

Project/Indep study title line 1
Project/Indep title line 2 (optional)

Mr./Ms. Firstname1 Lastname1
Mr./Ms. Firstname2 Lastname2
Mr./Ms. Firstname3 Lastname3

A Project Submitted in Partial Fulfillment
of the Requirements for
the Degree of Bachelor of Engineering (Computer Engineering)
Faculty of Engineering
King Mongkut's University of Technology Thonburi
202x

Project Committee

..... (Assoc.Prof. My main advisor name , Ph.D.)	Project Advisor
..... (Assoc.Prof. My Co-advisor name, Ph.D.)	Project Co-Advisor
..... (Asst.Prof. Committee2, Ph.D.)	Committee Member
..... (Asst.Prof. Committee3, Ph.D.)	Committee Member

Project Title	Project/Indep study title line 1 Project/Indep title line 2 (optional)
Credits	3
Member(s)	Mr./Ms. Firstname1 Lastname1 Mr./Ms. Firstname2 Lastname2 Mr./Ms. Firstname3 Lastname3
Project Advisor	Assoc.Prof. My main advisor name , Ph.D.
Co-advisor	Assoc.Prof. My Co-advisor name, Ph.D.
Program	Bachelor of Engineering
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	202x

Abstract

In a multihop ad hoc network, the interference among nodes is reduced to maximize the throughput by using a smallest transmission range that still preserve the network connectivity. However, most existing works on transmission range control focus on the connectivity but lack of results on the throughput performance. This paper analyzes the per-node saturated throughput of an IEEE 802.11b multihop ad hoc network with a uniform transmission range. Compared to simulation, our model can accurately predict the per-node throughput. The results show that the maximum achievable per-node throughput can be as low as 11% of the channel capacity in a normal set of α operating parameters independent of node density. However, if the network connectivity is considered, the obtainable throughput will reduce by as many as 43% of the maximum throughput.

Keywords: Multihop ad hoc networks / Topology control / Single-Hop Throughput

หัวข้อปริญญานิพนธ์	หัวข้อปริญญานิพนธ์บรรทัดแรก หัวข้อปริญญานิพนธ์บรรทัดสอง
หน่วยกิต	3
ผู้เขียน	นายสมศักดิ์ คอมพิวเตอร์ นางสาวสมศรี คอมพิวเตอร์2 นางสาวสมปอง คอมพิวเตอร์3
อาจารย์ที่ปรึกษา	รศ.ดร.ที่ปรึกษา วิทยานิพนธ์ รศ.ดร.ที่ปรึกษา วิทยานิพนธ์ร่วม
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	256x

บทคัดย่อ

การวิจัยครั้งนี้มีวัตถุประสงค์ เพื่อศึกษาความพึงพอใจในการให้บริการงานทั่วไปของสำนักวิชา พื้นฐานและภาษา เพื่อเปรียบเทียบระดับความพึงพอใจต่อการให้บริการงาน ทั่วไปของสำนักวิชาพื้นฐานและภาษา ของนักศึกษาที่มาใช้บริการสำนักวิชาพื้นฐานและภาษา สถาบัน เทคโนโลยีไทย-ญี่ปุ่น จาแนกตามเพศ คณะ และชั้นปีที่ศึกษา เพื่อศึกษาปัญหาและข้อเสนอแนะของ นักศึกษา มาเป็นแนวทางในการพัฒนาและปรับปรุงการให้บริการของสำนักวิชาพื้นฐานและภาษา

คำสำคัญ: การชูปเคลือบด้วยไฟฟ้า / การชูปเคลือบผิวเหล็ก / เคลือบผิวรังสี

กิตติกรรมประกาศ

ขอบคุณอาจารย์ที่ปรึกษา กรรมการ พ่อแม่พี่น้อง และเพื่อนๆ คนที่ช่วยให้งานสำเร็จ ตามต้องการ

สารบัญ

หน้า

สารบัญตาราง

ตารางที่

หน้า

สารบัญรูปภาพ

รูปที่

หน้า

สารบัญสัญลักษณ์

SYMBOL

α	Test variable
λ	Interarival rate
μ	Service rate

UNIT

m^2
jobs/ second
jobs/ second

สารบัญคำศัพท์ทางเทคนิคและคำย่อ

ABC	=	Adaptive Bandwidth Control
MANET	=	Mobile Ad Hoc Network

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ

โรคการบกพร่องทางการเรียนรู้ในเด็ก (Learning disorder, LD) คือ ความผิดปกติทางการเรียนรู้ที่เกิดจากการทำงานผิดปกติของสมอง ทำให้ผลการเรียนของเด็กต่ำกว่าศักยภาพที่แท้จริง โดยแบ่งออกเป็น 3 ประเภทตามความผิดปกติของกระบวนการเรียนรู้ที่แสดงออก นั่นคือ ความบกพร่องด้านการอ่าน ความบกพร่องทางการเขียนสะกดคำ และความบกพร่องทางด้านคณิตศาสตร์ โดยเด็กที่มีความบกพร่องด้านการอ่านจะไม่สามารถจดจำพยัญชนะ สระ และยังไม่สามารถสะกดคำได้จึงเป็นสาเหตุให้ เกิดการอ่านออกเสียงไม่ชัด ไม่สามารถผันวรรณยุกต์ได้ หรืออ่านไม่ออก ส่วนความบกพร่องด้านที่สอง คือ การเขียนสะกดคำ ความบกพร่องด้านนี้สามารถพบได้ร่วมกับความบกพร่องด้านการอ่าน เด็กมีความบกพร่องในการสะกดพยัญชนะ สระ หรือ วรรณยุกต์ จึงทำให้เกิดการเขียนหนังสือที่ไม่ถูกต้อง และความบกพร่องสุดท้ายคือ ความบกพร่องด้านคณิตศาสตร์ ลักษณะของเด็กประเภทนี้คือ ขาดทักษะการเข้าใจตัวเลข และจะเกิดการนับจำนวนหรือบวกคูณลบเลขผิด จึงไม่สามารถทำให้คำนวณเลขได้ สาเหตุของโรคการบกพร่องทางการเรียนรู้ที่เกิดจากการทำงานผิดปกติของสมองมีได้หลายสาเหตุด้วยกัน เช่น การทำงานของสมองบางตำแหน่งบกพร่อง กรรมพันธุ์ หรือความผิดปกติของโครโมโซม อ้างอิงจากข้อมูลที่ได้มาจาก พญ.วินัดดา ปิยะศิลป์ในพ.ศ. 2554 คาดว่ามีเด็กที่เป็นโรคการบกพร่องทางการเรียนรู้ หรือ LD (Learning Disorders) ประมาณ 500,000 คน ในช่วงที่เก็บข้อมูลสถิตินั้นมีอัตราเด็กเกิดใหม่ถึง 800,000 คนต่อปี แล้วคาดว่ามีโอกาสที่เด็กเป็น LD 40,000 คนต่อปี จากข้อมูลข้างต้นทำให้ทราบว่าเด็กที่เป็นโรคการบกพร่องทางการเรียนรู้มีจำนวนมาก โดยในปัจจุบันเด็กสามารถเข้ารับการบำบัดทดสอบเพื่อวินิจฉัยโรคการบกพร่องทางการเรียนรู้ได้ ซึ่งจะมีบุคลากรทางการแพทย์ควบคุมการทำแบบทดสอบและจำเป็นต้องให้แพทย์ผู้เชี่ยวชาญเป็นผู้วินิจฉัยกระบวนการนี้ใช้ระยะเวลานาน เนื่องจากบุคลากรทางการแพทย์มีจำกัด ทำให้ไม่สามารถรองรับเด็กเข้ามาทำแบบทดสอบได้เป็นจำนวนมากต่อวัน ซึ่งหากเด็กได้รับการรักษาที่ล่าช้า อาจจะทำให้ได้ผลลัพธ์การรักษาน้อยลง

จากสาเหตุข้างต้นจึงทำให้กลุ่มผู้พัฒนาจึงนำเสนอ “แอลดีสปอต หรือ ระบบตรวจจับอาการโรคการบกพร่องทางการเรียนรู้ด้านการเขียนสะกดคำ” ผ่านทางภาพการเขียนตัวอักษร สระ และ สะกดคำโดยใช้แอปพลิเคชันซึ่งเด็กต้องทำแบบทดสอบในรูปแบบของเกมด้วยการเขียนตัวอักษร สระ และสะกดคำ จากนั้นภาพแบบทดสอบจะถูกส่งให้ระบบแอลดีสปอต เพื่อคำนวณคะแนนและวินิจฉัยโรคการบกพร่องทางการเรียนรู้เบื้องต้น และนำไปแสดงผลในแอปพลิเคชันให้บุคลากรทางการแพทย์และผู้ปกครองสามารถดูผลลัพธ์ได้ ซึ่งในส่วนของการวินิจฉัยนั้นได้อ้างอิงหลักการวิเคราะห์ข้อมูลจากแพทย์มาใช้ในระบบวิเคราะห์ที่จะพัฒนา

แอลดีสปอต นั้นจะช่วยให้การวินิจฉัยโรคการบกพร่องทางการเรียนรู้เบื้องต้นในเด็กสามารถเข้าถึงได้ง่ายขึ้นโดยที่เด็กจะสามารถทำแบบทดสอบเบื้องต้นได้ผ่านทางแอปพลิเคชันก่อนที่จะเดินทางมาที่โรงพยาบาลเพื่อที่จะลดความซับซ้อนและระยะเวลาในการรอการวินิจฉัยเบื้องต้น อีกทั้งยังลดขั้นตอนหรือหน้าที่ของแพทย์หรือบุคลากร

1.2 วัตถุประสงค์

- เพื่อพัฒนาระบบวิเคราะห์รูปภาพลายมือเขียนของเด็กเพื่อวินิจฉัยโอกาสเป็นโรคการบกพร่องทางการเรียนรู้ด้านการเขียนและสะกดคำในเด็กได้อย่างแม่นยำ
- เพื่อพัฒนาแอปพลิเคชันที่อยู่ในรูปแบบเกมส์เพื่อดึงดูดความสนใจจากเด็ก และเด็กสามารถทำแบบทดสอบจนจบได้
- เพื่อลดความซับซ้อนและระยะเวลาในการรอเพื่อวินิจฉัยโรคการบกพร่องทางการเรียนรู้เบื้องต้นได้
- เพื่อช่วยให้บุคลากรทางการแพทย์สามารถทำงานได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

1.3 ขอบเขตของโครงการ

- แอปพลิเคชันในรูปแบบของเกมส์ที่รองรับระบบปฏิบัติการแอนดรอยด์ (Android) และ ไอโอเอส (iOS) ซึ่งรองรับเพียงภาษาไทย
- ระบบวิเคราะห์รูปภาพลายมือของเด็กซึ่งถูกสร้างขึ้นมาจากข้อมูลแบบทดสอบการเขียนของเด็กที่เป็นโรคการบกพร่องทางการเรียนรู้ จากหน่วยตรวจโรคจิตเวชเด็กและวัยรุ่น ภาควิชาจิตเวชศาสตร์ คณะแพทยศาสตร์ศิริราชพยาบาล
- ระบบวิเคราะห์รูปภาพลายมือเขียนของเด็กจะต้องรับรูปภาพลายมือของเด็ก โดยการเขียนผ่านทางแอปพลิเคชันที่ได้สร้างไว้

- ผลลัพธ์จะออกมาในรูปแบบจำนวนความผิดพลาดจากที่เขียนผิด และความน่าจะเป็นว่าเด็กมีความน่าจะเป็นที่โรคการบกพร่องทางการเรียนรู้เท่าใด โดยตัวระบบจะเรียนรู้จากภาพการเขียนทดสอบของเด็กที่เป็นโรคการบกพร่องทางการเรียนรู้และภาพการเขียนทดสอบของเด็กที่ไม่เป็นโรคการบกพร่องทางการเรียนรู้ จาก หน่วยตรวจโรคจิตเวชเด็กและวัยรุ่น ภาควิชาจิตเวชศาสตร์ คณะแพทยศาสตร์ ศิริราชพยาบาล
- ระบบจะแสดงผลลัพธ์ที่ได้จากการวินิจฉัยในแอปพลิเคชัน โดยที่ผู้ปกครองและบุคลากรทางการแพทย์จะสามารถเข้ามาดูผลลัพธ์แล้วนำไปใช้ประโยชน์ต่อได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

โครงการนี้จะเป็นประโยชน์กับใคร ยังไง ทั้งในเชิงรูปธรรมและนามธรรม ในปัจจุบันหรือในอนาคตถ้านำไป ต่อยอด

1.5 ตารางการดำเนินงาน

บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง

อธิบายทฤษฎี องค์ความรู้หลักที่ใช้ในงาน งานวิจัยที่นำมาใช้ในโครงงาน หรือเปรียบเทียบผลิตภัณฑ์ที่มีอยู่ในท้องตลาด Explain theory, algorithms, protocols, or existing research works and tools related to your work.

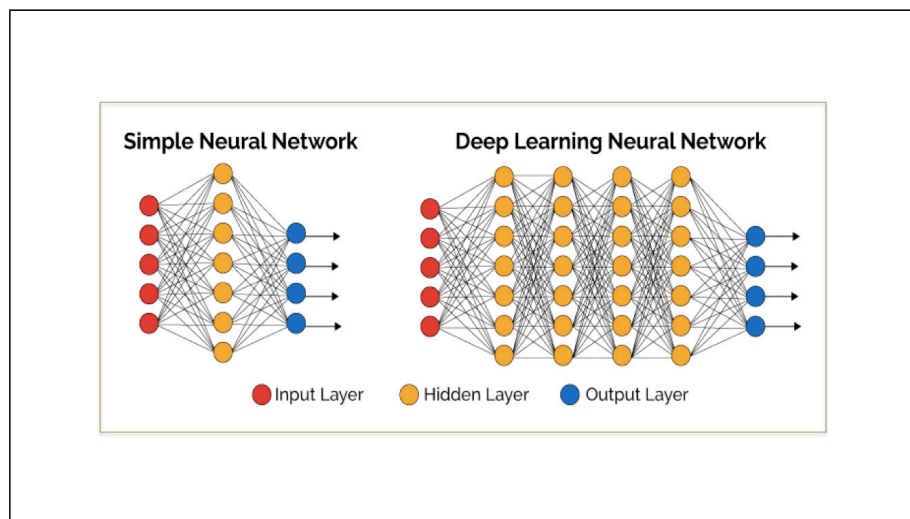
2.1 Core concept แนวคิดหลัก

เนื่องจากตัวระบบที่เราสร้างขึ้นมาเพื่อวินิจฉัยโรคการเรียนรู้บกพร่องในเด็กนั้นจะต้องทำการเรียนรู้ข้อมูลลักษณะจุดเด่นต่างๆ ของภาพผลแบบทดสอบการเรียนรู้บกพร่องในเด็กว่า มีลักษณะเด่นใดจึงจำแนกว่าเด็กคนนั้นมีโอกาสเป็นโรคการเรียนรู้บกพร่องในเด็ก จากการค้นคว้าหาข้อมูลจึงพบว่าเราจำเป็นต้องใช้ความรู้ในเรื่องของ Convolutional Neural Network ซึ่งเหมาะแก่การทำการจำแนกประเภทของรูปภาพ และเป็นส่วนหนึ่งของเรื่องการเรียนรู้เชิงลึกของคอมพิวเตอร์ (deep learning)

2.1.1 การเรียนรู้เชิงลึกของคอมพิวเตอร์ (deep learning)

การเรียนรู้เชิงลึกของคอมพิวเตอร์ (deep learning) เป็นหนึ่งในสาขาย่อยของ machine learning เป็นศาสตร์ที่พูดถึงการจำลองการทำงานของระบบโครงข่ายประสาทมนุษย์ โดย จะมีการแบ่งการทำงานข้างในเป็น layer ต่างๆ โดยเราจะมองเป็นสามส่วนหลักๆ ได้แก่

1. Input Layer มีหน้าที่สำหรับการรับข้อมูลป้อนเข้าโครงข่ายประสาท จากผู้ใช้ เช่น รูปภาพ
2. Hidden Layer มีหน้าที่สำหรับการประเมินข้อมูลที่ป้อนเข้ามา เพื่อหาข้อมูลต่างๆ ที่ใช้ในการจำแนกประเภทโดยตัว Hidden Layer นั้นสามารถมีได้มากกว่า 1 ชั้น
3. Output layer เป็นชั้นสุดท้ายมีหน้าที่สำหรับรับข้อมูลจาก Hidden Layer เพื่อใช้ในการบอกว่าท้ายที่สุดแล้วตัวข้อมูลที่รับเข้ามานั้นถูกจำแนกอยู่ในประเภทใด

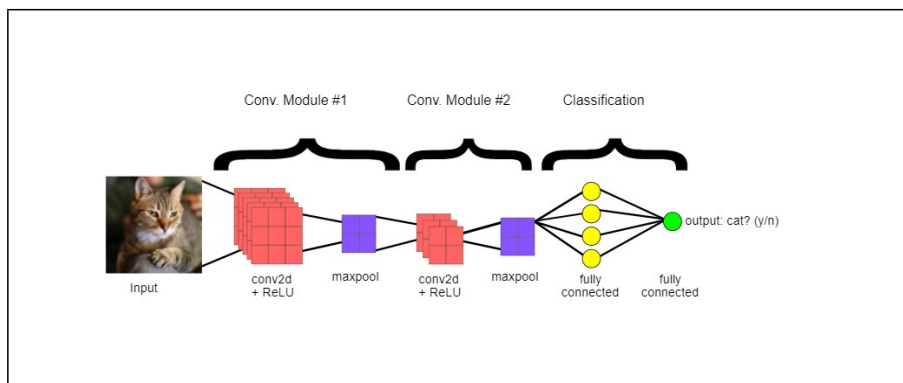


รูปที่ 2.1 ตัวอย่าง layer ของ การเรียนรู้เชิงลึกของคอมพิวเตอร์

[ที่มา : <https://verneglobal.com/news/blog/deep-learning-at-scale/>]

2.1.2 โครงข่ายประสาทเทียมแบบสังวัตนาการ (Convolutional Neural Network)

การจำแนกประเภทรูปภาพ สำหรับทางด้านการแพทย์กำลังเป็นที่สนใจ โครงข่ายประสาทเทียมแบบสังวัตนาการ หรือ CNN เลยได้รับความนิยมมากขึ้นโดย CNN เป็นรูปแบบหนึ่งของการเรียนรู้เชิงลึกของคอมพิวเตอร์ ที่เกิดจากการนำแนวคิดของ Neural Network มาเพิ่มในส่วนของ Convolutional layer ซึ่งเหมาะแก่การหาลักษณะต่างๆ ของข้อมูลต่างๆ เช่น รูปภาพ โดยตัวของ CNN นั้นจะประกอบด้วยหลายๆ layer ด้วยกัน

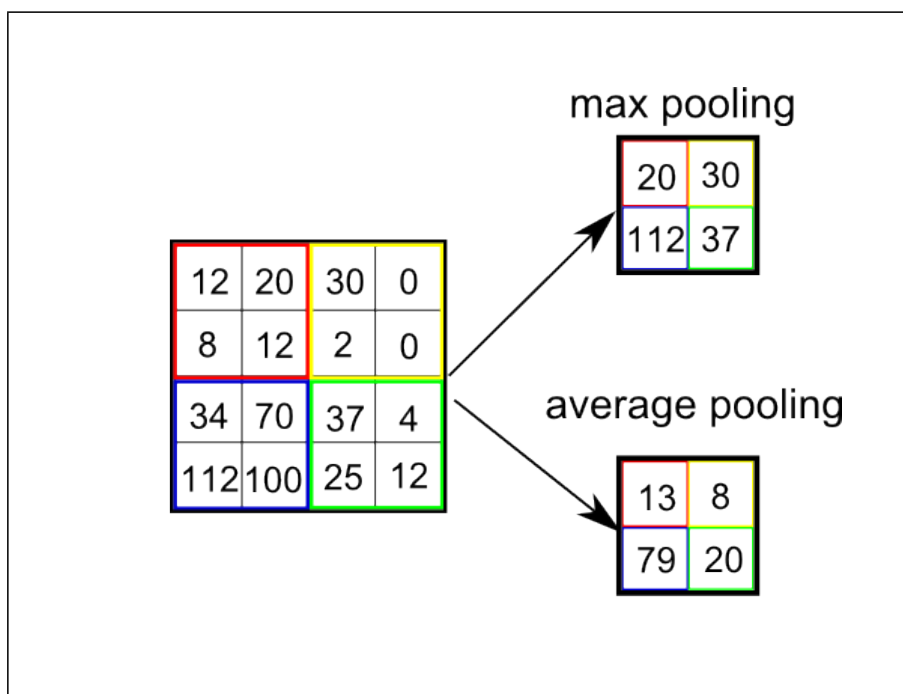


รูปที่ 2.2 แสดงตัวอย่างโครงข่าย CNN ที่ประกอบด้วย 1. convolutional layer 2. pooling layer 3. fully connected layer

[ที่มา : <https://developers.google.com/machine-learning/practica/image-classification/convolutional-neural-networks>]

โดย CNN จะมี layer หลักๆได้แก่

1. Convolutional layer ซึ่งมีหน้าที่ในการประมวลผลภาพเพื่อหาคุณลักษณะต่างๆ เช่น สี ขอบ ด้วย filters และนำไปเข้า activate function เพื่อแปลงผลลัพธ์ให้กลายเป็นข้อมูลนำเข้าสำหรับ layer ถัดไป โดยมี activate function ที่ได้รับความนิยมในการทำ การจำแนกประเภทรูปภาพ คือ Rectified Linear Unit (ReLU)
2. Pooling layer มีหน้าที่ในการลดมิติของข้อมูลที่เราได้จาก Convolutional layer ให้เล็กลง และคงไว้ซึ่งข้อมูลที่เป็นเพื่อที่จะ ทำให้การประมวลผลเร็วขึ้น โดยจะมีสองวิธีหลักๆได้แก่ Max pooling และ Mean pooling โดย Max pooling จะทำการเลือก ค่าที่มากที่สุดในขอบเขตที่สนใจ และ Mean Pooling จะทำการหาค่าเฉลี่ยของขอบเขตที่สนใจแล้วนำไปใช้ต่อ ดังรูป ??



รูปที่ 2.3 ตัวอย่างการทำ Max pooling และ mean pooling

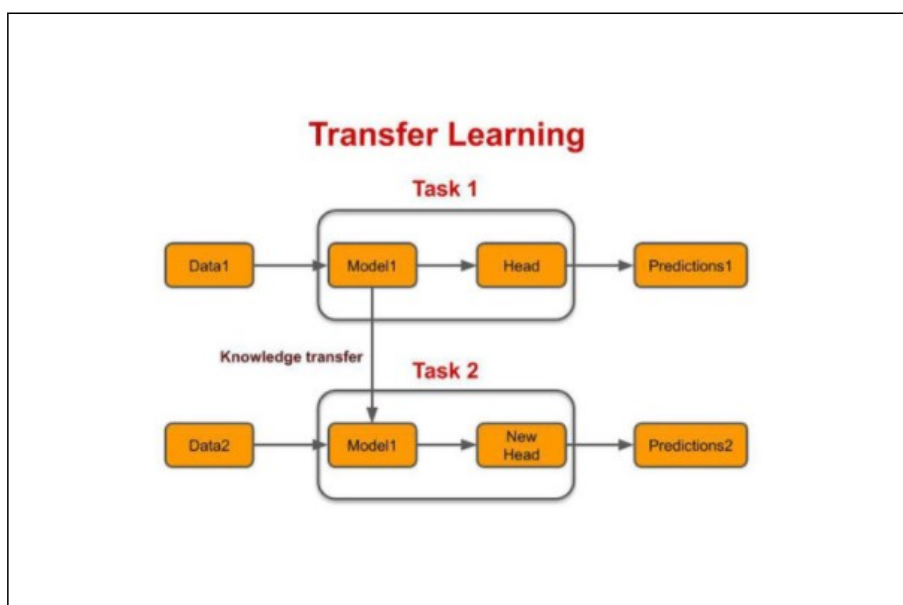
[ที่มา : <https://stackoverflow.com/questions/44287965>]

3. fully connected layer มีหน้าที่ในการรวบรวม output จาก layer ก่อนหน้า ที่ได้ทำการหา คุณลักษณะต่างๆมารวมและกำหนด ให้ผลลัพธ์ของ layer นี้มีจำนวนเท่ากับ จำนวนประเภทที่เราต้องการจำแนกรูปภาพ เพื่อดูว่าผลลัพธ์ท้ายสุดเราจำแนกรูปภาพนั้น ได้อยู่ในประเภทไหนซึ่ง CNN จะประกอบด้วยหลายๆ layer นี้เรียงกันไปจนถึง output ตามความเหมาะสมของ โมเดลนั้นๆ

และสามารถปรับ parameter ของแต่ละ layer ได้เพื่อให้การจำแนกประเภทนั้นออกมาแม่นยำที่สุด โดยในโครงการนี้ เราจะเลือกใช้ Convolutional neural network ในการสร้าง โมเดลเพื่อจำแนกข้อมูลภาพถ่ายของเรา

2.1.3 Transfer Learning

ในการทำ Convolutional neural network นั้น เราจำเป็นจะต้องออกแบบตัว layer และ parameter ต่างๆให้เหมาะสมเพื่อให้ได้ความแม่นยำในการจำแนกที่สูง ซึ่งเราจำเป็นต้องใช้ข้อมูลจำนวนมากในการ train ให้โมเดล CNN ของเรานั้นมีความแม่นยำ แต่ว่า Transfer Learning คือการที่เรานำ โมเดล CNN ที่มีการสร้างขึ้นมาไว้แล้วจากข้อมูลอื่น มาปรับแต่งในส่วน of fully connected layer เองใหม่ให้เหมาะสมกับข้อมูลที่เราจะทำการจำแนก ซึ่งจะทำให้เราประหยัดเวลาในการสร้างโมเดล และลดจำนวนข้อมูลที่ใช้ในการสร้างโมเดลเพื่อที่จะทำให้โมเดลนั้นมีความแม่นยำ

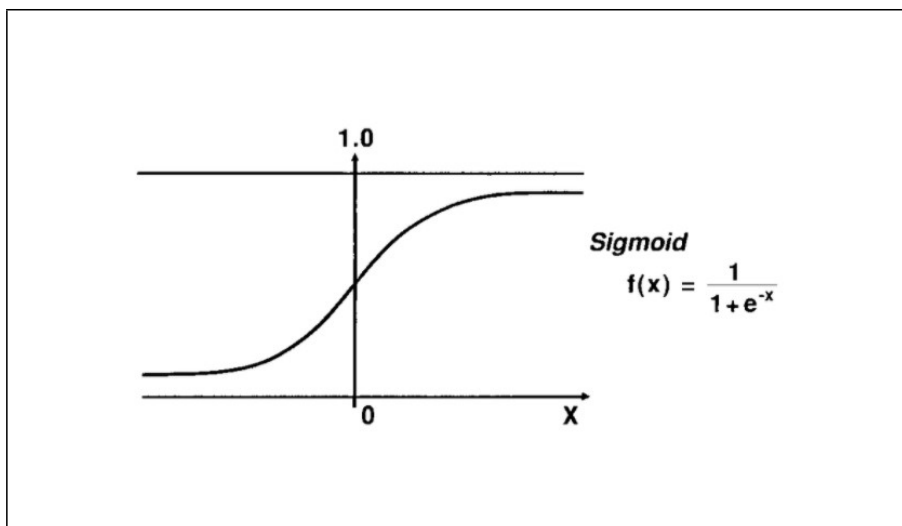


รูปที่ 2.4 ภาพอธิบายตัวอย่างของ Transfer Learning

[ที่มา : <https://www.topbots.com/transfer-learning-in-nlp/>]

2.1.4 Activate Function

Activate function มีหน้าที่ในการปรับผลลัพธ์ ของ neuron ในแต่ละ layer ก่อนจะส่งต่อไปเป็นข้อมูลนำเข้าสำหรับ layer ถัดไป โดย Activate function ที่เป็นที่นิยมคือ sigmoid function เนื่องจากตัว sigmoid function นั้นจะมีผลลัพธ์ออกมาอยู่ในช่วงของ 0 จนถึง 1 ทำให้เหมาะแก่การใช้ทำเรื่องความน่าจะเป็น แต่เนื่องจากกราฟของ sigmoid function เป็นดังรูป ?? เราจะเห็นว่าหากค่า $|x|$ มีค่าสูงมากขึ้นค่าของ sigmoid function จะมีการเปลี่ยนแปลงที่น้อยลง หรือมีค่าอนุพันธ์ที่น้อยลงทำให้การอัปเดตน้ำหนักของตัว Neural network ใน layer แรกๆนั้นมีค่าน้อยจนอาจทำให้การเรียนรู้หยุด ปัญหานี้มีชื่อเรียกว่า Vanishing gradient problem โดยสามารถแก้ไขด้วยการเปลี่ยน activate function ได้ยกตัวอย่างเช่น Rectified Linear Unit หรือ ReLU

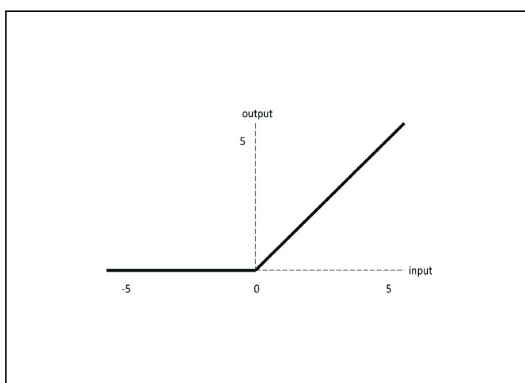


รูปที่ 2.5 ภาพอธิบายตัวอย่างของ Transfer Learning

[ที่มา : <https://www.topbots.com/transfer-learning-in-nlp/>]

2.1.5 Rectified Linear Unit (ReLU)

Rectified Linear Unit หรือ ReLU เป็น activate function ที่กำลังได้รับความนิยมเนื่องจากสามารถแก้ไขปัญหาในเรื่องของ vanishing gradient problem ได้ เพราะกราฟของ ReLU นั้นถ้าค่า x เป็นบวกจะได้ค่าของอนุพันธ์เท่ากับ 1 เสมอทำให้ความชันไม่หาย ซึ่งทำให้ตัวโมเดลของเรานั้นปรับค่าน้ำหนักได้ไวยิ่งขึ้น แต่ก็ยังมีข้อเสียเช่นกันคือผลลัพธ์จะออกมาอยู่ในช่วงตั้งแต่ 0 ถึง อินฟินิตี้ทำให้ไม่สามารถกำหนดขอบเขตได้ หรือผลลัพธ์สำหรับการที่ข้อมูลขาเข้าเป็นเลขติดลบจะเท่ากับ 0 เสมอทำให้ไม่สามารถแปลงค่าผลลัพธ์ที่เท่ากับ 0 กลับมาเป็นข้อมูลขาเข้าได้ เป็นต้น



รูปที่ 2.6 ภาพอธิบายตัวอย่างของ Transfer Learning

[ที่มา : <https://www.topbots.com/transfer-learning-in-nlp/>]

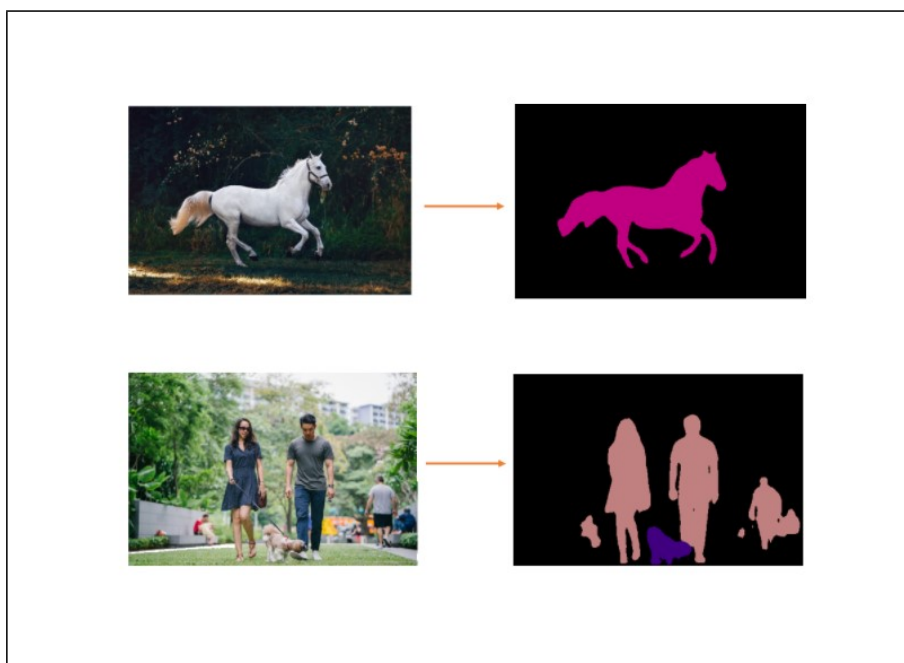
2.1.6 การปรับขนาดรูปภาพ (Image rescale)

การปรับขนาดรูปภาพ (Image rescale) นั้นเป็นส่วนหนึ่งของกระบวนการเตรียมข้อมูลเบื้องต้นสำหรับการทำโมเดล CNN เนื่องจากข้อมูลที่เราได้มาสำหรับการทำโมเดลนั้น อาจจะมีขนาดที่แตกต่างกันรวมถึงมีขนาดที่ใหญ่เกินไป ด้วยเหตุนี้จึงทำให้โมเดลใช้ระยะเวลาในการเรียนรู้นาน เราจึงกำหนดขนาดมาตรฐานและทำการปรับขนาดข้อมูลรูปภาพสำหรับการสร้างโมเดลก่อนที่จะนำไปใช้

2.1.7 การแยกบริเวณรูปภาพ (Image segmentation)

การแยกบริเวณรูปภาพ (Image segmentation) คือการแยกสิ่งที่เราสนใจออกมาจากพื้นหลังของรูปภาพเพื่อใช้ในการทำโมเดลต่อไป ซึ่งถูกนำไปใช้ประโยชน์ในหลายๆด้านด้วยกันได้แก่ การจับตัวหนังสือในภาพ การจับวัตถุแปลกปลอมในรูปภาพ เป็นต้น โดยมีหลายรูปแบบด้วยกันยกตัวอย่างเช่น Region-Based Segmentation, Edge Detection Segmentation เป็นต้น

- Region-Based Segmentation เป็นการแยกวัตถุออกจากภาพด้วยวิธีการใช้ค่า threshold เพื่อปรับภาพที่อยู่ในรูปแบบของ grayscale ให้กลายเป็น binary image โดยให้วัตถุเป็นหนึ่งสี และ พื้นหลังเป็นอีกสีหนึ่ง เพื่อที่เราจะได้รูปร่างของวัตถุขึ้นมา ซึ่งวิธีการเลือกค่า threshold ที่เหมาะสมมีมากมาย ยกตัวอย่างเช่น Otsu's thresholding method
- Edge Detection Segmentation ที่ใช้ในการหาขอบของวัตถุซึ่งใช้หลักการความไม่ต่อเนื่องภายใน pixel ของรูปภาพ โดยจุดที่เกิดการเปลี่ยนแปลงนั้นจะถูกระบุเป็นจุดขอบ
- Output layer เป็นชั้นสุดท้ายมีหน้าที่สำหรับรับข้อมูลจาก Hidden Layer เพื่อใช้ในการบอกว่าท้ายที่สุดแล้วตัวข้อมูลที่รับเข้ามานั้นถูกจำแนกอยู่ในประเภทใด



รูปที่ 2.7 ภาพตัวอย่างการทำ image segmentation บนภาพ

[ที่มา : <https://www.learnopencv.com/applications-of-foreground-background-separation-with-semantic-segmentation/>]

2.1.8 การแปลงรูปภาพเป็นข้อความ (Optical character recognition)

Optical character recognition หรือ OCR คือเทคโนโลยีที่ทำให้เราสามารถจับตัวอักษรที่อยู่ในภาพถ่ายยกตัวอย่างเช่น ภาพสแกนของเอกสาร หรือ สื่อสิ่งพิมพ์ต่างๆ เป็นต้น มาแปลงให้อยู่ในรูปแบบของตัวอักษรดิจิทัล ที่สามารถแก้ไขได้ และง่ายต่อการจัดเก็บนำไปใช้ต่อ ซึ่งเราสามารถนำเทคโนโลยีไปประยุกต์ใช้ได้หลากหลายด้าน เช่น วิเคราะห์ทะเบียนรถยนต์ ด้านการทำระบบค้นหาข้อมูล หรือระบบจัดเก็บรายละเอียดสินค้า

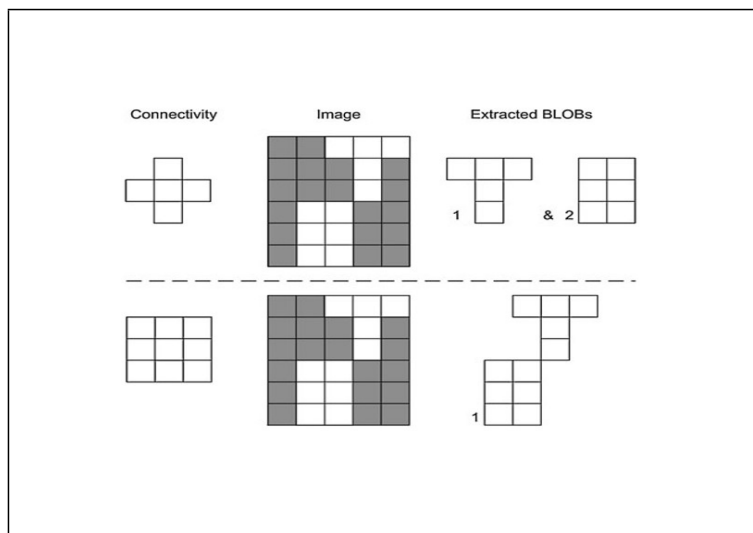


รูปที่ 2.8 ภาพตัวอย่างขั้นตอนการแปลงเอกสารมาอยู่ในรูปแบบข้อมูลด้วยกระบวนการ OCR

[ที่มา : <https://medium.com/states-title/using-nlp-bert-to-improve-ocr-accuracy-385c98ae174c>]

2.1.9 Blob coloring

ใช้ในการทำ OCR ของเราเป็นอัลกอริทึมที่ใช้ในการแบ่งขอบเขตของวัตถุ โดยจะทำการไล่ตั้งแต่ pixel บนสุดของภาพลงมาล่างสุดซึ่งแต่ละ pixel จะทำการจับว่า pixel รอบๆตัวนั้นเป็นสีดำหรือไม่ หากเป็นสีดำก็จะจับให้ pixel เหล่านั้นอยู่ใน label เดียวกัน ซึ่งวิธีการนี้จะทำให้เราสามารถแยกตัวอักษรแต่ละตัวออกจากกันได้ โดยแต่ละตัวก็จะมี label ของตัวมันเอง

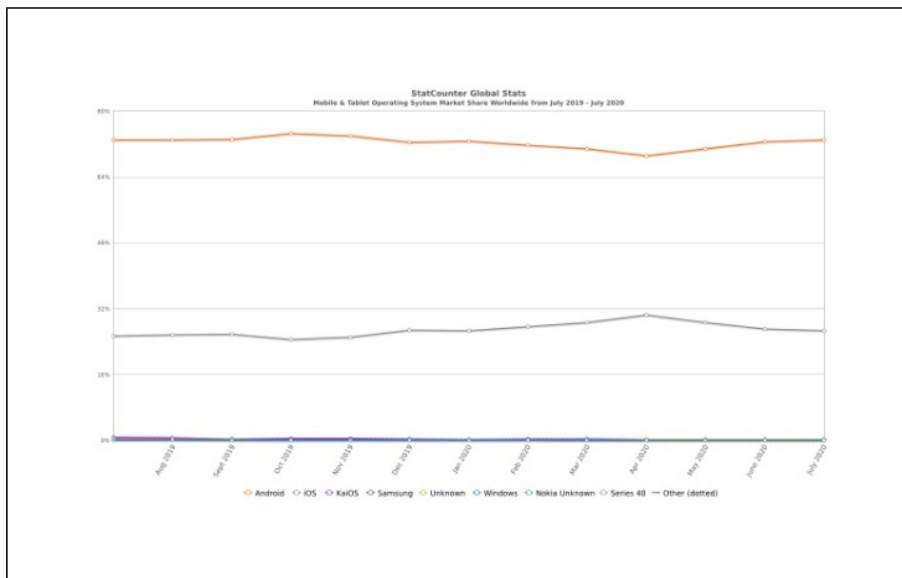


รูปที่ 2.9 ภาพตัวอย่างการทำงานของ Blob coloring

[ที่มา : <http://what-when-how.com/introduction-to-video-and-image-processing/blob-analysis-introduction-to-video-and-image-processing-part-1/>]

2.2 Languages and technologies ภาษาโปรแกรมและเทคโนโลยี

เนื่องด้วยด้วยเป้าหมายของโครงการที่ต้องการพัฒนาแอปพลิเคชันให้สามารถใช้งานได้หลายแพลตฟอร์ม โดยปัจจุบันระบบปฏิบัติการแอนดรอยด์และระบบปฏิบัติการไอโอเอสเป็นระบบปฏิบัติการที่ผู้คนใช้งานมากที่สุด โดยทั้งสองระบบปฏิบัติการครอบครองส่วนแบ่งทางตลาดมากกว่า 98% สำหรับโทรศัพท์มือถือและแท็บเล็ต จากสถิติระบบปฏิบัติการไอโอเอสและระบบปฏิบัติการแอนดรอยด์มีจำนวน



รูปที่ 2.10 ภาพตัวอย่างการทำงานของ Blob coloring

[ที่มา : <http://what-when-how.com/introduction-to-video-and-image-processing/blob-analysis-introduction-to-video-and-image-processing-part-1/>]

ผู้ใช้งานมากมาย ดังนั้นโครงการจึงพัฒนาแอปพลิเคชันให้สามารถใช้งานได้ทั้งสองระบบปฏิบัติการ ซึ่งรูปแบบในการพัฒนาแอปพลิเคชันให้สามารถใช้งานในหลายแพลตฟอร์มได้มีด้วยกันอยู่สองรูปแบบคือ Hybrid Application และ Web Application อย่างไรก็ตาม Hybrid Application สามารถทำงานได้ตามเป้าหมายของโครงการมากกว่าเพราะการเป็นรูปแบบแอปพลิเคชันทำให้สามารถใช้หน้าจอสัมผัสผ่านโทรศัพท์มือถือหรือแท็บเล็ตในการเขียนตัวอักษรได้ โดยมีเฟรมเวิร์กให้พัฒนามากมายเช่น React Native ,ionic และ Flutter เป็นต้น

2.2.1 React Native

React Native คือ เฟรมเวิร์กที่พัฒนาด้วยภาษา JavaScript สำหรับการพัฒนาแอปพลิเคชันสำหรับระบบปฏิบัติการไอโอเอสและระบบปฏิบัติการแอนดรอยด์โดยใช้เทคโนโลยี Cross platform ซึ่งในการพัฒนาด้วย React Native มีข้อดีคือในการพัฒนาสามารถพัฒนาแค่ครั้งเดียวแต่สามารถใช้งานได้ทั้งระบบปฏิบัติการไอโอเอสและระบบปฏิบัติการแอนดรอยด์ ด้วยการจัดการของ JavaScript ให้สามารถสื่อสารกับฝั่ง Native ของระบบปฏิบัติการทั้งสอง จึงได้ผลลัพธ์ออกมาเป็น Native Application ทั้งระบบปฏิบัติการไอโอเอสและระบบปฏิบัติการแอนดรอยด์

2.2.2 Keras

Keras คือเฟรมเวิร์กที่พัฒนาด้วยภาษา Python ที่ถูกสร้างขึ้นเพื่อให้เราสามารถจัดการกับการทำ การเรียนรู้เชิงลึกของคอมพิวเตอร์ (deep learning) ได้อย่างง่าย ข้อดีของ Keras คือ ใช้งานง่าย และสามารถดัดแปลงตัว layer ของ neural network ได้ง่าย โดยในโครงการนี้เราสามารถใส่ Keras ในการออกแบบตัว Layer ต่างๆ ของโมเดลได้รวมทั้งทำการสร้างโมเดลและทำนายด้วยภาษา Python ได้เลย

2.2.3 OpenCV

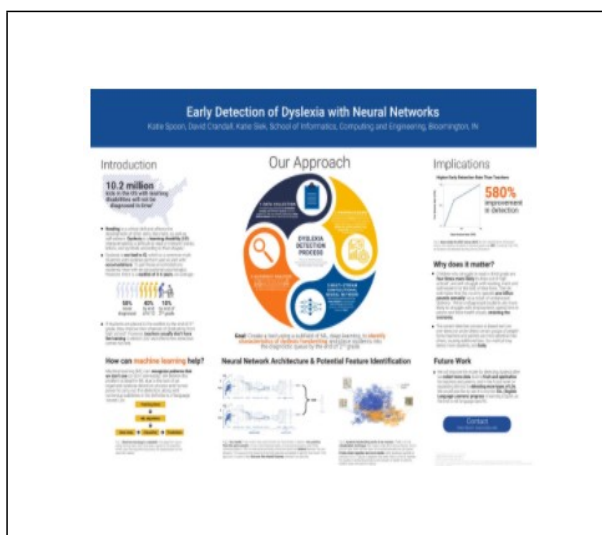
OpenCV เป็น library ที่มีจุดประสงค์เพื่อการแสดงผลด้วยคอมพิวเตอร์แบบเรียลไทม์ (Real - Time) รวมทั้งในส่วนของการทำ Image processing ที่รองรับการใช้งานบนหลายภาษาด้วยกัน โดยหนึ่งในนั้นคือ Python อีกทั้งยังสามารถใช้งานร่วมกับเฟรมเวิร์กการเรียนรู้เชิงลึกต่างๆ อาทิเช่น TensorFlow และ PyTorch เป็นต้น

2.2.4 Django Rest Framework

Django Rest Framework เป็น framework ที่พัฒนาขึ้นมาด้วยภาษา python ไว้ใช้สำหรับการสร้าง api ไว้คุยกับฐานข้อมูล เพื่อให้ website หรือ application สามารถเรียกใช้ตัว api เพื่อขอข้อมูล ซึ่งสามารถพัฒนาได้ง่ายด้วยภาษา python และที่เราเลือกเนื่องจากตัวโมเดลวินิจัยโรคของเราก็คพัฒนาขึ้นจากภาษา python เช่นเดียวกันทำให้สามารถเรียกใช้งานได้ง่าย

2.3 Related research / Competing solutions บทความที่เกี่ยวข้อง

2.3.1 Detecting Dyslexia Using Neural Networks



รูปที่ 2.11 ภาพตัวอย่างการทำงานของ Blob coloring

[ที่มา : <http://what-when-how.com/introduction-to-video-and-image-processing/blob-analysis-introduction-to-video-and-image-processing-part-1/>]

2.3.1.1 การใช้ภาพถ่ายมือในการวินิจฉัยโรค Dyslexia

มีงานจำนวนมากที่วินิจฉัยโรค Dyslexia โดยการใช้ข้อมูลคะแนนการสอบ ประวัติของผู้ทดสอบ หรือ แบบสอบถาม และมีอีกจำนวนหนึ่งที่ใช้ข้อมูลเช่นภาพการทำงานของสมอง หรือ การสื่อสารผ่านทางดวงตาเป็นต้น แต่วรรณกรรมนี้ได้เลือกที่จะใช้ข้อมูลลายมือเนื่องจาก เป็นข้อมูลที่ง่ายต่อการเก็บรวบรวม

2.3.1.2 การประมวลผลภาพ

ในส่วนของการประมวลผลภาพนั้น วรรณกรรมนี้ได้ทำการนำภาพถ่ายมือมาแบ่งเป็นบรรทัด หลังจากนั้นจึงได้นำแต่ละบรรทัดมาแบ่งเป็นอีก 50 ส่วน โดยวิธีการที่ใช้ในการแบ่งบรรทัด คือ Arvanitopoulos Susstrunk's seam carving แล้วจึงนำภาพแต่ละบรรทัดไปแบ่งเป็น 50 ส่วนโดยใช้ขนาด 113*113 ซึ่งยังมีบางภาพที่ยากต่อการทำ แล้วจำเป็นต้องใช้การแก้ไขโดยผู้จัดทำก่อน แต่ไม่ได้มีความยากในการแก้ไขสูง

2.3.1.3 Optical character recognition

Optical character recognition หรือ OCR นั้นเป็นการจับตัวอักษรภายในภาพแล้วจึงนำมาแปลงเป็นค่า วิธีนี้สามารถอ่านได้ว่าในภาพนั้นมีตัวอักษรตัวใดอยู่บ้าง แต่จากการทดลองของวรรณกรรมนี้พบว่า วิธีนี้ไม่เหมาะสมกับการนำมาอ่านภาพถ่ายลายมือของเด็กที่เป็นโรค เนื่องจากลายมือของเด็กนั้นมีความหลากหลายมาก ทำให้วิธีการตรวจจับด้วยระบบ OCR ไม่สามารถตรวจจับได้อย่างแม่นยำ

2.3.1.4 การทำโมเดลวินิจัย

เป็นส่วนที่ให้โมเดลนั้นได้ทำการระบุข้อมูลที่ป้อนเข้ามาเป็นโรค dyslexia หรือไม่ โดยตัวโมเดลนั้นอยู่ในรูปแบบของ โครงข่ายประสาทเทียมแบบสังวัตนาการ หรือ Convolutional Neural Network โดยประกอบด้วย convolutional layer จำนวน 5 layer

max-pooling จำนวน 3 layer fully-connected จำนวน 2 layer และ dropout layer จำนวน 1 layer หลังจากนั้นจึงได้แบ่งข้อมูลแบบ 3:1:1 โดยเป็นข้อมูลในส่วนของการ train 60

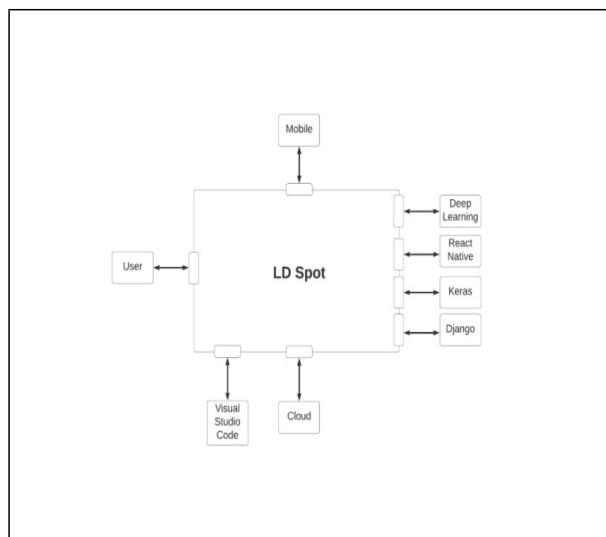
วรรณกรรมนี้เป็นวรรณกรรมที่ดีและมีคล้ายกับว่ามีข้อเสนอแนะว่าไม่ควรใช้อะไรบ้าง รวมถึงช่วยเรื่องการคิดระบบการทำงานว่าควรมีขั้นตอนแบบใดจากวิธีแรกถึงวิธีสุดท้าย เห็นได้ว่ามีหลายวิธีอย่างมากที่วินิจฉัยเรื่องของการเป็น LD แต่ว่าโปรเจ็คของเขาได้เลือกวิธีการวินิจฉัย ผ่านลายมือเนื่องจากสามารถเก็บรวบรวมได้ง่าย หลังจากนั้นนำภาพมาแบ่งเป็น 50 ส่วนตามขนาด 113*113 แต่ก็พบว่ายังมีบางภาพที่สามารถตัดแบ่งได้ยาก และใช้ OCR ในการระบุด้วย ผลออกมาคือมีความแม่นยำที่น้อย

บทที่ 3 วิธีการดำเนินงาน

Explain the design (how you plan to implement your work) of your project. Adjust the section titles below to suit the types of your work. Detailed physical design like circuits and source codes should be placed in the appendix.

3.1 Project Functionality

3.1.1 System Architecture



รูปที่ 3.1 ภาพตัวอย่างการทำงานของ Blob coloring

3.1.2 System requirements

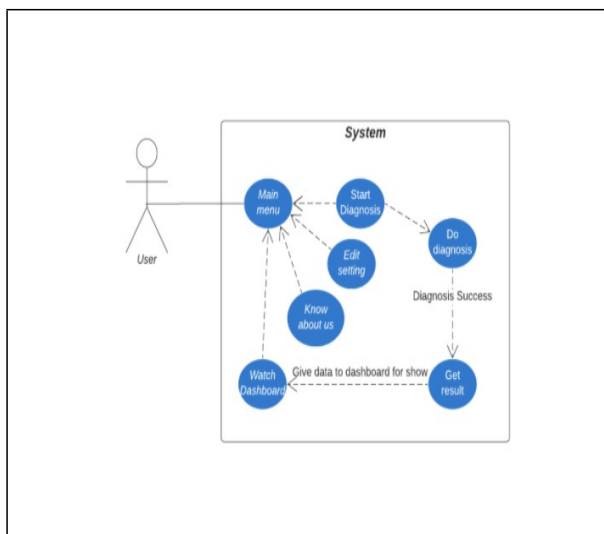
- รองรับระบบปฏิบัติการแอนดรอยด์ตั้งแต่ 4.1 ขึ้นไป
- รองรับระบบปฏิบัติการไอโอเอสตั้งแต่ 10.0 ขึ้นไป
- รองรับระบบสัมผัสหน้าจอ
- สามารถดูผลการวินิจฉัยย้อนหลังได้
- อนุญาตให้เก็บผลการวินิจฉัยบนระบบได้

3.1.3 (Use cases)

เมื่อผู้ใช้เข้าสู่แอปพลิเคชันของเราสิ่งแรกที่พบคือ หน้าหลัก (Main menu) เพื่อที่สามารถเชื่อมหรือใช้ฟังก์ชันอื่น ๆ โดยมี 4 ฟังก์ชัน อย่างแรกเลย การวินิจฉัย(Start Diagnosis) เมื่อผู้ใช้เลือกใช้ฟังก์ชันนี้ ทำให้เริ่มการวินิจฉัยโดยมีลักษณะคล้ายเกมส์ ให้เขียนตัวอักษร สระ และสะกดคำ จนเสร็จสมบูรณ์จากนั้น ก็วิเคราะห์ออกมาจากคำตอบที่ได้ตอบระหว่างเกมส์ เพื่อให้ได้ผลลัพธ์รวมถึง ส่งผลลัพธ์นั้นไปบอร์ดสถิติ (Dashboard) เพื่อที่แสดงข้อมูลให้ผู้ใช้คนอื่น ๆ เห็น นอกจากนี้ยังมีหน้าตั้งค่า(Setting) หน้าเกี่ยวกับเรา (About us)

3.2 โครงสร้างซอฟต์แวร์

ในส่วนของการใช้งานระบบ LDSpot นั้นจะแบ่งเป็นส่วนหลักๆได้แก่ แอปพลิเคชันทำแบบทดสอบ การประมวลผลภาพ การแยกภาพ การวินิจฉัย โดยมีขั้นตอนของตัวระบบดังนี้



รูปที่ 3.2 ภาพตัวอย่างการทำงานของ Blob coloring

1. ผู้ใช้จะต้องทำแบบทดสอบภายในแอปพลิเคชันโดยจะอยู่ในรูปแบบของเกมเขียน พยัญชนะ สระ และ สะกดคำ
2. หลังจากนั้นภาพแบบทดสอบที่ผู้ใช้ได้ทำจะถูกส่งเข้าไปภายในระบบ LDSpot เพื่อทำการปรับปรุงคุณภาพของรูปภาพได้แก่การปรับขนาดของภาพให้เหมาะสม การลดสัญญาณรบกวนในภาพ และการปรับสีให้อยู่ในรูปแบบของขาวดำ
3. เมื่อได้ภาพที่ผ่านการทำการปรับปรุงคุณภาพของภาพแล้ว ภาพจะถูกนำมาแบ่งเป็นช่องตามตัวอักษรโดยการสร้าง contour แล้วติดกรอบด้วย boundingbox ล้อมรอบแต่ละตัวอักษร หลังจากนั้นจึงตัดภาพตาม boundingbox ที่ได้สร้างไว้
4. นำภาพแต่ละตัวอักษรเข้าไปวินิจฉัย เพื่อนำผลลัพธ์จากโมเดลมาแสดงผลบนแอปพลิเคชัน

3.2.1 แอปพลิเคชันทำแบบทดสอบ (Application)

ในส่วนของแอปพลิเคชันทำแบบทดสอบนั้นเพื่อที่จะได้มาซึ่งภาพแบบทดสอบเราจึงออกแบบแอปพลิเคชันในรูปแบบของเกมให้ผู้ใช้ทำ ซึ่งในส่วนนี้ผู้ใช้จะต้องทำแบบทดสอบการเขียนพยัญชนะ สระ และสะกดคำ โดยจะมีกรอบขึ้นมาให้ผู้ใช้เขียนตามเสียงพูด

ตารางที่ 3.1 แสดงข้อมูลขาเข้าและขาออกของ แอปพลิเคชัน

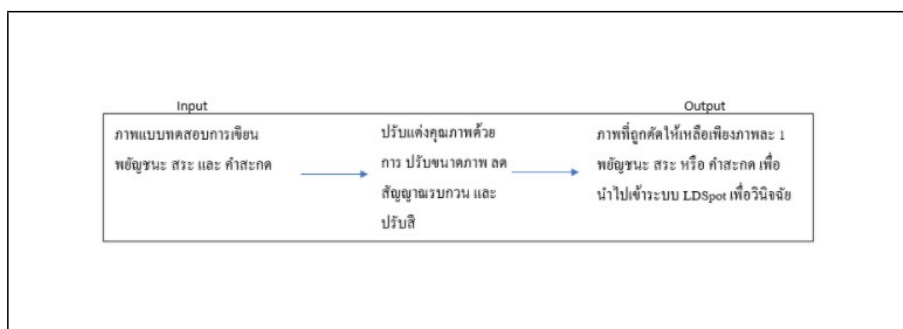
Input	ผู้ใช้ทำแบบทดสอบภายในแอปพลิเคชัน
Output	ภาพแบบทดสอบการเขียนพยัญชนะ สระ และ คำสะกด

3.2.2 การประมวลผลภาพ (Image processing)

ในส่วนนี้นั้นเราจะนำภาพแบบทดสอบที่ได้จากแอปพลิเคชันมาปรับปรุงคุณภาพของภาพเพื่อให้เหมาะสมแก่การนำไปวินิจฉัย โดยจะมีการปรับขนาดของภาพให้ตรงกับขนาดของภาพที่ระบบ LDSpot นั้นใช้ในการเรียนรู้ หลังจากนั้นจึงนำภาพไปทำการลดสัญญาณรบกวนด้วยวิธีการใช้ Gaussian blur และจึงปรับภาพให้อยู่ในสีขาวดำ เพื่อแยกตัวอักษรออกจากภาพพื้นหลัง

ตารางที่ 3.2 แสดงข้อมูลขาเข้าและขาออกของส่วนการประมวลผลภาพ

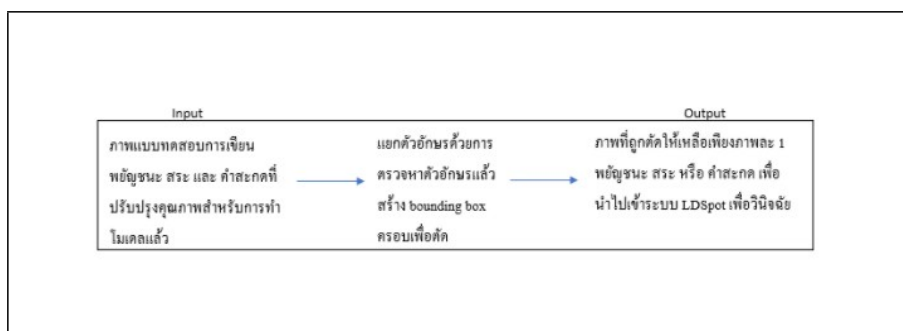
Input	ภาพแบบทดสอบการเขียนพยัญชนะ สระ และ คำสะกด
Output	ภาพแบบทดสอบการเขียนพยัญชนะ สระ และ คำสะกดที่ปรับปรุงคุณภาพสำหรับการทำโมเดลแล้ว



รูปที่ 3.3 แผนภาพแสดงข้อมูลนำเข้าและออกของการประมวลผลภาพ

3.2.3 การแยกภาพ (Image segmentation)

ในส่วนนี้เราจะทำการสร้าง contour ขึ้นมาจากภาพที่ได้ทำการปรับสีขาวดำแล้ว โดยเราจะนำ contour นั้นไปสร้าง bounding box เพื่อครอบแต่ละตัวอักษรให้แยกออกจากกัน เนื่องจากเราต้องการภาพตัวอักษรที่อยู่เดี่ยวๆ ไปใช้ในการวินิจฉัยโรคบกพร่องทางการเรียนรู้



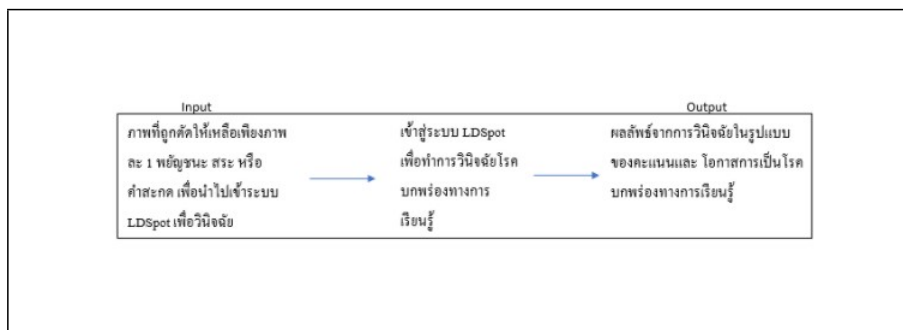
รูปที่ 3.4 แผนภาพแสดงข้อมูลนำเข้าและออกของการแยกภาพ

ตารางที่ 3.3 แสดงข้อมูลนำเข้าและขาออกของส่วนการแยกภาพ

Input	ภาพแบบทดสอบการเขียนพยัญชนะ สระ และ คำสะกดที่ปรับปรุงคุณภาพสำหรับการทำโมเดลแล้ว
Output	ภาพที่ถูกตัดให้เหลือเพียงภาพละ 1 พยัญชนะ สระ หรือ คำสะกด เพื่อนำไปเข้าระบบ LDSpot เพื่อวินิจฉัย

3.2.4 การวินิจฉัย (Learning disorder prediction)

เมื่อเราได้ภาพตัวอักษรเดี่ยวๆจากส่วนการแยกภาพแล้ว เราจะนำภาพตัวอักษรเดี่ยวๆนั้นไปโยนเข้าโมเดลที่เราได้ทำการสร้างไว้ เพื่อให้โมเดลวินิจฉัยโรคบกพร่องทางการเรียนรู้ แล้วนำผลลัพธ์ส่งกลับไปในฐานะข้อมูลเพื่อให้แอปพลิเคชันสามารถเรียกข้อมูลไปแสดงได้ต่อไป

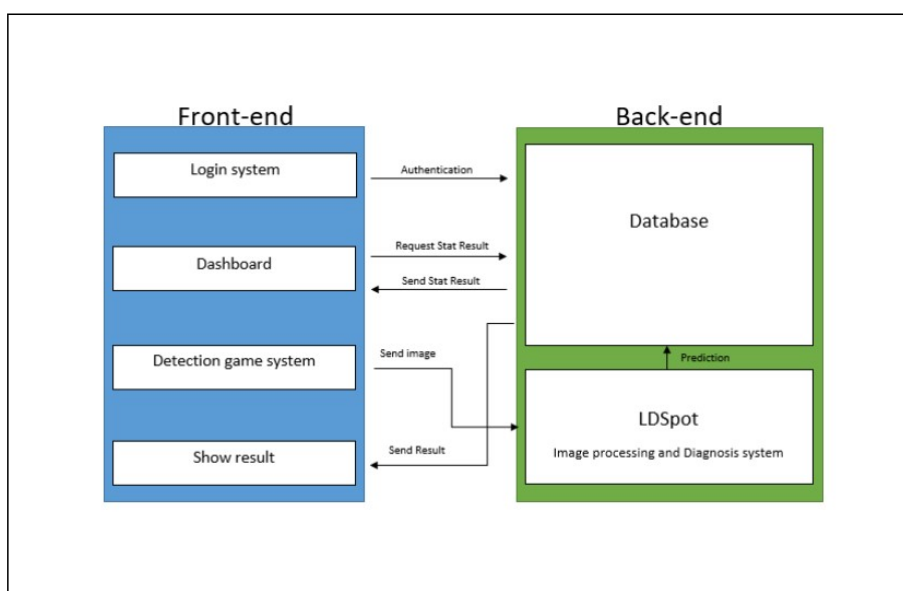


รูปที่ 3.5 แผนภาพแสดงข้อมูลขาเข้าและออกของการวินิจฉัย

ตารางที่ 3.4 แสดงข้อมูลขาเข้าและขาออกของส่วนการวินิจฉัย

Input	ภาพที่ถูกตัดให้เหลือเพียง ภาพละ 1 พยัญชนะ สระ หรือ คำสะกด เพื่อนำไปเข้าระบบ LDSpot เพื่อวินิจฉัย
Output	ผลลัพธ์จากการวินิจฉัยในรูปแบบของคะแนนและ โอกาสการเป็นโรคบกพร่องทางการเรียนรู้

3.3 Conceptual Design



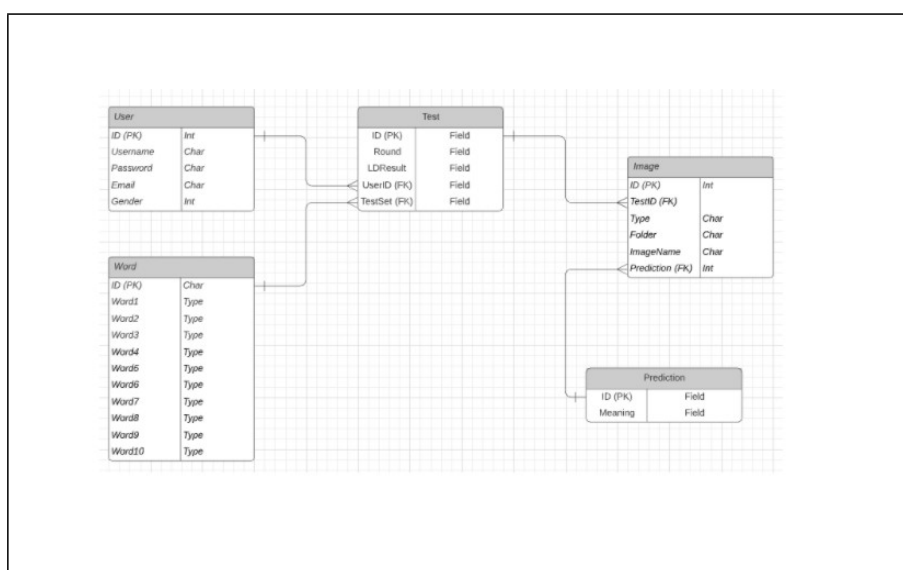
รูปที่ 3.6 แผนภาพแสดงข้อมูลขาเข้าและออกของการวินิจฉัย

การทำงานของตัวระบบ LDSpot นั้นจะอยู่สองส่วนด้วยกันได้แก่ front-end ที่ทำหน้าที่เป็นหน้าแอปพลิเคชันไว้สื่อสารกับผู้ใช้งาน และส่วนของ back-end ที่รับข้อมูลมาเพื่อประมวลผลแล้วหลังจากนั้นจึงนำข้อมูลไปเก็บใส่ฐานข้อมูลไว้ใช้งานต่อไป

- ในส่วนของ front-end นั้นจะประกอบไปด้วย แอปพลิเคชันในรูปแบบของเกม โดยที่ผู้ใช้งานจะสามารถเข้าสู่ระบบผ่านทางรหัสที่ได้ทำการสมัครสมาชิกไว้ โดยตัวรหัสจะถูกส่งไปเพื่อตรวจสอบความถูกต้องว่ารหัสนี้จริงในระบบหรือไม่กับฐานข้อมูลที่อยู่ภายในส่วนของ back-end หากตรวจสอบแล้วถูกต้องจึงจะสามารถเข้าสู่ระบบได้

- ผู้ใช้สามารถกดเข้ารับแบบทดสอบได้ โดยเมื่อเข้ารับแล้ว จะต้องทำตามขั้นตอนต่อไป ซึ่งผู้ใช้งานได้เขียนตัวอักษร สระ และคำสะกดตามเสียงไปเรื่อยๆ เมื่อเสร็จสิ้นแล้วภาพแบบทดสอบจะถูกส่งไปทางฝั่ง back-end ในส่วนของระบบ LDSpot เพื่อทำการวินิจฉัย หลังจากนั้นผลลัพธ์จะถูกส่งเก็บเข้าไปในฐานข้อมูล เพื่อให้ทางฝั่ง front-end สามารถดึงข้อมูลไปแสดงผลบนแอปพลิเคชันได้
- ผู้ใช้สามารถดูผลลัพธ์ย้อนหลังได้โดยกดดูผลลัพธ์ภายในแอปพลิเคชันหลังจากนั้น แอปพลิเคชันจะทำการติดต่อกับฐานข้อมูลเพื่อดึงผลลัพธ์ที่เคยได้มีการวินิจฉัยไว้ของผู้ใช้งานคนนั้นมาแสดงผล
- ผู้ใช้สามารถดูประวัติได้ โดยประวัตินั้นจะดึงข้อมูลสรุปจากฐานข้อมูลมาว่า มีผู้ใช้งานแอปพลิเคชันนี้แล้วกี่คน มีการทำนายว่าเป็นโรคบกพร่องทางการเรียนรู้กี่คน ไม่เป็นกี่คน เป็นต้น เพื่อใช้เป็นข้อมูลในเชิงสถิติต่อไป

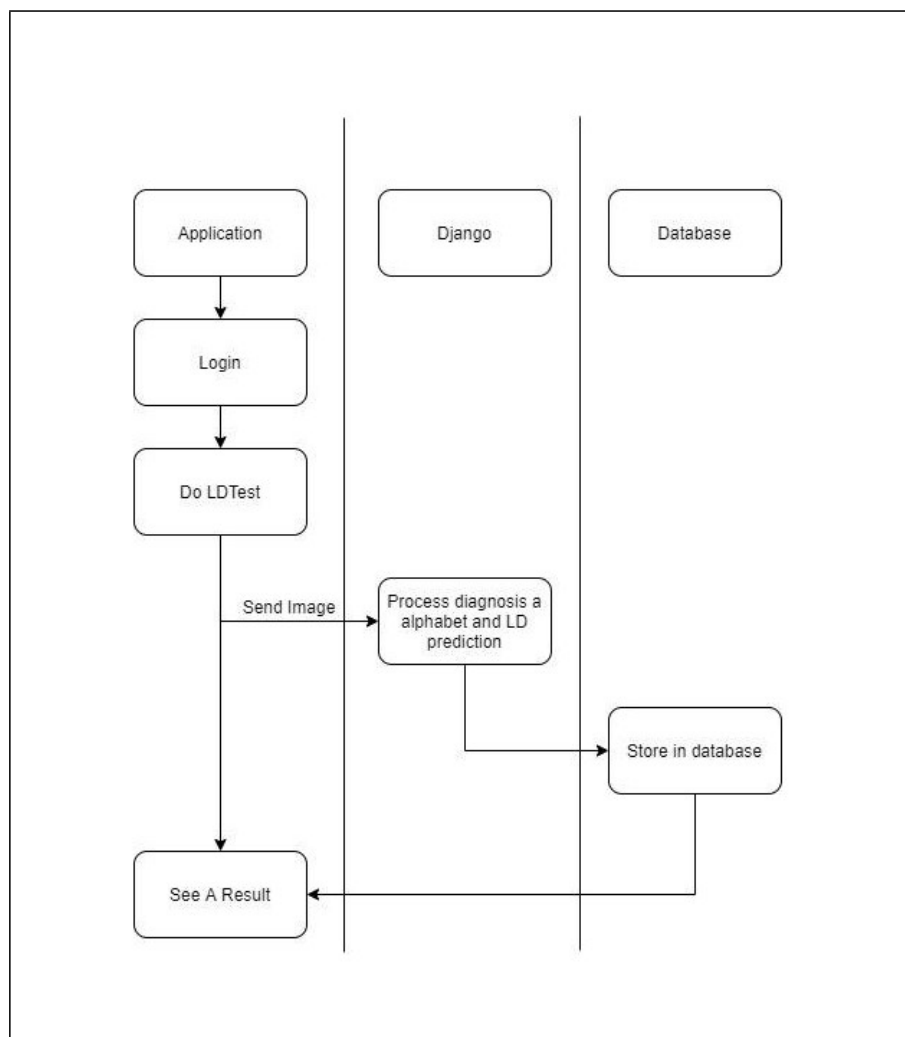
3.4 Database Design



รูปที่ 3.7 แผนภาพแสดงข้อมูลเข้าและออกของการวินิจฉัย

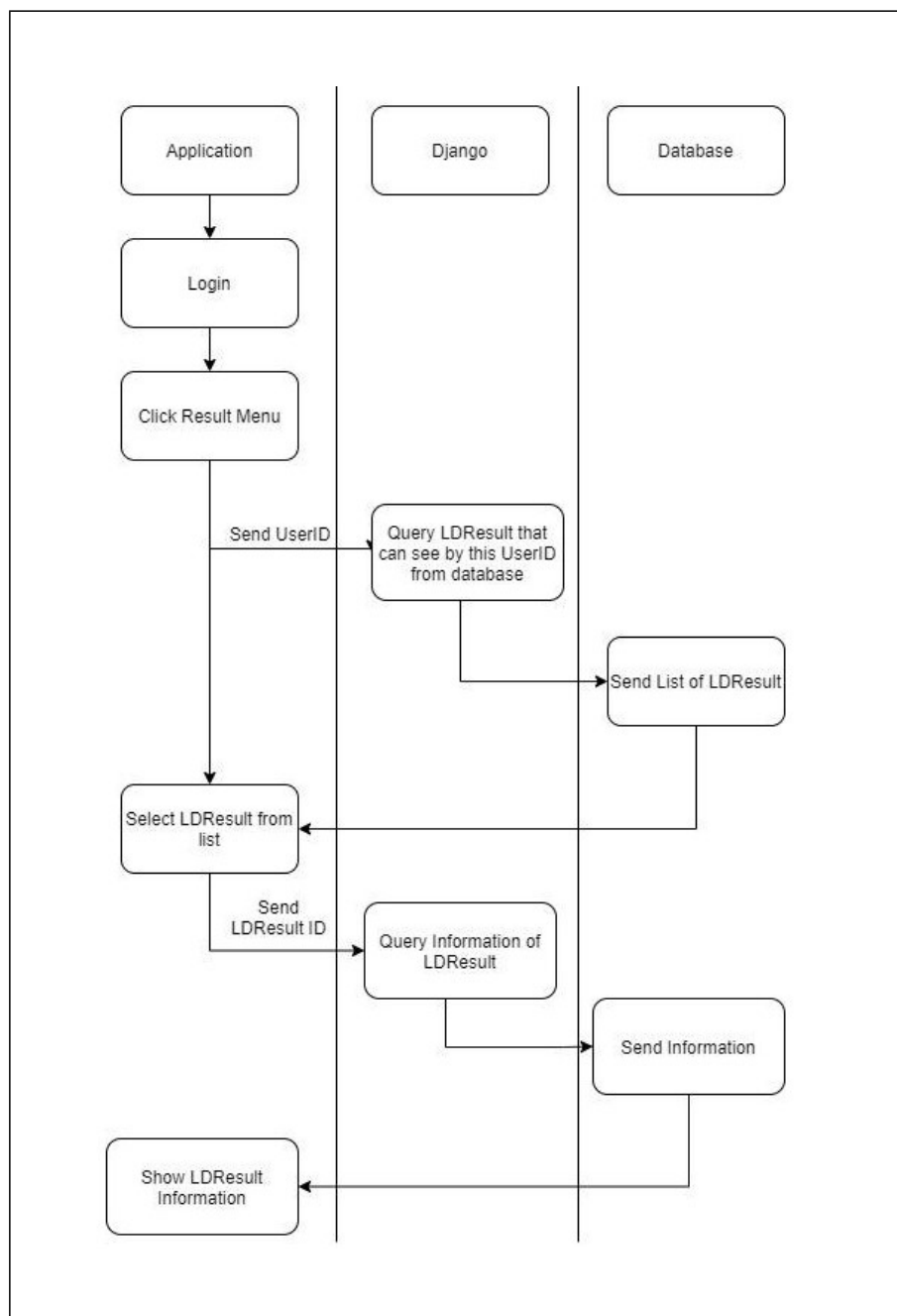
3.5 Activity Diagram Design

- ทำแบบทดสอบ



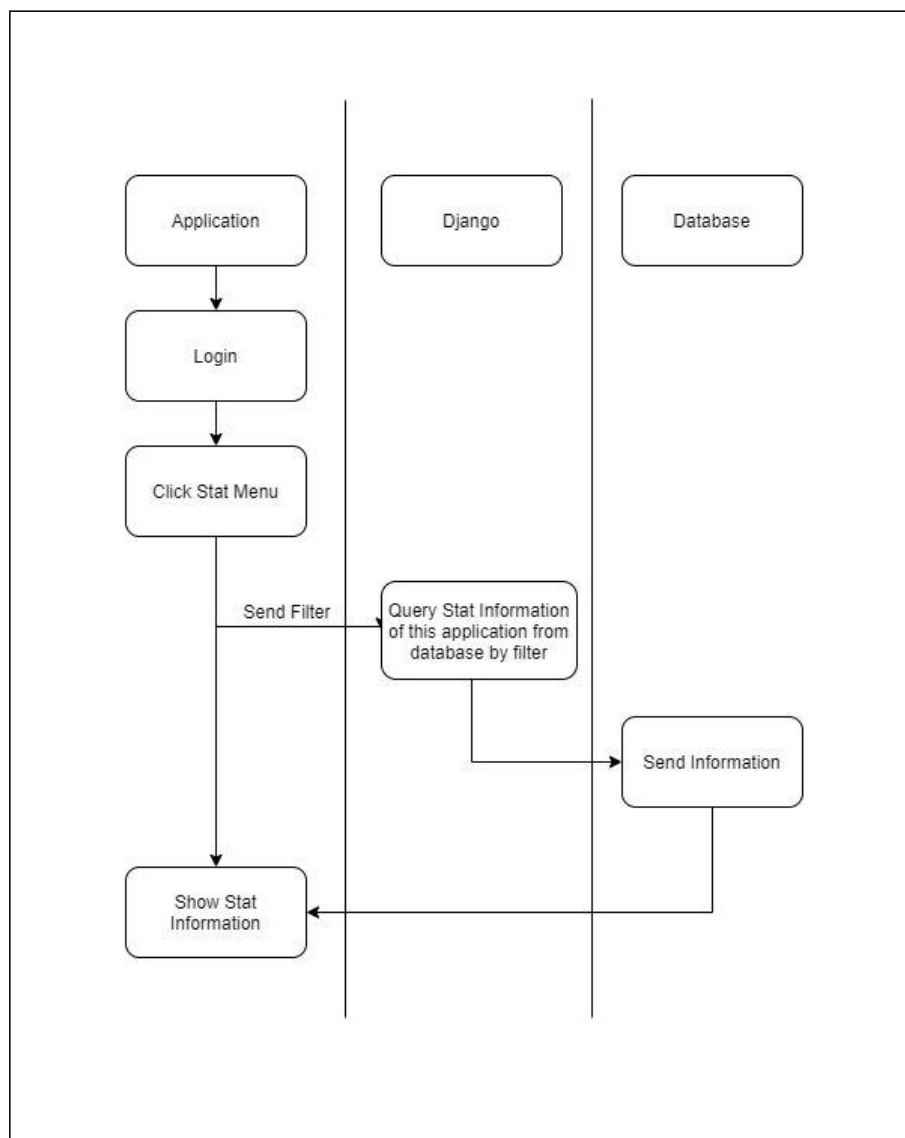
รูปที่ 3.8 แผนภาพแสดงข้อมูลเข้าและออกของการวินิจฉัย

- คู่มือลัพท์การทดสอบ



รูปที่ 3.9 แผนภาพแสดงข้อมูลเข้าและออกของการวินิจฉัย

- คู่มือการรวมของแอปพลิเคชัน



รูปที่ 3.10 แผนภาพแสดงข้อมูลเข้าและออกของการวินิจฉัย

- 3.6 สถาปัตยกรรมระบบ
- 3.7 Hardware Module 1
 - 3.7.1 Component 1
 - 3.7.2 Logical Circuit Diagram
- 3.8 Hardware Module 2
 - 3.8.1 Component 1
 - 3.8.2 Component 2
- 3.9 Path Finding Algorithm
- 3.10 Database Design
- 3.11 UML Design
- 3.12 GUI Design
- 3.13 การออกแบบการทดลอง
 - 3.13.1 ตัวชี้วัดและปัจจัยที่ศึกษา
 - 3.13.2 รูปแบบการเก็บข้อมูล

บทที่ 4 ผลการดำเนินงาน

You can title this chapter as **Preliminary Results** ผลการดำเนินงานเบื้องต้น or **Work Progress** ความก้าวหน้าโครงการ for the progress reports. Present implementation or experimental results here and discuss them. ใส่เฉพาะหัวข้อที่เกี่ยวข้องกับงานที่ทำ

- 4.1 ประสิทธิภาพการทำงานของระบบ
- 4.2 ความพึงพอใจการใช้งาน
- 4.3 การวิเคราะห์ข้อมูลและผลการทดลอง

บทที่ 5 บทสรุป

This chapter is optional for proposal and progress reports but is required for the final report.

5.1 สรุปผลโครงการ

สรุปว่าโครงการบรรลุตามวัตถุประสงค์ที่ตั้งไว้หรือไม่ อย่างไร

5.2 ปัญหาที่พบและการแก้ไข

State your problems and how you fixed them.

5.3 ข้อจำกัดและข้อเสนอแนะ

ข้อจำกัดของโครงการ What could be done in the future to make your projects better.

ภาคผนวก A

ชื่อภาคผนวกที่ 1

ใส่หัวข้อตามความเหมาะสม

This is where you put hardware circuit diagrams, detailed experimental data in tables or source codes, etc..

This appendix describes two static allocation methods for fGn (or fBm) traffic. Here, λ and C are respectively the traffic arrival rate and the service rate per dimensionless time step. Their unit are converted to a physical time unit by multiplying the step size Δ . For a fBm self-similar traffic source, Norros [?] provides its EB as

$$C = \lambda + (\kappa(H)\sqrt{-2\ln \epsilon})^{1/H} a^{1/(2H)} x^{-(1-H)/H} \lambda^{1/(2H)} \quad (\text{A.1})$$

where $\kappa(H) = H^H(1-H)^{(1-H)}$. Simplicity in the calculation is the attractive feature of (??).

The MVA technique developed in [?] so far provides the most accurate estimation of the loss probability compared to previous bandwidth allocation techniques according to simulation results. Consider a discrete-time queueing system with constant service rate C and input process λ_n with $\mathbb{E}\{\lambda_n\} = \lambda$ and $\text{Var}\{\lambda_n\} = \sigma^2$. Define $X_n \equiv \sum_{k=1}^n \lambda_k - Cn$. The loss probability due to the MVA approach is given by

$$\varepsilon \approx \alpha e^{-m_x/2} \quad (\text{A.2})$$

where

$$m_x = \min_{n \geq 0} \frac{((C - \lambda)n + B)^2}{\text{Var}\{X_n\}} = \frac{((C - \lambda)n^* + B)^2}{\text{Var}\{X_{n^*}\}} \quad (\text{A.3})$$

and

$$\alpha = \frac{1}{\lambda\sqrt{2\pi\sigma^2}} \exp\left(\frac{(C - \lambda)^2}{2\sigma^2}\right) \int_C^\infty (r - C) \exp\left(\frac{(r - \lambda)^2}{2\sigma^2}\right) dr \quad (\text{A.4})$$

For a given ε , we numerically solve for C that satisfies (??). Any search algorithm can be used to do the task. Here, the bisection method is used.

Next, we show how $\text{Var}\{X_n\}$ can be determined. Let $C_\lambda(l)$ be the autocovariance function of λ_n . The MVA technique basically approximates the input process λ_n with a Gaussian process, which allows $\text{Var}\{X_n\}$ to be represented by the autocovariance function. In particular, the variance of X_n can be expressed in terms of $C_\lambda(l)$ as

$$\text{Var}\{X_n\} = nC_\lambda(0) + 2 \sum_{l=1}^{n-1} (n-l)C_\lambda(l) \quad (\text{A.5})$$

Therefore, $C_\lambda(l)$ must be known in the MVA technique, either by assuming specific traffic models or by off-line analysis in case of traces. In most practical situations, $C_\lambda(l)$ will not be known in advance, and an on-line measurement algorithm developed in [?] is required to jointly determine both n^* and m_x . For fGn traffic, $\text{Var}\{X_n\}$ is equal to $\sigma^2 n^{2H}$, where $\sigma^2 = \text{Var}\{\lambda_n\}$, and we can find the n^* that minimizes (??) directly. Although λ can be easily measured, it is not the case for σ^2 and H . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.

ภาคผนวก B

ชื่อภาคผนวกที่ 2

ใส่หัวข้อตามความเหมาะสม

Next, we show how $\text{Var}\{X_n\}$ can be determined. Let $C_\lambda(l)$ be the autocovariance function of λ_n . The MVA technique basically approximates the input process λ_n with a Gaussian process, which allows $\text{Var}\{X_n\}$ to be represented by the autocovariance function. In particular, the variance of X_n can be expressed in terms of $C_\lambda(l)$ as

$$\text{Var}\{X_n\} = nC_\lambda(0) + 2 \sum_{l=1}^{n-1} (n-l)C_\lambda(l) \quad (\text{B.1})$$

Add more topic as you need

Therefore, $C_\lambda(l)$ must be known in the MVA technique, either by assuming specific traffic models or by off-line analysis in case of traces. In most practical situations, $C_\lambda(l)$ will not be known in advance, and an on-line measurement algorithm developed in [?] is required to jointly determine both n^* and m_x . For fGn traffic, $\text{Var}\{X_n\}$ is equal to $\sigma^2 n^{2H}$, where $\sigma^2 = \text{Var}\{\lambda_n\}$, and we can find the n^* that minimizes (??) directly. Although λ can be easily measured, it is not the case for σ^2 and H . Consequently, the MVA technique suffers from the need of prior knowledge traffic parameters.