การประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกในการฝังคำเพื่อสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรคจากบันทึกทางการแพทย์ของแผนกออร์โธปิดิกส์

นายธนากร รัตนจริยา

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2561

APPLYING DEEP LEARNING IN WORD EMBEDDING FOR MAKING A DIAGNOSIS PREDICTION MODEL FROM ORTHOPEDIC CLINICAL NOTE

Mr. Tanakorn Rattanajariya

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Computer Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2018

หัวข้อวิทยานิพนธ์	การประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกในการฝังคำเพื่อสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรคจากบันทึกทางการแพทย์ของแผนกออร์โธปิดิกส์
โดย	นายธนากร รัตนจริยา
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม	อาจารย์ นายแพทย์กฤษณ์ เจริญลาภ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ณัฐวุฒิ หนูไพโรจน์)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม
(อาจารย์ นายแพทย์กฤษณ์ เจริญลาภ)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.พีรพล เวทีกูล)

..... กรรมการภายนอกมหาวิทยาลัย
(นายแพทย์วิทวัส เจนบุญไทย)

ธนากร รัตนจริยา : การประยุกต์ใช้เทคนิคการเรียนรู้เชิงลึกในการฝังคำเพื่อสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรคจากบันทึกทางการแพทย์ของแผนกออร์โธปิดิกส์. (APPLYING DEEP LEARNING IN WORD EMBEDDING FOR MAKING A DIAGNOSIS PREDICTION MODEL FROM ORTHOPEDIC CLINICAL NOTE) อ.ที่ปรึกษาหลัก : ผศ. ดร.เกริก ภิรมย์โสภา, อ.ที่ปรึกษาร่วม : อ. นพ.กฤษณ์ เจริญลาภ

งานวิจัยนี้จะนำเสนอวิธีการฝังคำในการเรียนรู้เชิงลึกเพื่อสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรค ซึ่งปัจจัยสำคัญที่ทำให้ผลการวินิจฉัยโรคของแพทย์ผิดพลาดคือประสบการณ์ของแพทย์ที่ไม่เพียงพอ โดยการวินิจฉัยโรคที่ผิดพลาดนั้น นอกจากจะนำไปสู่การรักษาที่ผิดพลาดแล้ว ยังทำให้ผู้ป่วยเสียทั้งเงินและเวลา ดังนั้นเพื่อแก้ไขปัญหาการวินิจฉัยที่ผิดพลาด งานวิจัยนี้จึงนำเสนอวิธีการประยุกต์ใช้การเรียนรู้เชิงลึกกับการฝังคำ เพื่อทำนายผลการวินิจฉัยโรคจากระบบเวชระเบียน โดยจะสร้างแบบจำลองจากการใช้ข้อมูลในบันทึกของแพทย์ ซึ่งข้อมูลจะถูกนำไปวิเคราะห์ผ่านแบบจำลอง เพื่อทำนายผลการวินิจฉัยโรคที่มีความน่าจะเป็นออกมา เรียงตามลำดับความเชื่อมั่น และสุดท้ายจะใช้อัตราผลบวกจริง อัตราผลบวกเท็จ และค่าความแม่นยำมาเป็นตัววัดประสิทธิภาพของแบบจำลองที่ได้ ซึ่งพบว่าค่าความแม่นยำของแบบจำลองในงานวิจัยนี้มีค่าเท่ากับ 99.95% และอัตราผลบวกจริงมีค่าเท่ากับ 86.64% ด้วยการทำนายผลลัพธ์อันดับแรกเพียงอันดับเดียว

สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....

ปีการศึกษา 2561 ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

ลายมือชื่อ อ.ที่ปรึกษาร่วม .....

6070451521 : MAJOR COMPUTER ENGINEERING

KEYWORD: Diagnosis, Machine Learning, Deep Learning, Medical Record, Text Classification, Word Embedding

Tanakorn Rattanajariya : APPLYING DEEP LEARNING IN WORD EMBEDDING FOR MAKING A DIAGNOSIS PREDICTION MODEL FROM ORTHOPEDIC CLINICAL NOTE . Advisor: Asst. Prof. KRERK PIROMSOPA, Ph.D. Co-advisor: CHRIS CHAROENLAP, M.D.

We propose deep learning in word embedding for making a diagnostic prediction model. One factor that causes uncertainties in diagnostic is the inexperience of physicians. The diagnosis errors lead to incorrect and delay in treatment, waste of time and money. To solve the problem, a differential diagnosis is one critical tool. It is powerful and does not introduce additional work to physician. Our method uses a deep learning tool together with word embedding technique to make a differential diagnosis from existing diagnosis texts in medical system. The model will take the clinical notes from a physician. The note is then used to analyze the possibilities of diseases. The output is sorted by model confidence. In order to validate the model, we use True Positive Rate (Recall), False Positive Rate (Precision) and accuracy to compare to other works. Our model achieves a new record of accuracy at 99.95% The highest recall rate is at 86.64% in top first prediction.

Field of Study: Computer Engineering Student's Signature .....

Academic Year: 2018 Advisor's Signature .....

Co-advisor's Signature .....

กิตติกรรมประกาศ

ขอกราบขอบพระคุณ ผศ. ดร. เกริก ภิรมย์โสภา อาจารย์ที่ปรึกษาวิทยานิพนธ์ และ อ. นพ. กฤษณ์ เจริญลาภ อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม เป็นอย่างยิ่งที่ได้เสียสละเวลาในการให้คำปรึกษา คำแนะนำ และแนวทางในการดำเนินงาน ทั้งยังให้ความช่วยเหลืออย่างเต็มกำลังเมื่อเกิดปัญหาในการดำเนินงาน ทำให้การจัดทาวิทยานิพนธ์เสร็จสมบูรณ์ด้วยดี

ขอกราบขอบพระคุณ ผศ. ดร. ณัฐวุฒิ หนูไพโรจน์ ผศ. ดร. พีรพล เวทีกุล และ นพ. วิทวัส เจนบุญไทย คณะกรรมการสอบวิทยานิพนธ์เป็นอย่างยิ่ง ที่ได้กรุณาให้คำแนะนำสำหรับนำไปปรับปรุงแนวทางในการดำเนินงาน เพื่อให้งานเป็นไปอย่างราบรื่น

ขอขอบพระคุณ คณาจารย์ทุกท่านในภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย ที่ให้ความรู้ตลอดหลักสูตร

สุดท้ายนี้ขอขอบคุณ โรงพยาบาลจุฬาลงกรณ์ ที่ให้ความอนุเคราะห์ข้อมูลคำวินิจฉัย เพื่อนำมาใช้ใน วิทยานิพนธ์นี้

ธนากร รัตนจริยา

สารบัญ

หน้า

สารบัญรูป

หน้า

สารบัญตาราง

หน้า

สารบัญกราฟ

หน้า

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ปัจจุบันผลการวินิจฉัยโรคของแพทย์จะถูกเก็บบันทึกในรูปแบบของบัญชีจำแนกโรคระหว่างประเทศ (International Classification of Diseases and Related Health Problems: ICD) ซึ่งบ่งบอกถึงข้อมูลอาการ การบาดเจ็บภายนอก และความผิดปกติที่เกิดขึ้นกับผู้ป่วย ในรูปแบบของรหัสโรคที่ประกอบด้วยตัวอักษรภาษาอังกฤษและตัวเลขอารบิก เพื่อใช้เป็นรหัสโรคสากลในการสื่อสารกันระหว่างโรงพยาบาลและสถาบันทางการแพทย์ โดยบัญชีจำแนกโรคระหว่างประเทศที่ใช้กันอยู่ในปัจจุบัน คือฉบับปรับปรุงแก้ไขครั้งที่ 10 หรือไอซีดีเทน (ICD-10)

ในการวินิจฉัยโรค แพทย์จำเป็นต้องทราบประวัติผู้ป่วย และข้อมูลอาการหรือความผิดปกติที่เกิดขึ้นกับผู้ป่วย เพื่อนำข้อมูลเหล่านี้มาใช้ในการวินิจฉัยหาโรคที่ผู้ป่วยมีความน่าจะเป็น และทำการรักษาผู้ป่วยต่อไป แต่ในปัจจุบันเนื่องจากความรู้และประสบการณ์ของแพทย์ที่มีค่อนข้างจำกัด ทำให้ในบางครั้งการวินิจฉัยโรคของแพทย์อาจเกิดข้อผิดพลาดและไม่แม่นยำเสมอไป รวมถึงผู้ป่วยที่เข้ารับการรักษาอาจป่วยเป็นโรคที่แพทย์ไม่เคยพบ หรือแพทย์อาจลืมนึกถึงโรคบางโรคไป ทำให้แพทย์วินิจฉัยโรคผิด ดังนั้นเพื่อเป็นการช่วยให้แพทย์นึกถึงโรคที่มีความน่าจะเป็น และช่วยให้แพทย์สามารถวินิจฉัยโรคได้ง่ายขึ้นและมีประสิทธิภาพมากขึ้น งานวิจัยนี้จึงนำเสนอวิธีการสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรคโดยใช้เทคนิคการเรียนรู้เชิงลึก

การสร้างแบบจำลองในงานวิจัยนี้ จะประกอบด้วย 2 ขั้นตอนหลัก ๆ ขั้นตอนแรกคือการประมวลผลข้อมูลก่อน เพื่อกำจัดข้อมูลที่ไม่เกี่ยวข้องออก และแปลงข้อมูลให้อยู่ในรูปแบบที่เหมาะสมสำหรับนำไปใช้ในการสร้างแบบจำลอง ขั้นตอนต่อมาคือการนำข้อมูลที่ได้จากขั้นตอนแรกมาใช้ในการสร้างแบบจำลองด้วยเทคนิคการเรียนรู้เชิงลึก ซึ่งผลลัพธ์ที่ได้จะเป็นแบบจำลองเชิงทำนาย เนื่องจากชุดข้อมูลที่นำมาใช้ในการสร้างแบบจำลองเป็นชุดข้อมูลที่ทราบฉลากประเภทล่วงหน้า ซึ่งค่าฉลากประเภทที่ทราบล่วงหน้าในงานวิจัยนี้ คือรหัสไอซีดีเทนหรือผลการวินิจฉัยโรคของข้อมูลอาการในบันทึกทางการแพทย์ของผู้ป่วยแต่ละราย

ข้อมูลที่นำมาใช้ในงานวิจัยนี้จะเป็นข้อมูลในส่วนของบันทึกทางการแพทย์ของผู้ป่วยในโรงพยาบาลจุฬาลงกรณ์ โดยจะแบ่งชุดข้อมูลออกเป็น 2 ส่วน ส่วนแรกคือชุดข้อมูลสำหรับนำมาใช้เป็นชุดข้อมูลสอนเพื่อสร้างแบบจำลองเชิงทำนาย และอีกส่วนหนึ่งคือชุดข้อมูลสำหรับนำมาใช้เป็นชุดข้อมูลทดสอบเพื่อทดสอบความถูกต้องในการทำนายผลลัพธ์ของแบบจำลองเชิงทำนาย โดยใช้วิธีการแบ่งชุดข้อมูลออกเป็น 10 ส่วน (10-fold Cross Validation) ในการสร้างและทดสอบแบบจำลอง

สุดท้ายในส่วนของการวัดความแม่นยำในการทำนายของแบบจำลองจะพิจารณาจากรายชื่อโรคที่อยู่ในผลลัพธ์ที่แบบจำลองแสดงกับผลการวินิจฉัยโรคหรือรหัสไอซีดีเทนที่อยู่ในบันทึกทางการแพทย์แต่ละอัน โดยการวัดประสิทธิภาพของแบบจำลองจะคำนวณจากค่าความเที่ยง (Precision) ค่าการระลึกได้ (Recall) และค่าความแม่น (Accuracy)

1.2 วัตถุประสงค์ของงานวิจัย

1. เพื่อวิเคราะห์หาความสัมพันธ์ระหว่างผลการวินิจฉัยโรคกับข้อมูลอาการของโรค
2. เพื่อสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรค
3. เพื่อศึกษาการสร้างแบบจำลองเชิงทำนายด้วยเทคนิคการเรียนรู้เชิงลึก
4. เพื่อช่วยให้การวินิจฉัยโรคของแพทย์มีความถูกต้องแม่นยำมากขึ้น

1.3 ขอบเขตงานวิจัย

1. ข้อมูลที่นำมาใช้ในการสร้างและทดสอบแบบจำลองจะใช้เฉพาะข้อมูลที่เป็นภาษาอังกฤษเท่านั้น
2. งานวิจัยนี้จะใช้เทคนิคการเรียนรู้เชิงลึกในการสร้างแบบจำลองท่านั้น
3. ข้อมูลที่นำมาใช้ในการสร้างแบบจำลองเป็นข้อมูลผู้ป่วยที่มาจากโรงพยาบาลจุฬาลงกรณ์เท่านั้น
4. ข้อมูลที่นำมาใช้เป็นชุดข้อมูลทดสอบสำหรับวัดประสิทธิภาพการทำงานของแบบจำลอง จะใช้ข้อมูลผู้ป่วยจากโรงพยาบาลจุฬาลงกรณ์

1.4 ขั้นตอนและวิธีการดำเนินงานวิจัย

1. ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง
2. ศึกษาเครื่องมือที่นำมาใช้ในการสร้างแบบจำลอง
3. ออกแบบกระบวนการสร้างแบบจำลอง
4. เก็บข้อมูลที่จะนำมาใช้สร้างแบบจำลอง
5. สร้างแบบจำลองเชิงทำนาย
6. ประเมินผลการใช้งานแบบจำลอง และทดสอบความถูกต้องในการทำนายผลการวินิจฉัยโรค
7. วิเคราะห์ผลการประเมิน และปรับปรุงแบบจำลอง
8. สรุปผลและจัดทำเล่มวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถใช้แบบจำลองเพื่อทำนายผลการวินิจฉัยโรคจากข้อมูลอาการ
2. เรียนรู้การใช้เทคนิคการเรียนรู้เชิงลึกในการสร้างแบบจำลองเชิงทำนาย
3. ช่วยให้แพทย์สามารถตัดสินใจวินิจฉัยโรคผู้ป่วยได้ง่ายขึ้น
4. ช่วยเพิ่มประสิทธิภาพในการวินิจฉัยโรคของแพทย์ให้มีความถูกต้องแม่นยำมากขึ้น
5. สามารถนำเทคนิคที่ใช้ในงานวิจัยนี้ไปประยุกต์ใช้กับงานด้านอื่น ๆ ได้

1.6 โครงสร้างของเนื้อหาวิทยานิพนธ์

วิทยานิพนธ์เล่มนี้ประกอบด้วยโครงสร้างเนื้อหาทั้งหมด 5 บท โดยมีรายละเอียดดังต่อไปนี้

บทที่ 1 กล่าวถึงที่มาและความสำคัญของปัญหา วัตถุประสงค์ ขอบเขตของงาน ขั้นตอนและวิธีการดำเนินงาน ประโยชน์ที่คาดว่าจะได้รับ

บทที่ 2 กล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้อง

บทที่ 3 กล่าวถึงแนวคิดและวิธีการดำเนินงาน

บทที่ 4 กล่าวถึงการทดสอบและประเมินผล

บทที่ 5 กล่าวถึงบทสรุป

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

งานวิจัยนี้จะสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรค เพื่อช่วยให้แพทย์สามารถวินิจฉัยโรคผู้ป่วยได้อย่างมีประสิทธิภาพมากขึ้น ซึ่งขั้นตอนในการสร้างแบบจำลองจะมีการนำเอาทฤษฎีต่าง ๆ มาประยุกต์ใช้ ดังนี้

2.1.1 รหัสไอซีดีเทน

ICD ย่อมาจาก International Classification of Diseases and Related Health Problems เป็นบัญชีจำแนกโรคระหว่างประเทศที่จัดทำขึ้นโดยองค์การอนามัยโลก (WHO) เริ่มใช้ตั้งแต่ปี ค.ศ. 1893 โดยมีวัตถุประสงค์เพื่อใช้ในการจัดหมวดหมู่ของโรคและปัญหาสุขภาพต่าง ๆ ที่พบในมนุษย์ และใช้เป็นระบบรหัสโรคและรหัสปัญหาสุขภาพ ซึ่งมักจะถูกนำมาใช้ประโยชน์ในด้านระบาดวิทยา เวชสถิติ ระบบเวชสารสนเทศ การวางแผนยุทธศาสตร์ การวางแผนสุขภาพและการเบิกจ่ายค่ารักษาพยาบาล จากการบันทึกและรวบรวมข้อมูลทางสถิติ

ICD-10 ย่อมาจาก 10th Revision of ICD เป็นบัญชีจำแนกโรคระหว่างประเทศฉบับแก้ไขครั้งที่ 10 ซึ่งเป็นฉบับปรับปรุงครั้งล่าสุด ถูกปรับปรุงเมื่อปี ค.ศ. 2010 เป็นระบบที่มีองค์ประกอบสำคัญ 2 ส่วน คือ ระบบการจัดหมวดหมู่ของโรคและปัญหาสุขภาพต่าง ๆ ที่พบในมนุษย์ และระบบรหัสโรคและรหัสปัญหาสุขภาพ [1]

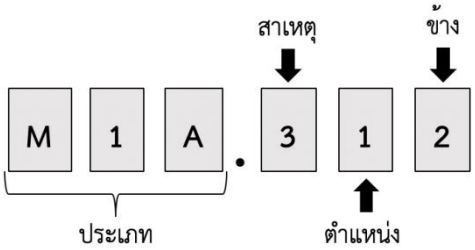
ผู้ป่วยที่เข้ารับการรักษาจากโรงพยาบาลจะได้รหัสโรค 1 รหัสต่อการเข้ารักษา 1 ครั้ง โดยปกติแพทย์ที่ทำการดูแลรักษาผู้ป่วยจะเป็นผู้สรุปว่าผู้ป่วยเป็นโรคอะไร หรือมีภาวะความผิดปกติเป็นอย่างไร แม้บางครั้งอาจมีความก้ำกึ่งซึ่งยากต่อการตัดสินใจ แต่แพทย์ก็ต้องสรุปให้ได้ เพื่อจะได้ทำการรักษาผู้ป่วยต่อไป

การให้รหัสโรคสำหรับผู้ป่วยแต่ละราย ผู้ให้รหัสจะต้องทราบว่าผู้ป่วยมีโรคทั้งหมดกี่โรค เป็นโรคอะไรบ้าง โดยทั่วไปการให้รหัสไอซีดีเทนนั้น จะให้รหัสต่อโรคที่ผู้ป่วยเป็น 1 โรค ผู้ให้รหัสจึงต้องการชื่อโรคที่ชัดเจนและมีคุณลักษณะที่ดี โดยคุณลักษณะของชื่อโรคที่ดีนั้นต้องประกอบด้วยองค์ประกอบ 3 อย่าง คือ

1. มีคำที่บอกว่าผู้ป่วยเป็นโรคอะไร
2. มีคำที่บอกว่าผู้ป่วยเป็นโรคที่ตำแหน่งใดหรือที่ระบบไหนของร่างกาย

3. มีคำที่บอกว่าผู้ป่วยเป็นโรคชนิดใด

ลักษณะของรหัสไอซีดีเทนเป็นรหัสที่ประกอบด้วยตัวเลขและตัวอักษร (Alphanumeric code) โดยรหัสแต่ละตัวจะขึ้นต้นด้วยอักษรภาษาอังกฤษ A-Z แล้วตามด้วยตัวเลขอารบิก 0-9 อีกประมาณ 2-4 ตัว จึงเป็นรหัสที่มีความยาว 3-5 อักขระ ตัวอย่างดังรูปที่ 1 สามหลักแรกจะสื่อถึงประเภทของโรค หลักแรกหลังจุดสื่อถึงสาเหตุของโรค หลักถัดมาสื่อถึงตำแหน่งของโรค และหลักสุดท้ายสื่อถึงข้าง เช่น ข้างซ้าย ข้างขวา เป็นต้น



รูปที่ 1 รูปแสดงส่วนประกอบของรหัสไอซีดีเทน

ไอซีดีเทนเป็นรายการรหัสโรคพร้อมคำอธิบาย ที่ได้มีการแบ่งเนื้อหาออกเป็นบทต่าง ๆ รวมทั้งสิ้น 21 บท [2] ดังนี้

กลุ่มที่ 1 โรคติดเชื้อ	ใช้รหัส A00-B99
กลุ่มที่ 2 เนื้องอกและมะเร็ง	ใช้รหัส C00-D49
กลุ่มที่ 3 โรคเลือด	ใช้รหัส D50-D89
กลุ่มที่ 4 โรคต่อมไร้ท่อ	ใช้รหัส E00-E89
กลุ่มที่ 5 โรคจิต โรคประสาท พฤติกรรม	ใช้รหัส F01-F99
กลุ่มที่ 6 โรคสมองและระบบประสาท	ใช้รหัส G00-G99
กลุ่มที่ 7 โรคตา	ใช้รหัส H00-H59
กลุ่มที่ 8 โรคหู	ใช้รหัส H60-H95
กลุ่มที่ 9 โรคหัวใจและหลอดเลือด	ใช้รหัส I00-I99
กลุ่มที่ 10 โรคปอดและระบบหายใจ	ใช้รหัส J00-J99
กลุ่มที่ 11 โรคระบบย่อยอาหาร	ใช้รหัส K00-K95
กลุ่มที่ 12 โรคผิวหนัง	ใช้รหัส L00-L99
กลุ่มที่ 13 โรคกล้ามเนื้อและกระดูก	ใช้รหัส M00-M99
กลุ่มที่ 14 โรคไตและระบบทางเดินปัสสาวะ	ใช้รหัส N00-N99
กลุ่มที่ 15 ตั้งครรภ์ การคลอด	ใช้รหัส O00-O9A
กลุ่มที่ 16 โรคของทารกแรกเกิด	ใช้รหัส P00-P96

กลุ่มที่ 17 พิการแต่กำเนิด ใช้รหัส Q00-Q99
กลุ่มที่ 18 อาการและอาการแสดงผิดปกติ ใช้รหัส R00-R99
กลุ่มที่ 19 การบาดเจ็บและการได้รับพิษ ใช้รหัส S00-T88
กลุ่มที่ 20 สาเหตุภายนอกของการบาดเจ็บ ใช้รหัส V01-Y98
กลุ่มที่ 21 การให้บริการสุขภาพ ใช้รหัส Z00-Z99

การให้รหัส จะเริ่มดำเนินการให้รหัสโรคได้ต้องมีข้อมูลทั้งหมดของผู้ป่วยจากใบสรุปการรักษาของแพทย์ซึ่งใช้เป็นหลักฐานในการให้รหัส การให้รหัสจะเริ่มจากการตรวจสอบโรคที่ปรากฏในใบสรุปการรักษาให้สอดคล้องกับข้อมูลในเวชระเบียน จากนั้นเปลี่ยนคำย่อทุกคำให้เป็นคำเต็ม และเลือกคำหลักของโรคทั้งหมด เพื่อใช้คำหลักในการเปิดหารหัสไอซีดีเทนจากดรรชนี และกำหนดรหัสโรคหลัก โรคร่วม โรคแทรก และโรคอื่น ๆ

การเรียงลำดับรหัสโรค ในการลงรหัสโรคของผู้ป่วยที่เข้ามารับการรักษาในโรงพยาบาล เมื่อค้นหาและใส่รหัสโรคทั้งหมดที่เกิดขึ้นกับผู้ป่วยแล้ว ผู้ให้รหัสยังมีหน้าที่เรียงลำดับรหัสโรคให้ตรงตามประเภทของรหัสโรค โดยในการให้รหัสโรคผู้ให้รหัสต้องเรียงลำดับรหัสโรค โดยเริ่มจาก รหัสโรคหลัก รหัสการวินิจฉัยร่วม รหัสโรคแทรกซ้อน รหัสการวินิจฉัยอื่น และรหัสสาเหตุของการบาดเจ็บ

การนำรหัสไอซีดีเทนไปใช้ประโยชน์ สามารถนำไปใช้ในการวินิจฉัยสาเหตุการตาย เพื่อประเมินสาเหตุการตายที่พบบ่อย ใช้ในการวินิจฉัยโรคในฐานข้อมูลของผู้ป่วย เพื่อประเมินสถานการณ์โรคที่พบบ่อย ประเมินผลลัพธ์ของผู้ป่วยเฉพาะโรค ประเมินความต่อเนื่องของบริการเฉพาะโรค ประเมินการเข้าถึงบริการของผู้ป่วยเฉพาะโรค และประเมินประสิทธิภาพของบริการเฉพาะโรค

2.1.2 การทำเหมืองข้อมูล

การทำเหมืองข้อมูล (Data mining) เป็นศาสตร์ที่จะนำไปสู่การค้นพบความรู้ในฐานข้อมูลขนาดใหญ่ (Knowledge discovery in large database) หมายถึงกระบวนการที่กระทำกับข้อมูลจำนวนมากเพื่อค้นหาแพท เทิร์น (patterns) และความสัมพันธ์ (associations) ที่ซ่อนอยู่ในชุดข้อมูลนั้น กระบวนการดังกล่าวมีความเป็นอัตโนมัติไม่สามารถประมวลผลได้ด้วยมือ ต้องใช้คอมพิวเตอร์เข้ามาช่วย เนื่องจากข้อมูลมีปริมาณมาก ผลลัพธ์จากการทำเหมืองข้อมูล คือความรู้ ซึ่งเป็นแพทเทิร์นและความสัมพันธ์ที่ซ่อนอยู่ในชุดข้อมูลหนึ่ง ๆ โดยแพทเทิร์นนั้นจะสะท้อนถึงเหตุการณ์หรือสิ่งที่เกิดขึ้นซ้ำแล้วซ้ำอีก (repeat) จนสามารถทำนายได้ (predictable) ตัวอย่างเช่น คนที่เป็นโรคชนิดหนึ่ง มักจะมียีนส์ลักษณะรูปแบบนี้ ซึ่งความรู้ดังกล่าวสามารถนำมาใช้ในการวินิจฉัยโรคทางการแพทย์ได้

ดังนั้นแพทเทิร์นหรือความสัมพันธ์ของลักษณะต่าง ๆ ที่พบในข้อมูลดิบนับเป็นความรู้ที่ช่วยให้มนุษย์นำไปใช้ประโยชน์ในด้านต่าง ๆ ได้ เช่น ในเชิงธุรกิจ การวินิจฉัยหรือรักษาโรคทางการแพทย์ การกีฬา เป็นต้น

การนำเสนอผลลัพธ์ความรู้จากการทำเหมืองข้อมูลประกอบด้วยรูปแบบต่าง ๆ ได้แก่ กฎความสัมพันธ์ (Association rule) กฎการจำแนกประเภท (Classification rule) ต้นไม้ตัดสินใจ เป็นต้น ซึ่งแบบจำลองผลลัพธ์จากการทำเหมืองข้อมูล สามารถแบ่งออกเป็น 2 ประเภท [3] ได้แก่

1. **แบบจำลองเชิงทำนาย (Predictive / Supervised modeling)**

   เป็นผลลัพธ์ที่สร้างจากการอนุมาน (inference) ชุดข้อมูลปัจจุบัน เพื่อใช้ในการทำนายประเภทตัวอย่างในอนาคต แบบจำลองในการทำนายเป็นผลลัพธ์จากการทำเหมืองจำแนกประเภทข้อมูลออกเป็นกลุ่มที่ทราบล่วงหน้าตามค่าคุณลักษณะที่เรียกว่า ฉลากประเภท (class label) ซึ่งถ้าค่าคุณลักษณะของฉลากมีค่าไม่ต่อเนื่องจะเรียกกระบวนการที่ใช้แยกแยะว่า การจำแนกประเภท (Classification) ถ้าค่าคุณลักษณะของฉลากมีค่าต่อเนื่อง จะเรียกกระบวนการที่ใช้แยกแยะว่า การถดถอย (Regression)

2. **แบบจำลองเชิงพรรณนา (Descriptive / Unsupervised modeling)**

   เป็นการหาความสัมพันธ์ต่าง ๆ หรือหาการจัดกลุ่มข้อมูล (Clustering) ซึ่งไม่ได้มีจุดมุ่งหมายเพื่อการทำนาย แต่เพื่อทำให้เข้าใจสาเหตุหรือปัจจัยของปัญหาหรือสิ่งที่สนใจได้ดีขึ้น เช่น เห็นการกระจายของกลุ่มข้อมูล หรือเห็นความสัมพันธ์ของข้อมูลในฐานข้อมูล

2.1.3 ตัวจำแนกประเภท (Classifier)

ผลลัพธ์ที่ได้จากแบบจำลองจะแตกต่างกันไปตามชนิดของตัวจำแนกประเภทที่นำมาใช้ในการทำเหมืองข้อมูล ซึ่งในปัจจุบันตัวจำแนกประเภทที่นิยมนำมาใช้ในการทำเหมืองข้อมูล ได้แก่

1. **ต้นไม้ตัดสินใจ (Decision Tree)**

   แบบจำลองจะเรียนรู้โดยการจำแนกประเภทข้อมูลออกเป็นกลุ่ม ๆ จากการวิเคราะห์คุณลักษณะของข้อมูล ซึ่งจะทำให้แบบจำลองทราบว่าคุณลักษณะใดของข้อมูลมีความสำคัญมากและคุณลักษณะใดมีความสำคัญน้อยต่อการจำแนกประเภทข้อมูล

2. การเรียนรู้เบย์ (Naïve Bayes)

แบบจำลองจะเรียนรู้โดยอาศัยทฤษฎีความน่าจะเป็นที่ข้อมูลน่าจะถูกจำแนกอยู่ในประเภทใด จากการวิเคราะห์ความน่าจะเป็นของชุดข้อมูล X ที่มีคุณลักษณะ n ตัวจะถูกจำแนกเป็นกลุ่ม C_i มีค่าเท่ากับ

$$P(C_i \mid A_1,\ldots,A_n) = \frac{P(A_1,\ldots,A_n \mid C_i) \cdot P(C_i)}{P(A_1,\ldots,A_n)}$$

3. ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine)

แบบจำลองจะเรียนรู้โดยการสร้างสมการเส้นตรง เพื่อแบ่งข้อมูลออกเป็นสองกลุ่ม จากการหาสัมประสิทธิ์ของสมการเพื่อสร้างเส้นแบ่งสำหรับจำแนกประเภทข้อมูล

4. โครงข่ายประสาทเทียม (Neural Network)

แบบจำลองจะเรียนรู้จากการรับข้อมูลนำเข้าเป็นเวกเตอร์จำนวนจริง แล้วนำมาคำนวณหาผลรวมเชิงเส้นแบบถ่วงน้ำหนัก และให้ข้อมูลส่งออกเป็นค่าคงที่ที่แตกต่างกันออกไปตามค่าผลรวมที่ได้จากฟังก์ชันกระตุ้น (activation function) ที่ใช้ โดยข้อมูลส่งออกที่ได้จะถูกนำกลับมาใช้คำนวณหาค่าผิดพลาด เพื่อปรับแก้น้ำหนักของข้อมูลนำเข้าต่อไป

5. การถดถอยแบบโลจิสติก (Logistic Regression)

เป็นการจำแนกข้อมูลที่มีคุณลักษณะเป็นค่าต่อเนื่อง โดยแบบจำลองจะเรียนรู้จากการสร้างสมการโลจิสติก เพื่อใช้คำนวณโอกาสที่จะเกิดของข้อมูลที่ต้องการทำนาย

2.1.4 การเตรียมความพร้อมข้อมูล (Data preprocessing)

การเตรียมความพร้อมข้อมูล เป็นเทคนิคที่ถูกนำมาใช้เพื่อเตรียมความพร้อมข้อมูลก่อนนำไปใช้ในการสร้างแบบจำลอง โดยจะทำการแปลงข้อมูลดิบให้อยู่ในรูปแบบที่เหมาะสมสำหรับนำไปใช้ในการสร้างแบบจำลอง ทำให้การทำเหมืองข้อมูลมีประสิทธิภาพและมีความแม่นยำมากขึ้น [4] โดยขั้นตอนในกระบวนการเตรียมความพร้อมข้อมูลจะประกอบด้วยขั้นตอน ดังนี้

1. การทำความสะอาดข้อมูล (Data cleaning)

เป็นการกำจัดข้อมูลที่ไม่เกี่ยวข้องหรือไม่เป็นประโยชน์ต่อการค้นหาออก ด้วยกระบวนการดังต่อไปนี้

- การตัดทิ้งระเบียนซ้ำซ้อน (duplicated records)
- การจัดการกับข้อมูลรบกวน (noise)
- การปรองดองความไม่สอดคล้อง (resolve inconsistencies)

- การเติมค่าขาดหาย
- การปรับเรียบข้อมูลรบกวน (smoothing)

2. การตัดคำ (Tokenization)

เป็นการแตกข้อมูลให้เป็นหน่วยย่อยของภาษา หรือโทเคน (Token) ซึ่งแต่ละโทเคนจะถูกแยกออกจากกันด้วยช่องว่าง

3. การตัดคำทั่วไปออก (Stop words)

คำทั่วไปคือคำที่มีความถี่สูงในเอกสารส่วนใหญ่ ดังนั้นคำเหล่านี้จึงไม่ช่วยเพิ่มประสิทธิภาพในการค้นหาข้อมูล

4. การแปลงคำให้เป็นรูปแบบดั้งเดิม (Lemmatization)

เป็นการตัดส่วนขยายของคำออก เช่น คำที่เติม s, es, ed หรือ ing โดยอาศัยบริบทของคำ เช่น คำที่เป็นคำนาม คำกริยา คำวิเศษณ์ เป็นต้น

5. การคัดเลือกคุณลักษณะ (Feature selection)

เป็นการคัดเอาเฉพาะข้อมูลที่มีคุณสมบัติที่เกี่ยวข้องหรือเป็นประโยชน์ต่อการค้นหามาใช้เท่านั้น เพื่อเป็นการเพิ่มประสิทธิภาพให้กับแบบจำลอง

2.1.5 การนับเวกเตอร์ (Count vectorization)

เป็นการนับจำนวนของคำแต่ละคำที่ปรากฏอยู่ในเอกสาร โดยแต่ละคำจะแทนหนึ่งคุณลักษณะของข้อมูล [5] ดังรูปที่ 2 ซึ่งการกำหนดคุณลักษณะจะช่วยให้การจัดการกับเอกสารจำนวนมากสามารถทำได้รวดเร็วและสะดวกยิ่งขึ้น

	Rome	Paris				word V
Rome =	[**1**,	0,	0,	0,	0,	..., 0]
Paris =	[0,	**1**,	0,	0,	0,	..., 0]
Italy =	[0,	0,	**1**,	0,	0,	..., 0]
France =	[0,	0,	0,	**1**,	0,	..., 0]

รูปที่ 2 รูปแสดงตัวอย่างการแทนคุณลักษณะของคำ

2.1.6 การให้ความสำคัญคำ (Word attention)

เป็นเทคนิคหนึ่งที่นำมาใช้ในการเรียนรู้เชิงลึกเพื่อสร้างแบบจำลอง โดยการให้น้ำหนักคำแต่ละคำเรียงตามลำดับความสำคัญของคำนั้น ๆ ทำให้แบบจำลองสามารถทำนายผลลัพธ์ได้แม่นยำมากกว่าการมองคำแต่ละคำเป็นเวกเตอร์ที่มีน้ำหนักเท่ากัน [6] [7]

2.1.7 ความถี่–ส่วนกลับของความถี่ของคำ (Term Frequency-Inverse Document Frequency)

การสร้างแบบจำลองโดยการทำเหมืองข้อมูล จะอาศัยหลักการวิเคราะห์หาคำสำคัญจากคำที่ปรากฏอยู่ในเอกสารหรือชุดข้อมูลสอน เพื่อสร้างแบบจำลองเชิงทำนาย โดยเทคนิคที่ใช้ในการวิเคราะห์หาคำสำคัญมี 2 วิธี [8] ดังนี้

1. **ความถี่ของคำ (Term Frequency: TF)**

   เป็นวิธีการนับความถี่ของคำแต่ละคำที่ปรากฏอยู่ในเอกสาร และใช้ค่าความถี่ของคำนั้น ๆ เป็นตัวชี้วัดความสำคัญของคำ

2. **ความถี่–ส่วนกลับของความถี่ของคำ**

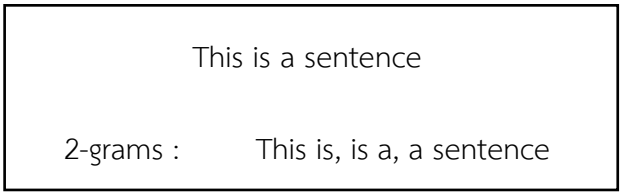
   เป็นวิธีการคำนวณจากผลคูณระหว่างค่าความถี่กับค่าส่วนกลับของความถี่ของคำ (Inverse Document Frequency: IDF) โดยส่วนกลับของความถี่ของคำจะแสดงให้เห็นว่าคำใดที่ปรากฏอยู่ในเอกสารหลาย ๆ ชุด คำนั้นย่อมมีความสำคัญลดลง ซึ่งถ้าผลคูณที่ได้มีค่ามากแสดงว่าคำนั้นมีความถี่สูงในเอกสารที่ใช้คำนวณ แต่มีความถี่ต่ำในเอกสารอื่น ๆ

2.1.8 เอ็นแกรม (N-gram)

เอ็นแกรมเป็นการสร้างแบบจำลองโดยการคำนวณค่าความน่าจะเป็นของชุดอักขระ (character sequence) ที่เกิดขึ้นร่วมกันเป็นคำ หรือค่าความน่าจะเป็นของคำที่เขียนเรียงกัน (word sequence) ที่เกิดขึ้นร่วมกันในประโยค โดยค่าความน่าจะเป็นของคำสามารถประมาณได้จากคลังข้อมูลที่สร้างไว้

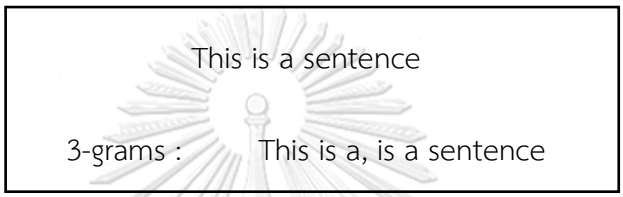
แกรมคือหน่วยที่ใช้ในการสร้างแบบจำลอง อาจเป็นคำหรืออักขระ โดยแกรมสามารถมีได้หลายขนาดตั้งแต่ 1 จนถึง n ตามแต่กำหนด เช่น แบบจำลองจากการประมาณค่าด้วย 2-แกรม 3-แกรม 4-แกรม เป็นต้น

การประมาณค่าด้วย 2-แกรม (Probability bi-grams) คือการประมาณค่าความน่าจะเป็นของสายพยางค์ที่เกิดขึ้นร่วมกันว่ามีค่าเท่ากับผลคูณของความน่าจะเป็นที่จะพบพยางค์ทีละ 2 พยางค์ ติดกันในสายพยางค์ ดังตัวอย่างในรูปที่ 3



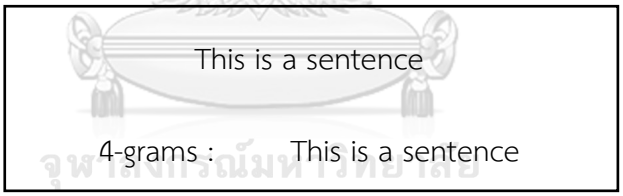
รูปที่ 3 รูปแสดงตัวอย่างของ 2-แกรม

การประมาณค่าด้วย 3-แกรม (Probability tri-grams) คือการประมาณค่าความน่าจะเป็นของสายพยางค์ที่เกิดขึ้นร่วมกันว่ามีค่าเท่ากับผลคูณของความน่าจะเป็นที่จะพบพยางค์ทีละ 3 พยางค์ ติดกันในสายพยางค์ ดังตัวอย่างในรูปที่ 4



รูปที่ 4 รูปแสดงตัวอย่างของ 3-แกรม

การประมาณค่าด้วย 4-แกรม (Probability quad-grams) คือการประมาณค่าความน่าจะเป็นของสายพยางค์ที่เกิดขึ้นร่วมกันว่ามีค่าเท่ากับผลคูณของความน่าจะเป็นที่จะพบพยางค์ทีละ 4 ตัว ติดกันในสายพยางค์ ดังตัวอย่างในรูปที่ 5



รูปที่ 5 รูปแสดงตัวอย่างของ 4-แกรม

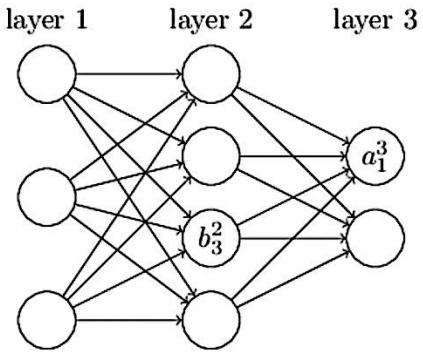
การประมาณค่าความน่าจะเป็นของสายอักขระโดยการใช้เอ็นแกรม คือการใช้สมมติฐานของมาร์คอฟ (Markov assumption) ที่ว่า การปรากฏของตัวอักษรตัวหนึ่งขึ้นอยู่กับตัวอักษรก่อนหน้าเพียง n-1 ตัว ซึ่งวิธีการนี้มักนิยมนำมาใช้ในงานทางด้านภาษาศาสตร์ ดังเห็นได้จากงานวิจัยของคาฟนาร์และเทรงเคิล [9], คอมบริคและโบธา [10], เปงและคณะ [11] มีการนำเอ็นแกรมมาใช้ เพื่อให้แบบจำลองสามารถระบุภาษาของข้อมูลได้อย่างมีประสิทธิภาพและสะดวกยิ่งขึ้น

ตัวอย่างงานของคาฟนาร์และเทรงเคิล ทำการจำแนกประเภทข้อความในแต่ละภาษา จากการคำนวณความน่าจะเป็นการต่อเนื่องกันของแต่ละข้อความ และสร้างแบบจำลองเอ็นแกรมของสายอักขระนั้น โดยใช้วิธีตั้งแต่ 1 ไปจนถึง 5-แกรม ซึ่งภายใน

แบบจำลองภาษาจะแสดงค่าความถี่และค่าการเกิดร่วมกันของเอ็นแกรมทั้งหมด จากนั้นใช้วิธีค่าความห่าง (distance measure) เพื่อเลือกประเภทที่ค่าความห่างน้อยสุดมาเป็นคำตอบ ซึ่งผลการทดลองของงานนี้พบว่าแบบจำลองสามารถระบุภาษาได้ถูกต้องมากกว่า 90% โดยผลการทดลองจะแตกต่างกันออกไปขึ้นอยู่กับขนาดของชุดข้อมูลสอนและข้อมูลทดสอบ

2.1.9 การเรียนรู้เชิงลึก (Deep learning)

เป็นวิธีการหนึ่งของการเรียนรู้ของเครื่องที่พยายามจะเรียนรู้วิธีการแทนข้อมูลอย่างมีประสิทธิภาพ ซึ่งหลักการของการเรียนรู้เชิงลึกคืออัลกอริทึมที่พยายามสร้างแบบจำลองเพื่อแทนความหมายของข้อมูลในระดับสูง จากการสร้างสถาปัตยกรรมข้อมูลดังรูปที่ 6 ซึ่งจะประกอบด้วยโครงสร้างย่อย ๆ หลายชั้น เช่น การแทนรูปภาพด้วยเวกเตอร์ของความสว่างต่อจุดพิกเซล (pixel) ซึ่งการแทน ความหมายจะทำให้การเรียนรู้การทำงานต่าง ๆ ทำได้ง่ายขึ้น เช่น การรู้จำใบหน้า การรู้จำการแสดงออกทางสีหน้า เป็นต้น [12]



รูปที่ 6 รูปแสดงโครงสร้างของการเรียนรู้เชิงลึกหรือโครงข่ายประสาทเทียม

จากรูปที่ 6 ค่าของแต่ละโหนดในโครงข่ายประสาทเทียมจะสามารถคำนวณได้จากสมการ

$$a_j^l = \sigma\left(\sum_k w_{jk}^l a_k^{l-1} + b_j^l\right)$$

[13]

โดย a_j^l แทนค่าของโหนด

l แทนลำดับชั้นในโครงข่ายประสาทเทียม

j แทนตำแหน่งในแต่ละชั้นซ่อนตัว

w_{jk}^l แทนค่าน้ำหนักของโหนด

b_j^l แทนค่าความเอนเอียง (bias)

2.1.10 การวัดความแม่นยำ

การวัดความแม่นยำในการทำนายกลุ่มตัวอย่างใหม่ของแบบจำลอง สามารถวัดได้ด้วย 2 วิธี [14] ดังนี้

1. **วิธีการแบ่งข้อมูลออกเป็น 2 ชุด (Hold Method)**

เป็นวิธีที่เหมาะกับชุดข้อมูลขนาดใหญ่ ตัวอย่างในชุดข้อมูลจะถูกแบ่งออกเป็น 2 ส่วนแบบสุ่ม ด้วยอัตราส่วนขนาดของชุดข้อมูลสอนเท่ากับ 2/3 และขนาดของชุดข้อมูลทดสอบเท่ากับ 1/3 โดยใช้ชุดข้อมูลสอนในการสร้างแบบจำลองการจำแนกประเภท และตรวจสอบความถูกต้องในการจำแนกประเภทข้อมูลใหม่หรือที่ไม่เคยเห็นมาก่อนด้วยชุดข้อมูลทดสอบ ค่าความแม่นยำคำนวณได้จากอัตราส่วนระหว่างจำนวนตัวอย่างในชุดข้อมูลทดสอบที่ทำนายกลุ่มได้อย่างถูกต้องกับจำนวนตัวอย่างทั้งหมดในชุดข้อมูลทดสอบ

2. **วิธีการแบ่งข้อมูลออกเป็น k ชุด (K-fold Cross Validation)**

เป็นวิธีที่เหมาะกับชุดข้อมูลจำนวนไม่มาก สมมติว่าขนาดของชุดข้อมูลเท่ากับ N ตัวอย่างในชุดข้อมูลจะถูกแบ่งออกเป็น k ส่วน โดยแต่ละชุดข้อมูลจะมีขนาด N/k วิธีนี้จะเรียนรู้ด้วยชุดข้อมูลสอนและตรวจสอบความถูกต้องในการจำแนกประเภทด้วยชุดข้อมูลทดสอบเป็นจำนวนทั้งหมด k รอบ โดยรอบที่ i จะใช้ชุดข้อมูลทดสอบชุดที่ i และใช้ชุดข้อมูลที่เหลือเป็นชุดข้อมูลสอน ค่าความแม่นยำคำนวณได้จากอัตราส่วนระหว่างจำนวนตัวอย่างในชุดข้อมูลทดสอบที่ทำนายกลุ่มได้อย่างถูกต้องทั้งหมด k รอบกับจำนวนตัวอย่างทั้งหมดในชุดข้อมูล

2.2 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับงานวิจัยนี้สามารถแบ่งออกได้เป็น 2 กลุ่มใหญ่ ๆ คือ กลุ่มงานที่เกี่ยวกับแอพพลิเคชั่นสำหรับวินิจฉัยโรค และกลุ่มงานที่เกี่ยวกับเทคนิคการฝังคำด้วยการเรียนรู้เชิงลึก

2.2.1 แอพพลิเคชั่นสำหรับวินิจฉัยโรค

ปัจจุบันการเรียนรู้ด้วยเครื่อง (Machine Learning) ได้ถูกนำมาใช้อย่างกว้างขวางในทุกสาขาวิชา โดยเฉพาะอย่างยิ่งในทางการแพทย์ การเรียนรู้ด้วยเครื่องจะเข้ามาช่วยให้การทำงานของแพทย์มีความแม่นยำและมีประสิทธิภาพมากขึ้น ตัวอย่างเช่น ในประเทศอินเดีย ประชากรที่มีอายุต่ำกว่า 30 ปีได้เสียชีวิตลงด้วยโรคหัวใจเป็นจำนวนมาก ดังนั้นเพื่อช่วยลดจำนวนผู้เสียชีวิตด้วยโรคหัวใจ การทำนายโอกาสเกิดโรคหัวใจได้ตั้งแต่เนิ่น ๆ ด้วยความรวดเร็วและความแม่นยำจึงถือเป็นปัจจัยสำคัญในการแก้ปัญหา ซึ่งมีงานวิจัยของคานนาน วาสานธี [15] ได้นำเอาเทคนิคการเรียนรู้ด้วยเครื่อง 4 เทคนิคมาสร้างแบบจำลองเพื่อเปรียบเทียบผลลัพธ์ที่ได้กัน ด้วยกราฟส่วนโค้งอาร์โอซี (ROC curve) เพื่อใช้สำหรับวินิจฉัยโรคหัวใจโดยใช้ตัวแปร 14 ตัวจากชุดข้อมูล

ก่อนหน้านี้ ได้มีงานวิจัยที่นำเอาการทำเหมืองข้อความมาประยุกต์ใช้เพื่อสร้างแบบจำลองสำหรับจำแนกประเภทโรค [16] โดยแบบจำลองในงานวิจัยนี้จะถูกสร้างมาจากตัวจำแนกประเภทที่แตกต่างกันทั้งหมด 4 ตัว ได้แก่ ต้นไม้ตัดสินใจ การเรียนรู้เบย์อย่างง่าย ซัพพอร์ตเวกเตอร์แมชชีน และโครงข่ายประสาทเทียม จากนั้นจะนำแบบจำลองทั้ง 4 อันมาเปรียบเทียบกันด้วยระยะเวลาที่ใช้ในการสร้าง ระยะเวลาที่ใช้ในการทำนาย กราฟเส้นโค้งอาร์โอซี อัตราผลบวกจริง อัตราผลบวกเท็จ ค่าความเที่ยง และค่าความแม่น โดยผลลัพธ์ที่ได้พบว่าแบบจำลองที่สร้างจากโครงข่ายประสาทเทียมจะให้อัตราผลบวกจริงสูงสุดที่ร้อยละ 89.03

2.2.2 เทคนิคการฝังคำด้วยการเรียนรู้เชิงลึก

งานด้านการทำเหมืองข้อความส่วนใหญ่ให้ผลลัพธ์ที่น่าพอใจ ตัวอย่างเช่น ชุมชนสุขภาพออนไลน์ (The online health communities) ถือเป็นจุดแลกเปลี่ยนข้อมูลเกี่ยวกับปัญหาสุขภาพต่าง ๆ เพื่อช่วยผู้ใช้จัดการกับปัญหาสุขภาพ แต่เนื่องด้วยด้วยขนาดของชุมชนสุขภาพออนไลน์มีขนาดใหญ่มาก ทำให้ผู้ดูแลไม่สามารถเข้าไปมีส่วนร่วมได้ในทุกบทสนทนา ซึ่งในบางบทสนทนานั้นจำเป็นต้องมีผู้เชี่ยวชาญเข้าไปช่วยให้ความรู้ ดังนั้นจึงมีงานวิจัยของเยทิสเจนยีดิซ และแพรท [17] ที่นำเอาการวิเคราะห์ความรู้สึก (Sentiment analysis) การ

คัดเลือกคุณลักษณะ (Feature selection) และการเกลี่ยข้อมูลสอน (Balancing training data) มาใช้เพื่อเพิ่มค่าพื้นที่ใต้เส้นโค้ง (AUC: Area Under Curve) และค่าเอฟ (F1-score)

มิโคลอฟ, เชน, คอร์ราโด, ดีน [18] นำเสนอสถาปัตยกรรมแบบจำลองสำหรับการคำนวณค่าตัวแทนเวกเตอร์ต่อเนื่องของคำในฐานข้อมูลขนาดใหญ่ โดยคุณภาพของตัวแทนจะถูกวัดจากความคล้ายคลึงของคำ ซึ่งผลลัพธ์ที่ได้จะถูกนำไปเปรียบเทียบกับประสิทธิภาพของเทคนิคก่อนหน้าตามชนิดของโครงข่ายประสาทเทียม และพบว่าค่าความแม่นยำสามารถมีค่าเพิ่มขึ้นได้ ด้วยการเพิ่มต้นทุนการคำนวณที่น้อยมาก

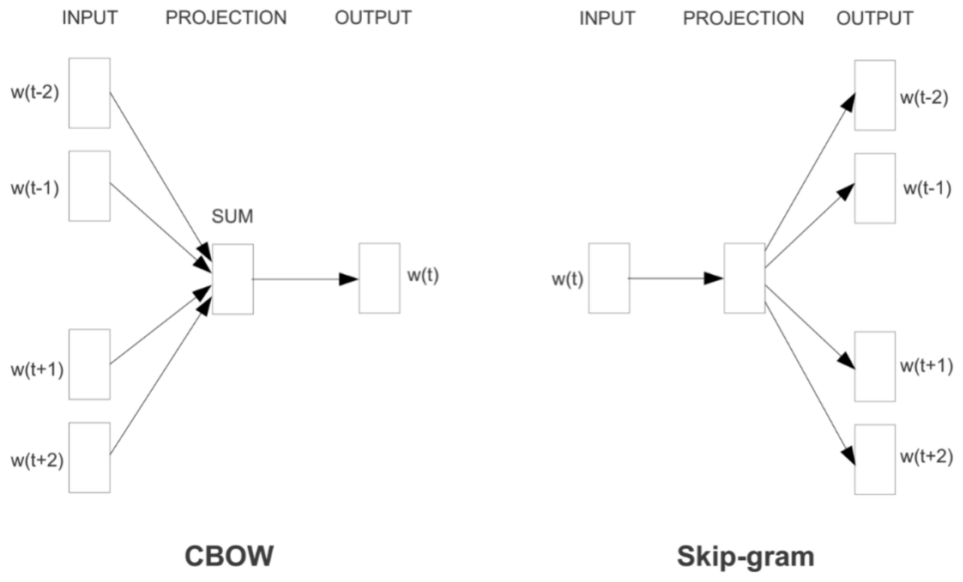
การทำเหมืองข้อความช่วยให้ผู้ใช้สามารถใช้ประโยชน์จากข้อมูลที่เป็นประโยชน์ที่ถูกซ่อนอยู่ในฐานข้อมูลขนาดใหญ่ได้อย่างมีประสิทธิภาพ โดยการกระจายของข้อมูลและความไวต่อความหมายของบริบทมักเป็นอุปสรรคต่อการจำแนกประเภทของข้อความขนาดสั้น ดังนั้นเพื่อจัดการกับอุปสรรคนี้ หวัง, ซู, เถียน, หลิว, เหา [19] จึงได้เสนอโครงร่างสำหรับขยายความข้อความขนาดสั้น โดยอาศัยการจัดกลุ่มคำและโครงข่ายประสาทเทียม ซึ่งคำที่เกี่ยวข้องกันมักอยู่ใกล้กัน ดังนั้นการหาความหมายของคำจึงสามารถทำได้รวดเร็วด้วยวิธีการจัดกลุ่ม โดยการเพิ่มองค์ประกอบของคำผ่านการฝังคำจากบริบทของคำ

ตัง, เหวย, หยาง, ซู, หลิว, ควิน [20] เสนอวิธีการฝังคำสำหรับจำแนกประเภททวิตเตอร์ ซึ่งส่วนใหญ่อัลกอริทึมสำหรับการเรียนรู้การเป็นตัวแทนของคำอย่างต่อเนื่องมักถูกจำลองมาจากบริบทของคำ และละเลยความรู้สึกของข้อความ จึงถือเป็นปัญหาสำหรับการวิเคราะห์ความรู้สึกของคำ ดังนั้นในงานวิจัยนี้จึงเสนอวิธีแก้ปัญหาด้วยการเรียนรู้การฝังคำเฉพาะ (SSWE: Sentiment Specific Word Embedding) จากการเข้ารหัสความรู้สึกของข้อมูลในตัวแทนของคำอย่างต่อเนื่อง โดยการพัฒนาโครงข่ายประสาทเทียม

เป้าหมายของการสร้างแบบจำลองภาษาเชิงสถิติคือการเรียนรู้ฟังก์ชันความน่าจะเป็นร่วมของลำดับคำในภาษา ซึ่งถือเป็นสิ่งที่ยากเนื่องจากมิติของคำในชุดข้อมูลทดสอบมักจะมีลำดับของคำที่แตกต่างจากลำดับของคำในขั้นตอนการเรียนรู้ โดยวิธีการดั้งเดิมที่ประสบความสำเร็จมักมาจากพื้นฐานของเอ็นแกรม (N-grams) ซึ่งมีการเรียงลำดับการทับซ้อนกันสั้น ๆ ของข้อมูลในชุดข้อมูลสอน ดังนั้นจึงมีงานของเบนจิโอ้ [21] ที่นำเสนอวิธีการทำลายมิติของคำ ด้วยการเรียนรู้การเป็นตัวแทนการกระจายตัวของคำ ซึ่งช่วยให้แต่ละประโยคสามารถสร้างค่าตัวแทนที่เกี่ยวข้องกับประโยคที่มีความหมายใกล้เคียงกันได้ พร้อมทั้งสามารถคำนวณหาฟังก์ชันความน่าจะเป็นของลำดับคำที่ถูกแสดงในรูปแบบตัวแทนได้อีกด้วย โดยการวางนัยของคำจะอาศัยหลักกการวางลำดับของคำที่ไม่เคยเห็นมาก่อน หากมีลำดับที่ใกล้เคียงกับคำในประโยคที่เคยเห็นก่อนหน้า ค่าตัวแทนของคำในประโยคนั้นก็จะคล้ายคลึงกัน โดยในงานวิจัยได้ทำการทดลองใช้โครงข่ายประสาทเทียมสำหรับสร้างฟังก์ชัน

ความน่าจะเป็นของคำ และพบว่าฟังก์ชันที่ได้สามารถเข้ามาช่วยปรับปรุงเรื่องนัยของคำในแบบจำลองเอ็นแกรมให้มีประสิทธิภาพมากขึ้นและยังสามารถนำไปใช้ประโยชน์กับคำในบริบทที่ยาวขึ้นได้

รูปที่ 7 แสดงโครงสร้างสคิปแกรม (Skip-gram) ซึ่งเป็นวิธีที่นิยมนำมาใช้ เนื่องจากเป็นวิธีที่มีประสิทธิภาพสำหรับการเรียนรู้การเป็นตัวแทนของเวกเตอร์แบบกระจาย และมีเวกเตอร์ที่เป็นตัวแทนตัวเลขขนาดใหญ่ของการความสัมพันธ์ทางไวยากรณ์และความหมายของภาษาที่แม่นยำ มิโคลอฟจึงได้นำเสนอวิธีการปรับปรุงประสิทธิภาพของเวกเตอร์และระยะเวลาที่แบบจำลองใช้ในการเรียนรู้ให้ดีและรวดเร็วยิ่งขึ้น โดยอาศัยความถี่ในการสุ่มคำใหม่บ่อย ๆ เพื่อเพิ่มความรวดเร็วและประสิทธิภาพในการเรียนรู้การแทนคำให้มากขึ้น ซึ่งในงานวิจัยได้อธิบายเทคนิคซอฟต์แม็กซ์ ว่าเป็นเทคนิคแบบลำดับชั้นที่อาศัยการสุ่มตัวอย่างเชิงลบ และพบว่าข้อจำกัดของการแทนคำที่พบได้ทั่วไป คือการไม่สนใจลำดับของคำและการไม่แสดงวลีสำนวนของคำ ตัวอย่างเช่น คำว่า “Canada” และ “Air” ไม่สามารถนำมารวมกันเพื่อให้เกิดเป็น “Air Canada” หรือวลีอย่างง่ายได้ ดังนั้นในงานวิจัยจึงได้นำเสนอวิธีการค้นหาวลีในข้อความอย่างง่ายขึ้น ทำให้พบว่าการเรียนรู้การเป็นตัวแทนเวกเตอร์ที่ดีสำหรับวลีหลายล้านนั้นมีความเป็นไปได้



รูปที่ 7 รูปแสดงโครงสร้างสคิปแกรม

วิธีการล่าสุดสำหรับการเรียนรู้การเป็นตัวแทนเวกเตอร์สเปซ (Vector space) ของคำศัพท์ ประสบความสำเร็จในการแสดงความหมายของคำและคำศัพท์ในเชิงวากยสัมพันธ์โดยใช้เวกเตอร์เชิงคณิตศาสตร์ แต่โครงสร้างของวิธีการเหล่านี้ยังไม่แน่นอน เพนนิงตัน จึงทำการวิเคราะห์เพื่อหาโครงสร้างที่ชัดเจนของวิธีการดังกล่าว และพบว่าผลลัพธ์ที่ได้จากการเรียนรู้คือรูปแบบการถดถอยโลจิสติส์ทั้งหมดแบบใหม่ ที่รวมเอาข้อดีของแบบจำลองสองหลักในอักษร การแยกตัวประกอบเมทริกซ์แบบทั่วไป และวิธีบริบทหน้าต่างแบบจำเพาะมาไว้ด้วยกัน ซึ่งแบบจำลองจะใช้ประโยชน์จากข้อมูลเชิงสถิติ โดยการเรียนรู้แบบเฉพาะในส่วนที่ไม่ใช่ศูนย์ในเมทริกซ์คำศัพท์ แทนที่จะใช้เมทริกซ์แบบกระจายหรือเมทริกซ์บริบทหน้าต่างแบบจำเพาะในฐานข้อมูลขนาดใหญ่ ซึ่งแบบจำลองนี้จะช่วยลดปริภูมิเวกเตอร์ที่โครงสร้างย่อย ๆ เห็นได้จากการเปรียบเทียบค่าประสิทธิภาพของงานมีค่าสูงถึง 75%

ไดและลี ได้นำเสนอวิธีการใช้ข้อมูลที่ไม่มีป้ายกำกับ 2 วิธี เพื่อปรับปรุงการเรียนรู้ลำดับของคำด้วยโครงข่ายที่เกิดขึ้นซ้ำ วิธีแรกคือการทำนายว่าจะเกิดอะไรขึ้นต่อไปในลำดับ ซึ่งเป็นรูปแบบภาษาดั้งเดิมในการประมวลผลภาษาทั่วไป ส่วนวิธีที่สองคือการใช้ตัวเข้ารหัสแบบอัตโนมัติ (Autoencoder) ตามลำดับ ซึ่งอาศัยการอ่านลำดับการป้อนข้อมูลลงในเวกเตอร์และทำการทำนายลำดับการป้อนข้อมูลอีกครั้ง โดยวิธีการทั้ง 2 วิธีนี้สามารถนำมาใช้เป็นขั้นตอนในการเตรียมพร้อมข้อมูลก่อนการเรียนรู้ของแบบจำลองได้ หรือกล่าวอีกนัยหนึ่งคือพารามิเตอร์ที่ได้จากขั้นตอนที่ไม่ได้รับการเรียนรู้สามารถนำมาใช้เป็นจุดเริ่มต้นสำหรับการเรียนรู้อื่น ๆ ของแบบจำลองได้ จากการทดลองพบว่าโครงข่ายที่เกิดขึ้นในระยะสั้น หลังจากหน่วยความจำถูกสั่งการด้วยวิธีการทั้ง 2 วิธีข้างต้น ทำให้โครงข่ายมีความเสถียรสูง และด้วยขั้นตอนการเตรียมพร้อมข้อมูลก่อนการเรียนรู้ ทำให้โครงข่ายนั้นยิ่งมีประสิทธิภาพมากขึ้นในการจัดหมวดหมู่ข้อความ

บทที่ 3
แนวคิดและวิธีดำเนินงาน

งานวิจัยนี้จะสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรคจากอาการของผู้ป่วย โดยใช้ข้อมูลจากเวชระเบียนในส่วนบันทึกของแพทย์มาทำการสร้างแบบจำลอง เพื่อให้แพทย์นำแบบจำลองไปใช้เป็นตัวช่วยในการวินิจฉัยโรค ทำให้การวินิจฉัยโรคของแพทย์มีประสิทธิภาพมากขึ้น ซึ่งแนวคิดในการสร้างแบบจำลองของงานวิจัยนี้สามารถแบ่งได้เป็น 4 ส่วน คือ การทำความสะอาดข้อมูล การสร้างแบบจำลอง การใช้งานแบบจำลอง และการวัดประสิทธิภาพแบบจำลอง

3.1 การเก็บข้อมูล

ข้อมูลที่นำมาใช้ในการสร้างแบบจำลองของงานวิจัยนี้นำมาจากข้อมูลเวชระเบียนผู้ป่วยในแผนกออร์โธปิดิกส์ของโรงพยาบาลจุฬาลงกรณ์ ซึ่งสามารถจำแนกโรคที่พบในเวชระเบียนแบ่งตามหมวดหมู่โรคได้ดังตารางที่ 1

ตารางที่ 1 ตารางแสดงจำนวนโรคในแต่ละหมวดหมู่

ลำดับ	หมวดหมู่โรค	ร้อยละ	จำนวน
1	Diseases of the musculoskeletal system and connective tissue	55.73%	7,526
2	Injury, poisoning and certain other consequences of external causes	32.28%	4,359
3	Neoplasms	4.27%	577
4	Congenital malformations, deformations and chromosomal abnormalities	2.51%	340
5	Diseases of the nervous system	1.74%	235
6	Factors influencing health status and contact with health services	1.15%	155
7	Diseases of the skin and subcutaneous tissue	0.84%	114
8	Certain infectious and parasitic diseases	0.49%	66
9	Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified	0.47%	63
10	Diseases of the circulatory system	0.22%	30

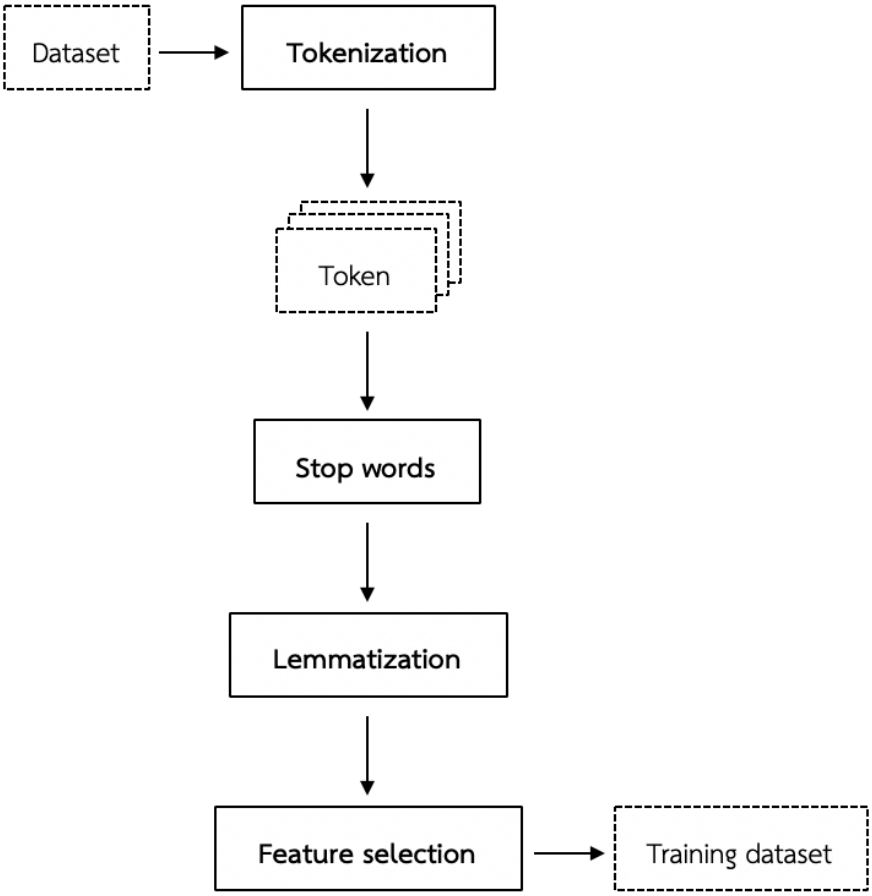
ลำดับ	หมวดหมู่โรค	ร้อยละ	จำนวน
11	Endocrine, nutritional and metabolic diseases	0.16%	21
12	Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism	0.06%	8
13	Diseases of the digestive system	0.03%	4
14	Mental, Behavioral and Neurodevelopmental disorders	0.01%	1
15	Diseases of the eye and adnexa	0.01%	1
16	Diseases of the ear and mastoid process	0.01%	1
17	Diseases of the respiratory system	0.01%	1
18	Diseases of the genitourinary system	0.01%	1
	รวม	100%	13,503

3.2 การทำความสะอาดข้อมูล

เริ่มจากการเก็บข้อมูลอาการและผลการวินิจฉัยโรคที่ถูกบันทึกอยู่ในส่วนบันทึกของแพทย์ในเวชระเบียนผู้ป่วยของโรงพยาบาลจุฬาลงกรณ์ ซึ่งข้อมูลอาการที่นำมาใช้ในงานวิจัยนี้จะเป็นข้อมูลที่เป็นภาษาอังกฤษเท่านั้น หากข้อมูลเป็นภาษาไทยจะทำการแปลภาษาข้อมูลก่อนนำมาใช้ และชื่อโรคที่นำมาใช้จะเป็นชื่อโรคที่อ้างอิงมาจากชื่อโรคที่อยู่ในฐานข้อมูลของรหัสไอซีดีเทน

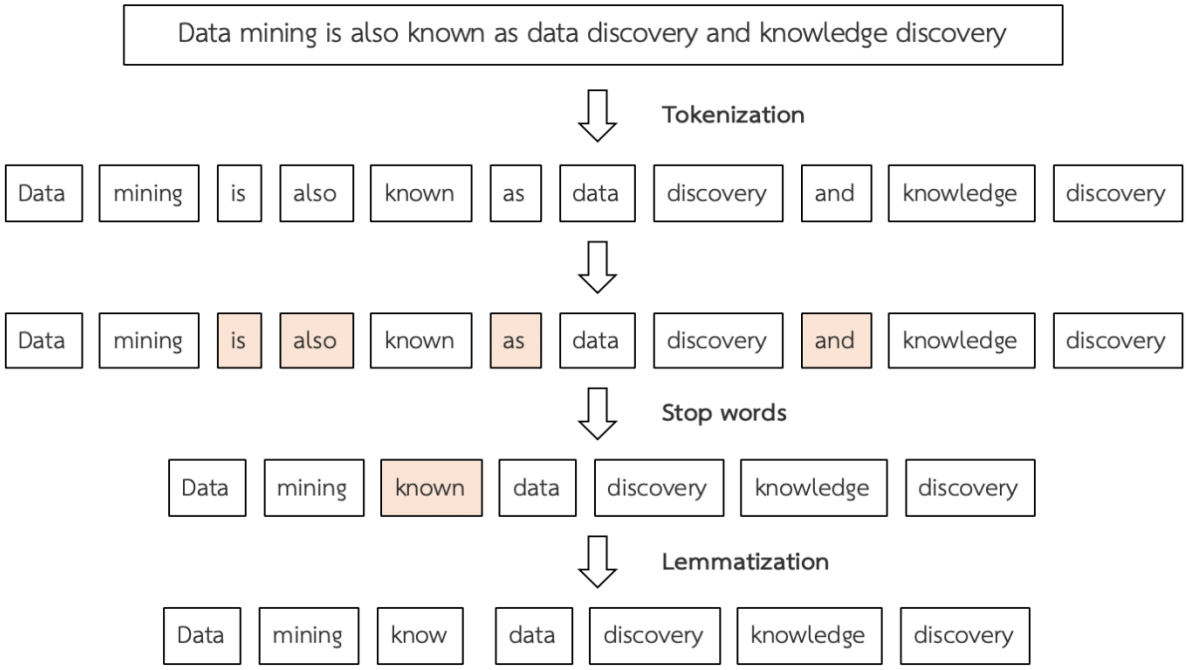
การทำความสะอาดข้อมูล จะเริ่มจากการตัดเวชระเบียนที่มีข้อมูลที่ไม่สมบูรณ์ทิ้ง และตัดเวชระเบียนที่ข้อมูลในส่วนบันทึกของแพทย์เป็นเพียงบันทึกการนัดหมายหรือวันที่ทิ้ง จากนั้นจะทำการแปลงข้อมูลให้เป็นหน่วยย่อยของภาษาหรือคำ ก่อนเข้าสู่กระบวนการประมวลผลข้อมูลก่อน

การประมวลผลข้อมูลก่อน เป็นการทำให้ข้อมูลอยู่ในรูปแบบที่เหมาะสมสำหรับนำไปใช้ในการสร้างแบบจำลอง ซึ่งจะประกอบไปด้วยขั้นตอนของการแตกคำ การตัดคำที่ไม่สำคัญ การตัดส่วนขยาย การเปลี่ยนคำให้อยู่ในรูปแบบดั้งเดิม และการคัดเลือกคำสำคัญสุดท้ายผลลัพธ์ที่ได้จะนำมาใช้เป็นชุดข้อมูลสอน สำหรับสร้างแบบจำลองในขั้นตอนการเรียนรู้ของแบบจำลอง โดยขั้นตอนในกระบวนการประมวลผลข้อมูลก่อนสามารถแสดงได้ ดังรูปที่ 8



รูปที่ 8 รูปแสดงกระบวนการประมวลผลข้อมูลก่อน

จากขั้นตอนในกระบวนการประมวลผลข้อมูลก่อน สามารถนำมาประยุกต์ใช้กับข้อมูลที่เป็นข้อความทางการแพทย์ในงานวิจัยนี้ได้ ดังรูปที่ 9



รูปที่ 9 รูปแสดงตัวอย่างกระบวนการประมวลผลข้อมูลก่อน

3.3 เวกเตอร์น้ำหนัก

ในงานวิจัยนี้ได้เลือกเทคนิคที่ใช้ในการสร้างเวกเตอร์น้ำหนักมา 2 เทคนิค ดังนี้

3.3.1 เวิร์ดทูเวก (Word2Vec)

เป็นการแปลงคำให้เป็นเวกเตอร์ โดยใช้หลักการสอนแบบจำลองด้วยบริบทของคำแต่ละคำ ซึ่งคำที่คล้ายกันจะถูกแสดงด้วยค่าตัวเลขที่คล้ายกัน เช่นเดียวกับหลักการของโครงข่ายประสาทเทียมแบบป้อน (Neural network feeder) ที่มีการเชื่อมต่อกันอย่างหนาแน่น จะมีชุดข้อมูลของตัวแปรอิสระและตัวแปรตามตามจำนวนเป้าหมายที่ต้องการทำนาย โดยมีการแบ่งประโยคออกเป็นคำย่อย ๆ และสร้างจำนวนคู่ของคำขึ้นมาตามขนาดของหน้าต่าง ตัวอย่างคู่ของคำเช่น (“New”, “York”)

กำหนด ให้“New” เป็นตัวแปรอิสระหรือตัวแปร X และ

ให้ “York” เป็นตัวแปรตามหรือตัวแปร Y ที่ตั้งเป้าหมายว่าจะทำนาย

เมื่อทำการป้อนคำว่า “New” ลงไปในโครงข่าย คำนั้นจะถูกนำไปผ่านชั้นการฝังคำที่กำหนดค่าเริ่มต้นไว้ด้วยค่าน้ำหนักแบบสุ่ม และถูกส่งผ่านชั้นซอฟต์แม็กซ์ เพื่อทำให้ผลการทำนายออกมาเป็นคำว่า “York” โดยเทคนิคที่ใช้ในการปรับน้ำหนักให้เหมาะสม คือ เทคนิค

เอสจีดี (SGD: Stochastic Gradient Descent) เป็นเทคนิคใช้เพื่อลดค่าฟังก์ชันการสูญเสีย หลักการคือพยายามลดค่าการสูญเสียของการทำนายคำเป้าหมายที่จะได้รับจากคำบริบท ซึ่งหากทำสิ่งนี้ด้วยจำนวนรอบที่เพียงพอในชั้นการฝังคำ จะทำให้ได้คำศัพท์ของเวกเตอร์คำหรือที่เรียกว่า พิกัดของคำ ในพื้นที่เวกเตอร์เรขาคณิต

3.3.2 โกรฟ (GLOVE)

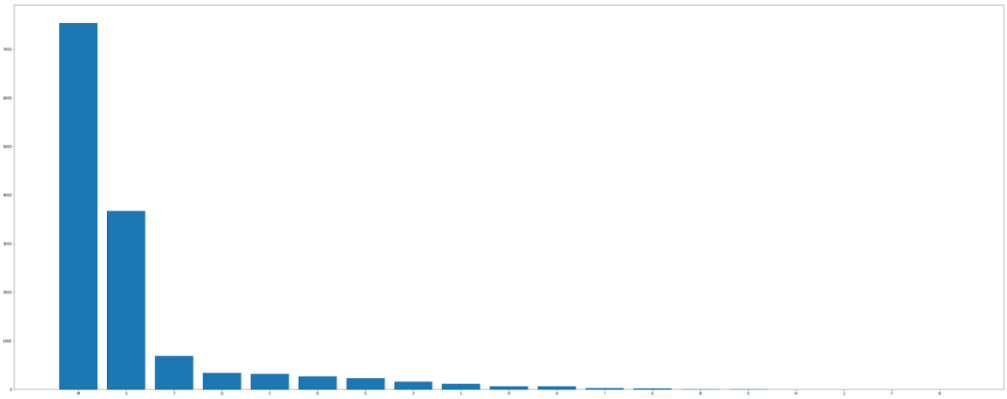
มีหลักการทำงานในลักษณะเดียวกับเวิร์ดทูเวก ซึ่งในเวิร์ดทูเวกจะเป็นการทำนายแบบคาดการณ์ที่ทำนายบริบทของคำจากคำที่กำหนด ส่วนโกรฟจะเป็นการเรียนรู้จากการสร้างเมทริกซ์ของคำที่เกิดขึ้นร่วมกัน โดยการนับจำนวนคำที่ปรากฏบ่อย ๆ ในแต่ละบริบท ทำให้ขนาดของเมทริกซ์จากที่มีขนาดใหญ่มากกลายเป็นเมทริกซ์ที่มีขนาดของมิติต่ำลง

3.4 การสร้างฉลากประเภท

จากข้อมูลผู้ป่วยที่ได้มาจากเวชระเบียนในแผนกออร์โธปิดิกส์ สามารถจำแนกฉลากประเภทได้ทั้งหมด 287 ประเภท โดยมีการกระจายตัวของข้อมูลในแต่ละฉลากประเภทที่แตกต่างกัน เช่น ในบางฉลากประเภทมีข้อมูลนับพันตัว ในขณะที่บางฉลากประเภทนั้นมีข้อมูลเพียงแค่ 1 หรือต่ำกว่า 10 ตัว ซึ่งอาจส่งผลให้การทำนายที่ได้จากการเรียนรู้เชิงลึกแย่ลง ดังนั้นในงานวิจัยนี้จึงทำการเพิ่มฉลากประเภทขึ้นมาใหม่ 1 อัน สำหรับข้อมูลที่มีจำนวนต่ำกว่า 10 ตัว นั่นคือ ฉลากประเภทโรคอื่น ๆ

3.4.1 การจำแนกฉลากประเภทโรคจากตัวอักษรแรกของรหัสไอซีดีเทน

ฉลากประเภทที่นำมาใช้ในงานวิจัยนี้ คือ รหัสไอซีดีเทน ซึ่งเป็นรหัสที่มีโครงสร้างแบบลำดับชั้น ที่ประกอบด้วยตัวอักษรและตัวเลข โดยตัวแรกของรหัสจะเป็นตัวอักษรภาษาอังกฤษ A ถึง Z ที่จะระบุถึงชนิดของโรคนั้น ๆ และตามด้วยตัวเลขซึ่งจะระบุถึงรายละเอียดย่อย ๆ ของโรคลงไป โดยถ้างานวิจัยนี้ทำการจำแนกฉลากประเภทตามตัวอักษรแรกของรหัสไอซีดีเทน จะทำให้ได้ฉลากประเภททั้งหมด 18 ประเภท ดังรูปที่ 10 เป็นกราฟแท่งแสดงจำนวนข้อมูลที่มีอยู่ในแต่ละฉลากประเภท



รูปที่ 10 รูปแสดงกราฟแท่งแสดงจำนวนข้อมูลที่พบในแต่ละฉลากประเภท

3.4.2 การจำแนกฉลากประเภทโรคด้วยฉลากโรคอื่น ๆ

จากรูปที่ 10 จะเห็นได้ว่า จำนวนข้อมูลที่พบในแต่ละฉลากประเภทมีจำนวนที่แตกต่างค่อนข้างมาก ซึ่งจะส่งผลกระทบต่อการนำข้อมูลนี้ไปใช้ในการเรียนรู้เชิงลึกเพื่อสร้างแบบจำลอง ดังนั้นเพื่อเป็นการแก้ไขปัญหาที่จะเกิดขึ้น งานวิจัยนี้จึงได้ทำการตัดฉลากประเภทที่มีจำนวนข้อมูลต่ำกว่า 10 ตัวออก และเปลี่ยนฉลากประเภทเหล่านั้นเป็น “Other” เพื่อให้ผลลัพธ์ที่ได้จากการเรียนรู้เชิงลึกมีความแม่นยำมากขึ้น

ฉลากประเภทที่ถูกตัดออกในงานวิจัยนี้ ได้แก่ รหัสไอซีดีเทนที่ขึ้นต้นด้วยตัวอักษร B, E, F, H, I, J, K, L, N และ R โดยแต่ละตัวอักษรแทนหมวดหมู่ของโรคดังนี้

- B แทน โรคติดเชื้อและปรสิตบางชนิด
- E แทน โรคของต่อมไร้ท่อ โภชนาการ และเมตาบอลิซึม
- F แทน โรคทางจิต พฤติกรรม และความผิดปกติเกี่ยวกับการพัฒนาทางระบบประสาท
- H แทน โรคของตาและอวัยวะข้างเคียง
- I แทน โรคของระบบไหลเวียนเลือด
- J แทน โรคของระบบหายใจ
- K แทน โรคของระบบย่อยอาหาร
- L แทน โรคของผิวหนังและเนื้อเยื่อใต้ผิวหนัง
- N แทน โรคของระบบสืบพันธุ์

3.4.3 ผลลัพธ์การจำแนกประเภทโรค

เมื่อทำการแทนที่ฉลากประเภทที่มีจำนวนข้อมูลต่ำกว่า 10 ตัวด้วยฉลากประเภท “Other” แล้ว จะทำให้ได้จำนวนฉลากประเภททั้งหมดที่นำมาใช้ในงานวิจัยนี้เท่ากับ 233 ประเภท จากในตอนแรกที่มีฉลากประเภททั้งหมด 287 ประเภท ดังนั้นจะเห็นได้ว่ามีโรคทั้งหมด 54 โรค ที่ถูกจัดอยู่ในฉลากประเภท “Other”

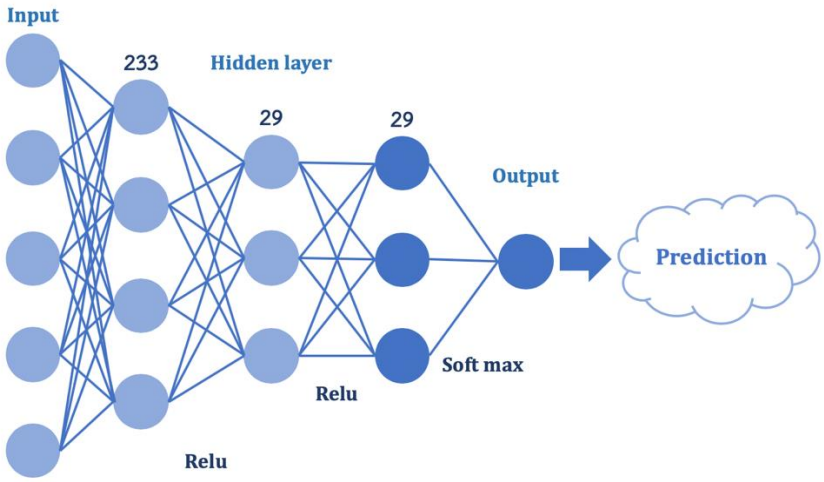
3.5 การสร้างเวกเตอร์น้ำหนักจำเพาะสำหรับทำนายผลการวินิจฉัยโรค

เนื่องจากการใช้เวกเตอร์น้ำหนักที่ได้จากเวิร์ดทูเวกและโกรฟ เป็นเวกเตอร์น้ำหนักที่ได้มาจากมาคำที่ไม่เฉพาะเจาะจงทางการแพทย์ ทำให้ค่าน้ำหนักที่ได้จากการใช้เวิร์ดทูเวกและโกรฟอาจยังไม่ดีพอสำหรับการนำไปใช้ในการสร้างแบบจำลองการเรียนรู้เชิงลึกเพื่อทำนายผลการวินิจฉัยโรค ดังนั้นในงานวิจัยนี้จึงได้สร้างเวกเตอร์น้ำหนักจำเพาะสำหรับใช้ในการวินิจฉัยโรคขึ้นมาอีก 1 เวกเตอร์ เพื่อนำเอาเวกเตอร์ที่ได้นี้ไปใช้คูณกับเวกเตอร์น้ำหนักของเวิร์ดทูเวกหรือโกรฟ และนำเวกเตอร์ผลคูณที่ได้ไปใช้เป็นน้ำหนักในการฝังคำของแบบจำลองการเรียนรู้เชิงลึก

3.6 การสร้างแบบจำลอง

แบบจำลองเชิงทำนายเป็นผลลัพธ์จากการวิเคราะห์ชุดข้อมูลปัจจุบัน เพื่อใช้ในการทำนายประเภทตัวอย่างของข้อมูลในอนาคต โดยแบบจำลองเชิงทำนายเกิดจากการทำเหมืองข้อมูลแบบจำแนกประเภท (Classification) ซึ่งเป็นการแบ่งข้อมูลออกเป็นกลุ่ม ๆ ตามฉลากประเภทที่ทราบล่วงหน้า โดยตัวจำแนกประเภทที่นิยมนำมาใช้ ได้แก่ ต้นไม้ตัดสิน การเรียนรู้เบย์ ซัพพอร์ตเวกเตอร์แมชชีน โครงข่ายประสาทเทียม และการถดถอยแบบโลจิสติก

ในงานวิจัยนี้ การสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรค จัดเป็นการทำเหมืองข้อมูลที่ให้ผลลัพธ์เป็นแบบจำลองเชิงทำนาย โดยข้อมูลที่นำมาใช้ในการสร้างแบบจำลองจะต้องเป็นข้อมูลที่ทราบค่าฉลากประเภท (label) ซึ่งจะเป็นตัวกำหนดประเภทของข้อมูลนั้น ๆ โดยค่าฉลากประเภทที่ใช้ในงานวิจัยนี้คือชื่อโรคหรือรหัสไอซีดีเทน ส่วนข้อมูลที่นำมาใช้ในขั้นตอนการเรียนรู้ของแบบจำลองคือข้อมูลในส่วนบันทึกของแพทย์ที่ได้จากโรงพยาบาล



รูปที่ 11 รูปแสดงโครงสร้างของแบบจำลองการวินิจฉัยแบบแตกต่างเชิงลึก

จากรูปที่ 11 การสร้างแบบจำลองจะเริ่มจากการแปลงข้อมูลทั้งหมดให้อยู่ในรูปเวกเตอร์ และนำเวกเตอร์ที่ได้ไปผ่านกระบวนการเรียนรู้เพื่อสร้างแบบจำลองโดยใช้เทคนิคการเรียนรู้เชิงลึก ซึ่งขั้นตอนการเรียนรู้เชิงลึกจะประกอบด้วยชั้นการเรียนรู้ทั้งหมด 3 ชั้น ได้แก่ ชั้นข้อมูลนำเข้า (Input layer) ชั้นซ่อนตัว (Hidden layer) และชั้นข้อมูลส่งออก (Output layer) โดยชั้นซ่อนตัวจะประกอบด้วยชั้นย่อย ๆ ทั้งหมด 3 ชั้น ดังนี้

- ชั้นย่อยที่ 1 ประกอบด้วยโหนดซ่อนตัว (Hidden node) 29 โหนด โดยฟังก์ชันกระตุ้น (Activation function) ที่ใช้ คือ หน่วยเชิงเส้นแก้ไข (Rectified linear unit)
- ชั้นย่อยที่ 2 ประกอบด้วยโหนดซ่อนตัว 233 โหนด โดยฟังก์ชันกระตุ้น (Activation function) ที่ใช้ คือ หน่วยเชิงเส้นแก้ไข (Rectified linear unit)
- ชั้นย่อยที่ 3 ประกอบด้วยโหนดซ่อนตัว 233 โหนด โดยฟังก์ชันกระตุ้น (Activation function) ที่ใช้ คือ ซอฟต์แมกซ์ (Softmax)

กระบวนการที่ใช้ในชั้นซ่อนตัว จะใช้การคำนวณค่าความสูญเสียย้อนกลับ (Back propagation loss) แบบครอสเอนโทรปีแน่ชัด (Categorical cross-entropy)

ตารางที่ 2 ตารางแสดงขั้นตอนการสร้างแบบจำลองการวินิจฉัยแบบแตกต่างเชิงลึก

Layer (type)	Output Shape	Param #
embedding_5 (Embedding)	(None, 13503, 100)	3515600
dropout_7 (Dropout)	(None, 13503, 100)	0
flatten_5 (Flatten)	(None, 1350300)	0
dense_8 (Dense)	(None, 29)	39158729

dropout_8 (Dropout)	(None, 29)	0
dense_9 (Dense)	(None, 233)	6990
dropout_9 (Dropout)	(None, 233)	0
dense_10 (Dense)	(None, 233)	54522

จากตารางที่ 2 แสดงให้เห็นว่าการสร้างแบบจำลองประกอบไปด้วย 5 ขั้นตอนทั้งหมด ดังนี้

- ขั้นตอนที่ 1 ข้อมูลการนำเข้ากับชั้นคำฝังตัวขนาด (13503,100) มีพารามิเตอร์จำนวน 3,515,600 ตัว ที่จะถูกนำมาผ่านการกำหนดน้ำหนักโดยใช้ค่าที่ได้จากเวิร์ดทูเวกหรือโกรฟ
- ขั้นตอนที่ 2 ทำการสุ่มข้อมูลออก (Drop out) 50% เพื่อลดความพอดีเกิน (Overfitting)
- ขั้นตอนที่ 3 ทำการปรับพื้น (Flatten) มิติของข้อมูลจากสองมิติให้เหลือหนึ่งมิติ
- ขั้นตอนที่ 4 ข้อมูลการนำเข้ากับโหนดภายในชั้นซ่อนตัวมีการเชื่อมต่อกันแบบทั่วถึง (Fully connected) โดยจะทำให้เกิดพารามิเตอร์ทั้งหมด 39,158,729 ตัว ซึ่งจะมีการแปลงข้อมูลในชั้นข้อมูลนำเข้าจาก (None, 1350300) เป็น (None, 29)
- ขั้นตอนที่ 5 ทำการสุ่มข้อมูลออก 50% เพื่อลดความพอดีเกิน
- ขั้นตอนที่ 6 ข้อมูลการนำเข้ากับโหนดภายในชั้นซ่อนตัวมีการเชื่อมต่อกันแบบทั่วถึง โดยจะทำให้เกิดพารามิเตอร์ทั้งหมด 6,990 ตัว ซึ่งจะมีการแปลงข้อมูลในชั้นซ่อนตัวก่อนหน้าจาก (None, 29) เป็น (None, 233)
- ขั้นตอนที่ 7 ทำการสุ่มข้อมูลออก 50% เพื่อลดความพอดีเกิน
- ขั้นตอนที่ 8 ข้อมูลการนำเข้ากับโหนดภายในชั้นซ่อนตัวมีการเชื่อมต่อกันแบบทั่วถึง (Fully connected) โดยจะทำให้เกิดพารามิเตอร์ทั้งหมด 54,522 ตัว ซึ่งจะมีการแปลงข้อมูลในชั้นซ่อนตัวก่อนหน้าจาก (None, 233) เป็น (None, 233)

3.7 เครื่องมือที่ใช้ในการพัฒนาแบบจำลอง

1. ฮาร์ดแวร์ที่ใช้ในการพัฒนาเครื่องมือ

   เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนา

   - หน่วยประมวลผล ความเร็ว 1.6 กิกะเฮิร์ท อินเทล คอร์ไอ 5
   - หน่วยความจำ 16 กิกะไบต์
   - ฮาร์ดดิสก์ ความจุ 256 กิกะไบต์

2. ซอฟต์แวร์ที่ใช้ในการพัฒนาเครื่องมือ

โปรแกรมสำหรับเครื่องคอมพิวเตอร์

- ระบบปฏิบัติการแมคโอเอส เวอร์ชัน 10.13.3
- เคอร์เนล ดาร์วิน เวอร์ชัน 17.4.0

ภาษาสำหรับพัฒนาเครื่องมือในการสร้างแบบจำลอง

- ภาษาไพทอน เวอร์ชัน 2.7

ไลบรารีสำหรับพัฒนาเครื่องมือในการสร้างแบบจำลอง

- ไลบรารีเอ็นแอลทีเค (NLTK)
- ไลบราลี่แม็ตพล็อตลิบ (matplotlib)
- ไลบราลี่ล็อกกิ้ง (logging)
- ไลบราลี่แพนดาส (pandas)
- ไลบราลี่นัมไพ (numpy)
- ไลบราลี่คีลาส (keras)
- ไลบรารีไซคิดเลิร์น (Scikit-learn)
    - ไลบรารีเนอีฟเบส์
    - ไลบรารีดีซีชันทรี
    - ไลบรารีเอสวีเอ็ม
    - ไลบรารีนิวรอลเน็ตเวิร์ก

บทที่ 4
การทดสอบเครื่องมือ และการอภิปราย

4.1 การประเมินประสิทธิภาพของอัลกอริทึมที่ใช้ในการสร้างแบบจำลอง

การวัดประสิทธิภาพแบบจำลอง จะคำนวณจากค่าความเที่ยง ค่าการระลึกได้ และค่าความแม่นของแบบจำลอง โดยใช้บันทึกทางการแพทย์ของผู้ป่วยมาเป็นชุดข้อมูลทดสอบ เพื่อตรวจสอบความถูกต้องในการทำนายของแบบจำลอง บันทึกทางการแพทย์ที่นำมาใช้จะประกอบด้วยข้อมูลที่เป็นภาษาไทยปนภาษาอังกฤษและอักษรย่อทางการแพทย์ ดังนั้นก่อนนำข้อมูลมาใช้จะต้องนำข้อมูลไปผ่านการประมวลผลข้อมูลก่อนเช่นเดียวกับในขั้นตอนการสร้างแบบจำลอง เพื่อแปลงให้ข้อมูลอยู่ในรูปแบบที่เหมาะสมสำหรับนำมาใช้เป็นชุดข้อมูลทดสอบ

การวัดประสิทธิภาพแบบจำลองจะเริ่มจากการสร้างคอนฟิวชันเมทริกซ์ (confusion matrix) ขนาด 2 x 2 ซึ่งเป็นตารางแบบจัตุรัสที่มีจำนวนแถวเท่ากับจำนวนคอลัมน์และเท่ากับจำนวนประเภท ดังตารางที่ 3 ซึ่งข้อมูลที่อยู่ในคอลัมน์จะเป็นประเภทที่อยู่ในชุดข้อมูลสอน ส่วนข้อมูลที่อยู่ในแถวจะเป็นประเภทที่แบบจำลองทำนายได้

ตารางที่ 3 ตารางแสดงคอนฟิวชันเมทริกซ์ที่ข้อมูลมี 2 ประเภท

	ชื่อโรคที่ถูกต้อง	ชื่อโรคที่ไม่ถูกต้อง
แบบจำลองแสดงชื่อโรค	TP (ผลลัพธ์ที่ถูกต้อง)	FP (ผลลัพธ์ที่เกินคาด)
แบบจำลองไม่แสดงชื่อโรค	FN (ผลลัพธ์ที่หายไป)	TN (ผลลัพธ์ที่ถูกต้อง)

จากตารางที่ 3 ประกอบด้วยค่าต่าง ๆ ดังนี้

- **ผลบวกจริง** (True Positive หรือ TP) คือ กรณีที่ผู้ป่วยเป็นโรคแล้วแบบจำลองทำนายว่าเป็นโรค
- **ผลบวกเท็จ** (False Positive หรือ FP) คือ กรณีที่ผู้ป่วยไม่เป็นโรคแต่แบบจำลองทำนายว่าเป็นโรค
- **ผลลบจริง** (True Negative หรือ TN) คือ กรณีที่ผู้ป่วยไม่เป็นโรคแล้วแบบจำลองทำนายว่าไม่เป็นโรค

- **ผลลบเท็จ** (False Negative หรือ FN) คือ กรณีที่ผู้ป่วยเป็นโรคแต่แบบจำลองทำนายว่าไม่เป็นโรค

ซึ่งแบบจำลองที่มีประสิทธิภาพ ควรมีค่าผลบวกจริงและผลลบจริงสูง ผลบวกเท็จและผลลบเท็จต่ำ

หลังจากสร้างตารางคอนฟิวชันเมทริกซ์แล้วจะสามารถคำนวณค่าพรีซิชั่น ค่ารีคอลล์ ค่าเอฟ และค่าความถูกต้องรวม ได้จากค่าที่อยู่ในตารางคอนฟิวชันเมทริกซ์ ตามสูตรดังนี้

1. **ค่าความเที่ยง** สามารถคำนวณได้จาก

$$\textbf{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}}$$

ค่าความเที่ยงเป็นอัตราส่วนระหว่างจำนวนชื่อโรคที่เกี่ยวข้องและถูกแสดงผลกับจำนวนชื่อโรคที่ถูกแสดงผลทั้งหมด ซึ่งค่านี้จะแสดงให้เห็นว่าแบบจำลองสามารถทำนายชื่อโรคได้อย่างแม่นยำมากน้อยเพียงใด โดยแบบจำลองที่ดีควรมีค่าความเที่ยงที่สูง กล่าวคือแบบจำลองต้องแสดงชื่อโรคที่ไม่เกี่ยวข้องให้น้อยที่สุด

2. **ค่าการระลึกได้** สามารถคำนวณได้จาก

$$\textbf{Recall} = \frac{\text{True Positive}}{\text{True Positive + False Negative}}$$

ค่าการระลึกได้เป็นอัตราส่วนระหว่างจำนวนชื่อโรคที่เกี่ยวข้องและถูกแสดงผลกับจำนวนชื่อโรคที่เกี่ยวข้องทั้งหมด ซึ่งค่านี้จะแสดงให้เห็นว่าแบบจำลองสามารถทำนายชื่อโรคที่ถูกต้องได้มากน้อยเพียงใด โดยแบบจำลองที่ดีควรมีค่าการระลึกได้ที่สูง กล่าวคือทุกชื่อโรคที่แบบจำลองแสดงผลออกมาต้องเป็นชื่อโรคที่เกี่ยวข้อง

3. **ค่าความแม่น** สามารถคำนวณได้จาก

$$\textbf{Accuracy} = \frac{\text{True Positive + True Negative}}{\text{True Positive + True Negative + False Positive + False Negative}}$$

ค่าความแม่นเป็นจำนวนชื่อโรคที่ทำนายถูกต้องทั้งหมด ทั้งในกรณีของชื่อโรคที่เกี่ยวข้องและชื่อโรคที่ไม่เกี่ยวข้อง

เมื่อได้ค่าความเที่ยง ค่าการระลึกได้ และค่าความแม่นของแต่ละชื่อโรคแล้ว จะนำค่าเหล่านี้ของทุกโรคมารวมกันเพื่อหาค่าเฉลี่ย โดยใช้วิธีการหาค่าเฉลี่ยขนาดเล็ก (Micro-average) กำหนดให้ μ แทนค่าเฉลี่ยขนาดเล็ก และ C แทนจำนวนประเภทหรือจำนวนโรค ดังสมการต่อไปนี้

1. **ค่าเฉลี่ยขนาดเล็กของค่าความเที่ยง** สามารถคำนวณได้จาก

$$Precision^{\mu} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|}(TP_i+FP_i)} \quad [22]$$

2. **ค่าเฉลี่ยขนาดเล็กของค่าการระลึกได้** สามารถคำนวณได้จาก

$$Recall^{\mu} = \frac{\sum_{i=1}^{|C|} TP_i}{\sum_{i=1}^{|C|}(TP_i+FN_i)} \quad [22]$$

3. **ค่าเฉลี่ยขนาดเล็กของค่าความแม่น** สามารถคำนวณได้จาก

$$Accuracy^{\mu} = \frac{\sum_{i=1}^{|C|} TP_i+TN_i}{\sum_{i=1}^{|C|}(TP_i+TN_i+FP_i+FN_i)} \quad [22]$$

การวัดประสิทธิภาพในการทำนายของแบบจำลอง จะนำข้อมูลอาการที่อยู่ในบันทึกทางการแพทย์ของผู้ป่วยมาป้อนให้กับแบบจำลอง เพื่อให้แบบจำลองทำการประมวลผลและทำนายชื่อโรคจากข้อมูลอาการที่ป้อนเข้ามา จากนั้นจะนำชื่อโรคที่แบบจำลองทำนายได้มาเทียบกับผลการวินิจฉัยโรคในบันทึกทางการแพทย์ เพื่อคำนวณหาค่าความเที่ยง ค่าการระลึกได้ และค่าความแม่นของแบบจำลอง

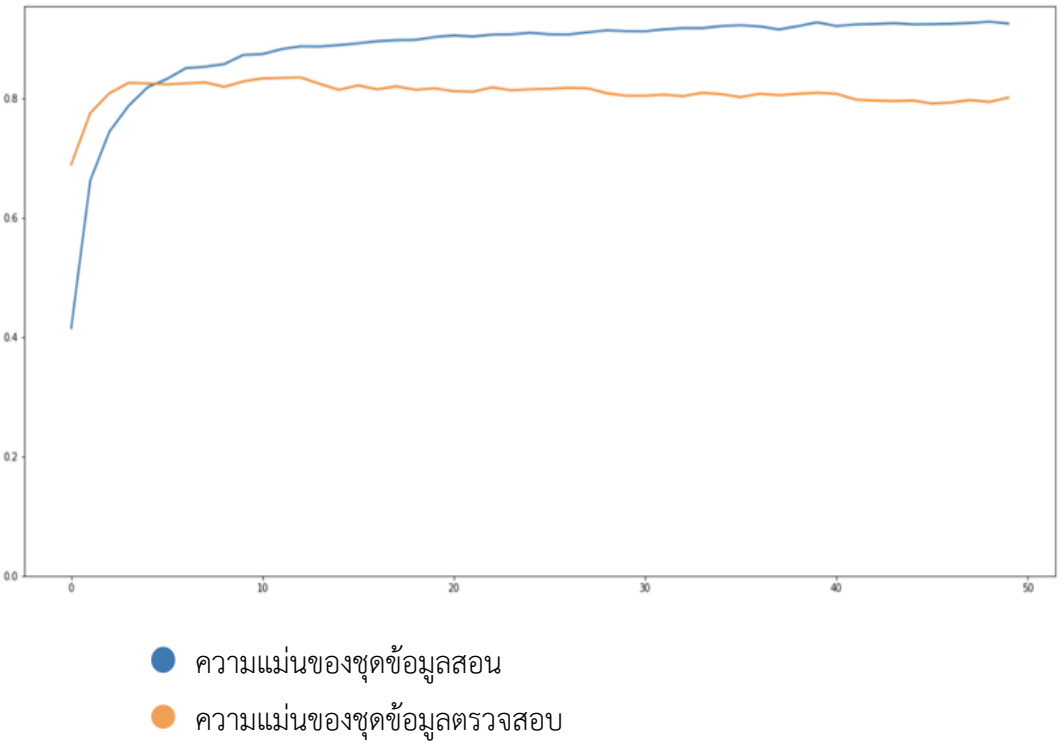
4.2 ผลการประเมินประสิทธิภาพของอัลกอริทึมที่ใช้ในการสร้างแบบจำลอง

ในงานวิจัยนี้ได้ทดลองสร้างแบบจำลองโดยใช้การเรียนรู้เชิงลึกร่วมกับหลาย ๆ เทคนิค เช่น โกรฟ เวิร์ดทูเวก รวมถึงใช้การเรียนรู้เชิงลึกเพียงอย่างเดียวในการสร้างแบบจำลอง เพื่อนำผลลัพธ์ที่ได้จากแต่ละแบบจำลองมาเปรียบเทียบกัน

4.2.1 แบบจำลองการเรียนรู้เชิงลึกที่ไม่ได้ใช้เทคนิคการฝังคำ

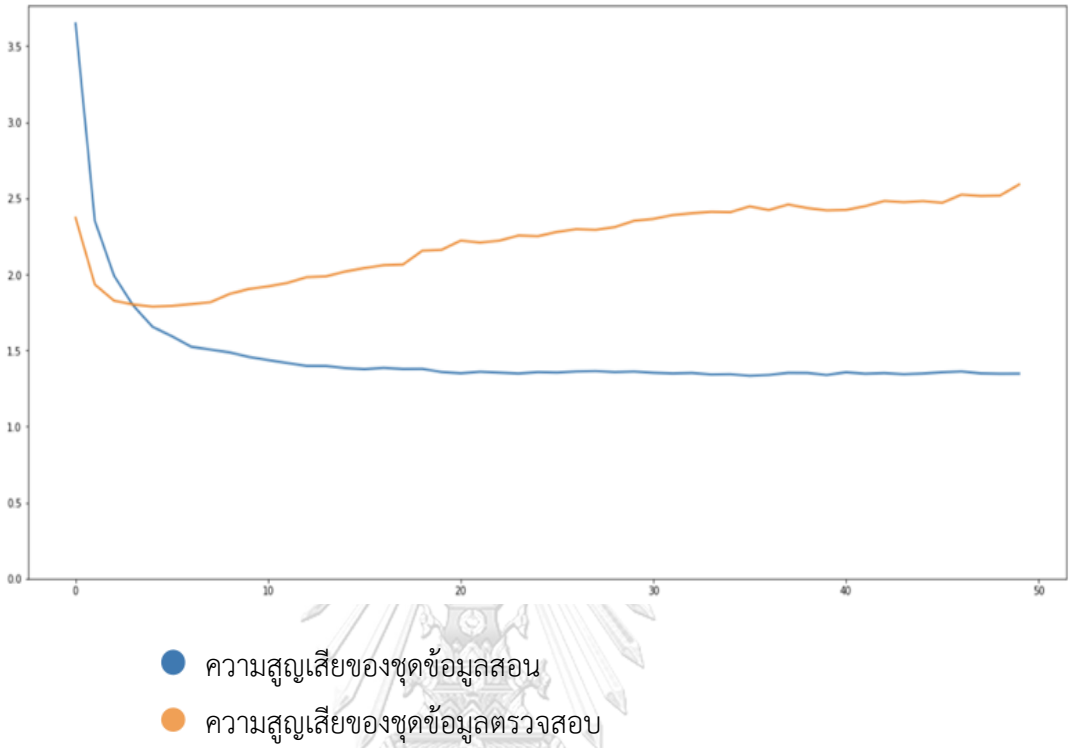
ผลลัพธ์ของแบบจำลองที่สร้างจากการใช้เทคนิคการเรียนรู้เชิงลึกเพียงอย่างเดียว สามารถแสดงได้ดังกราฟที่ 1 และกราฟที่ 2

กราฟที่ 1 กราฟแสดงค่าความแม่นของชุดข้อมูลสอนและชุดข้อมูลตรวจสอบ (Validating data) ในแบบจำลองการเรียนรู้เชิงลึกที่ไม่ได้ใช้เทคนิคการฝังคำ



จากกราฟที่ 1 จะเห็นว่าค่าความแม่นที่ได้จากชุดข้อมูลสอนและชุดข้อมูลตรวจสอบมีแนวโน้มสูงขึ้นเรื่อย ๆ จนถึงในช่วงหลัง ๆ ค่าความแม่นของชุดข้อมูลสอนเริ่มมีค่าคงที่หรือมีค่าเกือบเท่ากับ 1 ทำให้งานวิจัยนี้พบว่าจำนวนครั้งในการสอนแบบจำลองที่เหมาะสม คือ 50 รอบ

กราฟที่ 2 กราฟแสดงค่าความสูญเสียของชุดข้อมูลสอนและชุดข้อมูลตรวจสอบในแบบจำลองการเรียนรู้เชิงลึกที่ไม่ได้ใช้เทคนิคการฝังคำ

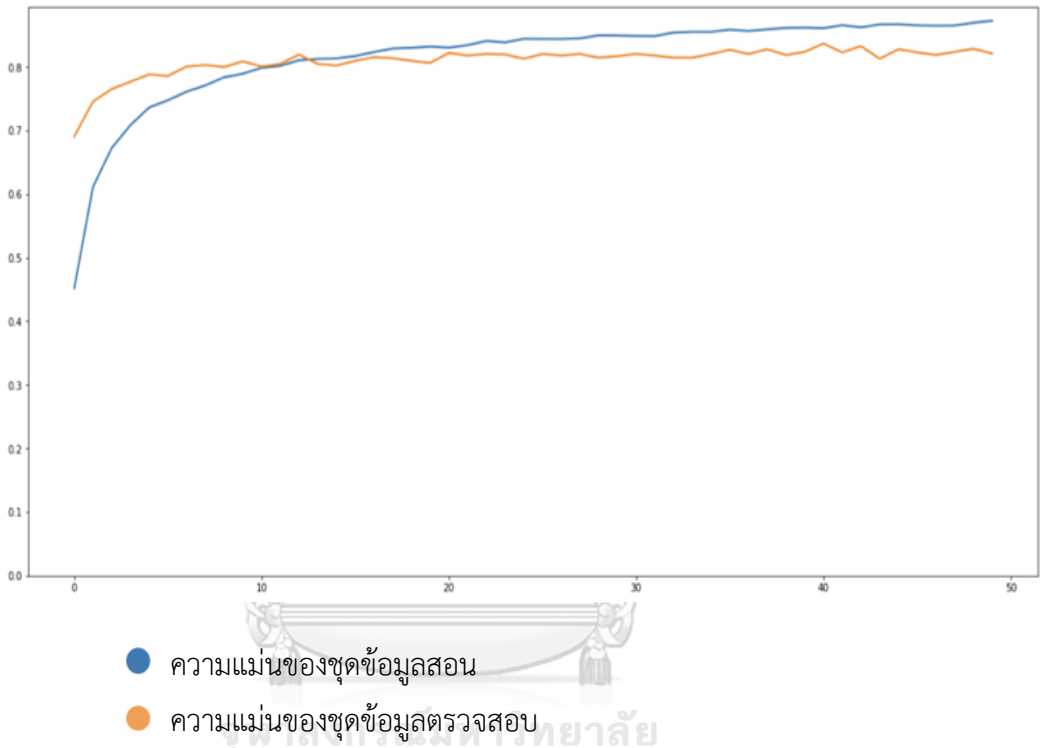


จากกราฟที่ 2 จะเห็นว่าค่าความสูญเสียที่ได้จากชุดข้อมูลสอนมีแนวโน้มลดลงเรื่อย ๆ จนถึงในช่วงหลัง ๆ ค่าความสูญเสียของชุดข้อมูลสอนเริ่มมีค่าคงที่ ในขณะที่ค่าความสูญเสียที่ได้จากชุดข้อมูลตรวจสอบมีแนวโน้มที่จะเพิ่มขึ้นเรื่อย ๆ ทำให้งานวิจัยนี้พบว่าจำนวนครั้งในการสอนแบบจำลองที่เหมาะสม คือ 50 รอบ

4.2.2 แบบจำลองการเรียนรู้เชิงลึกจากการใช้ค่าน้ำหนัก 100 มิติของโกรฟ

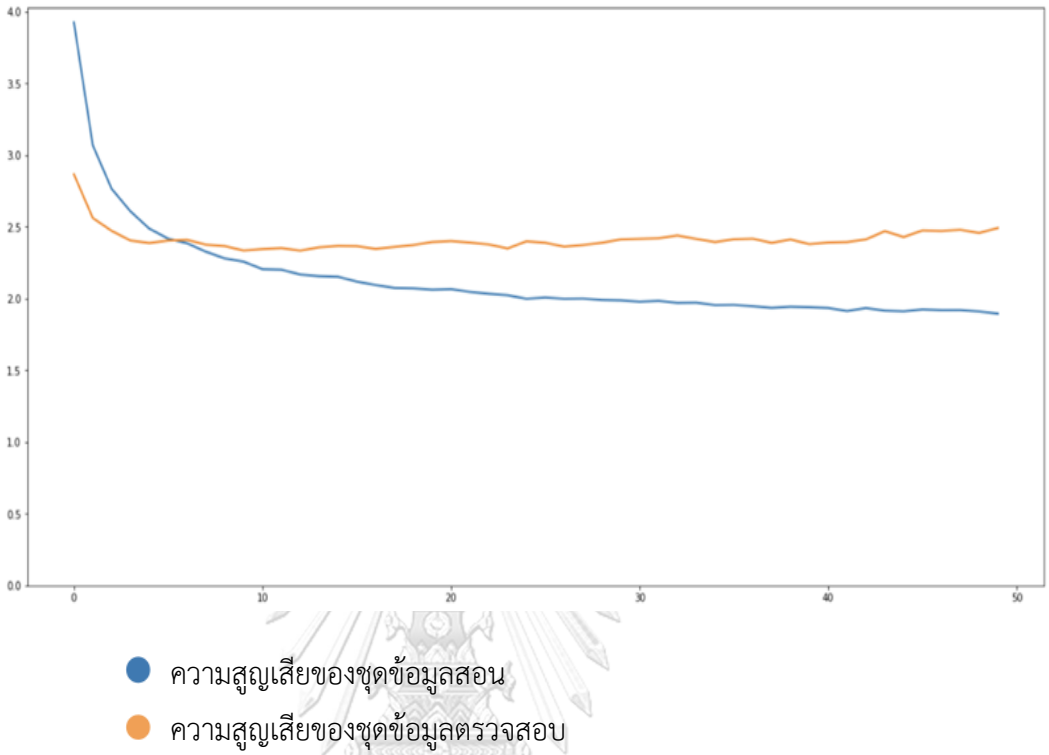
ผลลัพธ์ของแบบจำลองที่ได้จากการใช้การเรียนรู้เชิงลึกร่วมกับการใช้ค่าน้ำหนัก 100 มิติของโกรฟ สามารถแสดงได้ดังกราฟที่ 3 และกราฟที่ 4

กราฟที่ 3 กราฟแสดงค่าความแม่นของชุดข้อมูลสอนและชุดข้อมูลตรวจสอบในแบบจำลองการเรียนรู้เชิงลึกจากการใช้ค่าน้ำหนัก 100 มิติของโกรฟ



จากกราฟที่ 3 จะเห็นว่าค่าความแม่นที่ได้จากชุดข้อมูลสอนและชุดข้อมูลตรวจสอบมีแนวโน้มสูงขึ้นเรื่อย ๆ จนถึงในช่วงหลัง ๆ ค่าความแม่นของชุดข้อมูลสอนเริ่มมีค่าคงที่หรือมีค่าเกือบเท่ากับ 1 ทำให้งานวิจัยนี้พบว่าจำนวนครั้งในการสอนแบบจำลองที่เหมาะสม คือ 50 รอบ

กราฟที่ 4 กราฟแสดงค่าความสูญเสียของชุดข้อมูลสอนและชุดข้อมูลตรวจสอบในแบบจำลองการเรียนรู้เชิงลึกจากการใช้ค่าน้ำหนัก 100 มิติของโกรฟ

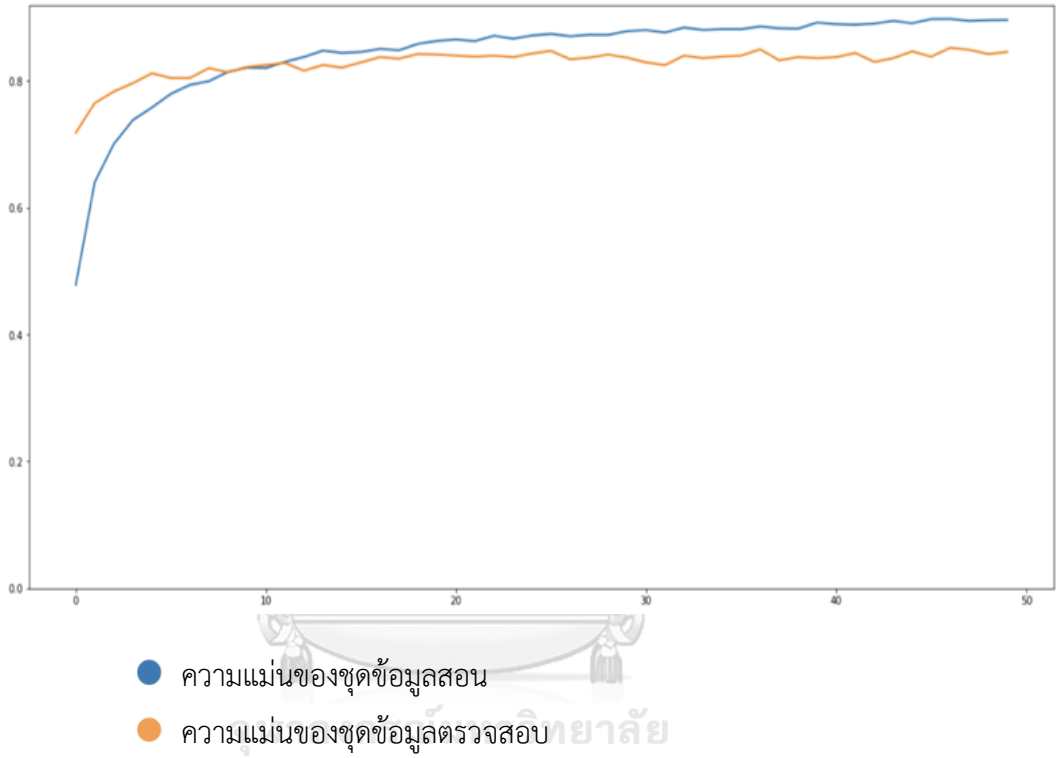


จากกราฟที่ 4 จะเห็นว่าค่าความสูญเสียที่ได้จากชุดข้อมูลสอนมีแนวโน้มลดลงเรื่อย ๆ จนถึงในช่วงหลัง ๆ ค่าความสูญเสียของชุดข้อมูลสอนเริ่มมีค่าคงที่ ในขณะที่ค่าความสูญเสียที่ได้จากชุดข้อมูลตรวจสอบมีแนวโน้มที่จะเพิ่มขึ้นเรื่อย ๆ ทำให้งานวิจัยนี้พบว่าจำนวนครั้งในการสอนแบบจำลองที่เหมาะสม คือ 50 รอบ

4.2.3 แบบจำลองการเรียนรู้เชิงลึกจากการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวก

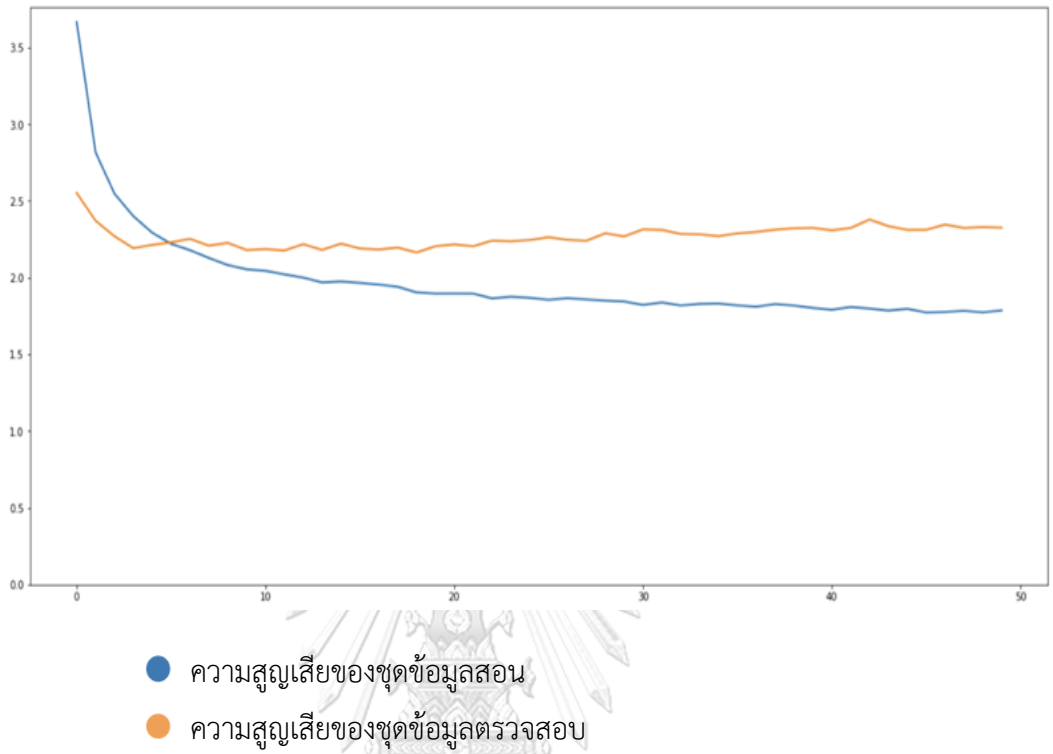
ผลลัพธ์ของแบบจำลองที่ได้จากการใช้การเรียนรู้เชิงลึกร่วมกับการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวก สามารถแสดงได้ดังกราฟที่ 5 และกราฟที่ 6

กราฟที่ 5 กราฟแสดงค่าความแม่นของชุดข้อมูลสอนและชุดข้อมูลตรวจสอบในแบบจำลองการเรียนรู้เชิงลึกจากการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวก



จากกราฟที่ 5 จะเห็นว่าค่าความแม่นที่ได้จากชุดข้อมูลสอนและชุดข้อมูลตรวจสอบมีแนวโน้มสูงขึ้นเรื่อย ๆ จนถึงในช่วงหลัง ๆ ค่าความแม่นของชุดข้อมูลสอนเริ่มมีค่าคงที่หรือมีค่าเกือบเท่ากับ 1 ทำให้งานวิจัยนี้พบว่าจำนวนครั้งในการสอนแบบจำลองที่เหมาะสม คือ 50 รอบ

กราฟที่ 6 กราฟแสดงค่าความสูญเสียของชุดข้อมูลสอนและชุดข้อมูลตรวจสอบในแบบจำลองการเรียนรู้เชิงลึกจากการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวก

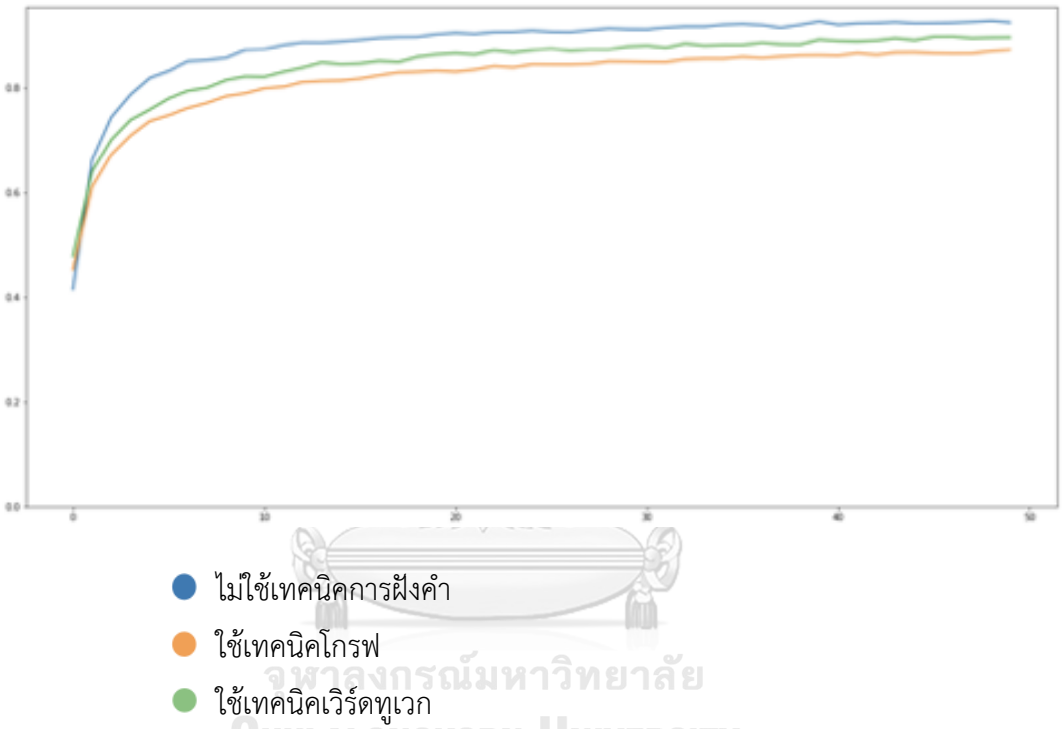


จากกราฟที่ 6 จะเห็นว่าค่าความสูญเสียที่ได้จากชุดข้อมูลสอนมีแนวโน้มลดลงเรื่อย ๆ จนถึงในช่วงหลัง ๆ ค่าความสูญเสียของชุดข้อมูลสอนเริ่มมีค่าคงที่ ในขณะที่ค่าความสูญเสียที่ได้จากชุดข้อมูลตรวจสอบมีแนวโน้มที่จะเพิ่มขึ้นเรื่อย ๆ ทำให้งานวิจัยนี้พบว่าจำนวนครั้งในการสอนแบบจำลองที่เหมาะสม คือ 50 รอบ

4.3 ผลการเปรียบเทียบประสิทธิภาพของอัลกอริทึมที่ใช้ในการสร้างแบบจำลอง

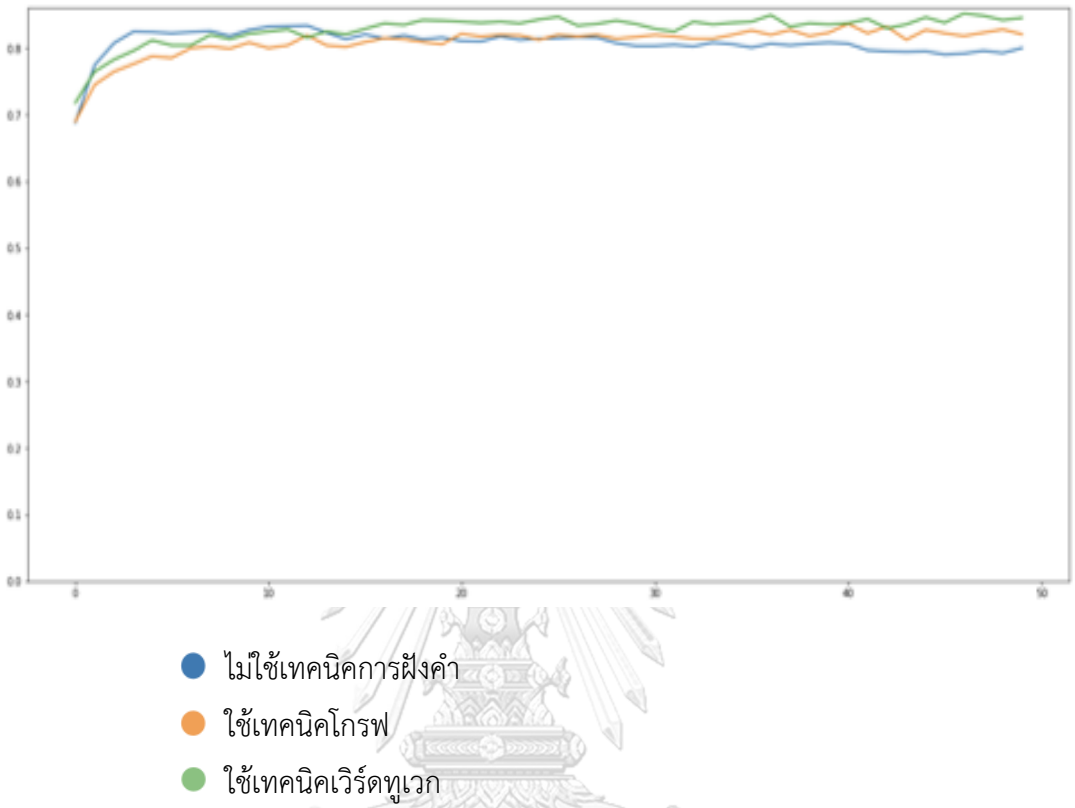
แบบจำลองที่ได้ในงานวิจัยนี้จะมีทั้งหมด 3 แบบ ได้แก่ แบบจำลองที่ได้จากการใช้การเรียนรู้เชิงลึก แบบจำลองที่ได้จากการใช้ค่าน้ำหนัก 100 มิติของโกรฟ และแบบจำลองที่ได้จากการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวก โดยเมื่อนำผลลัพธ์ที่ได้จากทั้ง 3 แบบจำลองมาเปรียบเทียบกัน จะได้ผลลัพธ์ดังกราฟที่ 7 8 9 และ 10

กราฟที่ 7 กราฟแสดงผลการเปรียบเทียบค่าความแม่นของชุดข้อมูลสอนของแบบจำลองทั้ง 3 แบบ



จากกราฟที่ 7 จะเห็นว่าเส้นโค้งของแบบจำลองที่ได้จากการใช้การเรียนรู้เชิงลึกเพียงอย่างเดียวให้ค่าความแม่นมากที่สุด เนื่องจากเป็นเส้นโค้งที่มีความชันและโค้งเข้าใกล้มุมบนซ้ายมากที่สุด

กราฟที่ 8 กราฟแสดงผลการเปรียบเทียบค่าความแม่นของชุดข้อมูลตรวจสอบของแบบจำลองทั้ง 3 แบบ

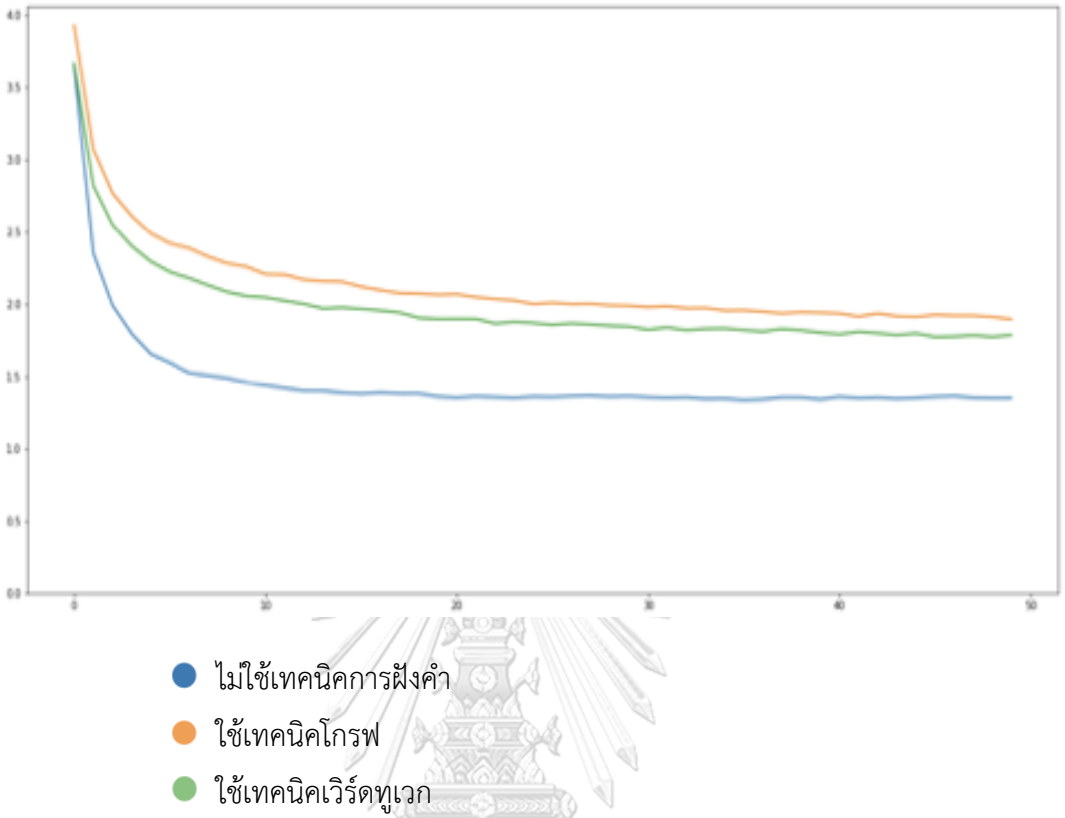


จากกราฟที่ 8 จะเห็นว่าเส้นโค้งของแบบจำลองที่ได้จากการใช้การเรียนรู้เชิงลึกร่วมกับการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวกให้ค่าความแม่นของชุดข้อมูลตรวจสอบมากที่สุด ด้วยการสอนแบบจำลอง 50 ครั้ง เนื่องจากเป็นเส้นโค้งที่เข้าใกล้ค่า 1 หรือ 100% มากที่สุด ดังเห็นได้จากค่าความแม่นในตารางที่ 4 ดังนั้นในงานวิจัยนี้จึงเลือกใช้วิธีการสร้างแบบจำลองโดยใช้การเรียนรู้เชิงลึกร่วมกับการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวก

ตารางที่ 4 ตารางแสดงค่าความแม่นของชุดข้อมูลตรวจสอบของแบบจำลองทั้ง 3 แบบ

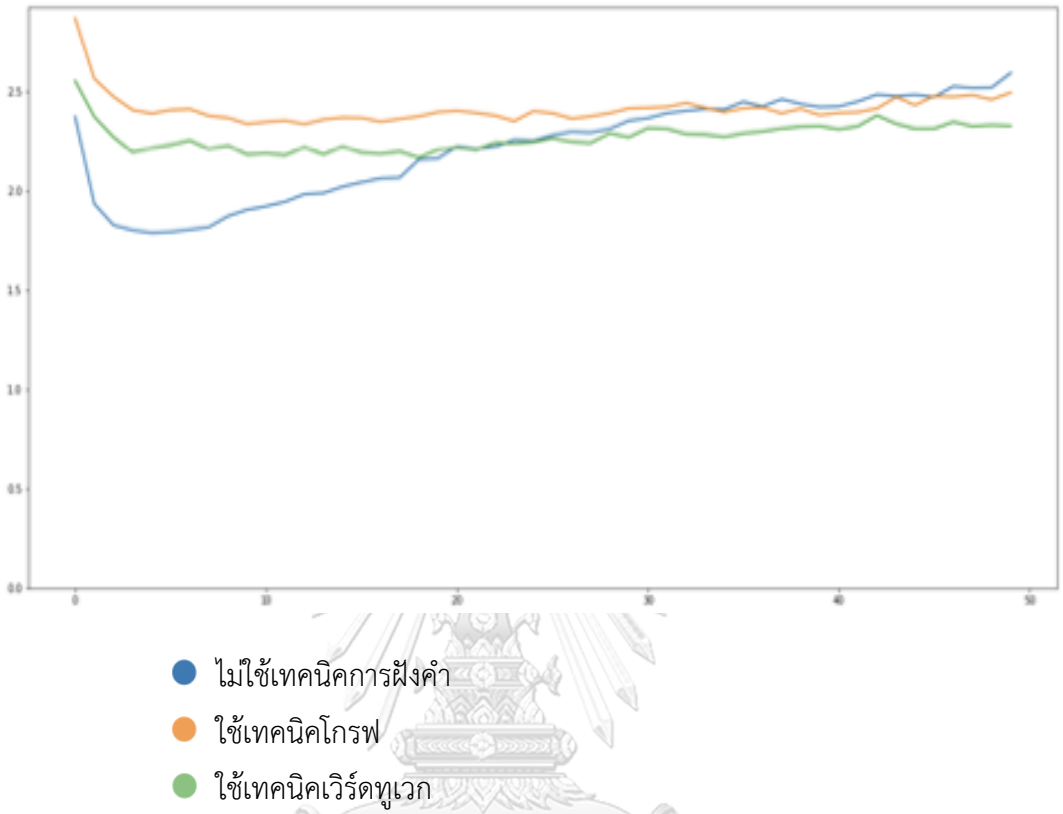
การสร้างแบบจำลองการเรียนรู้เชิงลึก	**ค่าความแม่น**
ไม่ใช้เทคนิคการฝังคำ	81.67%
ใช้เทคนิคโกรฟ	83.65%
ใช้เทคนิคเวิร์ดทูเวก	86.64%

กราฟที่ 9 กราฟแสดงผลการเปรียบเทียบค่าความสูญเสียของชุดข้อมูลสอนของแบบจำลองทั้ง 3 แบบ



จากกราฟที่ 9 จะเห็นว่าเส้นโค้งของแบบจำลองที่ได้จากการใช้การเรียนรู้เชิงลึกเพียงอย่างเดียวให้ค่าความสูญเสียน้อยที่สุด เนื่องจากเป็นเส้นโค้งที่โค้งเข้าหามุมล่างซ้ายมากที่สุดหรือมีค่าเข้าใกล้ 0 มากที่สุด

กราฟที่ 10 กราฟแสดงผลการเปรียบเทียบค่าความสูญเสียของชุดข้อมูลตรวจสอบของแบบจำลองทั้ง 3 แบบ



จากกราฟที่ 10 จะเห็นว่าเส้นโค้งของแบบจำลองที่ได้จากการใช้การเรียนรู้เชิงลึกร่วมกับการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวกให้ค่าความสูญเสียน้อยที่สุด ด้วยการสอนแบบจำลอง 50 ครั้ง เนื่องจากเป็นเส้นโค้งมีค่าเข้าใกล้ 0 มากที่สุด ดังนั้นในงานวิจัยนี้จึงเลือกใช้วิธีการสร้างแบบจำลองโดยใช้การเรียนรู้เชิงลึกร่วมกับการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวก

4.4 ผลการเปรียบเทียบประสิทธิภาพของแบบจำลองในงานวิจัยนี้กับงานวิจัยอื่นที่เกี่ยวข้อง

แบบจำลองในงานวิจัยนี้สร้างจากการใช้การเรียนรู้เชิงลึกร่วมกับการใช้ค่าน้ำหนัก 100 มิติของเวิร์ดทูเวกซึ่งเมื่อนำผลลัพธ์ที่ได้จากแบบจำลองไปเปรียบเทียบกับแบบจำลองในงานวิจัยที่เกี่ยวข้อง พบว่าแบบจำลองในงานวิจัยนี้ให้ค่าอัตราผลบวกเท็จที่น้อยกว่า ค่าความเที่ยงที่มากกว่า และค่าความแม่นที่มากกว่า ดังตารางที่ 5 ด้วยการแสดงผลการทำนายหรือผลการวินิจฉัยโรคเพียงอันดับแรกอันดับเดียว

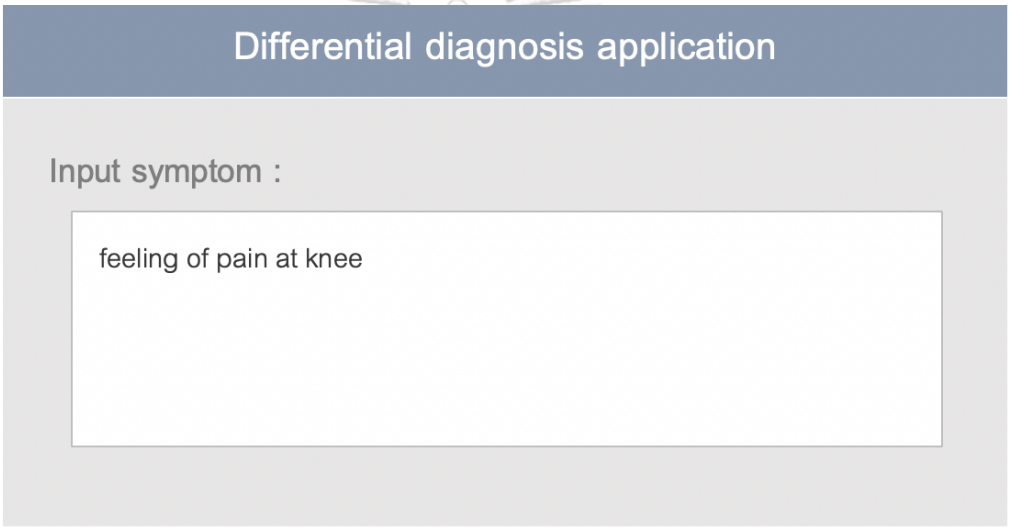
ตารางที่ 5 ตารางแสดงผลการเปรียบเทียบผลลัพธ์ที่ได้จากแบบจำลองในงานวิจัยนี้กับงานวิจัยอื่น

	งานวิจัยเรื่อง “การประยุกต์ใช้การทำเหมืองข้อความเพื่อจำแนกประเภทโรคจากอาการ”				งานวิจัยนี้
	ต้นไม้ตัดสินใจ (1 อันดับ)	การเรียนรู้เบย์ (10 อันดับ)	ซัพพอร์ตเวกเตอร์แมชชีน (6 อันดับ)	โครงข่ายประสาทเทียม (7 อันดับ)	การเรียนรู้เชิงลึก (1 อันดับ)
อัตราผลบวกจริง	50.87%	72.33%	86.63%	**89.03%**	86.64%
อัตราผลบวกเท็จ	1.64%	7.06%	14.51%	11.08%	**0.02%**
ความเที่ยง	50.30%	37.17%	24.21%	27.11%	**68.10%**
ความแม่น	97.13%	92.86%	86.45%	89.50%	**99.95%**

4.5 ตัวอย่างผลลัพธ์ที่ได้จากการทดสอบแบบจำลอง

การใช้งานแบบจำลอง เริ่มต้นจากการป้อนข้อมูลอาการให้กับแบบจำลอง จากนั้นแบบจำลองจะนำข้อมูลอาการที่ได้ไปทำการประมวลผล เพื่อหาผลการวินิจฉัยโรคที่ความน่าจะเป็นมากที่สุดจากข้อมูลอาการที่ได้รับ

รูปที่ 12 เป็นตัวอย่างการใช้งานแบบจำลอง โดยการใส่ “feeling of pain at knee” เป็นข้อมูลนำเข้าให้กับแบบจำลอง หลังจากนั้นแบบจำลองจะแตกข้อความออกเป็นคำย่อย ๆ หรือโทเคน ดังนี้ “feeling”, “of”, “pain”, “at”, “knee” ซึ่งหลังจากที่แปลงคำเป็นโทเคนแล้ว แบบจำลองจะทำการแปลงโทเคนเหล่านั้นให้เป็นตัวเลขทางคณิตศาสตร์โดยใช้เทคนิคความถี่ของคำแบบย้อนกลับ (Term Frequency Inverse Document Frequency) ซึ่งจะได้ผลลัพธ์ออกมาดังรูปที่ 13

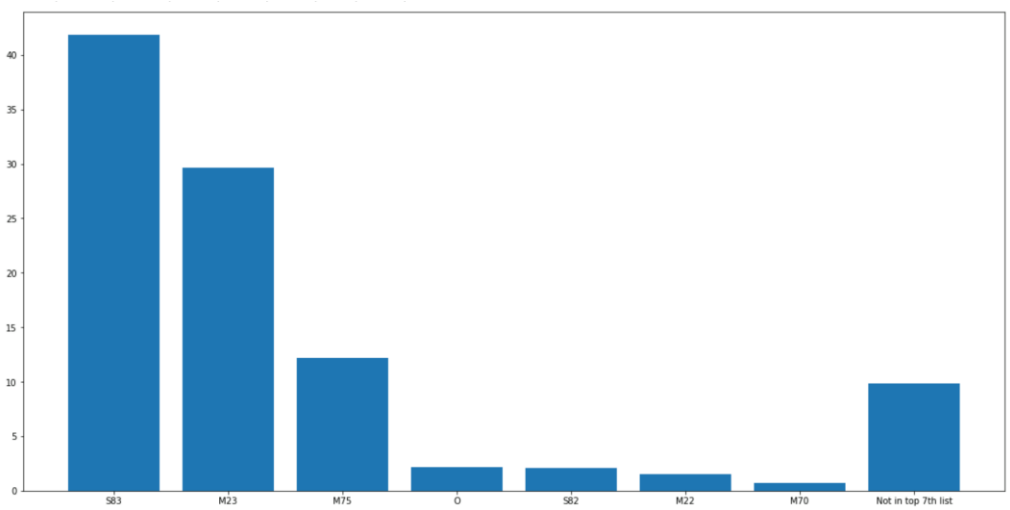


รูปที่ 12 รูปแสดงส่วนต่อประสานกับผู้ใช้ของแบบจำลองในงานวิจัยนี้

```
[[0.         0.         0.         0.96699072 1.3939983  0.
  0.         1.16057414 0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.
```

รูปที่ 13 รูปแสดงเวกเตอร์ของประโยค “feeling of pain at knee”

จากนั้นแบบจำลองจะนำข้อมูลมาประมวลผลและแสดงผลลัพธ์ดังรูปที่ 14 โดยผลการวินิจฉัยโรคที่แบบจำลองแสดง จะเรียงตามลำดับความน่าจะเป็นจากมากไปน้อย ในรูปแบบของกราฟแท่ง และเปอร์เซ็นต์ ซึ่งจากรูปที่ 14 เห็นว่าผลวินิจฉัยโรคที่มีความน่าจะเป็นมากที่สุดสำหรับข้อมูลนำเข้านี้คือ S83: Sprain and strain involving cruciate ligament of knee ด้วยความน่าจะเป็น 41.83%



Sprain and strain involving cruciate ligament of knee
S83 41.83%
Derangement of meniscus due to old tear or injury
M23 29.61%
Adhesive capsulitis of shoulder
M75 12.16%
Other diseases
O 2.17%
Fracture of upper end of tibia
S82 2.12%
Recurrent subluxation of patella
M22 1.55%
Chronic crepitant synovitis of hand and wrist Forearm
M70 0.7%

รูปที่ 14 รูปแสดงผลลัพธ์ที่ได้จากแบบจำลองในงานวิจัยนี้

บทที่ 5
บทสรุป

5.1 สรุปผลวิทยานิพนธ์

งานวิจัยนี้มีเป้าหมายที่จะช่วยแพทย์ในการวินิจฉัยโรคให้มีความแม่นยำและรวดเร็วยิ่งขึ้น จึงได้ทำการศึกษาค้นคว้า และพัฒนาแบบจำลองสำหรับทำนายผลการวินิจฉัยโรคขึ้น จากการใช้ข้อมูลเวชระเบียนผู้ป่วยในแผนกออร์โธปิดิกส์ของโรงพยาบาลจุฬาลงกรณ์มาเป็นชุดข้อมูลสำหรับสร้างแบบจำลอง โดยแบบจำลองในงานวิจัยนี้จะแสดงผลลัพธ์เป็นชื่อโรคพร้อมทั้งรหัสไอซีดีเทน ของโรค เพื่อช่วยแพทย์ให้สามารถวินิจฉัยโรคได้ง่ายขึ้น และช่วยแพทย์ในการจำแนกรหัสไอซีดีเทน ซึ่งในงานวิจัยนี้ได้นำเอาเทคนิคการฝังคำที่อยู่ในการเรียนรู้เชิงลึกมาใช้ และพบว่าทำให้ผลลัพธ์ที่ได้จากแบบจำลองมีประสิทธิภาพและมีความแม่นยำมากขึ้น

5.2 ปัญหาและข้อจำกัดในการทำวิทยานิพนธ์

1. แบบจำลองในงานวิจัยนี้สร้างมาจากข้อมูลเวชระเบียนผู้ป่วยในแผนกออร์โธปิดิกส์ของโรงพยาบาลจุฬาลงกรณ์
2. แบบจำลองสามารถทำนายผลการวินิจฉัยโรคในรูปแบบของรหัสไอซีดีเทนได้มากสุดเพียง 3 หลักเท่านั้น
3. ข้อมูลที่นำมาใช้ในงานวิจัยนี้เป็นข้อมูลภาษาอังกฤษเท่านั้น
4. แบบจำลองจำกัดกลุ่มผู้ใช้เป็นกลุ่มแพทย์เท่านั้น

5.3 แนวทางในการปรับปรุงวิทยานิพนธ์

งานวิจัยนี้เป็นการสร้างแบบจำลองสำหรับทำนายผลการวินิจฉัยโรคที่พบในแผนกออร์โธปิดิกส์เท่านั้น ดังนั้นในอนาคตอาจมีการนำแนวคิดที่ได้จากงานวิจัยนี้ไปประยุกต์ใช้กับข้อมูลผู้ป่วยในแผนกอื่น ๆ เพื่อช่วยแพทย์ในการวินิจฉัยโรค ซึ่งการนำแนวคิดในการสร้างแบบจำลองนี้ไปประยุกต์ใช้อาจต้องมีการปรับค่าตัวแปรต่าง ๆ เพื่อให้เหมาะสมกับชุดข้อมูลที่จะนำไปใช้ต่อไป

นอกจากนี้ในส่วนของฉลากประเภทโรคที่เป็นประเภทอื่น ๆ ในงานวิจัยนี้ อาจมีการนำเทคนิคอื่น ๆ มาประยุกต์ใช้ เพื่อให้แบบจำลองสามารถทำนายประเภทโรคได้ถูกต้องแม่นยำยิ่งขึ้น เช่น เทคนิค เค-เพื่อนบ้านที่ใกล้ที่สุด (K-NN: K-Nearest Neighbors), เทคนิคกฎพื้นฐาน (base rule) เป็นต้น

บรรณานุกรม

1. สำนักงานพัฒนาธุรกรรมทางอิเล็กทรอนิกส์. บทความรหัส *ICD*-10 ที่เป็นมาตรฐานคืออะไร. [cited 2017 15 June]; Available from: https://www.etda.or.th/content/1231.html.
2. World Heatlh Organization. *Classification of Diseases (ICD).* [cited 2017 15 June]; Available from: http://apps.who.int/classifications/icd10/browse/2010/en.
3. Jiawei Han, Micheline Kamber, and Jian Pei, *Data Mining: Concepts and Techniques.* The Morgan Kaufmann series in data management systems, 2012.
4. Pereira, L., et al., *ICD9-based Text Mining Approach to Children Epilepsy Classification.* Procedia Technology, 2013. 9: p. 1351-1360.
5. Fabian Pedregosa, et al., *Scikit-learn: Machine Learning in Python.* Journal of Machine Learning Research, 2011. 12: p. 2825-2830.
6. Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, *Neural Machine Translation by Jointly Learning to Align and Translate.* 2014.
7. Shuangfei Zhai, et al., *DeepIntent: Learning Attentions for Online Advertising with Recurrent Neural Networks.* 2016.
8. HO CHUNG WU, et al., *Interpreting TF-IDF Term Weights as Making Relevance Decisions.* ACM Transactions on Information Systems, 2008. 26.
9. Cavnar, W.B. and J.M. Trenkle. *N-gram-based text categorization*. in *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*. 1994. Citeseer.
10. Combrinck, H.P. and E. Botha. *Text-based automatic language identification*. in *Proceedings of the 6th Annual Symposium of the Pattern Recognition Association of South Africa*. 1995.
11. Peng, F. and D. Schuurmans. *Combining naive Bayes and n-gram language models for text classification*. in *European Conference on Information Retrieval*. 2003. Springer.
12. Schmidhuber, J., *Deep learning in neural networks: An overview.* Neural Networks, 2015. 61: p. 85-117.
13. Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, *Deep learning.* Nature, 2015.

521: p. 436–444.

14. James, G., Witten, D., Hastie, T., Tibshirani, R., *An Introduction to Statistical Learning: with Applications in R.* 2014.
15. Kannan, R. and V. Vasanthi, *Machine Learning Algorithms with ROC Curve for Predicting and Diagnosing the Heart Disease*. 2019. p. 63-72.
16. Ketpupong, P. and K. Piromsopa, *Applying Text Mining for Classifying Disease from Symptoms*. 2018. 467-472.
17. Huh, J., M. Yetisgen-Yildiz, and W. Pratt, *Text classification for assisting moderators in online health communities.* Journal of Biomedical Informatics, 2013. 46(6): p. 998-1005.
18. Mikolov, T., et al., *Efficient estimation of word representations in vector space.* 2013.
19. Wang, P., et al., *Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification.* 2016. 174: p. 806-814.
20. Tang, D., et al. *Learning sentiment-specific word embedding for twitter sentiment classification*. in *Proceedings of the* 52*nd Annual Meeting of the Association for Computational Linguistics (Volume* 1: *Long Papers)*. 2014.
21. Bengio, Y., et al., *A neural probabilistic language model %J J. Mach. Learn. Res.* 2003. 3: p. 1137-1155.
22. Fabrizio Sebastiani, *Machine Learning in Automated Text Categorization.* ACM Computing Surveys, 2002. 34(1): p. 1-47.

ภาคผนวก

ภาคผนวก ก

รหัสไอซีดีเทนซีเอ็มทั้งหมดที่นำมาใช้ในงานวิจัย

ลำดับ	หมวดหมู่โรค	รหัสไอซีดีเทนซีเอ็ม
1	Certain infectious and parasitic diseases: A00-B99	A17, A18, A46, A52, A80, B18, B20, B37, B47, B91
2	Neoplasms: C00-D49	C15, C16, C18, C20, C22, C24, C34, C40, C41, C43, C44, C47, C48, C49, C50, C53, C61, C70, C73, C75, C76, C77, C78, C79, C82, C83, C84, C85, C90, C91, C96, D16, D17, D18, D21, D23, D35, D36, D48
3	Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism: D50-D89	D61, D64, D66, D68, D75, D76
4	Endocrine, nutritional and metabolic diseases: E00-E89	E11, E22, E46, E64, E75, E78, E83, E87
5	Mental, Behavioral and Neurodevelopmental disorders: F01-F99	F01, F80
6	Diseases of the nervous system: G00-G99	G03, G04, G06, G20, G31, G35, G47, G52, G54, G56, G57, G58, G62, G71, G80, G81,

ลำดับ	หมวดหมู่โรค	รหัสไอซีดีเทนซีเอ็ม
		G82, G83, G95, G96, G97, G98
7	Diseases of the eye and adnexa: H00-H59	H16, H26
8	Diseases of the ear and mastoid process: H60-H95	H81, H91
9	Diseases of the circulatory system: I00-I99	I05, I07, I10, I20, I21, I25, I33, I48, I50, I62, I63, I69, I70, I72, I74, I77, I80, I87, I89, I97
10	Diseases of the respiratory system: J00-J99	J18, J45, J96
11	Diseases of the digestive system: K00-K95	K22, K56, K70, K80, K81
12	Diseases of the skin and subcutaneous tissue: L00-L99	L02, L03, L04, L08, L50, L59, L60, L72, L81, L89, L90, L91, L92, L97, L98
13	Diseases of the musculoskeletal system and connective tissue: M00-M99	M00, M01, M02, M04, M05, M06, M07, M08, M10, M11, M12, M13, M14, M1A, M15, M16, M17, M18, M19, M20, M21, M22, M23, M24, M25, M26, M27,

ลำดับ	หมวดหมู่โรค	รหัสไอซีดีเทนซีเอ็ม
		M30, M31, M32, M33, M34, M35, M36, M40, M41, M42, M43, M45, M46, M47, M48, M49, M50, M51, M53, M54, M60, M61, M62, M63, M65, M66, M67, M70, M71, M72, M75, M76, M77, M79, M80, M81, M83, M84, M85, M86, M87, M88, M89, M90, M91, M92, M93, M94, M95, M96, M97, M99
14	Diseases of the genitourinary system: N00-N99	N18, N19, N31, N32
15	Congenital malformations, deformations and chromosomal abnormalities: Q00-Q99	Q06, Q27, Q28, Q65, Q66, Q67, Q68, Q69, Q70, Q71, Q72, Q74, Q76, Q77, Q78, Q79, Q82, Q87
16	Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified: R00-R99	R19, R22, R26, R60, R94

ลำดับ	หมวดหมู่โรค	รหัสไอซีดีเทนซีเอ็ม
17	Injury, poisoning and certain other consequences of external causes: S00-T88	S00, S02, S04, S06, S09, S12, S13, S14, S20, S22, S24, S25, S27, S29, S30, S32, S33, S34, S35, S36, S37, S38, S39, S40, S41, S42, S43, S44, S45, S46, S48, S49, S51, S52, S53, S54, S55, S56, S58, S60, S61, S62, S63, S64, S65, S66, S67, S68, S69, S70, S72, S73, S76, S79, S80, S81, S82, S83, S84, S85, S86, S87, S88, S89, S91, S92, S93, S96, S97, S98, S99, T14, T23, T63, T79, T81, T82, T84, T85, T86, T87, T88
18	Factors influencing health status and contact with health services: Z00-Z99	Z01, Z03, Z09, Z42, Z47, Z48, Z73, Z88, Z95, Z96
	รวม	332

ภาคผนวก ข
ตัวอย่างข้อมูลบนเวชระเบียนและรหัสไอซีดีเทนซีเอ็มที่นำมาใช้ในงานวิจัย

ลำดับ	บันทึกของแพทย์	รหัสไอซีดีเทนซีเอ็ม
1	7-8 ปีก่อน เดินได้ รองเท้าส้นสูงเท้าพลิก หลังจากนั้นปวดเข่าซ้ายมาก เดินไม่ได้ประมาณ 1 wk. 5-6 ปีก่อน ปวดมากขึ้น pain when activity	M23
2	ผู้ป่วยปวดหลังร้าวลงขาซ้าย 5 ปี ปวดชณะเดิน ปวดที่หลังมากกว่าขา ยืนนานจะปวด เดินกระเผก ปัสสาวะ ถ่ายอุจจาระปกติ มีโรคประจำตัวเป็นความดันโลหิตสูง	M41
3	case ผู้ป่วยเด็กหญิงไทยอายุ12ปี CC: หลังคด 1 ปี PI: แม่สังเกตว่าไหล่ไม่เท่ากัน เดินได้ปกติ ไม่ปวด FC-I มีประจำเดือนตั้งแต่อายุ 11 ปี PH: ปฏิเสธประวัติแพ้ยา	M41
4	ปวดหลัง 3 ปี ยกของแล้วมีเสียงลั่นที่หลัง แล้วมีปวดหลังร้าวถึงปลายเท้าทั้งสองข้าง เดินได้ 10 เมตร จะมีอาการปวดร้าวลงปลายเท้า ขวามากกว่าซ้าย นั่งพักแล้วดีขึ้น	M43
5	ปวดหลัง 10 ปี เตะฟุตบอลล้มแล้วมีเสียงลั่นที่หลัง 1 ปี ปวดหลังมากขึ้น ปวดร้าวลงขาขวามากกว่าซ้าย motor gr 5 all	M48
6	ปวดหลังร้าวลงต้นขาขวา 6 เดือน กินยายังไม่ดีขึ้น motor grade V all SLRT negative	M48
7	case ผู้ป่วยหญิงปวดข้อมือซ้ายมา 1 ปี เป็นๆหายๆ	M70
8	case ปวดคอร้าวลงแขน 2 ปี เวลายกแขนแล้วมีอาการปวด	M75
9	1 อาทิตย์ ขณะวิ่งออกกำลังกาย ปวดสะโพกซ้าย แต่ยังออกกำลังกายได้ ปวดมากขึ้นเรื่อยๆ เดินลงน้ำหนักได้ แต่ปวด	M84
10	1 ปี ประสบอุบัติเหตุ แขนขวาหักใส่เฝือก แล้วแขนผิดรูป แพทย์นัดมาผ่าตัด	M84
11	3 ปี ปวดสะโพกขวาเวลาเดินไกล ๆ เดินเซ แพทย์นัดมาผ่าตัด	M87
12	MCA accident 2 mo PTA มีปวดต้นคอ + อ่อนแรงและชาแขน 2 ข้าง มากขึ้น อุจจาระปัสสาวะปกติ	S12

ภาคผนวก ค
ตัวอย่างโปรแกรมที่พัฒนาขึ้นมาใช้ในงานวิจัย

โปรแกรมสำหรับสร้างแบบจำลอง

```
import matplotlib
import matplotlib.pyplot as plt
import logging
import pandas as pd
import numpy as np
from numpy import random
import gensim


from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.metrics import accuracy_score, confusion_matrix
import matplotlib.pyplot as plt
import nltk
from nltk.corpus import stopwords
import re
from bs4 import BeautifulSoup
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import Pipeline
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.metrics import precision_recall_fscore_support
from sklearn.model_selection import KFold
from keras import layers
import itertools
import os
os.environ["CUDA_DEVICE_ORDER"] = "PCI_BUS_ID"   # see issue #152
os.environ['CUDA_VISIBLE_DEVICES'] = '-1'
```

```
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '2'
os.environ['KMP_DUPLICATE_LIB_OK']='True'
df = pd.read_csv('MedicalRecord.csv')
df = df[pd.notnull(df['c_group'])]
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import os
os.environ["CUDA_DEVICE_ORDER"] = "PCI_BUS_ID"   # see issue #152
os.environ['CUDA_VISIBLE_DEVICES'] = '-1'
import tensorflow as tf

from sklearn.preprocessing import LabelBinarizer, LabelEncoder
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
from tensorflow import keras
from keras.models import Sequential
from keras.layers import Dense, Activation, Dropout
from keras.preprocessing import text, sequence
from keras.callbacks import Callback, ModelCheckpoint
from keras import utils
from keras.optimizers import SGD
from keras.layers import Embedding
from keras.layers import LSTM,Conv1D,GlobalAveragePooling1D,Flatten
from keras.metrics import top_k_categorical_accuracy

REPLACE_BY_SPACE_RE = re.compile('[/(){}\[\]\|@,;]')
BAD_SYMBOLS_RE = re.compile('[^0-9a-z #+_]')
STOPWORDS = set(stopwords.words('english'))

def clean_text(text):
```

```
    """
        text: a string

        return: modified initial string
    """
    text = BeautifulSoup(text, "lxml").text # HTML decoding
    text = text.lower() # lowercase text
    text = REPLACE_BY_SPACE_RE.sub(' ', text) # replace REPLACE_BY_SPACE_RE
symbols by space in text
    text = BAD_SYMBOLS_RE.sub('', text) # delete symbols which are in
BAD_SYMBOLS_RE from text
    text = ' '.join(word for word in text.split() if word not in STOPWORDS) # delete
stopwors from text
    return text

df['full_text'] = df[['full_text','diag_name']].apply(lambda x: ''.join(x), axis=1)
df['full_text'] = df['full_text'].apply(clean_text)
train_size = int(len(df))

#train_posts = df['diag_name'].apply(clean_text)

train_posts = df['full_text']
max_words = len(df.full_text.unique())
tokenize = text.Tokenizer(num_words=max_words, char_level=False)
tokenize.fit_on_texts(train_posts) # only fit on train
x_train = tokenize.texts_to_matrix(train_posts,mode="tfidf")
batch_size = 50
epochs = 10
kf = KFold(n_splits=10, random_state=None, shuffle=True)
def top_7_accuracy(y_true, y_pred):
    return top_k_categorical_accuracy(y_true, y_pred, k=7)
```

```
#for train_index, test_index in kf.split(x_train):
# Build the model
d = {}
arr = ["L","I","E","K","R","B","J","N","F","H"]
for (i,v) in enumerate(df.c_group):
    if len(list(filter(lambda x:x == v[:1],arr))) > 0:
        df.c_group[i] = "O"

train_tags = df['c_group']
encoder = LabelEncoder()
encoder.fit(train_tags)
y_train = encoder.transform(train_tags)
num_classes = np.max(y_train) + 1
y_train = utils.to_categorical(y_train, num_classes)

X_train = x_train
Y_train = y_train
#X_train, X_test, Y_train, Y_test = train_test_split(x_train, y_train, test_size=0.1)
plt.plot(df.c_group)
plt.show()

#train word vector
from keras.regularizers import l1
arr = list(map(lambda x:x[:1],df.c_group))
encoder = LabelEncoder()
encoder.fit(arr)
arr = encoder.transform(arr)
class_len = np.max(arr) + 1
arr = utils.to_categorical(arr, class_len)
from sklearn.model_selection import KFold
from numpy import zeros
```

```
from numpy import asarray
max_length = 0
for i in X_train:
    if max_length < len(i):
        max_length = len(i)
vocab_size = len(tokenize.word_index) + 1
kf = KFold(n_splits=10, random_state=None, shuffle=True)
word_index = tokenize.word_index
MAX_NUM_WORDS = 35155
EMBEDDING_DIM = 100
embeddings_index = {}
with open(os.path.join("", 'enwiki_20180420_100d.txt')) as f:
    for line in f:
        values = line.split()
        word = values[0]
        try:
            coefs = np.asarray(values[1:], dtype='float32')
            embeddings_index[word] = coefs
        except ValueError:
            print(ValueError)

print('Found %s word vectors.' % len(embeddings_index))
# prepare embedding matrix
num_words = min(MAX_NUM_WORDS, len(word_index)) + 1
embedding_matrix = np.zeros((num_words, EMBEDDING_DIM))
for word, i in word_index.items():
    if i > MAX_NUM_WORDS:
        continue
    embedding_vector = embeddings_index.get(word)
    if embedding_vector is not None:
        # words not found in embedding index will be all-zeros.
```

```
        embedding_matrix[i] = embedding_vector
print(embedding_matrix)

from keras.layers import GRU
from keras.regularizers import l1
from sklearn.metrics import classification_report
from hyperopt import Trials, STATUS_OK, tpe
from hyperas import optim
from hyperas.distributions import choice, uniform

with tf.device('/gpu:0'):
    for train_index, test_index in kf.split(X_train):
        model = Sequential()
        model.add(Embedding(vocab_size-1, 100,
input_length=max_length,weights=[embedding_matrix],trainable=False))
        model.add(Dropout(0.5))
        model.add(Flatten())

model.add(Dense(int(num_classes/8),activation="tanh",activity_regularizer=l1(0.001),
input_shape=(max_words,)))
        model.add(layers.Dropout(0.5))
        model.add(Dense(int(num_classes/1),activation="tanh"))
        model.add(Dropout(0.5))
        model.add(Dense(num_classes,activation='softmax'))
        model.compile(loss='categorical_crossentropy',
                      optimizer="adam",
                      metrics=['accuracy'])
        checkpoint = ModelCheckpoint('best_model_word2vec.h5', verbose=1,
monitor='val_loss',save_best_only=True, mode='auto')
        # with tf.device('/gpu:0'):
        model.summary()
```

```
        history = model.fit(X_train[train_index], Y_train[train_index],
                        batch_size=batch_size,
                        epochs=50,
                        verbose=1,
                        validation_split=0.1,callbacks=[checkpoint])
        score = model.evaluate(X_train[test_index], Y_train[test_index],
                          batch_size=batch_size)
        print('Test accuracy:', score)
        Y_pred = model.predict_classes(X_train[test_index])
        test = utils.to_categorical(Y_pred, num_classes)
        report = classification_report(Y_train[test_index],
test,target_names=sorted(df.c_group.unique()))
        print(report)
        break
print(history.history['acc'])
print(history.history['val_acc'])
print(history.history['loss'])
print(history.history['val_loss'])
```


โปรแกรมสำหรับใช้แบบจำลองเพื่อการทำนาย

```
import matplotlib
import matplotlib.pyplot as plt
import logging
import pandas as pd
import numpy as np
from numpy import random
import gensim
import nltk
from sklearn.model_selection import train_test_split
```

```
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.metrics import accuracy_score, confusion_matrix
import matplotlib.pyplot as plt
from nltk.corpus import stopwords
import re
from bs4 import BeautifulSoup
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import Pipeline
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.metrics import precision_recall_fscore_support
import itertools
import os

df = pd.read_csv('MedicalRecord.csv')
df = df[pd.notnull(df['c_group'])]
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import os
os.environ["CUDA_DEVICE_ORDER"] = "PCI_BUS_ID"   # see issue #152
os.environ['CUDA_VISIBLE_DEVICES'] = '-1'
os.environ['KMP_DUPLICATE_LIB_OK']='True'
import tensorflow as tf

from sklearn.preprocessing import LabelBinarizer, LabelEncoder
from sklearn.metrics import confusion_matrix

from tensorflow import keras
from keras.models import Sequential
from keras.layers import Dense, Activation, Dropout
from keras.preprocessing import text, sequence
```

```
from keras.callbacks import Callback, ModelCheckpoint
from keras import utils
from keras.engine.topology import Layer
REPLACE_BY_SPACE_RE = re.compile('[/(){}\[\]\|@,;]')
BAD_SYMBOLS_RE = re.compile('[^0-9a-z #+_]')
STOPWORDS = set(stopwords.words('english'))

def clean_text(text):
    """
        text: a string

        return: modified initial string
    """
    text = BeautifulSoup(text, "lxml").text # HTML decoding
    text = text.lower() # lowercase text
    text = REPLACE_BY_SPACE_RE.sub(' ', text) # replace REPLACE_BY_SPACE_RE
symbols by space in text
    text = BAD_SYMBOLS_RE.sub('', text) # delete symbols which are in
BAD_SYMBOLS_RE from text
    text = ' '.join(word for word in text.split() if word not in STOPWORDS) # delete
stopwors from text
    return text
df['full_text'] = df['full_text'].apply(clean_text)
df['full_text'] = df[['full_text','diag_name']].apply(lambda x: ''.join(x), axis=1)
df = df[pd.notnull(df['c_group'])]
df['diag'] = df[['c_group','diag_name']].apply(lambda x: ':'.join(x), axis=1)
train_size = int(len(df))
train_posts = df['full_text'][:train_size]
max_words = len(df.full_text.unique())
tokenize = text.Tokenizer(num_words=max_words, char_level=False)
tokenize.fit_on_texts(train_posts) # only fit on train
```

```
min = int(0*len(df['full_text']))
max = int(1*len(df['full_text']))
#train_posts = df['full_text'][0]
train_posts = ["feeling of pain at knee"]
x_train = tokenize.texts_to_matrix(train_posts,mode="tfidf")
x_test = x_train[:]

from keras.models import load_model
from IPython.display import SVG
from keras.utils.vis_utils import model_to_dot
arr = []
arr = ["L","I","E","K","R","B","J","N","F","H"]
for (i,v) in enumerate(df.c_group):
    if len(list(filter(lambda x:x == v[:1],arr))) > 0:
        df.c_group[i] = "O"
model = load_model('best_model_word2vec.h5')
y_pred = model.predict_classes(x_test)
#print(y_pred)
from heapq import nlargest
y_pred = list(map(lambda x: sorted(df.c_group.unique())[x], y_pred))
sum = 0
index = 0
result = []
for (i,v) in enumerate(model.predict_proba(x_test)):
    arr = []
    for j in nlargest(7, enumerate(v),key=lambda x: x[1]):
        arr.append(j)
    result.append(arr)
#print(result)
y_actual = df.c_group[min:max]
x = []
```

```
y = []
for (i,v) in enumerate(y_pred):
    for j in result[0]:
        if (j[0] == 124):
            print("Other diseases")
        else:

print(df[df['c_group'].str.match(sorted(df.c_group.unique())[j[0]])]['diag_name'].tolist()[0])
        print(sorted(df.c_group.unique())[j[0]],str(round(j[1]*100,2))+"%")
        x.append(sorted(df.c_group.unique())[j[0]])
        y.append(round(j[1]*100,2))
print(y)
print(y)
s = 0
for v in y:
    s += v
y.append(100-s)
x.append("Not in top 7th list")
print(y)
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
f, ax = plt.subplots(1,figsize=(20,10))
lineWidth = 2
n = np.arange(len(y))
ax.plot(x,y,linewidth=lineWidth)
ax.set_ylim(bottom=0)
plt.show(f)
```

ภาคผนวก ง
เหตุและผลในการสร้างแบบจำลอง

เหตุผลการใช้เวกเตอร์น้ำหนัก

จำนวนมิติของเวกเตอร์น้ำหนักจะส่งผลต่อคุณสมบัติ (property) ของคำที่ถูกนำมาใช้ในการสร้างแบบจำลอง กล่าวคือถ้าจำนวนมิติของเวกเตอร์ยิ่งมีมากเท่าไร คุณสมบัติของคำที่ถูกนำมาใช้ก็จะดีมากขึ้นเท่านั้น แต่ในขณะเดียวกันหากจำนวนมิติของเวกเตอร์ยิ่งมีมาก ระยะเวลาที่ถูกใช้ในการสร้างแบบจำลองก็จะยิ่งเพิ่มมากขึ้น ดังนั้นสาเหตุที่งานวิจัยนี้เลือกใช้เวกเตอร์น้ำหนักขนาด 100 มิติเพื่อสร้างแบบจำลอง เนื่องจากเมื่อทำการทดลองและวิเคราะห์แล้วพบว่า เมื่อมิติของเวกเตอร์มีขนาดเกิน 100 มิติขึ้นไป คุณสมบัติของคำที่ดีขึ้นจะไม่คุ้มค่ากับระยะเวลาที่เสียไปในขั้นตอนการสร้างแบบจำลอง บวกกับข้อจำกัดทางด้านทรัพยากรในงานวิจัยนี้ที่ใช้เพียงซีพียู (CPU) ในการสร้างแบบจำลอง ทำให้เครื่องไม่สามารถรองรับการประมวลที่มีจำนวนมิติของเวกเตอร์น้ำหนักเกิน 100 มิติได้ ซึ่งในงานวิจัยนี้มีเป้าหมายเพื่อหาวิธีการสร้างแบบจำลองที่ดีที่สุดจากการใช้เวกเตอร์น้ำหนักเท่านั้น ดังนั้นการเพิ่มจำนวนมิติของเวกเตอร์เพื่อให้ได้มาซึ่งคุณสมบัติของคำที่ดีขึ้นจึงไม่ใช่วัตถุประสงค์หลักของงานวิจัยนี้ แต่สามารถนำไปใช้เป็นแนวทางในการต่อยอดการสร้างแบบจำลองในอนาคตเพื่อให้ได้แบบจำลองที่มีประสิทธิภาพดีขึ้นได้

อิทธิพลของการกำหนดน้ำหนักเริ่มต้นในชั้นฝังคำต่อแบบจำลอง

ก่อนการสร้างแบบจำลองจะต้องมีการกำหนดค่าน้ำหนักเริ่มต้นให้กับคำที่จะถูกนำมาใช้ในการสอนแบบจำลอง โดยการกำหนดค่าน้ำหนักของคำในงานวิจัยนี้จะอ้างอิงมาจากค่าน้ำหนักในเวกเตอร์ของเวิร์ดทูเวกหรือโกรฟ ซึ่งถือเป็นเวกเตอร์มาตรฐานที่ถูกนำมาใช้เพื่อกำหนดค่าน้ำหนักเริ่มต้นของคำ โดยการกำหนดค่าน้ำหนักของคำในงานวิจัยนี้ จะนำเอาค่าน้ำหนักของคำที่ตรงกับคำในเวิร์ดทูเวกมาใช้ และหากพบคำที่ไม่ตรงกับคำในเวิร์ดทูเวก งานวิจัยนี้จะกำหนดค่าน้ำหนักของคำนั้นเป็น 0 ซึ่งจากผลการทดลองพบว่าจำนวนของคำที่ตรงกันระหว่างคำในงานวิจัยนี้กับเวิร์ดทูเวกมีจำนวนเท่ากับ 16,924 คำจากทั้งหมด 35,156 คำ หรือคิดเป็น 55.9% ของทั้งหมด

ความลึกของโครงข่ายประสาทในการเรียนรู้เชิงลึก

โครงสร้างของการเรียนรู้เชิงลึก ประกอบไปด้วยชั้นต่าง ๆ เช่น ชั้นกำหนดค่าน้ำหนักเริ่มต้น ชั้นซ่อนตัว และชั้นการจำแนกประเภท ซึ่งในการสร้างแบบจำลองการเรียนรู้เชิงลึกจำเป็นต้องมีการกำหนดจำนวนชั้นที่จะใช้ในชั้นซ่อนตัว เพื่อให้ได้แบบจำลองที่มีความแม่นยำ โดยจำนวนชั้นที่มากขึ้นอาจไม่ทำให้ได้แบบจำลองที่แม่นยำขึ้นเสมอไป ดังนั้นในงานวิจัยนี้จึงได้ทำการทดลองเพื่อหาจำนวนชั้นที่เหมาะสมที่สุดสำหรับนำมาใช้สร้างแบบจำลอง โดยการทดลองเพิ่มทีละ 1 ชั้นไปเรื่อย ๆ จนได้ค่าความแม่นยำสูงสุด ซึ่งผลลัพธ์ที่ได้พบว่าจำนวนชั้นที่เหมาะสมที่สุดคือ 2 ชั้น เพราะเมื่อมีจำนวนชั้นมากกว่า 2 ชั้นแล้ว ค่าความแม่นยำของแบบจำลองจะลดลงเรื่อย ๆ จากการทดลองพบว่า ถ้าใช้จำนวนชั้น 3 ชั้น ผ่านการฝึกฝนด้วยจำนวนรอบที่มากกว่า 2 ชั้น 5 เท่าแต่ได้ผลลัพธ์เป็นความแม่นยำเพียง 73.42 %

ความแม่นยำจากการแปลภาษาด้วยตัวแปลภาษาของกูเกิล (Google translate)

ชุดข้อมูลที่นำมาใช้เพื่อสร้างแบบจำลองในงานวิจัยนี้ประกอบไปด้วยข้อมูลที่เป็นภาษาอังกฤษ ภาษาอังกฤษปนไทย และภาษาไทย แต่เนื่องจากเวกเตอร์น้ำหนักของเวิร์ดทูเวกที่นำมาใช้ในการกำหนดค่าน้ำหนักเริ่มต้นของคำ รองรับเฉพาะคำที่เป็นภาษาอังกฤษเท่านั้น ดังนั้นในงานวิจัยนี้จึงจำเป็นต้องทำการแปลภาษาของข้อมูลทั้งหมดให้เป็นภาษาอังกฤษก่อน โดยไลบรารี่ที่งานวิจัยนี้เลือกใช้คือตัวแปลภาษาของกูเกิล ซึ่งจากการทดลองสุ่มตัวอย่างของข้อมูลจำนวน 50 ระเบียนไปทำการแปลภาษาผ่านตัวแปลภาษาของกูเกิล และนำผลลัพธ์ของข้อมูลที่ได้ซึ่งเป็นภาษาอังกฤษทั้งหมดไปให้แพทย์ทำการตรวจสอบพบว่าตัวแปลภาษาของกูเกิลมีความแม่นยำในการแปลภาษา 90% โดยพบว่าถ้าตัวอย่างข้อมูลที่ถูกนำมาแปลเป็นภาษาอังกฤษทั้งหมด การแปลของกูเกิลจะแม่นยำ 100%

ตัวอย่างผลการให้คะแนนการแปลภาษาโดยใช้ตัวแปลภาษาของกูเกิล

รหัสไอซีดีเทน	คำวินิจฉัย	คำวินิจฉัยที่ผ่านตัวแปลภาษา	คะแนน
M46	Chronic osteomyelitis เป็น spondylolithiasis 5 ปี จากอุบัติเหตุ ได้ทำ Laminectomy with instrument 3 ครั้ง fail surgery หลังมา F/U ตามนัด F/U Film พบ LS-spine : Osteolytic lesion at L4 with separated R/O Osteomyelitis then -> admit for debridement and fusion at L3-4 admit 1/11/47 ทาง Ortho ได้ให้ Antibiotic เป็น Clinda และ Set Or for explore wound + debridement at 18/11/47 -> ให้ clinda until 29/11/47 Culture found Staphylo. coagulase negative จึงได้ consult Med (infectious) เปลี่ยน Antibiotic เป็น vancomycin ต่อมา 8/12/47 Ste OR for explore wound with debridement ได้ fluid ที่ anterior iliac crest ผล culture : bacteroides fragilis 23/12/47 ผู้ป่วย Stuporous + dyspnea + BP drop -> คิดถึง Septic shock ย้ายมา Med on Levophed ในผู้ป่วยรายนี้คิดถึงสาเหตุอาจเกิดจาก 1. Instruemnt infection 2.CNS infection -> LP cell 160 (N90%) 3.Other source ได้ start ATB : Meropenem + Vancomycin ได้ถึง 25/11/47 เกิด generalized MP rash off Meropenem switch เป็น Moxifloxacin แทน 26/12/47 clinical เริ่มดีขึ้น consciousness 29-30/12/47 clinical ดีขึ้น รู้สึกตัว รับ feed ได้ Hemo culture negative 2. Acute Renal Failure มีปัญหา Acute renal failure 23/12/47 + severe aicdosis + hyperkalemia then ย้ายเข้า ICU 2 on CVVH -> OK off CVVH ได้ในวันที่ 29/21/47 30/12/47 on hemodialysis ต่อ 3. Convulsion 23/12/47 มี convulsion หลายครั้ง -> on Valium IV + Dialntin -> not improve add Phenobarbe -> หยุดชัก คิดว่าสาเหตุ convulsion จาก 1. severe acidosis 2. R/O CNS infeciton 3. Uremia และเกิด	Chronic osteomyelitis is a spondylolithiasis 5 years after the accident. Laminectomy with instrument 3 times failed after surgery Follow up F / U Film found LS-spine: Osteolytic lesion at L4 with separated R / O Osteomyelitis then -> admit for debridement and fusion at L3-4 admit 1/11/47 Ortho has given Antibiotic to Clinda and Set Or for explore wound + debridement at 18/11/47 -> to clinda until 29/11/47 Culture found Staphylo. coagulase negative Consult Med (infectious) Change of Antibiotic to vancomycin. 8/12/47 Ste. OR for explore wound with debridement. Fluid in the anterior iliac crest. culture: bacteroides fragilis 23/12/47 Stuporous patients + dyspnea + BP drop -> miss Septic shock moves Med on Levophed in this patient. Causes may be caused by 1. Instruemnt infection 2.CNS infection -> LP cell 160 (N90%) 3.Other source has started ATB: Meropenem + Vancomycin up to 25/11/47 Generalized MP rash off Meropenem switch to Moxifloxacin 26/12/47 clinical Improved consciousness 29-30 / 12/47 clinical Improved feed intake Hemo culture negative 2. Acute Renal Failure Acute renal failure 23/12/47 + severe aicdosis + hyperkalemia Move on to ICU 2 on CVVH -> OK off CVVH on 29/21/47 30/12/47 on hemodialysis. 3. Convulsion 23/12/47 has many convulsions -> on Valium IV + Dialntin -> not. improve add Phenobarbe -> stop convulsions think	4/5

	infected instrument เกิด septic shock ได้ antibiotic : Moxifloxacin + Vancomycin then switch to Fosfomycin + Teicoplanin + Fusidic acid ได้รักษาให้ inotrope จนเกิด digital dry gangrene ทางศัลยกรรมแนะนำให้รอ autoamputation ทาง ortho ได้ผ่าตัด remove instrument ที่หลังออกเมื่อ 27/1/48 หลังผ่าตัดอาการโดยรวมดีขึ้น ไข้ดีขึ้น และมี plan ให้ Antibiotic ครบ 1 ปี รักษาแบบ chronic osteomyelitis เปลี่ยนยาเป็น Ciprofloxacin, Fusidic acid, Rifampicin	convulsion cause of 1. severe acidosis 2. R / O CNS infeciton 3. Uremia An infected instrument septic shock caused by antibiotic: Moxifloxacin + Vancomycin then switch to Fosfomycin + Teicoplanin + Fusidic acid treatment has a digital dry gangrene inotrope until the surgery is recommended to wait. Remove the instrument after 27/1/48 after surgery, improve overall symptoms, improve fever and have a plan to Antibiotic complete one year treatment of chronic osteomyelitis. Drug change Ciprofloxacin, Fusidic acid, Rifampicin	
M51	during admission ENT and psychiatric consultations were done for sinusitis and adjustment disorder	During admission ENT and psychiatric consultations were done for sinusitis and adjustment disorder	5/5
S42	1/2 hr PTA หล่น รถมอเตอร์ไซด์ แขนขวายันพื้น ไม่มีแขนชา ปวดต้นแขนขวา กระดกข้อมือและกำมือขวาได้ Film ==> Spiral Fx Right humerus with butterfly fragment	1/2 hr Prior to admits dropping motorbike Right arm, no floor, no arm, pain, right arm, right wrist, right wrist and right hand. Film ==> Spiral fracture Right humerus with butterfly fragment	2/5
M87	anterior approach : intra-op difficult to insert femoral stemp-> accidental tear posterior capsule-> post op immobilization for 2 wk before ambulation	anterior approach: intra-op difficult to insert femoral stem-> accidental tear posterior capsule-> post op immobilization for 2 wk before ambulation	5/5
M17	piriformis syndrome -> kenakot with xylocain injection -> improve	piriformis syndrome -> kenakot with xylocain injection -> improve	5/5
Q67	consult Pediatric: fix split S2 SEM at LUSB plan ECHO -> OPD case 23/03/2548 pulmonary function test: normal wait for MRI 22/03/2548	Consult Pediatric: fix split S2 SEM at LUSB plan ECHO -> OPD case 23/03/2005 pulmonary function test: normal wait for MRI 22/03/2005	5/5
S72	Urine c/s [26/1] -- peudomonas aeruginosa ----- ciprofloxacin [500] 1*2 bed sore at buttock ---- consult ostomy clinicc	Urine Cesarian section [26/1] - peudomonas aeruginosa ----- ciprofloxacin [500] 1 * 2 bed sore at buttock ---- consult ostomy clinicc	5/5
M51	CT brain: old encephalomalacia at	Brain CT: old encephalomalacia at	5/5

	anterior temporal lobe plan wait for MRI-> F/U OPD neuro med	anterior temporal lobe plan wait for MRI-> Follow up OPD neuro med	
M51	หลังผ่าตัด ผู้ป่วยสามารถลุกนั่งได้ทันทีที่หายเจ็บแผล total drain 160 cc แผลดี ไม่มี complication ใดๆ	After surgery, the patient can sit immediately to cure the wound. Total drain 160 cc.	2/5
T84	No Underlying disease CC: Left knee swelling with tender 1 wk PTA PI : 2 1/2 mo PTA Dx Left OA Knee ==> Left HTO 4/11/47 1 1/2 mo PTA had Pin tract infection C/S : Staph coag neg => sens Clinda Admit 23/11/47 D/C 26/12/47 admit day had swellingwith tender Left knee ---------- ---------- Hospital course : 14/1/48 off External fixator 18/1/48 Tapping Left knee ==> 20 ml Clear yellow well vecosity C/S ==> No growth 1/2/48 Tapping Left knee yellow turbid fluid 15 cc Culture no growth	Nobelium Underlying disease cc left knee swelling with tender Footnote 1 wk Prior to admits Pi : 2 1.2 Molybdenum Prior to admits Dx left Osteoarthritis Knee ==> left HTO 4/11/47 Footnote 1 1.2 Molybdenum Prior to admits had Pin tract infection Cesarian section : Staph coag neg => sens Clinda Admit 23/11/47 Discontinue 26/12/47 admit day had swellingwith tender left knee ---------- ---------- Hospital course : 14/1/48 off External fixator 18/1/48 Tapping left knee ==> 20 ML Clear yellow well vecosity Cesarian section ==> Nobelium Hussman Strategic Growth 1/2/48 Tapping left knee yellow turbid fluid 15 cc Culture Nobelium growth	1/5
M51	MRI L-S spine [24/1] -- posterior disc bulging L4/5 , L5/S1 , degenerative disc L3/4 , L4/5 , L5/51	MRI L-S spine [24/1] - posterior disc bulging L4 / 5, L5 / S1, degenerative disc L3 / 4, L4 / 5, L5 / 51	5/5
M51	หลังผ่าตัด ผู้ป่วยสามารถลุกนั่งได้ทันทีที่หายปวดแผล แผลดี ไม่มี complication ใด ๆ total drain 90 cc	After surgery, the patient can sit up immediately, healing pain, ulcer, wounds well, no complication. Total drain 90 cc	5/5
M23	History : 9 month PTA : MC rider accident, Left knee contact floor, pain at left knee with swelling , after that the swelling was resolved but loose knee, abnormal gait, no pain and had swelled on and off esp. walk so much PE : Not pale, no icteric sclera. Heart : Normal S1,S2 , no murmur Left Knee : Quadriceps power grade IV Anterior Drawer sign Positive	history : 9 Production Month Prior to admits : MC rider accident, left knee Contact Us floor, pain AT left knee with swelling , after that The. swelling was resolved shoes loose knee, abnormal gait, Nobelium pain Adnexa had swelled ON Adnexa off esp. walk SO much Physical examination : Not pale, Nobelium icteric sclera. Heart : Normal S1,S2 , Nobelium murmur left	2/5

	Pivot shift test Positive Varus, Valgus stress test Negative Ballotement Positive Mc Murrey's test Negative ---------- OR for ACL Reconstruction 19/01/2548 ----------	Knee : Quadriceps power Grade IV. Anterior Drawer sign Positive Pivot Shift 6 months order 10\% off Positive Varus, Valgus stress 6 months order 10\% off Negative Ballotement Positive mc Murrey's 6 months order 10\% off Negative ---------- OR for ACL Reconstruction 19/01/2548 ----------	
S34	มาด้วยอาการปวดหลังมา 1 เดือนหลังจากล้มขณะเล่นฟุตบอล ปวดหลังร้าวลงขา 2 ข้าง ชาหลังเท้าเป็นบางครั้ง แนะนำให้ผ่าตัดแต่ผู้ป่วยปฏิเสธการผ่าตัด	With back pain one month after falling while playing football. Back pain on the legs 2 times the numbness after the foot is sometimes. Recommended surgery, but patients refuse surgery.	4/5
M87	Underlying Aplastic anemia S/P BM Transplantation 6 yr at Klang Hospital ---------- CC: Both Hip pain 2 yr PI : 2 yr PTA Both hips pain, increase pain when walking PE : Right Left SLRT Pos Pos Rolling Pos Pos Thomas Pos Pos Hip Ext 20 5 degrees Hip Flex 100 90 degrees Hip Int Rot 5 5 degrees Hip Ext Rot 20 15 degrees Hip Adduct 45 45 degrees Hip Abduct 30 0 degrees Film Both hip : AVN of both hips (Left >> Right) ---------- 27/01/2548 OR For Total Hip Replacement	Underlying Aplastic anemia Status post BM Transplantation 6 yr AT Klang Hospital ---------- cc Both Hip pain 2 yr Pi : 2 yr Prior to admits Both hips pain, Increase the chances for success pain when walking Physical examination : Align Right left SLRT POS POS Rolling POS POS Thomas POS POS Hip Ext 20 5 degrees Hip FLEX-radiation 100 90 degrees Hip Int Scramble (Rot 13) 5 5 degrees Hip Ext Rot 20 15 degrees Hip Adduct 45 45 degrees Hip Abduct 30 0 degrees Film Both hip : AVN of both hips left >> Align Right ---------- 27/01/2548 OR For Total Hip Replacement	5/5
T84	small bowel obstruction due to adhesion band ได้ lysis adhesion band แต่เนื่องจาก adhesion ที่เกาะ อยู่ลึก ใน pelvic cavity จึงได้ ทำ small bowel resection and anastomosis	Small bowel obstruction due to adhesion band due to adhesion band, but due to adhesion in deep pelvic cavity, small bowel resection and anastomosis	5/5

ประวัติผู้เขียน

ชื่อ-สกุล	ธนากร รัตนจริยา
วัน เดือน ปี เกิด	26 มีนาคม 2537
สถานที่เกิด	จ. นครราชสีมา
วุฒิการศึกษา	สำเร็จการศึกษาระดับปริญญาตรี หลักสูตรวิศวกรรมศาสตร์บัณฑิต (วศ.บ.) สาขา วิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ที่อยู่ปัจจุบัน	70/29 เพชรบุรี ซอย 5 แขวง ทุ่งพญาไท เขต ราชเทวี กรุงเทพฯ 10400