial Neural Network) แล้วนำข้อมูลที่ได้ไปสร้างโมเดลเพื่อทำนายความชอบผู้ใช้งาน และค้นหาเพลงที่คาดว่าตรงกับความชอบของผู้ใช้งาน แล้วแนะนำเพลงนั้นๆให้กับผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PREDICTING USER'S FAVORITE MUSICAL STYLE BY SMART MUSIC PLAYER

Mr. Tanatip	Boonloy	56010572
Miss. Arisara	Thongpuek	56010654
Assoc.Prof. Dr.	Kietikul Jearanaitanakij	Advisor

Academic Year 2016

ABSTRACT

Nowadays, Artificial Intelligence (AI) has involved in many applications with continuous develops and improvements. It has been used in many fields to reduce workload and save time. AI can make a prediction on most difficult problems by learning behavior from the data set and making models to solve them. This project applies AI to an android application for inventing a music mobile application that can predict and suggest a music which tends to be favored by the user.

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ จะไม่เสร็จสมบูรณ์ได้ หากปราศจากการช่วยเหลือของ อาจารย์ที่ปรึกษา รศ.ดร.เกียรติกูล เจียรนัยธนะกิจ ซึ่งคอยให้คำปรึกษา , ความรู้และการแนะแนวทางปริญญานิพนธ์ให้สามารถพัฒนาจนเสร็จสมบูรณ์ได้ ทางคณะผู้จัดทำขอขอบพระคุณเป็นอย่างสูง

ขอขอบคุณ ห้องวิจัย ISAG ที่เอื้อเฟื้อสถานที่ในการทำปริญญานิพนธ์

ขอขอบคุณอาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่าน ที่ช่วยประสิทธิ์ประสาทวิชาจนสามารถนำมาประยุกต์ใช้กับปริญญานิพนธ์ฉบับนี้ได้

ขอขอบคุณบิดา มารดา ที่คอยสนับสนุนในด้านต่างๆ สำหรับการทำปริญญานิพนธ์

ขอขอบคุณนายนพกร ใช้บุญเรือง, รุ่นพี่, เพื่อนๆ ที่คอยช่วยแหลือ ให้ข้อมูลและคำแนะนำต่างๆ เพื่อใช้ในการทำปริญญานิพนธ์

สุดท้ายนี้ทางคณะผู้จัดทำหวังเป็นอย่างยิ่งว่าปริญญานิพนธ์ฉบับนี้จะเป็นประโยชน์ต่อผู้อ่านไม่มากก็น้อย

ธนาธิป บุญลอย

อริศรา ทองเผือก

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ I

ABSTRACT II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญรูป VI

สารบัญตาราง VII

บทที่ 1 บทนำ 1

1.1 ความสำคัญและที่มาของโครงงาน 1

1.2 วัตถุประสงค์ 1

1.3 ขอบเขตของโครงงาน 1

1.4 วิธีการดำเนินงาน 2

1.5 ประโยชน์ที่ได้รับ 2

1.6 ส่วนประกอบของรายงาน 3

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 4

2.1 โครงข่ายประสาทเทียม 4

2.2 ระบบปฏิบัติการแอนดรอยด์ 6

2.3 แอตทริบิวต์ที่ได้จากเพลง 7

2.4 Spring Framework 9

2.5 Java Programming Language 10

2.6 Integrated Development Environment 11

2.7 RESTful Web Service 13

2.8 Favor Level 13

2.9 Amazon Web Services 14

2.10 Docker 16

2.11 Streaming Service 16

2.12 Content Delivery Network (CDN) 17

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

บทที่ 3 การออกแบบและการพัฒนา 18
3.1 ภาพรวมของระบบ 18
3.2 ผังงานของระบบฝั่งแอพพลิเคชันแอนดรอยด์ 20
3.3 ผังงานของระบบฝั่งเซิฟเวอร์ 22
3.4 โครงสร้างของโครงข่ายประสาทเทียม 24
3.5 ออกแบบฐานเก็บข้อมูล 25

บทที่ 4 การประเมินและใช้งานระบบ 26
4.1 การกำหนดข้อมูลที่ใช้ในการเทรนของ Artificial Neural Network 26
4.2 การทดลองใช้แอพลิเคชั่น Smart Music Player บนระบบปฏิบัติการแอนดรอยด์ 26
4.3 การทดลองการหาจำนวน Node ของ Hidden Layer ใน Artificial Neural Network 32
4.4 การเทรนด้วย Artificial Neural Network 33
4.5 ผลการทดลอง 34

บทที่ 5 บทสรุป 37
5.1 บทสรุปของโครงงาน 37
5.2 ผลที่ได้จากการทำโครงงาน 37
5.3 ปัญหาและอุปสรรคที่เกิดจากการทำโครงงาน 37
5.4 แนวทางในการพัฒนาต่อ 38
บรรณานุกรม 39

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป หน้า

2.1 โครงข่ายเซลล์ประสาท 4
2.2 ภาพไดอะแกรมของ Neuron....5
2.3 ตัวอย่างการประกอบกันของ Neuron จนเป็นโครงข่ายประสาทเทียม....5
2.4 โลโก้แอนดรอยด์....6
2.5 การแบ่งเสียงในการฟังออกเป็นช่วงๆ 8
2.6 โลโก้สปริงเฟรมเวิร์ค 9
2.7 โลโก้ภาษาจาวา 10
2.8 โลโก้ Android studio....11
2.9 โลโก้ IntelliJ....12
2.10 โลโก้ Amazon Web Services....14
2.11 โลโก้ Docker....16
3.1 ภาพรวมของระบบ 18
3.2 ผังงานระบบฝั่งแอพพลิเคชัน....20
3.3 แผนผังการทำงานฝั่งเซิฟเวอร์ 1 22
3.4 แผนผังการทำงานฝั่งเซิฟเวอร์ 2 (ต่อจากรูป 3.3)....23
3.5 โครงสร้างโครงข่ายประสาทเทียม....24
3.6 แบบจำลองแสดงการออกแบบฐานข้อมูล 25
4.1 หน้าเลือก Login....27
4.2 หน้าระบบ Register 27
4.3 หน้าระบบ Login 28
4.4 หน้าแสดงรายการเพลงทั้งหมดและส่วนการแนะนำเพลงจากระบบ....28
4.5 หน้าควบคุมการเล่นเพลง โดยมีสามารถ กดเล่น, พัก, เพลงถัดไป 29
4.6 กดเพิ่มเพลงไปที่ Playlist....29
4.7 หน้าแสดง Playlist ทั้งหมด....30
4.8 สร้าง Playlist ใหม่ 30
4.9 หน้าจัดการ Playlist 31
4.10 หน้า Seach....31
4.11 ตัวอย่างโค้ด Artificial Neural Network ภาษา Python 34

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง หน้า

2.1 Favor Level 2 ระดับ 14
4.1 ใช้วิธี Decreasing Learning Rate 32
4.2 ใช้วิธีไม่ใช้ Decreasing Learning Rate 33
4.3 Favor Level 2 ระดับ 34
4.4 ผลการทดลองเมื่อกำหนดค่า Output เป็น 2 ค่า 35
4.5 Favor Level 4 ระดับ 35
4.6 ผลการทดลองเมื่อกำหนดค่า Output เป็น 4 ค่า 36

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของโครงงาน

ปัจจุบันข้อมูลข่าวสารที่เกิดขึ้นในแต่ละวันมีจำนวนมหาศาล การเลือกสรรข้อมูลให้ได้ตามความต้องการจึงจำเป็นต้องใช้เวลานาน โดยเพลงก็เป็นข้อมูลอีกชนิดหนึ่งที่เกิดขึ้นตลอดเวลาซึ่งต้องใช้เวลา หากต้องการที่จะหาเพลงที่ต้องการตามแนวเพลงที่สนใจจากเพลงจำนวนมหาศาลที่มีอยู่ในปัจจุบัน ซึ่งระบบปัญญาประดิษฐ์ที่เป็นระบบที่มีการพัฒนามาอย่างต่อเนื่อง และในปัจจุบันก็ได้มีการประยุกต์ใช้กับระบบอัจฉริยะต่างๆ มากมาย เพื่อช่วยแก้ปัญหาในด้านต่างๆ รวมถึงช่วยในการกรองข้อมูล เพื่อให้ได้ข้อมูลที่ต้องการ ลดระยะเวลาที่จำเป็นต้องใช้ในการเลือกสรรข้อมูลลงไป

พฤติกรรมการใช้งานสมาร์ทโฟนในปัจจุบันก็มีแนวโน้มการใช้งานในอัตราที่สูงขึ้นเรื่อยๆ โดยมีอัตราการใช้งานที่สูงกว่าการใช้อุปกรณ์อื่นๆ การพัฒนาแอพพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ซึ่งเป็นระบบปฏิบัติการที่เป็นที่นิยมบนสมาร์ทโฟน ให้เป็นแอพพลิเคชั่นที่ใช้สำหรับเล่นเพลงและมีความสามารถในการเรียนรู้ความชอบ และพฤติกรรมการฟังเพลงของผู้ใช้งาน แล้วสามารถแนะนำเพลงใหม่ๆ ที่ตรงตามแนวความชอบของผู้ใช้งานจึงเป็นอีกสิ่งหนึ่งที่น่าสนใจ

1.2 วัตถุประสงค์

1) สามารถสร้างแอพพลิเคชันสำหรับเล่นเพลงบนระบบปฏิบัติการแอนดรอยด์ได้
2) สามารถใช้ระบบโครงข่ายประสาทเทียมในการเรียนรู้ความชอบของผู้ใช้งาน และทำนายความชอบของผู้ใช้งานได้
3) สามารถนำระบบปัญญาประดิษฐ์ที่ได้เข้ามาประยุกต์ใช้กับแอพพลิเคชันเล่นเพลงได้

1.3 ขอบเขตของโครงงาน

1) Smart Music Player สามารถใช้ได้ในระบบปฏิบัติการแอนดรอยด์เท่านั้น
2) เพลงที่ใช้งานจะต้องมีแอตทริบิวต์ที่กำหนดไว้ครบถ้วน
3) เพลงที่ใช้งานจะต้องเป็นเพลงสากลซึ่งมีความนิยมในระดับหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 วิธีการดำเนินงาน

1) ออกแบบและกำหนดขอบเขตของโครงงาน
2) ศึกษาเกี่ยวกับโครงข่ายประสาทเทียม (Artificial Neural Network)
3) ศึกษาและค้นหาแอตทริบิวต์ต่างๆ ที่เกี่ยวข้องกับเพลง
4) ศึกษาและพัฒนาการสร้างแอพพลิเคชันบนระบบปฏิบัติการแอนดรอยด์
5) ออกแบบและพัฒนาการสร้างโครงข่ายประสาทเทียมที่ใช้ทำนายแนวเพลงที่ผู้ใช้งานชอบ
6) ศึกษาและพัฒนาการสร้างเซิฟเวอร์ที่ติดต่อกับ Smart Music Player
7) ทำการทดลองกับผู้รับการทดลองพร้อมบันทึกผล
8) ปรับปรุงความแม่นยำในการทำนายเพลงที่ผู้ใช้ชอบ
9) ตรวจสอบและปรับปรุงข้อผิดพลาดต่างๆ ของแอพพลิเคชัน

1.5 ประโยชน์ที่ได้รับ

1) ความรู้และความสามารถในการพัฒนาแอพพลิเคชันบนระบบปฏิบัติการแอนดรอยด์
2) ความรู้และความสามารถในการออกแบบและทำการทดลองโดยใช้โครงข่ายประสาทเทียม
3) สามารถสร้างแอพพลิเคชันสำหรับเล่นเพลงบนระบบปฏิบัติการแอนดรอยด์ที่มีการประยุกต์ใช้โครงข่ายประสาทเทียมเพื่อทำนายเพลงที่ผู้ใช้งานชอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 ส่วนประกอบของรายงาน

รายงานเล่มนี้ประกอบด้วยส่วนประกอบ 5 ส่วน คือ

บทที่ 1 บทนำ กล่าวถึง ความเป็นมา วัตถุประสงค์ ขอบเขตการดำเนินงานของโครงงาน วิธีการดำเนินงาน ผลประโยชน์ที่คาดว่าจะได้รับ และส่วนประกอบของรายงาน

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง กล่าวถึง ทฤษฎีพื้นฐานที่ใช้ในการทำโครงงานประกอบด้วยโครงข่ายประสาทเทียม, ระบบปฏิบัติการแอนดรอยด์, คุณลักษณะที่ได้จากเพลง, Spring Framework, Java Programming Language, IDE (Integrated Development Environment), RESTful Web Service, Favor Level, Amazon Web Services, Docker, Streaming Service, Content Delivery Network

บทที่ 3 การออกแบบและพัฒนา กล่าวถึง ภาพรวมของระบบ, ผังงานของระบบฝั่งแอพพลิเคชันแอนดรอยด์, ผังงานของระบบฝั่งเซิฟเวอร์, โครงสร้างโครงข่ายประสาทเทียม, การออกแบบฐานข้อมูล

บทที่ 4 การทดลองและผลการทดลอง กล่างถึง การกำหนดข้อมูลที่ใช้ในการเทรนของ Artificial Neural Network, การทดลองใช้แอพลิเคชั่น Smart Music Player, การทดลองหาจำนวน Node ของ Hidden Layer ใน Artificial Neural Network, การเทรนด้วย Artificial Neural Network และ ผลการทดลอง

บทที่ 5 บทสรุป กล่าวถึง บทสรุปของโครงงาน ปัญหาและอุปสรรคต่างๆ ของโครงงาน แนวทางการแก้ไขและพัฒนาต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

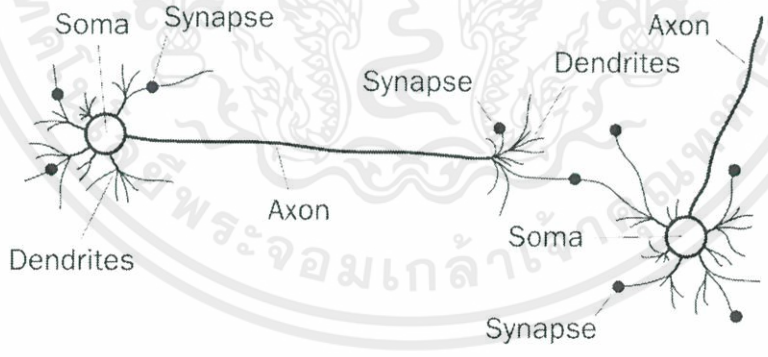
2.1 โครงข่ายประสาทเทียม

ระบบประสาทสามารถกำหนดเป็นรูปจำลองได้โดยอ้างอิงจากสมองของมนุษย์ ซึ่งสมองของมนุษย์นั้นประกอบไปด้วยเซตของเซลล์ประสาทที่เชื่อมต่อกัน หรือหน่วยการประมวลผลข้อมูลแบบง่าย ถูกเรียกว่า เซลล์ประสาท (Neurons) สมองของมนุษย์นั้นมีเซลล์ประสาทประมาณ 10 พันล้านเซลล์ และ มีการเชื่อมต่อ 60 ล้านล้านการเชื่อมต่อ โดยจุดเชื่อมต่อของแต่ละเซลล์ประสาทถูกเรียกว่า จุดประสานประสาท (Synapses)

ถึงแม้ว่าในแต่ละเซลล์ประสาทจะมีโครงสร้างอย่างง่าย แต่ในแต่ละองค์ประกอบถือว่าเป็นพลังในการประมวลผลขนาดใหญ่ เซลล์ประสาทมีส่วนประกอบต่างๆดังนี้

1) โซมา (Soma) เป็นจุดศูนย์กลางเซลล์
2) เดนไดรท์ (Dendrites) เป็นแขนงของเซลล์ประสาทซึ่งมีหลายแขนง มีลักษณะเป็นรากไม้
3) แอกซอน (Axon) เป็นแขนงเส้นยาวเส้นเดียวที่ออกมาจากตัวเซลล์

เดนไดรท์จะแตกแขนงเซลล์เป็นเครือข่ายรอบๆ โซมา และแอกซอนจะยืดไปหาเดรนไดรท์และโซมาของเซลล์ประสาทอื่น ตามรูป 2.1



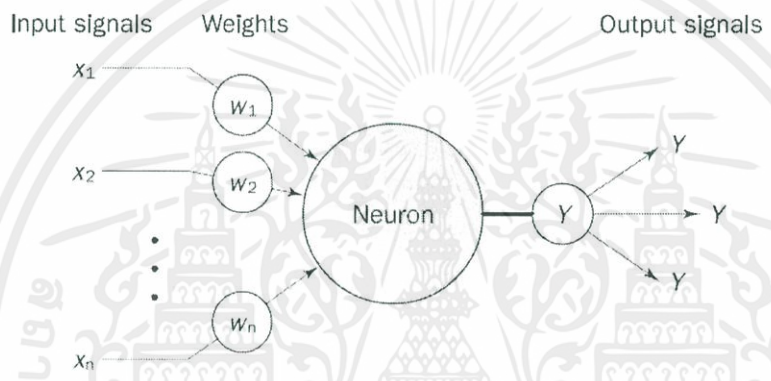
รูป 2.1 โครงข่ายเซลล์ประสาท

ในการส่งสัญญาณหากันของเซลล์ประสาท เซลล์จะใช้ปฏิกิริยาเคมีไฟฟ้า กล่าวคือ สารเคมีถูกปล่อยจากจุดประสานประสาท เพราะการเปลี่ยนแปลงศักย์ไฟฟ้าในเซลล์ เมื่อค่าศักย์ไฟฟ้าถึงจุดที่กำหนดไว้ จะเกิด กระแสประสาทขึ้น (action potential) ซึ่งจะถูกส่งผ่านแอกซอน เมื่อส่งไปถึงจุดปลายประสาท จะทำให้เกิดการเพิ่มหรือลดศักย์ไฟฟ้าของโครงข่ายประสาท

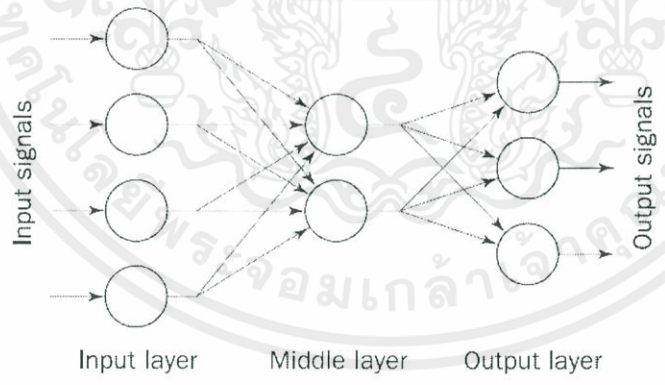
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปแบบการกระตุ้นที่กล่าวข้างต้น เซลล์ประสาทแสดงให้เห็นถึงจุดเด่นของการเชื่อมต่อในรูปแบบนี้ เซลล์ประสาทสามารถเกิดการเชื่อมต่อใหม่ๆกับเซลล์ประสาทอื่นๆ แม้ว่ากลุ่มเซลล์ประสาทนี้บางครั้งต้องย้ายจากที่หนึ่งไปอีกที่หนึ่ง เป็นรูปแบบกลไกของการเรียนรู้ของสมอง

ในโครงข่ายประสาทเทียม (Artificial Neural Networks) ประกอบไปด้วยจำนวนของหน่วยประมวลผลที่เชื่อมต่อกันในรูปอย่างง่าย ถูกเรียกว่า neurons ซึ่งคล้ายคลึงกับเซลล์ประสาทของสมองมนุษย์จริงๆ โดย neuron แต่ละตัว ประกอบด้วย input, weight, neuron และ output เปรียบได้กับ เดนไดร์ท, จุดประสานประสาท, โซมา และแอกซอน ของเซลล์ประสาทของมนุษย์ ตามรูปที่ 2.2 และ neurons ที่ประกอบกันจนเป็นโครงข่ายประสาทเทียมดังรูปที่ 2.3



รูป 2.2 ภาพไดอะแกรมของ Neuron



รูป 2.3 ตัวอย่างการประกอบกันของ Neuron จนเป็นโครงข่ายประสาทเทียม

ในปัจจุบันโครงข่ายประสาทเทียมเป็นปัญญาประดิษฐ์ที่มีโครงสร้างคล้ายคลึงกับโครงข่ายประสาทของมนุษย์ สามารถที่จะเรียนรู้ได้ และใช้ประสบการณ์ในการพัฒนาประสิทธิภาพตัวเอง เมื่อให้โครงข่ายประสาทเทียมเรียนรู้กลุ่มตัวอย่างที่เพียงพอแล้ว ทำให้สามารถแสดงความคิดเห็นเกี่ยวกับสิ่งที่ยังไม่เคยพบเห็นได้ ตัวอย่างในการใช้งานคือ สามารถจำแนกลายมือและคำพูดของมนุษย์ อีกทั้งสามารถตรวจสอบวัตถุระเบิดที่สนามบินได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ระบบปฏิบัติการแอนดรอยด์

รูป 2.4 โลโก้แอนดรอยด์

แอนดรอยด์เป็นระบบปฏิบัติการที่มีพื้นฐานมาจากระบบปฏิบัติการลินุกซ์ (Linux) พัฒนาโดยบริษัทกูเกิล (Google) ออกแบบมาสำหรับอุปกรณ์ที่ใช้จอสัมผัส เช่น สมาร์ทโฟน หรือ แท็บเล็ต สมาร์ทโฟนที่เป็นระบบปฏิบัติการแอนดรอยด์นั้น มีอุปกรณ์หลากหลายให้เลือก กล่าวคือ มีความหลากหลายทั้งทางด้านราคา คุณภาพ คุณสมบัติด้านความสามารถของอุปกรณ์ให้เลือกสรร สามารถเข้าถึงผู้บริโภคได้ง่ายกว่า เนื่องจากมีราคาหลายระดับ ความสามารถของสมาร์ทโฟน จะขึ้นตามระดับราคานั้นด้วย ข้อดีของการใช้ระบบปฏิบัติการแอนดรอยด์ มีดังนี้

1) มีตัวเลือกหลากหลายให้เลือก ตามความเหมาะสมของการเลือกใช้ และยังมีราคาที่ถูกหากเทียบกับความสามารถของอุปกรณ์ที่ได้
2) ในการพัฒนาแอพพลิเคชันในระบบปฏิบัติการแอนดรอยด์ใช้ภาษา Java ในการพัฒนาซึ่งง่ายต่อการติดตั้งบนระบบปฏิบัติการอื่นๆ เช่น อุปกรณ์ระบบปฏิบัติการ Windows10 เป็นต้น
3) มี IDE ที่ช่วยในการพัฒนาแอพพลิเคชันให้ง่ายมากขึ้น คือ โปรแกรม Android Studio ช่วยในเรื่องของการดีไซน์และ การพัฒนาทำให้ประสิทธิภาพและความเร็วในการทำงานสำหรับนักพัฒนาเพิ่มมากขึ้น
4) มีการเติบโตในตลาดที่สูง และมีผู้ใช้ระบบปฏิบัติการมือถือแอนดรอยด์ เป็นจำนวนมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 แอตทริบิวต์ที่ได้จากเพลง

ในการทำนายเพลงที่ผู้ใช้ชื่นชอบ ต้องมีการเก็บแอตทริบิวต์ของเพลงแต่ละเพลง เพื่อเก็บข้อมูลของเพลงมาวิเคราะห์ในโครงข่ายประสาทเทียม ซึ่งทางผู้จัดทำได้ทำการศึกษาแอตทริบิวต์แต่ละตัว ซึ่งสามารถนำมาหาความแตกต่างของแต่ละเพลงได้ ดังนี้

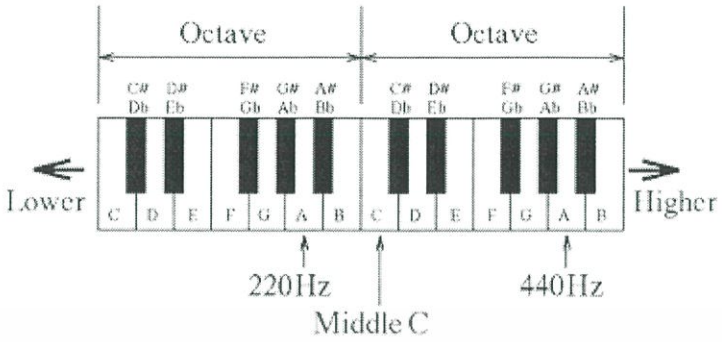
2.3.1 Genre คือประเภทของเพลง จะจำแนกตามสไตล์เพลง, ประเภทเครื่องดนตรี, เทคนิค, เนื้อหาของเพลง, จังหวะของเพลง, ประวัติหรือความสลักสำคัญตามภูมิภาคซึ่งมาจากวัฒนธรรมต่างๆ โดยประเภทของเพลงที่ผู้จัดทำได้เลือกมาเป็นตัวแปร มีการกำหนดประเภทเพลงดังนี้

1) Alternative
2) Pop
3) Rock
4) Acoustic
5) Hip Hop
6) R&B

2.3.2 Key คือเสียงคีย์ที่อยู่ในเพลง โดยมีทั้งหมด 12 เสียง ดังนี้

1) C
2) C# หรือ Db
3) D
4) D# หรือ Eb
5) E
6) F
7) F# หรือ Gb
8) G
9) G# หรือ Ab
10) A
11) A# หรือ Bb
12) B

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.5 การแบ่งเสียงในการฟังออกเป็นช่วงๆ

2.3.3 Tempo คือ การก้าวหรือความเร็วของเพลงที่เล่น โดยประมาณจากจำนวนเสียงจังหวะ (Beat) ในเพลง ซึ่งมีหน่วยเป็น beats per minute (BPM)

2.3.4 Acousticness คือการวัดว่าเพลงนี้มีความเป็น Acoustic มากเท่าไร

2.3.5 Danceability คือการวัดว่าเพลงนี้เหมาะสำหรับการเต้นหรือไม่ ซึ่งจะอยู่บนพื้นฐานของการรวมกันขององค์ประกอบทางดนตรีและจังหวะ เสถียรภาพของจังหวะ ความชัดเจนของเสียง Beat และความสม่ำเสมอโดยรวม

2.3.6 Energy คือการวัดความรับรู้ของความเข้มเสียงและการฮึกเหิม โดยปกติแล้ว เพลงที่เป็นให้ความรู้สึกฮึกเหิม มีลักษณะเป็นเพลงเร็ว เสียงดัง และอึกทึก ยกตัวอย่างเช่น Death Metal เป็นเพลงที่มี Energy สูง ในขณะที่เพลงBach มีคะแนนน้อยในส่วนนี้
คุณลักษณะที่ช่วยสนับสนุนในการรับรู้ของ Energy คือ ช่วงของ Dynamic , การรับรู้ระดับเสียงและเสียงร้องหรือเสียงจากเครื่องดนตรี (Timbre)

2.3.7 Instrumentalness คือ การวัดว่าเพลงนั้นๆเป็นเพลงที่ไม่มีคำร้อง โดยในที่นี้ เสียง หรือคำว่า “Ooh” หรือ “Aah” จะถูกนับว่าเป็นเสียงเครื่องดนตรีชนิดหนึ่ง

2.3.8 Liveness คือ เป็นการตรวจจากเสียงเพลงว่ามีเสียงของผู้ชมหรือไม่ หากมีเสียงของผู้ชมสามารถกำหนดได้ว่าการบันทึกเพลงนี้เป็นการแสดงสด

2.3.9 Loudness คือ ความดังเสียงโดยรวม มีหน่วยเป็นเดซิเบล (dB) ค่าของ Loundness เป็นค่าเฉลี่ยของเพลง และมีประโยชน์ในการเปรียบเทียบกับความสัมพันธ์ความดังเสียงของเพลง ค่าความดังของเพลงเป็นคุณภาพของเสียงที่เป็นหลักจิตวิทยาและเกี่ยวพันกับร่างกายด้วย

2.3.10 Mode คือ การกำหนดว่าคีย์เสียงเป็น Major หรือ Minor ของเพลง

2.3.11 Speechiness คือ เป็นการตรวจจาก Track ว่าเป็นเสียงพูดหรือเปล่า ถ้าเสียงบันทึกไหนที่มีเสียงพูดเยอะจะเป็น recording (เช่น Talk Show, Audio Book, บทกวี) จะมีค่าเป็น 1.0 และ ถ้ามีค่ามากกว่า 0.66 มีความเป็นไปได้ว่าเป็นTrack เสียงพูด หากมีค่าระหว่าง 0.33 หรือ 0.66 อาจจะเป็นทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพลงหรือเสียงพูด เช่นกรณีเพลง Rap หากค่าน้อยกว่า 0.33 มีแนวโน้มว่า Track เป็นเพลงหรือเพลงบรรเลง

2.3.12 Time Signature คือ ค่าประมาณของ notational convention เพื่อระบุจำนวน beats ในแต่ละแถบ

2.3.13 Valence คือ ค่าประมาณการแสดงความหมายของเพลงว่าเป็นเพลง Positive หรือ Negative ซึ่งเพลงที่มีค่า valence สูงให้ความรู้สึก Positive มากกว่า เช่น รู้สึกมีความสุข ร่าเริง หรือเบิกบาน ในขณะที่เพลงที่มีค่า valence น้อย ให้ความรู้สึก Negative มากกว่า เช่น รู้สึกเศร้า หดหู่ หรือโกรธ เป็นต้น

2.3.14 Play Time คือ ค่าอัตราส่วนของระยะเวลาที่ผู้ใช้งานฟังเพลงนั้นๆ ต่อระยะเวลาทั้งหมดของเพลง ซึ่งใช้แสดงถึงอัตราความชอบหรือความถูกใจของผู้ใช้งานที่มีผลต่อเพลงนั้น

2.4 Spring Framework

รูป 2.6 โลโก้สปริงเฟรมเวิร์ค

เป็นเฟรมเวิร์คที่เอื้ออำนวยได้ครอบคลุมในการเขียนโปรแกรมและมีรูปแบบการตั้งค่าสำหรับการใช้งานภาษาJava ในแต่ละชนิดของแพลทฟอร์มการพัฒนาโปรแกรม องค์ประกอบสำคัญของสปริงคือการให้การสนับสนุนโครงสร้างพื้นฐานในระดับแอพพลิเคชัน สปริงทำให้ทีมพัฒนาสามารถให้ความใส่ใจในระดับตรรกะทางธุรกิจของแอพพลิเคชัน ไม่ต้องกังวลเกี่ยวกับเรื่องสภาพแวดล้อมในการพัฒนาโปรแกรมที่เฉพาะเจาะจง มีตัวอย่างฟังก์ชันการทำงานดังนี้

1) Dependency Injection ช่วยในการจัดการ Dependency ของคลาสต่างๆ ให้ง่ายมากยิ่งขึ้น
2) การเขียนโปรแกรมเชิง Oriented รวมถึงตัวจัดการในการติดต่อกันที่สปริงได้ประกาศไว้
3) Spring MVC web application และ RESTful web service framework
4) พื้นฐานการสนับสนุนสำหรับ JDBC, JPA, JMS

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 Java Programming Language

รูป 2.7 โลโก้ภาษาจาวา

เป็นภาษาในการเขียนโปรแกรมโดยใช้หลักการเชิงวัตถุ (Object-Oriented Programming) มีการพัฒนาเพื่อให้ใช้แทนภาษา C++ Java เป็นภาษา High level ทำให้Syntax มีความคล้ายคลึงกับภาษามนุษย์ จึงทำให้สามารถเขียนได้ง่าย มีความซับซ้อนน้อยกว่าภาษา C++ ในcompile time และ run time ภาษาจาวาจะมีการตรวจสอบข้อผิดพลาดทั้งสองขั้นตอน ง่ายต่อการ debug รวมทั้งมีฟังก์ชันการทำงานภายในที่ช่วยในการเขียนให้ง่ายขึ้น เช่น Garbage Collection เป็นต้น นอกจากนี้ยังมี IDE และ Library ต่างๆ สำหรับภาษาจาวาในการช่วยพัฒนาโปรแกรมต่างๆ ไม่ต้องเสียค่าใช้จ่าย และสะดวกง่ายดายในการนำ Library ต่างๆมาช่วยลดเวลาในการเขียนโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 Integrated Development Environment

รูป 2.8 โลโก้ Android studio

2.6.1 Android Studio

เป็น IDE(Integrated Development Environment)ของ Android official ซึ่งเป็นเครื่องมือที่ช่วยในการดีไซน์และการพัฒนาโดยเฉพาะสำหรับแอพพลิเคชันในระบบปฏิบัติการแอนดรอยด์ ซึ่งช่วยให้สามารถทำงานได้เร็วและมีประสิทธิภาพมากยิ่งขึ้น มีเครื่องมือที่สามารถกำหนดเองได้ใน ตัวอย่างเช่น

1) Code templates และ ตัวอย่าง application ช่วยให้ง่ายต่อการเข้าใจในการเขียนโค้ดและการออกแบบต่างๆ
2) Testing tools & Frameworks
3) Lintelligence
4) Layout Editor
5) APK Analyzer

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 InteliJ

รูป 2.9 โลโก้ IntelliJ

เป็น IDE (Integrated Development Environment) ที่ช่วยเพิ่มประสิทธิภาพในการพัฒนาโปรแกรมต่างๆ ด้วยประสิทธิภาพการวิเคราะห์โค้ด และเหมาะกับการทำงานพัฒนาโปรแกรมภาษา Java ยกตัวอย่างฟังก์ชันที่ช่วยในการทำงานดังนี้

1) ช่วยตรวจโค้ดเมื่อเกิดความผิดพลาด และขึ้น pop up เพื่อแสดงและเลือกเพื่อที่จะกระทำ และช่วยตรวจหาโค้ดที่ซ้ำได้
2) แสดงชื่อตัวแปรหรือฟังก์ชันขณะที่กำลังเขียนโค้ด เพื่อช่วยลดระยะเวลาในการเขียนลง
3) มีอินเตอร์เฟสสำหรับ Version control หลากหลาย เช่น Git , SVN เป็นต้น
4) มีเครื่องมือในการสร้าง ที่สนับสนุนหลายตัว เช่น Maven, Gradle, Ant เป็นต้น
5) มีเครื่องมือสนับสนุนในการทำแอพพลิเคชันฝั่งเซิฟเวอร์เช่น Tomcat, JBoss และอื่นๆ
6) ให้การสนับสนุนเฟรมเวิร์คของทั้ง server-side และ front-end ชื่อดังต่างๆ เช่น Spring, Android, Angular JS เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 RESTful Web Service

REST ย่อมาจาก (Representational State Transfer) เป็นแนวทางในการใช้ HTTP Protocol ในการสร้าง Web Service ที่นิยมใช้กันมากในปัจจุบัน โดยมี Method เรียกใช้ในการติดต่อระหว่างเว็บเซอร์วิซ และ ไคลเอนท์เรียกว่า HTTP Method ประกอบไปด้วย 4ส่วนคือ

1) Method GET เป็นการเรียกข้อมูล

2) Method Post เป็นการสร้างข้อมูลขึ้นมาใหม่

3) Method Put เป็นการแก้ไขอัพเดทข้อมูล

4) Method Delete เป็นการลบข้อมูล

จาก HTTP Method นั้น และการทำงานในการส่งข้อมูลกลับมานั้น อยู่ในรูปแบบ JSON ทำให้เห็นได้ว่ามีความง่ายในการใช้งาน ข้ามเพลตฟอร์มต่างๆ มีไลบรารีหลากหลายตัวที่ออกมาสนับสนุนการทำงานของ RESTful Web Service เช่น Spring Framework , Jersey Framework เป็นต้น

2.8 Favor Level

$$favor_level = Discretize\left(\frac{\sum_{i=1}^{T} pt_i}{dt \times T}\right) \tag{2.1}$$

โดย T คือจำนวนครั้งของการเล่นเพลง

dt คือระยะเวลาทั้งหมดของเพลง

i คือครั้งที่เล่นเพลง

pt_i คือระยะเวลาในการฟังเพลงของผู้ใช้ในครั้งที่ i

นำค่า favor_level มาเปรียบเทียบตามตารางที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.1 Favor Level 2 ระดับ

Class	Range
Dislike	0.00-0.25
Like	0.26-1.00

2.9 Amazon Web Services

รูป 2.10 โลโก้ Amazon Web Services

Amazon Web Service ให้บริการทางด้านบริการผลิตภัณฑ์ที่ใช้งานร่วมกับ Cloud, การคำนวณ, การเก็บข้อมูล, ฐานข้อมูล, การวิเคราะห์, ระบบเครือข่าย, โทรศัพท์, โปรแกรมเครื่องมือที่ช่วยในการพัฒนา IoT และด้านความปลอดภัยเป็นต้น บริการเหล่านี้ช่วยทำให้การทำงานเป็นไปได้เร็วขึ้น ลดงบประมาณในการทำงานและขนาดของงาน

AWS เป็นที่เชื่อถือได้ในเหล่าบริษัทขนาดใหญ่ และ Start-up ต่างๆ ในการรองรับปริมาณงานที่หลากหลาย ซึ่งในโครงงานนี้ เราได้ใช้ AWS ดังนี้

2.9.1 Amazon Relational Database Service (Amazon RDS)

Amazon RDS คือ การให้บริการการจัดการ Relational Database ที่ให้ความสะดวกในการเลือก Database Engines ซึ่งมี 6 ชนิด คือ Amazon Aurora, MySQL, MariaDB, Oracle, Microsoft SQL Server, and PostgreSQL และมีการจัดการโค้ด, แอพลิเคชั่นและเครื่องมือที่พร้อมใช้งานกับ Database หรือ Amazon RDS ที่มีอยู่แล้ว ซึ่ง Amazon RDS สามารถจัดการกับคำสั่งในการจัดการ Database ต่างๆ เช่น Provisioning, Patching, Backup, Recovery, Failure Detection และการ Repair

เช่นเดียวกันกับหลายๆ การบริการของ Amazon Web Services ซึ่งไม่ต้องการการลงทุนล่วงหน้า และสามารถจ่ายเงินเฉพาะสำหรับทรัพยากรที่ใช้เท่านั้น สำหรับในโครงงานนี้เราเลือกใช้ RDS ที่มี Database Engine เป็น MariaDB

2.9.2 Amazon CloudFront

Amazon CloudFront คือ Global Content Delivery Network (CDN) Service ซึ่งสามารถจัดส่งไฟล์ไปยังเว็บไซต์, APIs, Content Video หรือ ส่วนประกอบอื่นๆ ของเว็บ โดย Amazon CloudFront สามารถใช้งานร่วมกับผลิตภัณฑ์อื่นๆ ของ Amazon Web Services ได้ และสำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นักพัฒนาหรือนักธุรกิจ สามารถจัดการการใช้งานได้ง่าย เนื่องจากไม่มีการใช้งานขั้นต่ำเป็นข้อผูกมัด

Amazon CloudFront เป็น Web Service Interface ที่สามารถแบ่ง Object เป็นส่วนๆได้ การแบ่งนั้นต้องระบุตำแหน่งที่ตั้งของไฟล์ข้อมูลนั้นๆ โดยการแบ่งจะมี CloudFront.net Domain Name ที่ไม่ซ้ำกัน เช่น abc123.cloudfront.net ที่สามารถอ้างอิงถึง Object นั้นๆผ่านเครือข่ายทั่วโลก ที่สามารถดึงข้อมูลที่ไหนก็ได้ที่สะดวก สามารถเพิ่ม Domain Name ของตัวเองได้ในการแบ่ง Object และยังสามารถสร้างการแบ่งการดาวน์โหลด Object โดยใช้โปรโตคอล HTTP หรือ HTTPS หรือ Stream Object ได้

2.9.3 Amazon Simple Storage Service (Amazon S3)

Amazon Simple Storage Service (Amazon S3) คือ Object Storage ซึ่งเป็น Web Service Interface ในการเก็บและกู้ข้อมูลจำนวนใดๆจากที่ไหนก็ได้ในเว็บ และ Amazon S3 ออกแบบเพื่อให้มีความทนทาน 99.999999999% และการปรับขนาดประมาณล้านล้าน Object

โดยในโครงงานนี้ ได้ใช้ Amazon S3 ในการเก็บไฟล์เพลงเริ่มต้น และไฟล์เพลงที่ผ่านการ Transcode เพื่อเตรียมพร้อมสำหรับการ Streaming

2.9.4 Amazon Elastic Compute Cloud (Amazon EC2)

Amazon Elastic Compute Cloud (Amazon EC2) คือ Web Service ที่ให้บริการทางด้าน Cloud ที่สามารถปรับขนาดการประมวลผลได้ และมีความปลอดภัย ถูกออกแบบมาให้สามารถประมวลผล Web-Scale Cloud ได้ง่ายสำหรับนักพัฒนา

Amazon EC2 เป็น Web Service Interface อย่างง่ายที่อนุญาติให้สามารถทำการตั้งค่าปริมาณการประมวลผลได้ด้วยฟังค์ชันคำสั่งเพียงเล็กน้อย สามารถควบคุมทรัพยากรการประมวลผลและสามารถรันบนสภาพแวดล้อมการประมวลผลที่ Amazon พิสูจน์มาแล้ว คือ Amazon EC2 ลดเวลาในการเปิดใช้งาน Server ตัวใหม่ในเวลาไม่กี่นาที, อนุญาติให้ปรับขนาดการประมวลผลอย่างรวดเร็วได้, เปิดเครื่องและปิดเครื่อง เมื่อเกิดการเปลี่ยนแปลงข้อกำหนดของการประมวลผล

Amazon EC2 เปลี่ยนแปลงระบบการเศรษฐกิจของการประมวลผล โดยอนุญาติให้จ่ายเท่าที่ได้ใช้ไป ซึ่ง Amazon EC2 ให้บริการเครื่องมือสำหรับการทำแอพลิเคชั่นที่สามารถกู้คืนสภาพได้หากเกิดข้อผิดพลาด และสามารถทำ Isolate Theme สำหรับเหตุการณ์ข้อผิดพลาดที่เกิดบ่อยๆ

2.9.5 Amazon Elastic Transcoder

Amazon Elastic Transcoder เป็นเปลี่ยนแปลงรูปแบบไฟล์ใน Cloud ซึ่งถูกออกแบบให้สามารถปรับขนาดได้มากๆ ง่ายในการใช้งานและคุ้มค่าในการงานสำหรับนักพัฒนาและธุรกิจ ในการเปลี่ยนแปลงไฟล์จากรูปแบบเดิมให้อยู่ในรูปแบบที่สามารถเล่นได้บนอุปกรณ์ต่างๆ เช่น Smartphones, tablets และ PC เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโครงงานนี้ได้ใช้ในการเปลี่ยนแปลงไฟล์มีเดีย (.mp3) ให้เป็นไฟล์ที่สามารถเล่นได้ด้วย Streaming Protocol โดยไฟล์ทั้งหมดเก็บอยู่บน Amazon S3

2.10 Docker

รูป 2.11 โลโก้ Docker

นักพัฒนาที่ใช้ Docker ไม่จำเป็นต้องติดตั้งและตั้งค่าฐานข้อมูลที่ซับซ้อน รวมถึงไม่ต้องกังวลเกี่ยวกับการสลับการทำงานระหว่างเวอร์ชั่นของภาษาเครื่องมือที่เข้ากันไม่ได้ เมื่อแอพลิเคชั่นเป็น Dockerized มีความซับซ้อนในการใส่ข้อมูลใน Containers ซึ่งสามารถ Build ได้ง่าย รวมถึงการแชร์และรัน ในการเริ่มการใช้งานของผู้ร่วมงานใน Codebase อันใหม่ ไม่ต้องเสียเวลากับการติดตั้งซอฟแวร์และอธิบายขั้นตอนการตั้งค่าต่างๆ โค้ดสามารถส่งผ่านใน Docker file ได้เลย โดย Dependency จะถูกดึงคล้ายกับ Package ของ Docker และติดตั้ง Editor ที่สามารถ Build และ Debug แอพลิเคชั่นในเวลาไม่กี่นาที

2.11 Streaming Service

Streaming Service คือการให้บริการการสตรีมมิ่งไฟล์มัลติมีเดียต่างๆผ่านทางอินเตอร์เน็ต โดยเทคนิคการสตรีมมิ่งนั้น คือการที่แบ่งไฟล์มัลติมีเดียออกเป็นส่วนเล็กๆ ให้สามารถดาวน์โหลดได้รวดเร็วขึ้น และสามารถเล่นเพลงหรือวีดีโอได้เลย โดยไม่ต้องรอดาวน์โหลดไฟล์ทั้งหมดให้เสร็จก่อนแล้วค่อยเล่น ทำให้การใช้งานทำได้ราบรื่นมากขึ้นและไม่ต้องรอเวลานาน

โปรโตคอลที่ให้บริการการสตรีมมิ่งมีหลายอัน ยกตัวอย่าง Real-Time Messaging Protocol (RTMP), HTTP Live Streaming (HLS) เป็นต้น ในโครงงานนี้เราใช้ HLS ในการสตรีมมิ่ง

HLS คือโปรโตคอลที่สามารถส่งไฟล์มัลติมีเดียวผ่านโปรโตคอล HTTP ได้ โดยส่งไฟล์มาจาก Web Server ในการเล่นเพลงบนอุปกรณ์ต่างได้ HLS สนับสนุนทั้ง Live Broadcasts และไฟล์มัลติมีเดียที่บันทึกไว้แล้ว และสนับสนุน Multiple Alternate Streams ที่ความต่างกันของ bit rate และซอฟแวร์ทางฝั่งของผู้ใช้สามารถสลับการสตรีมตามการเปลี่ยนแปลงของ Bandwidth ได้อย่างมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

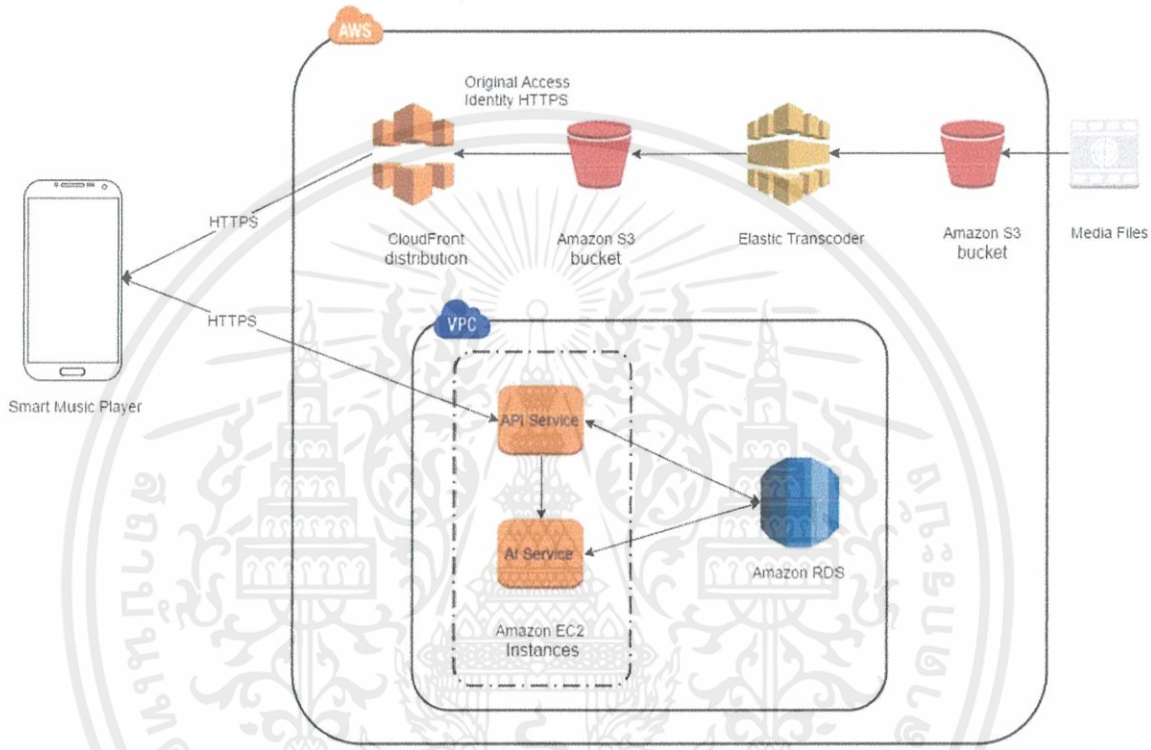
2.12 Content Delivery Network (CDN)

CDN คือระบบเครือข่ายขนาดใหญ่ของเครื่อง Server ที่เชื่อมต่อกันทั่วโลก ในการให้บริการการดาวน์โหลด Content ต่างๆ เช่น ไฟล์มัลติมีเดียต่างๆ และรูปภาพ เป็นต้น โดย Content เหล่านี้จะถูกคัดลอกไปไว้เครื่อง Server แต่ละแห่งทั่วโลก ที่เป็น Edge of Internet ดังนั้น หากเกิดการดาวน์โหลดขึ้น เครื่องของผู้ใช้สามารถเลือกดาวน์โหลดจากเครื่อง Server ที่อยู่ใกล้ที่สุด เพื่อลดระยะทางในการดาวน์โหลดลง เพิ่มประสิทธิภาพในการดาวน์โหลด และทำให้ไฟล์ Content ต่างๆ ถูกดาวน์โหลดไปถึงผู้ใช้ในเวลาที่รวดเร็วที่สุด ซึ่งในโครงงานนี้เราได้ใช้ CDN ของ AWS CloudFront ในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการพัฒนา

3.1 ภาพรวมของระบบ



รูป 3.1 ภาพรวมของระบบ

ในส่วนนี้จะออกแบบองค์ประกอบต่างๆของระบบ ซึ่งจะใช้ AWS ทั้งหมดในส่วนของ Back-End จะมีส่วนของ Streaming และ VPC Service ต่างๆของระบบ ดังนี้

3.1.1. Streaming

ในส่วนนี้ ระบบต้องเก็บไฟล์เพลงทั้งหมดไว้ที่ Amazon S3 bucket ก่อน จากนั้นใช้ Elastic Transcoder ในการ Transcode ไฟล์ให้อยู่ในรูปแบบที่เหมาะสมสำหรับการเล่นเพลงแบบ Streaming แล้วเก็บไว้ที่ Amazon S3 bucket ที่ใช้เก็บไฟล์ streaming ในการส่งข้อมูลให้ Smart Music Player จะใช้ CloudFront Distribution ในการส่งข้อมูลแบบ Streaming

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2. RESTful API Service

ทำหน้าที่ในการติดต่อระหว่าง Back-End กับฝั่งผู้ใช้ ในการจัดการข้อมูลต่างๆ ซึ่งจะเก็บข้อมูลไว้ที่ Amazon RDS และติดต่อกับ AI Service ในการเทรนข้อมูลที่เก็บมาจากฝั่งผู้ใช้งาน ซึ่ง API Service และ AI Service เป็น Instance ที่ถูกเก็บไว้ใน Amazon EC2 ใน Virtual Private Cloud และใช้โปรโตคอล HTTPS ในการสื่อสารระหว่าง Server และฝั่งผู้ใช้งาน เขียนด้วยภาษา Java

3.1.3. AI Service

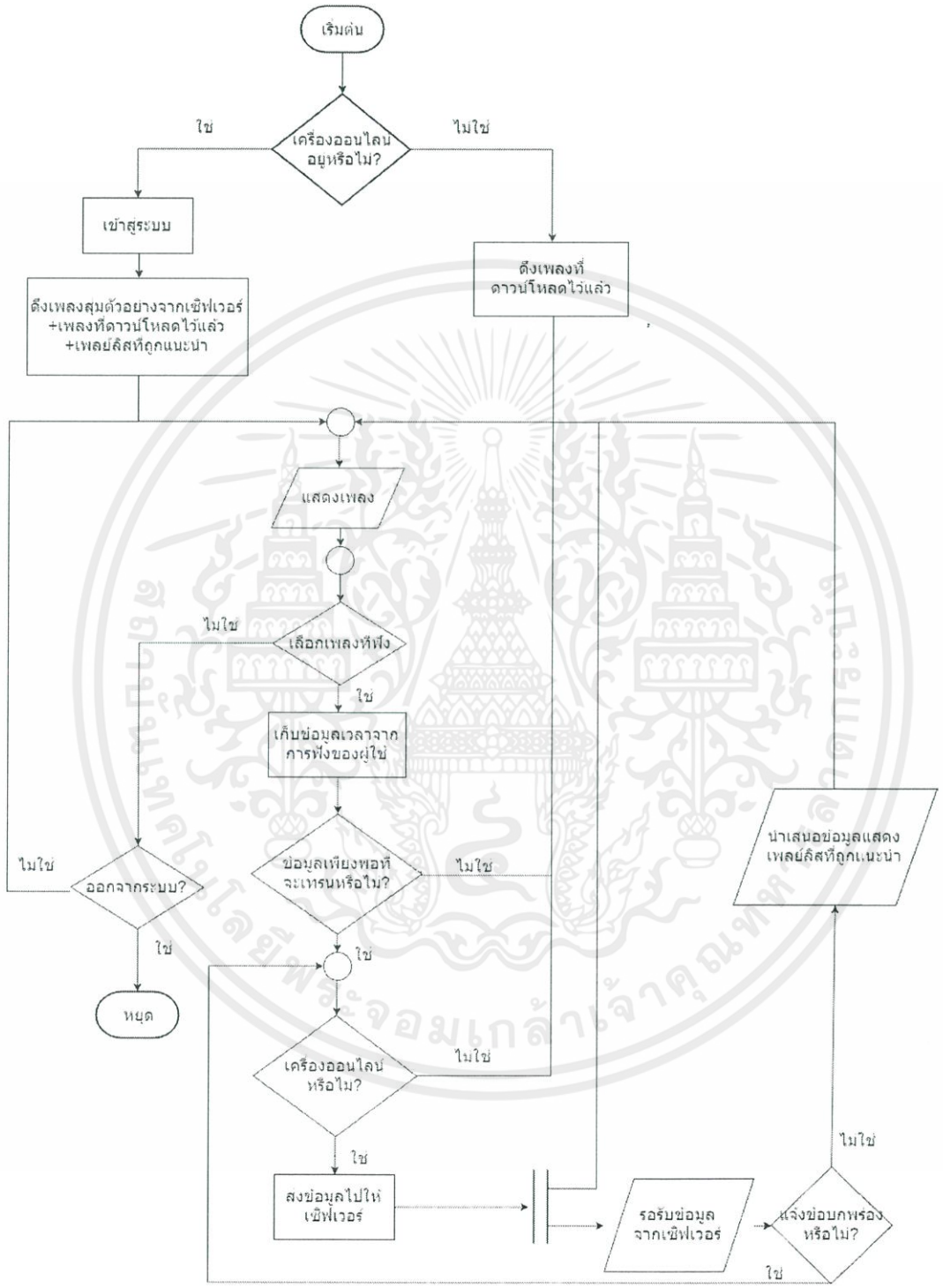
ในส่วนนี้จะเป็น Artificial Neural Network ทำหน้าที่ในการเทรนข้อมูลที่ถูกส่งต่อมาจาก RESTful API Service และเก็บข้อมูลการเทรนไว้ที่ Amazon RDS เขียนด้วยภาษา Python โดยในการเทรนนั้นจะเทรนแยกคนละ Model กัน จึงทำให้ทำงานหนัก และควรจะแยกออกมาจาก Back-End Server ปกติ

3.1.4. Amazon RDS

เป็น Relational Database ที่อยู่ใน AWS ทำหน้าที่เก็บข้อมูลต่างๆ จากทั้ง RESTful API Service และ AI Service ซึ่ง Amazon RDS นี้เป็น Instance ที่อยู่ใน Virtual Private Cloud ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ผังงานของระบบฝั่งแอพพลิเคชันแอนดรอยด์



รูป 3.2 ผังงานระบบฝั่งแอพพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบายผังงาน

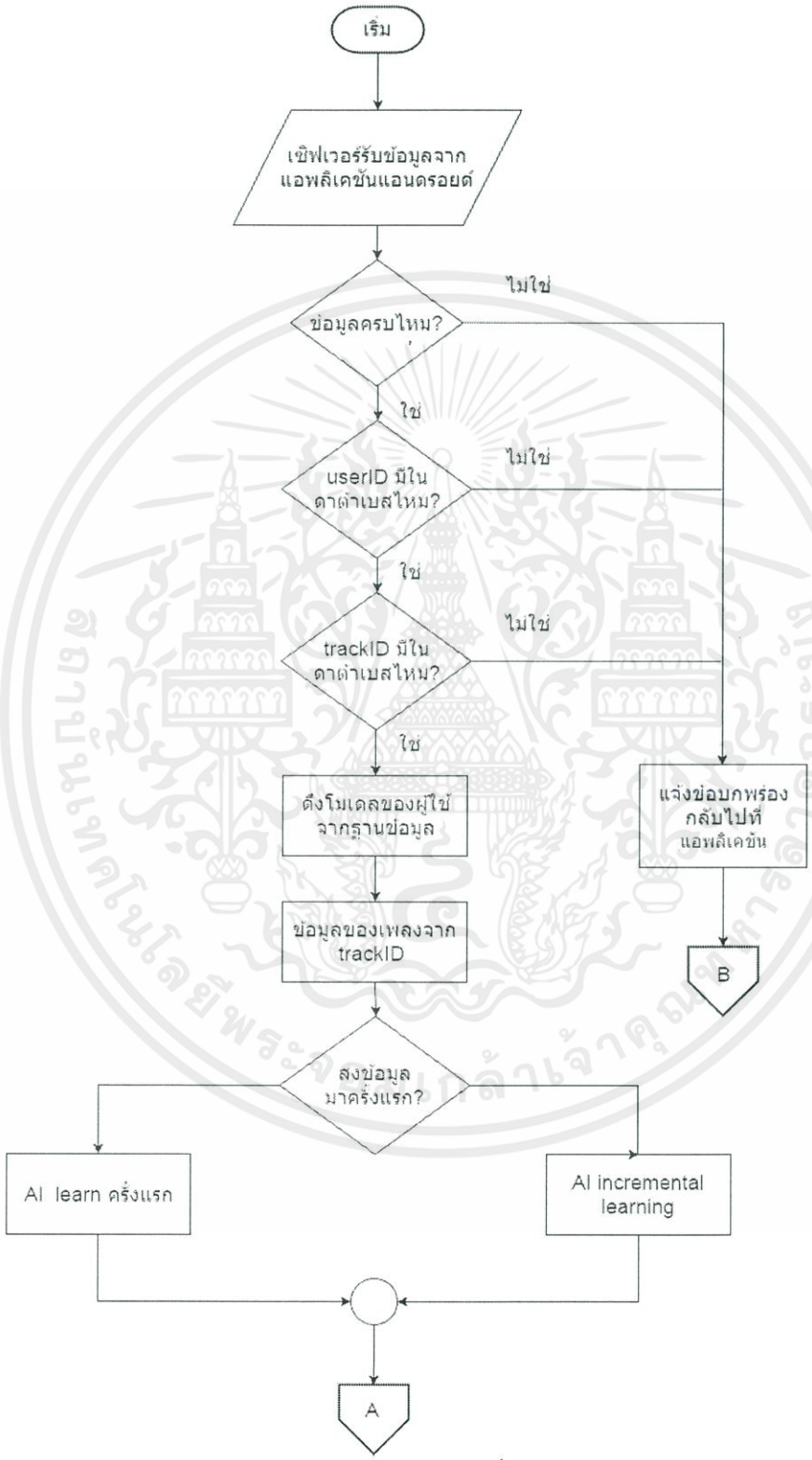
ในส่วนนี้อธิบายถึงการทำงานของฝั่งแอพพลิเคชันแอนดรอยด์ เริ่มจากผู้ใช้เปิดแอพพลิเคชัน แล้วตัวโปรแกรมจะเช็คว่าเครื่องออนไลน์อยู่หรือไม่ หากออนไลน์อยู่ ก็ให้ล็อกอินเข้าสู่ระบบและทำการดึงเพลงจากฝั่งเซิฟเวอร์มาแสดง ซึ่งเพลงที่ดึงมานั้น มีทั้งส่วนที่เป็นเพลงสุ่มตัวอย่าง เพลย์ลิสที่แนะนำ และดึงเพลงที่ผู้ใช้ดาวน์โหลดมาแล้วจากเครื่อง นำมาแสดงผล ถ้าไม่ออนไลน์ก็จะดึงเพลงที่ดาวน์โหลดไว้แล้วมาแสดงเท่านั้น

เมื่อผู้ใช้กดฟังเพลง และฟังเพลงจบแล้ว แอพพลิเคชันทำการเก็บเวลาที่ผู้ใช้ฟังก่อนกดออก และเก็บจนกว่าจะได้จำนวนที่เพียงพอที่จะให้โครงข่ายประสาทเทียมเรียนรู้ในครั้งแรก เมื่อเก็บครบแล้ว จึงทำการเช็คว่าออนไลน์หรือไม่ หากออนไลน์แอพพลิเคชันจะทำการส่งข้อมูลที่เก็บไว้ส่งไป ให้เซิฟเวอร์ ในขั้นตอนนี้แอพพลิเคชันแอนดรอยด์จะทำงานแบบไม่พร้อมกัน คือรอเซิฟเวอร์ตอบกลับ และยังสามารถกดเล่นเพลงได้ปกติ เมื่อเซิฟเวอร์ตอบกลับมา จึงนำเสนอเพลย์ลิสที่ถูกแนะนำให้กับผู้ใช้

ในการส่งข้อมูลที่เก็บไว้ให้กับเซิฟเวอร์ จะทำการส่งให้ก็ต่อเมื่อเป็นการเรียนรู้ครั้งแรกของบัญชีผู้ใช้นี้ และเก็บตามรอบเวลาที่กำหนด

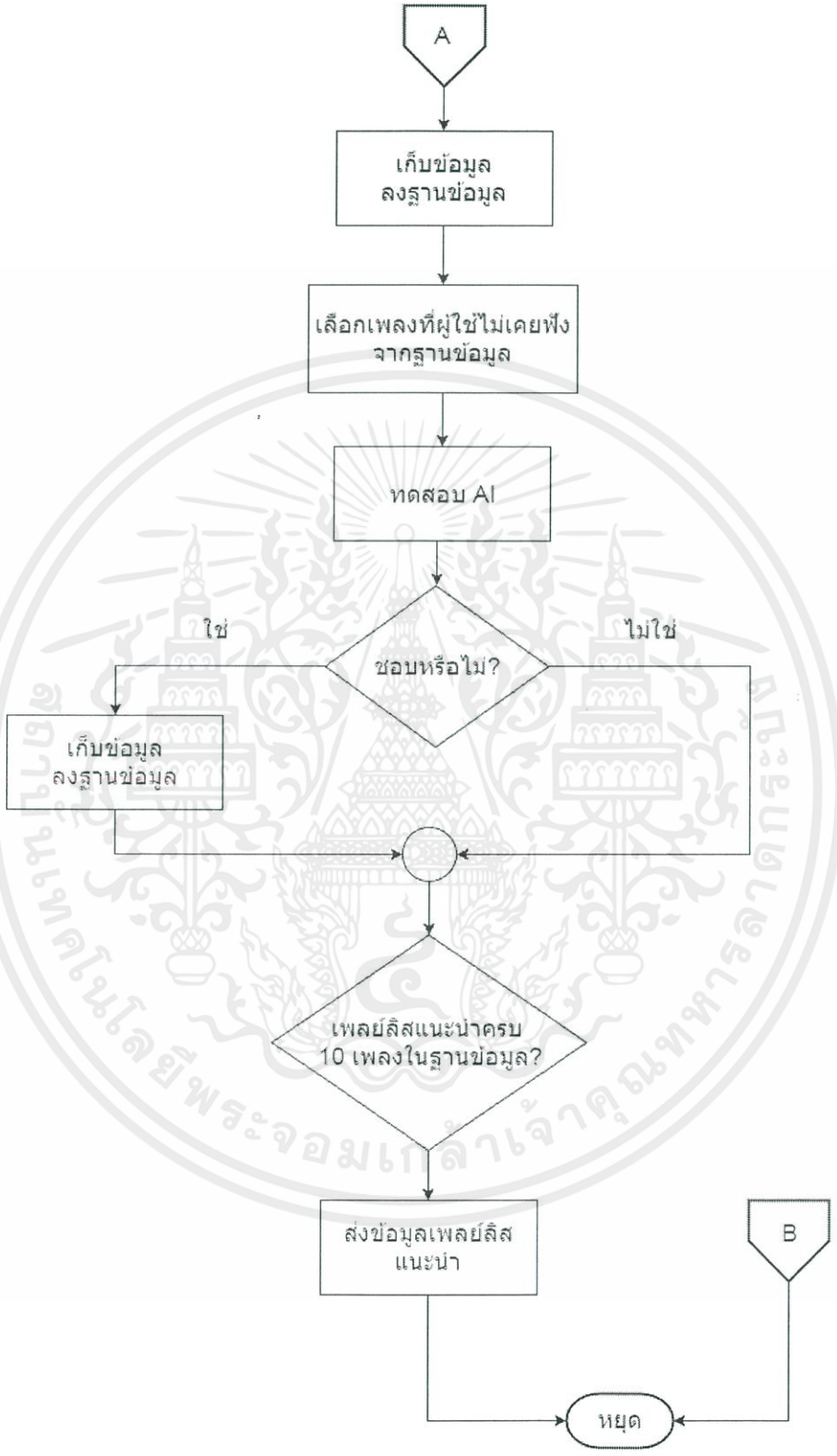
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ผังงานของระบบฝั่งเซิฟเวอร์



รูป 3.3 แผนผังการทำงานฝั่งเซิฟเวอร์ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.4 แผนผังการทำงานฝั่งเซิฟเวอร์ 2 (ต่อจากรูป 3.3)

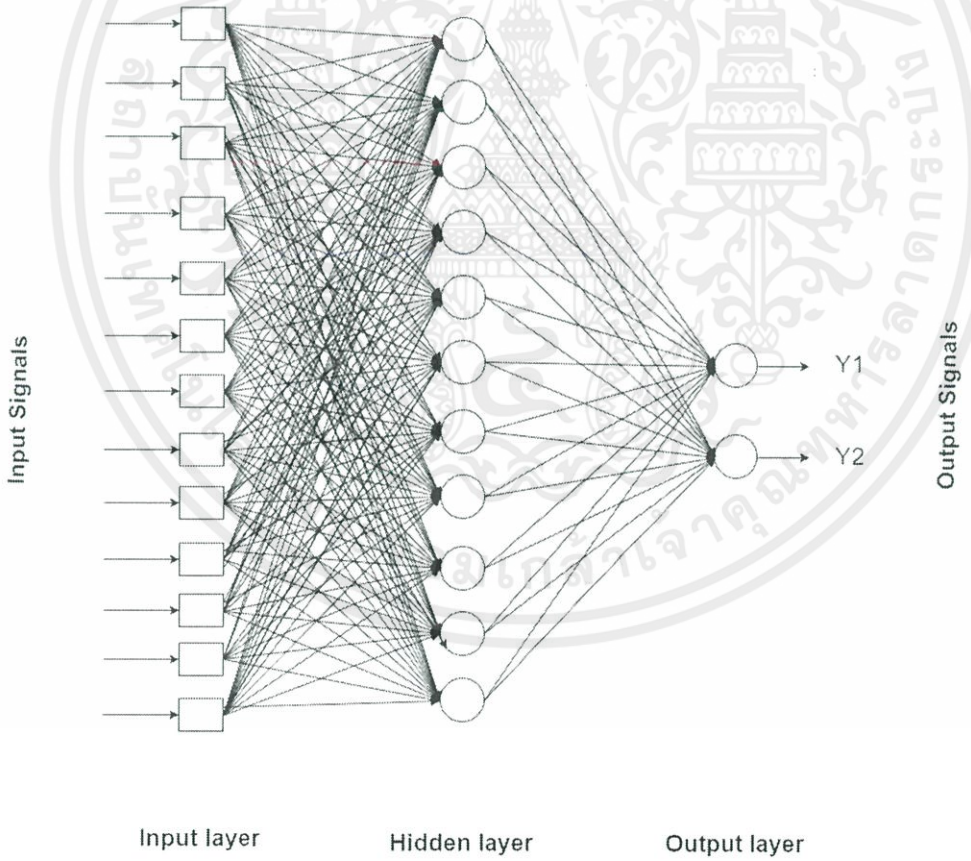
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบายผังงาน

เซิฟเวอร์รับข้อมูลจากแอพพลิเคชัน จากนั้นจะทำการตรวจสอบว่าข้อมูลที่ได้รับมาครบหรือไม่ UserID และ TrackID นี้มีในระบบหรือไม่ หากมีไม่ครบให้แจ้งข้อผิดพลาดกลับไปที่แอพพลิเคชัน เมื่อได้รับข้อมูลครบแล้ว จะทำการดึงข้อมูลโมเดลของผู้ฟังและข้อมูลเพลง จากนั้นเช็คว่าส่งข้อมูลมาครั้งแรกหรือไม่ หากส่งมาครั้งแรก จะไม่มีโมเดลผู้ฟังในระบบ ต้องให้โครงข่ายประสาทเทียมทำการนำข้อมูลที่ได้มาเรียนรู้ก่อน หากไม่ใช่ จะทำการเรียนรู้เพิ่มขึ้นจากของเดิม

เมื่อทำการเรียนรู้เรียบร้อยแล้ว จะเก็บข้อมูลโมเดลที่ได้จากการเรียนรู้ครั้งใหม่ลงฐานข้อมูล จากนั้นจะเลือกเพลงที่ผู้ใช้ไม่เคยฟังโดยการสุ่ม โดยอ้างอิงจากตารางที่ผู้ใช้เคยฟังแล้ว และทดสอบโครงข่ายประสาทเทียมให้ทำนายเพลงว่าผู้ใช้จะชอบหรือไม่ หากชอบจะเก็บลงฐานข้อมูลเพลย์ลิสที่ถูกแนะนำ เมื่อครบ10 เพลงในฐานข้อมูลแล้ว จึงจะส่งข้อมูลกลับไปฝั่งแอพพลิเคชันแอนดรอยด์

3.4 โครงสร้างของโครงข่ายประสาทเทียม



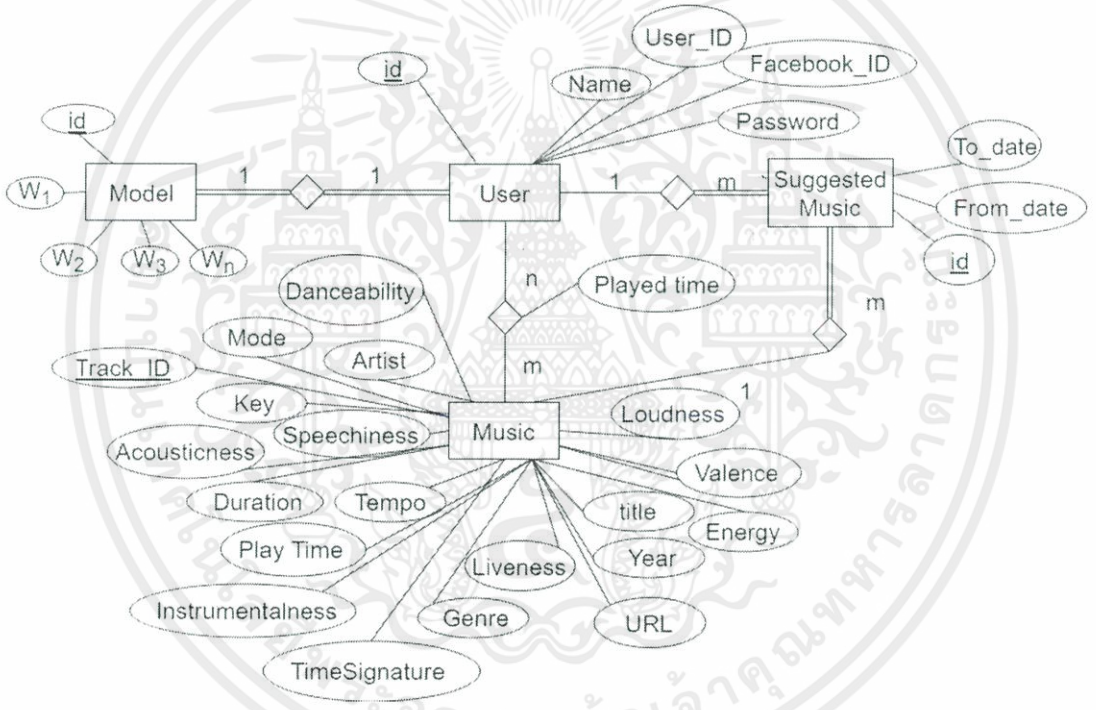
รูป 3.5 โครงสร้างโครงข่ายประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนนี้แสดงการออกแบบโครงข่ายประสาทเทียมโดยคำนึงถึง Input Signals และ Output Signals โครงข่ายประสาทเทียมที่ใช้ในปริญญานิพนธ์นี้ ใช้เพียง 3 Layer คือ Input Layer, Hidden Layer และ Output Layer

จากทฤษฎีที่เกี่ยวข้องในบทที่ 2 ได้กล่าวถึงทฤษฎีโครงข่ายประสาทเทียมและแอตทริบิวต์ที่ได้จากเพลง ใน Input Layer มีจำนวน Input Signals ทั้งหมด 13 ตัวตามจำนวนแอตทริบิวต์ที่ได้จากเพลง ในส่วนของ Output Layer มีเพียง 2 Neuron และในส่วนของ Hidden Layer มี 11 Node ซึ่งได้มาจากการทดลอง

3.5 ออกแบบฐานเก็บข้อมูล



รูป 3.6 แบบจำลองแสดงการออกแบบฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การประเมินและใช้งานระบบ

ในส่วนนี้กล่าวถึงการใช้งานแอพลิเคชั่น Smart music Player และนำข้อมูลการฟังเพลงของผู้ใช้กลับไปเทรนที่ฝั่ง Back-End Server รวมทั้งผลลัพธ์ที่ได้จากการเทรนด้วย Artificial Neural Network ด้วย

4.1 การกำหนดข้อมูลที่ใช้ในการเทรนของ Artificial Neural Network

ในส่วนของข้อมูลที่ให้ผู้ใช้ฟังเพื่อเก็บข้อมูลการฟังของผู้ใช้ไปเทรนในส่วนของ Back-End Server มีเพลงทั้งหมด 200 เพลง และแบ่งเพลงเป็น 6 ประเภท คือ

1) Pop 100 เพลง
2) Acoustic 20 เพลง
3) Alternative 20 เพลง
4) Rock 20 เพลง
5) R&B 20 เพลง
6) Hip-hop 20 เพลง

โดยสัดส่วนของเพลงแต่ละประเภทนั้น นำมาจากผลการสัมภาษณ์โดยการสุ่มจาก นักศึกษาคณะวิศวกรรมคอมพิวเตอร์ เมื่อนำมารวมและคิดเป็นสัดส่วนแล้ว จะให้ผลตามการแจกแจงที่กล่าวมาข้างต้น

ในการทดลองฟังเพลงในแอพลิเคชั่น Smart music Player นั้น เรามีอาสาสมัครทั้งหมด 10 คน มาช่วยทดสอบระบบ

4.2 การทดลองใช้แอพลิเคชั่น Smart Music Player บนระบบปฏิบัติการแอนดรอยด์

สามารถทำงานได้ในระบบปฏิบัติการแอนดรอยด์ รุ่น Android 4.4 (KITKAT) ขึ้นไป โดยมี Feature การทำงานดังนี้

1) สามารถเล่นเพลงด้วยระบบ Streaming ได้
2) สามารถควบคุมการทำงานต่างๆของ Music Player ได้ เช่น เล่น, พัก เปลี่ยนเพลง
3) สามารถเลือกโหมดการเล่นได้ (เล่นตามลำดับ, เล่นแบบสุ่ม)
4) สามารถสุ่มรายชื่อเพลงที่จะฟังจากระบบได้
5) สามารถเก็บเพลงเป็น Playlist ของตนเองได้
6) สามารถจัดการ Playlist ได้

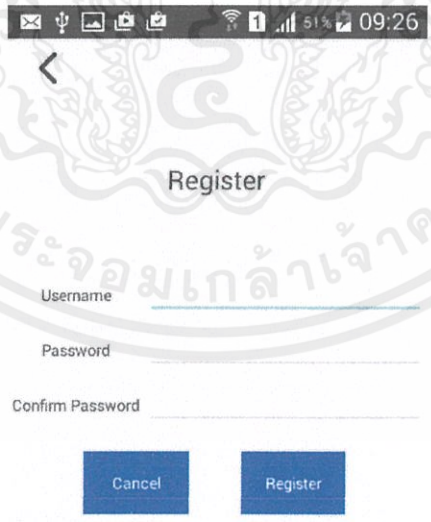
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) สามารถ Search เพลงตาม Keyword ที่ต้องการ

8) ระบบสามารถแนะนำเพลงที่ผู้ใช้มีแนวโน้มว่าจะชอบได้

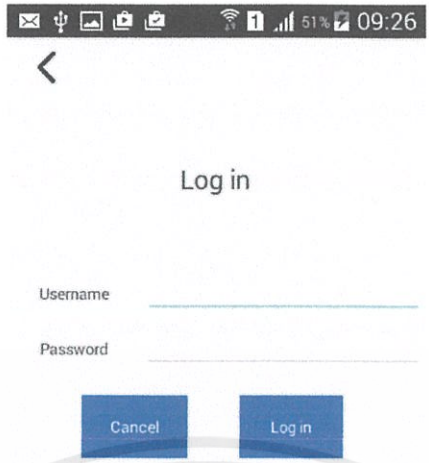
โดยมี User Interface ดังนี้

รูป 4.1 หน้าเลือก Login



รูป 4.2 หน้าระบบ Register

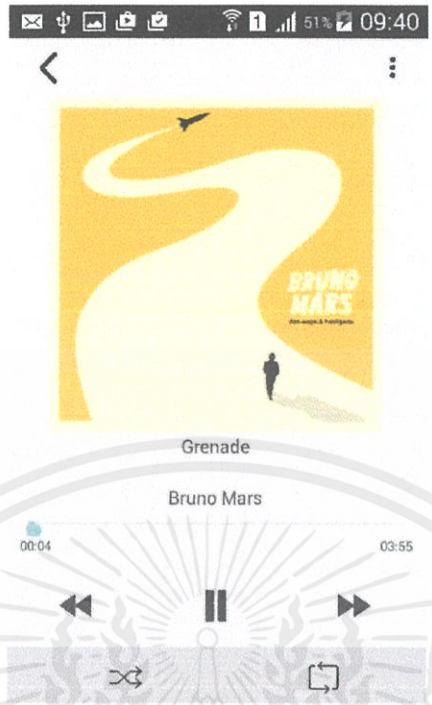
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



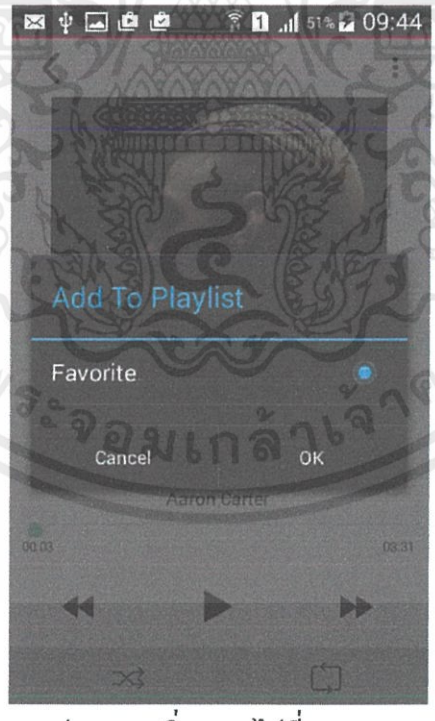
รูป 4.3 หน้าระบบ Login

รูป 4.4 หน้าแสดงรายการเพลงทั้งหมด และส่วนการแนะนำเพลงจากระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

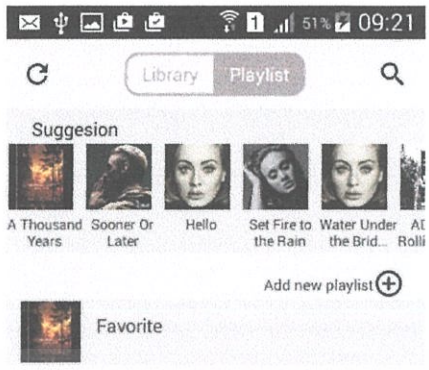


รูป 4.5 หน้าควบคุมการเล่นเพลง โดยมีสามารถ กดเล่น, พัก, เพลงถัดไป, เพลงก่อนหน้า, เล่นเพลงแบบสุ่ม และเล่นเพลงแบบซ้ำ

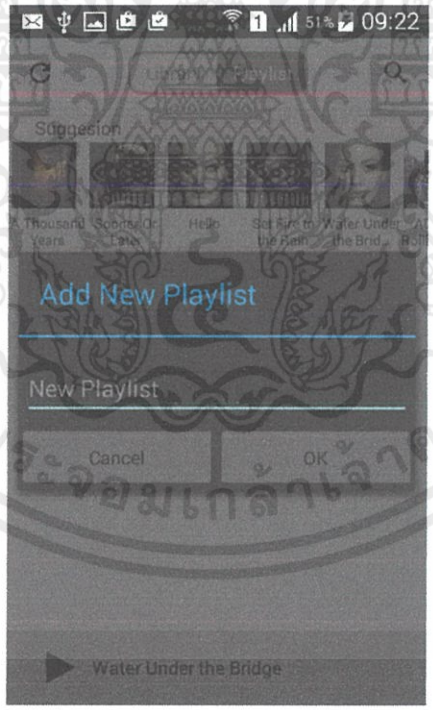


รูป 4.6 กดเพิ่มเพลงไปที่ Playlist

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

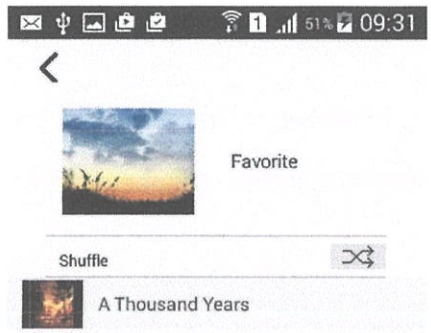


รูป 4.7 หน้าแสดง Playlist ทั้งหมด

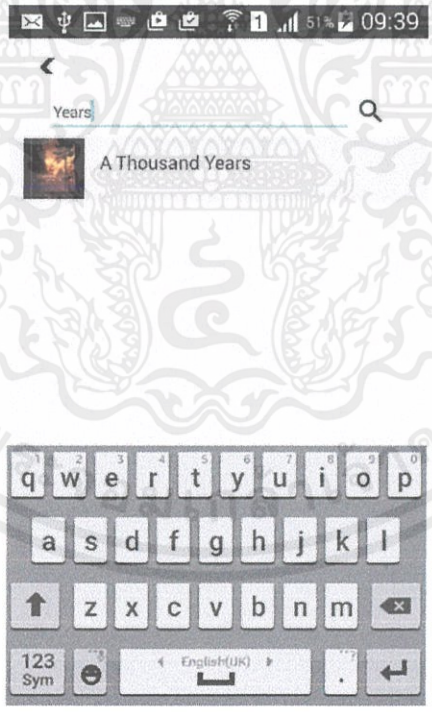


รูป 4.8 สร้าง Playlist ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.9 หน้าจัดการ Playlist



รูป 4.10 หน้า Seach

ในการทดลองใช้ระบบ เราให้อาสาสมัครทั้ง 10 คน ทดลองฟังเพลงผ่าน Smart Music Player โดยกดเลือกเล่นเพลงที่ผู้ใช้ต้องการ และกดข้ามเพลงที่ผู้ใช้ไม่ชอบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้ฟังเพลงเกิดกว่าค่าที่กำหนดแล้ว ระบบจะส่งไฟล์ที่เก็บข้อมูลการฟังของผู้ใช้ไปยังฝั่ง Back-End Sever และเริ่มเทรนด้วย Artificial Neural Network

4.3 การทดลองการหาจำนวน Node ของ Hidden Layer ใน Artificial Neural Network

เป็นการทดลองเพื่อหาจำนวน Node ใน Hidden Layer ที่ดีที่สุด และทดสอบว่ากระบวนการเทรน Artificial Neural Network ด้วยวิธีการใช้ Learning Rate เป็นค่าคงที่ หรือใช้ Learning Rate ลดลงเรื่อยๆตามจำนวน Epoch ที่เพิ่มขึ้น วิธีใดจะได้ค่า Accuracy สูงที่สุด ซึ่งได้ผลการทดสอบดังนี้

ตาราง 4.1 ใช้วิธี Decreasing Learning Rate

Decreasing Learning Rate								
Volunteer ID	**Node Size**	**Epochs**	**SSE**	**ACC**	**TP**	**TN**	**FP**	**FN**
1	3	7800	40.74566647	83.74384236	142	28	25	8
2	**11**	8500	22.33612121	91.13300493	116	69	15	3
3	2	4600	56.7360942	82.75862069	154	14	35	0
4	6	6800	40.97666279	85.22167488	119	54	17	13
5	**11**	8300	22.55421482	95.07389163	80	113	2	8
6	9	9900	40.93818036	84.2364532	45	126	9	23
7	12	8000	17.04391738	96.55172414	101	95	2	5
8	7	7900	24.17755496	93.10344828	138	51	7	7
9	**11**	6500	14.55399318	97.04433498	78	119	1	5
10	10	6000	9.504658934	98.52216749	82	118	1	2

จากตาราง 4.1 พบว่า ค่าฐานนิยมของจำนวน Node ที่ดีที่สุดในแต่ละการทดลองของผู้ใช้มีค่าเท่ากับ 11 และจากตารางข้างต้น สามารถหาค่าเฉลี่ยของ Accuracy ได้ที่ 90.73891626

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.2 ใช้วิธีไม่ใช้ Decreasing Learning Rate

Without Decreasing Learning Rate								
Volunteer ID	**Node Size**	**Epochs**	**SSE**	**ACC**	**TP**	**TN**	**FP**	**FN**
1	3	2400	46.93764744	83.25123153	146	23	30	4
2	11	8200	12.63089308	94.58128079	115	77	7	4
3	3	7300	46.16237551	83.25123153	151	18	31	3
4	6	9500	24.80174947	91.62561576	127	59	12	5
5	11	8200	21.14513312	90.64039409	76	108	7	12
6	6	7400	16.01386589	95.07389163	60	133	2	8
7	11	9300	8.258114991	98.02955665	104	95	2	2
8	9	4600	25.78352221	93.59605911	144	46	12	1
9	14	9800	9.331440463	97.04433498	82	115	5	1
10	11	9900	10.61499379	97.04433498	81	116	3	3

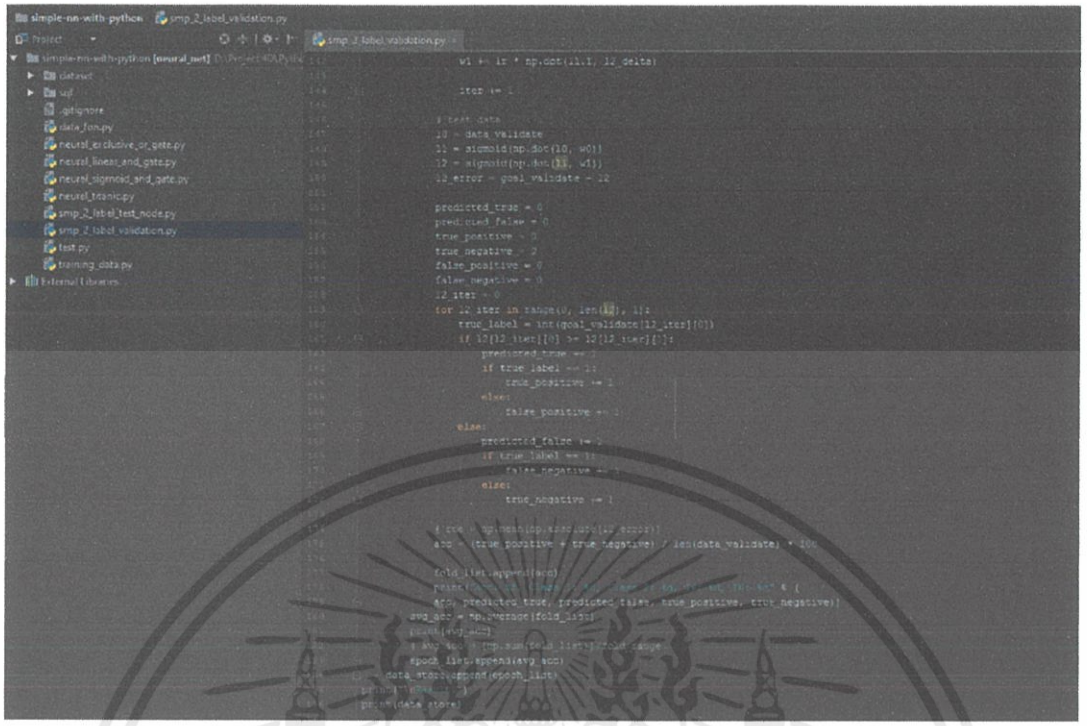
จากตาราง 4.2 พบว่า ค่าฐานนิยมของจำนวน Node ที่ดีที่สุดในแต่ละการทดลองของผู้ใช้มีค่าเท่ากับ 11 และจากตารางข้างต้น สามารถหาค่าเฉลี่ยของ Accuracy ได้ที่ 92.4137931

จากทั้ง 2 ตาราง จะเห็นได้ว่า จำนวน Node ที่ดีที่สุดที่ใช้ใน Hidden Node คือ 11 และค่า Accuracy เฉลี่ย ของตาราง 4.2 มีค่าสูงกว่า ซึ่งเป็นการทดลองที่ใช้ค่า learning rate แบบคงที่ ในโครงงานนี้เราจึงเลือกใช้จำนวน Hidden Node ที่ 11 Node และเทรนข้อมูลโดยใช้ค่า learning rate คงที่

4.4 การเทรนด้วย Artificial Neural Network

Artificial Neural Network ที่ใช้ในโครงงานนี้ สร้างจากภาษา Python และทำ Ten- Fold Cross Validation ในการประมาณค่าความแม่นยำของการทำนายค่า input แอตทริบิวท์ส่วนใหญ่มากจาก Spotify API และใช้ Goal Predicate เป็น Favor Level ซึ่งคำนวณมากจากข้อมูลการฟังเพลงของผู้ใช้แต่ละคน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.11 ตัวอย่างโค้ด Artificial Neural Network ภาษา Python

4.5 ผลการทดลอง

4.5.1 การทดลองที่ 1

กำหนดค่า Output เป็น 2 ค่า คือชอบและไม่ชอบ โดยหากยึดตาม Favor Level คือแบ่งเป็นตามตารางดังนี้

ตาราง 4.3 Favor Level 2 ระดับ

Class	Range
Dislike	0.00-0.25
Like	0.26-1.00

ได้ผลการทดลองดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.4 ผลการทดลองเมื่อกำหนดค่า Output เป็น 2 ค่า

Volunteer No.	ANN (%)
1	69.45
2	50.73
3	69.45
4	60.59
5	46.3
6	65.51
7	63.54
8	60.59
9	64.03
10	51.23
Average	60.14
StdDev	8.09

4.5.2 การทดลองที่ 2

กำหนดค่า Output เป็น 4 ค่า คือชอบมาก, ชอบ, เฉยๆและไม่ชอบ โดยหากยึดตาม Favor Level คือแบ่งเป็น

ตาราง 4.5 Favor Level 4 ระดับ

Class	Range
Dislike	0.00-0.25
Average	0.26-0.50
Favor	0.51-0.75
Very favor	0.76-1.00

ได้ผลการทดลองดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.6 ผลการทดลองเมื่อกำหนดค่า Output เป็น 4 ค่า

Volunteer No.	ANN (%)
1	28.57
2	34.48
3	46.8
4	40.39
5	38.92
6	40.88
7	61.58
8	57.63
9	46.31
10	44.34
Average	43.99
StdDev	9.91

จากผลการทดลองทั้งสองตาราง จะเห็นได้ว่า การทดลองเมื่อ Output เป็น 2 ค่าทำให้มีโอกาสในการที่ Artificial Neural Network ตอบถูกมากกว่า ดังนั้น ในการใช้งานจริงจะเหลือ Output 2 ค่า คือ Dislike และ Like เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

5.1 บทสรุปของโครงงาน

Smart Music Player เป็นแอพลิเคชั่นที่เก็บข้อมูลการฟังเพลงของผู้ใช้งาน โดยให้รบกวนผู้ใช้ให้น้อยที่สุด เมื่อเก็บข้อมูลผู้ใช้ได้จำนวนหนึ่งแล้วส่งไปที่ฝั่ง Back-End Server เพื่อให้ Artificial Neural Network ทำการเทรนข้อมูล จากนั้นจะนำเพลงที่ผู้ใช้ไม่เคยฟัง มาทำนายว่าผู้ใช้จะชอบฟังหรือไม่ แล้วนำเพลงที่มีผลการทำนายว่าผู้ใช้ชอบฟัง ส่งกลับมาเป็น Playlist เพลงที่แนะนำให้ผู้ใช้ โดยในการเก็บข้อมูลนั้น แอพพลิเคชั่นจะเก็บเวลาที่ฟังเพลงแต่ละเพลงของผู้ใช้เพื่อดูว่าผู้ใช้มีแนวโน้มจะชอบเพลงรูปแบบใด เนื่องจากพฤติกรรมการฟังเพลงโดยส่วนใหญ่ของผู้ใช้นั้น หากเจอเพลงที่ไม่ชอบ ผู้ใช้อาจกดข้ามหรือเปลี่ยนไปเป็นเพลงอื่นๆ ที่ผู้ใช้ต้องการฟัง ทำให้เวลาที่ฟังเพลงที่ไม่ได้มีความชอบเป็นพิเศษนั้นเป็นเวลาน้อย จึงสามารถคาดเดาได้ว่าผู้ใช้ไม่ได้ชอบเพลงที่มีรูปแบบดังกล่าว

5.2 ผลที่ได้จากการทำโครงงาน

1) ได้รับความรู้ในการพัฒนาแอพพลิเคชั่นบนระบบปฏิบัติการแอนดรอยด์
2) ได้นำความรู้ในการเขียนโปรแกรมและออกแบบระบบมาใช้งาน
3) ได้นำความรู้ในการจัดการและออกแบบ Database มาใช้งาน
4) ได้รับความรู้ในการทำ Streaming Application
5) มีความคิดสร้างสรรค์ในการออกแบบ User Interface และ User Experience

5.3 ปัญหาและอุปสรรคที่เกิดจากการทำโครงงาน

1) ในการเก็บข้อมูลของอาสาสมัครนั้น การหาอาสาสมัครที่ใช้ Smartphone ที่มีระบบปฏิบัติการ Android ซึ่งจำเป็นในการรันแอพพลิเคชั่นที่สร้างมาเพื่อเก็บข้อมูลตัวอย่างการฟังเพลงค่อนข้างยาก จึงทำให้ลำบากในการเก็บข้อมูลการทดสอบสมมติฐาน
2) ในการคัดเลือกเพลงที่จะนำไปให้อาสาสมัครทดลองฟังและนำไปเทรนนั้น การจำแนกเพลงค่อนข้างยาก เนื่องจากเพลง 1 เพลง สามารถมีได้หลากหลายประเภท เช่นเพลงหนึ่งเพลงมีประเภทของเพลงเป็น ทั้ง Pop และ Rock รวมถึงในระบบที่มีการให้บริการเกี่ยวกับเพลงหลายๆ ระบบ ยังมีการจำแนกประเภทของเพลงต่างกันอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 แนวทางในการพัฒนาต่อ

1) แอพพลิเคชั่นสามารถใช้งานได้หลากหลายแพลตฟอร์มมากขึ้น
2) พัฒนา Artificial Neural Network ให้มีความแม่นยำเพิ่มมากขึ้น
3) เพิ่มจำนวนเพลงที่ให้บริการ
4) อัพเกรดระบบการค้นหาเพลง ให้เจอเพลงที่ผู้ใช้มีโอกาสชอบ มากขึ้น
5) ย้ายการเก็บข้อมูลต่างๆ ของผู้ใช้ ลงบน Server ทั้งหมด เพื่อง่ายต่อการใช้งานของ User
6) ปรับเปลี่ยน UI และ UX ของแอพพลิเคชั่นให้เป็นมิตรกับผู้ใช้มากขึ้น
7) เพิ่ม Security ในการเชื่อมต่อ และใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Michael Negnevitsky. 2005. **Artificial Intelligence A Guide to Inteeligent Systems.** England: Pearson Education Limited

International Data Corporation. **Smartphone OS Market Share, 2016 Q2.** [Online]. Available:http://www.idc.com/prodserv/smartphone-os-market-share.jsp

Obaro Ogbo, 2555. **7 reasons why you should develop apps for Android rather than iOS**. [Online]. Available:http://www.androidauthority.com/develop-apps-for-android-rather-than-ios-607219/

Musicgenreslist. **The most comprehensive list of genres of music available on the Internet.** [Online]. Available:http://www.musicgenreslist.com/

Liz Diamond-Manlusoc. **Tempo: Definition and Uses in Musical Forms**. [Online]. Available:http://study.com/academy/lesson/tempo-definition-and-uses-in-musical-forms.html

Spotify. **Get Audio Features for Several Tracks**. [Online]. Available:https://developer.spotify.com/web-api/get-several-audio-features/

Android. **Features:Everything you need to build on Android**. [Online]. Available:https://developer.android.com/studio/features.html?hl=eo

Jetbrains. **Features-MAKING DEVELOPMENT AN ENJOYABLE EXPERIENCE**. [Online]. Available:https://www.jetbrains.com/idea/features/

Wikipedia. **เซลล์ประสาท.** [Online]. Available:https://th.wikipedia.org/wiki/เซลล์ประสาท

Spring. **Spring Framework**. [Online]. Available:http://projects.spring.io/spring-framework/

ORACLE. **ORACLE DATA SHEET-MySQL ENTERPRISE EDITION**. [Online]. Available:https://www.mysql.com/products/enterprise/mysql-datasheet.en.pdf

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Arnondora. **REST API**. [Online]. Available: https://www.arnondora.in.th/dev-tip-rest-api-in-php-httpful/

Docker. **What is Docker** [Online]. Available: https://www.docker.com/what-docker

Amazon Web Service. **AWS Documentation**. [Online]. Available: https://aws.amazon.com/documentation/

HTTP Live Service. **HTTP Live Streaming Overview**. [Online]. Avialable: https://developer.apple:com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/Introduction/Introduction.html

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้