## 2.1 สาระสำคัญ

การศึกษา ถือเป็นรากฐานที่สำคัญในการพัฒนาทรัพยากรมนุษย์ เปรียบเสมือนเครื่องมือหลักในการช่วยพัฒนาตั้งแต่การวางรากฐาน ศักยภาพ และขีดความสามารถ จนก่อให้เกิดเป็นพลังสร้างสรรค์ในการพัฒนาประเทศอย่างยั่งยืน ดังนั้น การศึกษาจึงเป็นอีกหนึ่งปัจจัยสำคัญที่มีความจำเป็นต่อการพัฒนาประเทศ ซึ่งปัจจุบันการศึกษาถูกใช้เป็นนโยบายหลักสำหรับพัฒนาประเทศในหลายประเทศ

ในประเทศไทย การพัฒนาการศึกษาในระดับประถมศึกษาและมัธยมศึกษา ถือเป็นหนึ่งปัญหาสำคัญที่เกิดขึ้น และรอแนวทางที่จะเข้ามาช่วยพัฒนาและแก้ไข ซึ่งในปัจจุบันรัฐบาลได้มีการให้ทุนสนับสนุนสำหรับการศึกษาค้นคว้า ทดลองหาวิธีการต่างๆ ที่จะช่วยเพิ่มประสิทธิภาพทั้งการสอนของคุณครู และการเรียนของเด็กนักเรียนให้สูงขึ้น โดยผลที่ได้จากการวิจัยหรือทดลองนี้ จะถูกเขียนออกมาเป็นรูปเล่มรายงานอย่างหนา ในรูปแบบของไฟล์ PDF ที่ประกอบไปด้วย รายงานสำหรับผู้บริหารที่แสดงถึงตัวเลขสถิติต่างๆของการทดลองนั้น และรายงานสำหรับการเรียนการสอน ที่อธิบายรายละเอียดแนวทางการเรียนการสอนในการทดลองนั้นๆ เช่น เทคนิคการสอน การดึงความสนใจของนักเรียนในเรื่องต่างๆ หรือการนำสื่อการเรียนการสอนมาใช้ให้เป็นประโยชน์ เป็นต้น อย่างไรก็ตาม เอกสารรายงานเหล่านี้ไม่ได้มีการเผยแพร่ และกระจัดกระจายอยู่ตามโรงเรียนหรือหน่วยงานต่างๆ ส่งผลให้การใช้ประโยชน์จากเอกสารกเหล่านั้นเป็นไปได้ยากมาก เนื่องจากในปัจจุบันยังไม่มี knowledge sharing platform สำหรับคุณครู ที่จะสามารถเข้าไปสืบค้นหาข้อมูลได้ อีกทั้งจำนวนหน้าของรายงานที่ค่อนข้างมาก ทำให้การกรองข้อมูลต่างๆเพื่อที่จะศึกษาเรื่องใดเรื่องหนึ่งนั้นทำได้ช้ามาก

ทั้งนี้ ทางผู้จัดทำได้เล็งเห็นถึงความสำคัญในการรวบรวมและคัดกรองข้อมูล ซึ่งการที่จะทำให้ข้อมูลเหล่านี้สามารถนำไปใช้ให้เกิดประโยชน์สูงสุดในเชิงของ knowledge sharing ได้นั้น จะต้องอาศัยการสรุปใจความในส่วนที่สำคัญของรายงานแต่ละเล่ม เพื่อให้คุณครูสามารถสืบค้น เรียนรู้จากตัวอย่างการสอนที่ดี (best practice) และสามารถใช้ประโยชน์จากข้อมูลในรายงานเหล่านั้นได้ในเวลาอันรวดเร็ว ดังนั้น จึงได้ออกแบบระบบ Building a knowledge sharing platform with Text mining and knowledge extraction ขึ้น ภายใต้ความร่วมมือกับมูลนิธิสดศรี-สฤษดิ์วงศ์และมูลนิธิโรงเรียนรุ่งอรุณ เพื่อรวบรวมรายงาน เคล็ดลับตัวอย่างวิธีการสอนที่ดีเหล่านี้ รวมถึงวิเคราะห์ข้อมูลในรายงาน สรุปใจความสำคัญ และแบ่งออกเป็นหมวดหมู่ เพื่อให้ง่ายต่อการสืบค้นและศึกษาเรียนรู้ อย่างไรก็ตาม ข้อมูลที่อยู่ในไฟล์ PDF หรือไฟล์ text นั้นเป็นข้อมูลขนาดใหญ่ที่ไม่มีโครงสร้าง (Schemaless) จึงไม่สามารถนำมาประมวลผลเพื่อวิเคราะห์ด้วยซอฟท์แวร์ทั่วๆไปได้ ดังนั้น เพื่อให้ข้อมูลเหล่านี้สามารถนำมาใช้ให้เกิดประโยชน์ได้ จำเป็นต้องมีการพัฒนาระบบบริหารจัดการข้อมูลชนิดใหม่ที่สามารถดึงองค์ความรู้ออกจากข้อมูล text และนำไปบูรณาการกับข้อมูลอื่นๆ เพื่อนำไปใช้ประโยชน์ในการวิเคราะห์เชิงสถิติ และการวิเคราะห์เชิงทำนาย (Predictive Analytics) ได้

โดยการประมวลผลข้อมูล text หรือ Text Mining นี้ ต้องอาศัยเทคโนโลยีทางด้าน Machine Learning มาช่วยในการทำงาน

ในการพัฒนาระบบในระยะเริ่มต้นนั้น จะมีคุณครูอาสาสมัครจากมูลนิธิ เข้ามาช่วยอ่านและทำการเน้นข้อความ หรือพารากราฟ ในส่วนที่เป็นเนื้อหาใจความสำคัญของรายงานนั้นๆ อีกทั้งสร้าง tag เพื่อบ่งบอกหัวเรื่องของเนื้อหานั้นด้วย เพื่อใช้สำหรับการจัดหมวดหมู่ เช่น สอนยังไงให้เด็กเข้าใจง่ายที่สุด เทคนิคการดึงดูดความสนใจของนักเรียน เป็นต้น โดยคุณครูจะช่วยทำการวิเคราะห์รายงานนี้ทำแค่บางส่วนเท่านั้น หลังจากนั้นข้อมูลที่คุณครูวิเคราะห์เหล่านี้ จะถูกนำมาพัฒนา Machine Learning Model โดย model จะถูกสอนให้เรียนรู้คำต่างๆ (train) และ tag ของพารากราฟนั้นๆ ที่อยู่ในรายงาน เมื่อมีรายงานเล่มใหม่เข้ามาในระบบ ระบบจะทำการวิเคราะห์เนื้อหา (Topic discovery) และสามารถแสดงส่วนที่เป็นใจความสำคัญ รวมถึงจัดประเภทหมวดหมู่ (Document classification) ของรายงานได้อัตโนมัติ โดยไม่ต้องอาศัยคุณครูมานั่งอ่านหมดทั้งเล่ม ซึ่งวิธีนี้ช่วยให้ประหยัดทั้งเวลาและจำนวนทรัพยากรบุคคลเป็นอย่างมาก เมื่อผู้ใช้งานเข้ามาใช้ระบบนี้ จะสามารถสืบหาข้อมูลที่เกี่ยวข้องกับการพัฒนาการเรียนการสอน ด้วยการใส่คำ (tag) ที่ต้องการค้นหา จากนั้น web application จะให้ผลลัพธ์ออกมาเป็นย่อหน้าพารากราฟที่เกี่ยวข้องกับ tag ที่ผู้ใช้งานใส่ พร้อมทั้งแนบลิงค์สำหรับดาวน์โหลดเอกสาร ผู้ใช้งานสามารถอ่านพารากราฟใจความสำคัญที่ระบบแสดงก่อนได้ หากตรงกับความสนใจสามารถดาวน์โหลดรายงานทั้งเล่มไปเพื่อศึกษารายละเอียดต่อไป

สำหรับความท้าทายของการพัฒนาระบบนี้คือ รูปแบบประโยคของภาษาไทย เนื่องจากภาษาไทยไม่มีการแบ่งคำที่ชัดเจนเหมือนภาษาอังกฤษที่จะมีการใช้เว้นวรรคคั่นระหว่างคำ ทำให้การวิเคราะห์รูปประโยคการแบ่งพารากราฟต่างๆในรูปแบบไฟล์ PDF ค่อนข้างยาก จึงต้องมีการใช้ Image processing รวมกับ Text analytic algorithm เข้ามาช่วยในการจำแนกคำต่างๆออกมากจากไฟล์ PDF อีกทั้งยังต้องทดลองหา Model ที่เหมาะสมสำหรับการเรียนรู้ tag ของแต่ละพารากราฟในแต่ละเอกสารอีกด้วย

ทางผู้จัดทำคาดหวังว่าระบบที่กล่าวมานี้ จะเป็นอีกหนึ่งแหล่งรวบรวมข้อมูลความรู้ ที่จะช่วยให้คุณครูสามารถศึกษาค้นคว้าเพื่อพัฒนาการเรียนการสอน และช่วยผลักดันให้การศึกษาในประเทศไทยสามารถพัฒนาก้าวหน้าไปได้อีกขั้น

## 2.2 คำสำคัญ

Machine Learning, Clustering, Latent Semantic Analysis, Classification, One vs Rest, Big data, Spark, Impala, PDF to Text, Keyword extraction, Text Analysis, Tag Box Extraction from PDF

**3. หลักการและเหตุผล**

จากสถิติการศึกษาขั้นพื้นฐานของประเทศไทยในรายงานประจำปีของ World Economic Forum ปี 2014-2015 [1] พบว่าประเทศไทยอยู่ในลำดับที่ 90 จาก 144 ประเทศทั่วโลกที่ได้รับการจัดอันดับ ซึ่งถือได้ว่าอยู่ในลำดับค่อนข้างล่าง ในขณะเดียวกันผลการวิเคราะห์ในรายงานของ International Institute of Management Development และ Pearson-The Economist Intelligence Unit พบว่าการศึกษาของไทยถูกจัดให้อยู่ในกลุ่มต่ำสุดเช่นเดียวกัน ซึ่งรายงานเหล่านี้ล้วนเป็นตัวบ่งชี้ให้เห็นว่าการศึกษาของไทยยังอ่อน แอ และมีจุดบกพร่องเป็นอย่างมาก ควรที่จะต้องได้รับการพัฒนาอย่างเร่งด่วน

หนึ่งในปัจจัยสำคัญที่มีผลต่อการพัฒนาการศึกษา คือการสอนของคุณครู เพราะคุณครูเปรียบเสมือนผู้ที่ถ่ายทอดความรู้ต่างๆและพัฒนาเด็กให้เติบโตไปเป็นทรัพยากรมนุษย์ที่มีคุณภาพ ดังนั้น รัฐบาลไทยจึงมีการให้ทุนสนับสนุนกับครูในการศึกษาค้นคว้า ทดลองหาวิธี ในการพัฒนาการเรียนการสอนให้มีประสิทธิภาพมากยิ่งขึ้น และเข้าถึงเด็กนักเรียนได้มากขึ้น โดยเฉพาะการศึกษาขั้นพื้นฐานในระดับประถามและมัธยมศึกษา ซึ่งในปัจจุบัน ได้มีเอกสารที่ถูกเขียนออกมาเพื่อรายงานผลการทดลอง และวิธีในการพัฒนาการเรียนการสอนที่ดี (best practice) ซึ่งรายงานเหล่านี้มักจะประกอบไปด้วยจำนวนหน้าที่มาก และอยู่ในรูปแบบของไฟล์ PDF ทำให้ครูสามารถสืบค้นข้อมูลได้ยาก และต้องเสียเวลาในการอ่านหนังสือหลายร้อยหน้าจำนวนหลายเล่มเพราะเอกสารไม่มีการรวบรวมและจัดเป็นหมวดหมู่ ทำให้ประสิทธิภาพในการสืบค้นข้อมูลนั้นไม่ดี อีกทั้งยังอาจได้ข้อมูลที่ไม่ครบถ้วน

ทางผู้จัดทำจึงจะทำการรวบรวมเอกสารรายงานเหล่านี้ เพื่อทำให้เกิดเป็น knowledge sharing platform ที่คุณครูสามารถเข้ามาสืบค้นหามูล และศึกษาค้นคว้าได้อย่างง่าย ภายใต้โครงการ Building a knowledge sharing platform with Text mining and knowledge extraction โดยได้รับความร่วมมือกับมูลนิธิมูลนิธิสดศรี-สฤษดิ์วงศ์และมูลนิธิโรงเรียนรุ่งอรุณ ซึ่งระบบนี้ จะช่วยรวบรวมและทำการจัดระเบียบเอกสารให้มีความเป็นระบบมากขึ้น รวมทั้ง ทำการวิเคราะห์ คัดแยกเนื้อหาส่วนต่างๆ ในไฟล์เอกสาร และทำการ tag ข้อความเหล่านั้นให้โดยอัตโนมัติว่า เนื้อหาในส่วนนั้นๆ มีความเกี่ยวข้องกับเรื่องอะไรบ้าง และทำการจัดเก็บข้อมูลเหล่าลงไปยังระบบฐานข้อมูล ซึ่งเปรียบเสมือนเครื่องมือที่ช่วยคัดกรองเนื้อหาขั้นต้นในเรื่องที่คุณครูสนใจศึกษา เพื่อให้สามารถทำการสืบค้นได้ง่ายและรวดเร็วยิ่งขึ้น สามารถนำความรู้เหล่านี้ไปพัฒนาการเรียนการสอน และพัฒนาให้การศึกษาของไทยก้าวไปสู่ในระดับต้นๆของโลกได้

**4. วัตถุประสงค์**

- เพื่อสร้างฐานข้อมูลที่ก่อให้เกิดเป็น knowledge sharing platform ของประเทศไทยได้
- เพื่อให้คุณครูสามารถสืบค้นข้อมูลได้อย่างมีประสิทธิภาพและรวดเร็ว สามารถนำไปพัฒนาการเรียนการสอน เพื่อช่วยให้การศึกษาของไทยดีขึ้น
- เพื่อศึกษาการทำ Document clustering และ Topic discovery สำหรับการคัดแยกเนื้อหา และจัดหมวดหมู่เอกสาร
- เพื่อสร้าง Machine Learning Model สำหรับเรียนรู้เอกสารภาษาไทยและ tag ที่กำหนดเพื่อใช้สำหรับการประมวลผลออกมาเป็นพารากราฟที่สำคัญ และ tag ที่เกี่ยวข้องจากเอกสาร
- เพื่อสร้าง Web Application สำหรับค้นหา tag ที่สนใจ และแสดงผลลัพธ์ออกมาเป็น Paragraph และ Tag ที่เกี่ยวข้องพร้อมเอกสารฉบับสมบูรณ์ในรูปแบบ PDF

**5. ปัญหาหรือประโยชน์ที่เป็นเหตุผลให้ควรพัฒนาโปรแกรม**

ปัจจุบันการศึกษาค้นคว้าหาข้อมูลเพื่อที่จะนำมาพัฒนาการเรียนการสอนของคุณครูนั้นทำได้ยาก เนื่องจากเอกสารข้อมูลต่างๆถูกเก็บอย่างกระจัดกระจาย ทำให้ถูกนำมาใช้ประโยชน์ได้เท่าที่ควร และในขณะเดียวกันเอกสารรายงานเหล่านี้มักจะประกอบไปด้วยจำนวนหน้าที่มาก และอยู่ในรูปแบบของไฟล์ PDF ทำให้ครูสามารถสืบค้นข้อมูลได้ยาก และต้องเสียเวลาในการอ่านเพื่อหาใจความสำคัญเฉพาะในส่วนที่ต้องการเรียนรู้เพื่อนำไปพัฒนา ตามที่ได้กล่าวมาแล้วในข้อที่ 3

ดังนั้น โครงการ Building a knowledge sharing platform with Text mining and knowledge extraction จะช่วยสร้างฐานข้อมูลรวบรวมความรู้ที่เกี่ยวกับการพัฒนาการเรียนการสอน และก่อให้เกิดเป็น knowledge sharing platform ของประเทศไทย เพื่อให้คุณครูสามารถเข้ามาค้นคว้าหาความรู้ได้ง่ายยิ่งขึ้น ซึ่งระบบนี้จะช่วยคัดแยก จัดหมวดหมู่เอกสารเป็นประเภท รวมถึงช่วยคัดเลือกส่วนที่เป็นความสำคัญของเนื้อหาในรายงานแต่ละเล่ม เพื่อมาแสดงให้เห็นสาระสำคัญของรายงานเล่มนั้นๆ ซึ่งเปรียบเสมือนเป็นเครื่องมือที่ช่วยคัดกรองเนื้อหาที่คุณครูสนใจได้อีกทางหนึ่ง ซึ่งจะช่วยให้คุณครูสามารถเข้าถึงองค์ความรู้เหล่านี้ได้อย่างครบถ้วน สามารถค้นหาข้อมูลได้อย่างรวดเร็ว และนำความรู้ไปพัฒนาการเรียนการสอนเพื่อให้การศึกษาของเด็กไทยก้าวหน้าต่อไป

ทั้งนี้ หลังจากโครงการพัฒนาสำเร็จ ระบบนี้ยังสามารถนำไปประยุกต์ใช้กับเรื่องอื่นๆได้ เช่น วิธีการเพาะปลูกให้ได้ผลิตผลที่ดี เป็นต้น โดยการเปลี่ยนข้อมูลของการ train model (ไฟล์ PDF และข้อมูล tag)

**6. เป้าหมายและขอบเขตของโครงการ**

เป้าหมายของโครงการนี้ คือการสร้าง knowledge sharing platform ที่จะช่วยรวบรวมข้อมูลรายงานตัวอย่างการเรียนการสอนที่ดี (best practice) มาทำการคัดแยก จัดหมวดหมู่ เพื่อให้คุณครูสามารถสืบค้นข้อมูลได้อย่างมีประสิทธิภาพและรวดเร็ว สามารถนำความรู้ไปพัฒนาและปรับใช้ตาม เพื่อให้เกิดการเรียนการสอนที่ดีที่จะช่วยพัฒนาศักยภาพของเด็กนักเรียนได้ โดย platform มีขอบเขต ดังนี้

- สามารถรับไฟล์ PDF ภาษาไทย เพื่อทำการระบุและแบ่งพารากราฟต่างๆจากไฟล์นั้น และทำการดึงข้อความภาษาไทยในแต่ละพารากราฟออกมาผ่านกระบวนการ Text processing เพื่อให้ได้ผลลัพธ์สุดท้ายเป็นคำที่สำคัญต่างๆ สำหรับนำไปใช้ในการทำ Machine Learning
- สร้าง Machine learning model ที่สามารถรับข้อมูล text file ภาษาไทยที่ได้จากขั้นตอนที่ 1 และทำการติด tag ของแต่ละพารากราฟในไฟล์นั้นๆ โดยวิธีการ supervised classification ซึ่งจะแบ่งเป็น 2 ขั้นตอนคือ
    - การทำ training model จะรับ text และ tag ของ paragraph
    - การทำ prediction จาก model โดยจะรับ text เป็น paragraph และแสดง tag เป็นผลลัพธ์ และทำการเก็บลงใน Database
- Web application เพื่อที่จะใช้ในการสืบค้นข้อมูลที่ทำการ tag มาแล้วจากขั้นตอน prediction จากใน Database

**รายละเอียดของการพัฒนา**

**7.1 Story board**

เมื่อต้องการที่จะค้นหาเอกสารที่เกี่ยวข้องกับเรื่องใดเรื่องหนึ่งขึ้นมาใช้งาน ผู้ใช้จะสามารถค้นหาข้อมูลได้อย่างรวดเร็วด้วย platform ที่ทางกลุ่มพัฒนาขึ้น โดย platform ที่พัฒนาขึ้นจะแบ่งออกได้เป็น 2 ส่วนหลักๆ

***- ส่วนของผู้พัฒนาและผู้ดูแลระบบ*** โดยผู้พัฒนาจะทำการเตรียมเอกสารที่เป็นไฟล์ PDF ตัวอย่าง ซึ่งเอกสารเหล่านี้จะมีผู้เชี่ยวชาญเฉพาะมาช่วยในการระบุคำสำคัญต่างๆเพื่อทำการเตรียม machine learning model โดยหลังจากที่ทำการสร้าง model เสร็จแล้ว เอกสารที่เหลือจะทำการ tag เอกสารได้โดยอัตโนมัติ โดยใช้ machine learning model ข้างต้น เช่น ถ้าต้องการให้ตัว model สามารถทำการจำแนกเนื้อหาที่เกี่ยวข้องกับเรื่อง “การดึงความสนใจนักเรียน” ผู้พัฒนา/ผู้ดูแลจะต้องเตรียมเอกสารที่มีเนื้อหาที่เกี่ยวข้องกับเรื่องการดึงความสนใจของนักเรียน และให้ผู้เชี่ยวชาญช่วยระบุว่า มีคำใดบ้างที่สามารถระบุได้ว่า ข้อความนี้มีความ

เกี่ยวข้องกับ "การดึงความสนใจของนักเรียน" และนำไปทำการเตรียม model โดยหลังจากสร้าง model เสร็จแล้ว ผู้พัฒนาสามารถนำเอกสารที่เกี่ยวข้องกับ "การดึงความสนใจนักเรียน" มาทำการ tag เอกสารโดยอัตโนมัติได้

***- ส่วนของผู้ใช้งาน*** ผู้ใช้สามารถเข้ามาใช้งานผ่าน web application ที่ทางกลุ่มพัฒนาขึ้นมา แล้วทำการค้นหาเนื้อหาที่เกี่ยวข้องกับสิ่งที่ผู้ใช้ต้องการ แล้วเนื้อหาส่วนนั้นก็จะปรากฏขึ้นมา และมีไฟล์เอกสารนั้นให้ผู้ใช้สามารถ download ไปอ่านได้ ยกตัวอย่างเช่น ครูสมศรีต้องการที่จะหาข้อมูลเรื่อง “การดึงความสนใจนักเรียน” เพื่อนำไปเตรียมการเรียนการสอนสำหรับชั้นเรียน สิ่งที่คุณครูต้องทำก็คือ ค้นหาด้วยคำว่า “ดึงความสนใจนักเรียน” ในหน้าเว็บ แล้วเว็บก็จะทำการแสดงผลย่อหน้าที่เกี่ยวข้องกับการดึงความสนใจนักเรียนจากเอกสารต่างๆในระบบ รวมถึงแสดง tag ที่เกี่ยวข้องกับย่อหน้านั้นๆ โดยแต่ละย่อหน้าก็จะมี tag ที่เกี่ยวข้องเป็นของตัวเอง และมีลิงค์สำหรับดาวน์โหลดเอกสารที่มีข้อความนั้นอยู่ให้คลิกเพื่อดาวน์โหลดได้

โดย platform นี้จะมีจุดเด่นที่เราสามารถนำ model นี้ ไปประยุกต์ใช้กับหัวข้ออื่นๆได้ โดยไม่จำเป็นต้องออกแบบโปรแกรมใหม่ทั้งหมด เพียงแค่เตรียมเอกสารที่เกี่ยวข้องและกำหนดคำสำคัญของหัวข้อนั้นๆให้กับเอกสารตัวอย่างและให้ระบบทำการเรียนรู้ด้วยตัวเองกับเอกสารที่เหลือ

ในปัจจุบันนี้ มีโปรแกรมสำหรับการแปลง unstructured information(เอกสาร,รูปภาพ) ให้เป็น structured information(SQL tables) โดยใช้ machine learning ในการแปลงข้อมูลคือ Deepdive Stanford University จะเป็นโปรแกรมที่สามารถอ่านข้อมูลในหลากหลายรูปแบบ เช่น ข้อความในรูปแบบ text file หรือข้อมูลที่อยู่ในฐานข้อมูล แล้วสามารถนำข้อมูลต่างๆ เหล่านั้นมาเชื่อมโยงกันโดยใช้ machine learning และนำมาทำการวิเคราะห์ข้อมูลต่างๆได้ โดยใช้หลักการทำ document clustering และการทำ Topic Discovery ต่างๆ เช่น การนำบทความที่เขียนไว้และฐานข้อมูลมาสรุปผลร่วมกัน ซึ่งนอกจาก Deepdive [2] แล้ว จะมีโปรแกรมสำหรับดึงข้อมูลจาก unstructured information ได้แก่ AlchemyLangage API [3] ซึ่งใช้ IBM Watson ในการทำ Machine Learning โดยจะสามารถอ่านข้อมูลที่เป็น text file ต่างๆ โดยใช้ข้อมูลเหล่านั้น เทียบกับ public model หรือ Custom model โดยผลลัพธ์ที่ได้ออกมาจากการใช้ Alchemy API ได้แก่ Sentiment ของคำ, Name Entity Recognition และ Keywords ต่างๆ เป็นต้น หรือ Aylien [4] ที่เป็นโปรแกรมที่รับ text file และทำการตรวจสอบคำสำคัญ, สรุปของบทความหรือการสร้าง hashtag จาก model ของทางระบบที่สร้างไว้ ซึ่งโดยส่วนใหญ่ของโปรแกรมเหล่านี้ จะรองรับสำหรับภาษาในภาษาอังกฤษหรือภาษาที่รากศัพท์มาจากภาษาละติน เนื่องจากมี Library ในการจัดการทางภาษาศาสตร์จาก NLP Stanford

สำหรับเครื่องมือและเทคโนโลยีต่างๆที่ทางกลุ่มได้เลือกใช้ในขั้นตอนต่างๆ จะแบ่งเป็นการทำ Word Segmentation ซึ่งในปัจจุบันมีโปรแกรมสำหรับตัดคำภาษาไทยต่างๆได้แก่ LexTo เป็นโปรแกรมในการจัดคำ

ที่จะใช้วิธี Dictionary base ในการที่จะเลือกแบ่งคำจากประโยค และ TLex เป็นโปรแกรมในการตัดคำภาษาไทยโดยใช้ machine learning ชื่อว่า Condition Random Fields โดยทางกลุ่มได้เลือกใช้ Lexto สำหรับการตัดคำเนื่องจากมีการเพิ่มคำใหม่ต่างๆ การใช้ Dictionary base จะง่ายกว่าในการเรียนรู้, การทำ Topic Discovery นั้น จะมีวิธีการทำ Clustering สำหรับ Text Analysis 2 ตัว ได้แก่ Latent Dirichlet Allocation [5] โดยจะเป็นการทำ topic discovery จากข้อมูลต่างๆ โดยจะตรวจสอบเนื้อหาภายในนำมาเปรียบเทียบกับเอกสารอื่นๆใน topic ที่เกี่ยวข้อง และ Latent semantic analysis [6] จะทำการสร้าง matrix สำหรับเก็บจำนวน frequency ของคำและใช้วิธีการทางคณิตศาสตร์เรียกว่า Singular value decomposition (SVD) ในการลดจำนวนมิติของ array ใน matrix เพื่อหาค่าที่มีความเกี่ยวข้องมากที่สุดจากในเอกสารนั้นๆ ซึ่งทางกลุ่มได้เลือกใช้ LDA เนื่องจากสามารถตรวจสอบหาความเกี่ยวเนื่องระหว่างเอกสารได้ดีกว่า [7] และสุดท้ายการทำ Classification จะมีเทคนิคต่างๆ เช่น One-vs-Rest สามารถให้ผลลัพธ์การจัดกลุ่มได้หลายผลลัพธ์ (multiclass classification) โดยจะใช้วิธีการจัดกลุ่มข้อมูลนั้นๆ เข้าในแต่ละหมวดหมู่แล้วทำการเปรียบเทียบผลกับหมวดหมู่อื่นๆ แล้วเลือกผลลัพธ์ที่ทำให้การจัดกลุ่มมีความแม่นยำสูงที่สุด โดยการจัดกลุ่มอันนี้จะทำให้ข้อมูล 1 ย่อหน้าสามารถมี tag ได้หลายอย่าง, Neural network เป็น classification algorithm ที่เลียนแบบการทำงานของระบบประสาทของมนุษย์ และสุดท้าย Decision Tree เป็น rule-based classification คือการสร้างต้นไม้ของกฎต่างๆ เพื่อที่จะจัดกลุ่มข้อมูล โดยทางกลุ่มได้เลือกใช้ Neural Network [8] เนื่องจากเป็น machine learning algorithm ที่มีความยืดหยุ่นสูง และมีการปรับปรุงประสิทธิภาพของ model ได้เรื่อยๆ ระหว่างที่กำลังทำงานอยู่ ซึ่งต่างจาก rule-based algorithm ที่จะตายตัวเมื่อการสร้าง model เสร็จสิ้น

**7.2 เทคนิคหรือเทคโนโลยีที่ใช้**

- **Word Segmentation** เป็นวิธีการในการแบ่งคำต่างๆออกจากประโยค โดยในภาษาไทยนั้น รูปแบบของประโยคจะเป็นคำต่อกันโดยไม่มีตัวระบุการจบคำหรือประโยคเหมือนกับภาษาอังกฤษ หรือมีตัวอักษรที่มีความหมายหรือคำที่ชัดเจนแบบภาษาญี่ปุ่น ทำให้จำเป็นจะต้องใช้โปรแกรมเฉพาะในการตัดคำ
- **bag-of-words model** เป็นโมเดลในการทำ mapping ของคำต่างๆให้กลายเป็นตัวเลข เพื่อที่จะสามารถนำไปใช้ประโยชน์ในเชิงคณิตศาสตร์และการทำสถิติต่างๆต่อไป
- **Term frequency – Inverse document frequency (TF-IDF)** เป็นวิธีทางสถิติที่จะทำการตรวจสอบคำต่างๆในบทความเพื่อนำไปเปรียบเทียบกับบทความทั้งหมด เพื่อหาอัตราส่วนว่าคำๆนี้มีความสำคัญต่อบทความโดยรวมมากน้อยแค่ไหน โดย TF-IDF จะแบ่งขั้นตอนเป็น 2 ส่วนคือ

Term frequency โดยในขั้นตอนนี้จะทำการนับจำนวนครั้งที่คำต่างๆปรากฏในบทความหนึ่งๆ และการทำ Inverse document frequency โดยในขั้นตอนนี้จะเป็นนำคำต่างๆในบทความมาเปรียบเทียบกับบทความทั้งหมดและคำนวณหาค่าน้ำหนักความสำคัญนั้นๆจากบทความทั้งหมด โดยการทำ TF-IDF สามารถใช้ประโยชน์ในการหาคำสำคัญในบทความต่างๆ ซึ่งสามารถนำไปประยุกต์ใช้ได้อย่างหลากหลายเช่น การทำ Search engine หรือการทำ Text Summarization

- **Latent Dirichlet Allocation** เป็น clustering algorithm ที่ใช้สำหรับการทำ topic discovery จากข้อมูลต่างๆ ที่ใส่เข้าไป ซึ่งจะมีการเรียกใช้ vector ของคำที่ได้จากการทำ bag-of-word model มาทำการหาความถี่ของคำเทียบกับเอกสารต่างๆ และทำการแปลงสร้าง model ความเกี่ยวข้องของคำต่างๆ เทียบกับเอกสารอื่นๆที่ได้ทำการเรียนรู้ เพื่อค้นหา Keyword ที่สำคัญสำหรับนำไปใช้งานต่อ ซึ่ง LDA นั้นจะมองเอกสารเป็นการรวมกันของ topics ต่างๆที่ซ่อนอยู่ โดยแต่ละ topic จะมีค่าคำต่อความน่าจะเป็น ซึ่งจะบ่งบอกคำนี้มีความเกี่ยวข้องกับ topic ดังกล่าวมากน้อยเพียงใด โดยจะใช้สำหรับการดึง tag ที่เกี่ยวข้องต่างๆจาก paragraph เพื่อนำไปใช้สำหรับการ train model ในขั้นตอนการทำ classification
- **Neural network** เป็น machine learning algorithm ที่มีหลักการทำงานที่เลียนแบบการทำงานของโครงสร้างในระบบประสาทของมนุษย์ โดยมีการส่งข้อมูลที่ทำการเรียนรู้อยู่ในระบบเข้าสู่ node ต่างๆ และหาค่าน้ำหนักในแต่ละ node แล้วทำการส่งข้อมูลไปยัง node ย่อยๆ ต่างๆ ไปเรื่อยๆ จนได้ผลลัพธ์การจัดกลุ่มที่ดีที่สุด โดยการทำ neural network จะช่วยทำให้การระบุว่า tag เรื่องหนึ่งๆ มีความเกี่ยวข้องกับ paragraph ที่เรียนรู้หรือไม่ มีความแม่นยำในระดับที่น่าพึงพอใจ

### 7.3 เครื่องมือที่ใช้ในการพัฒนา

- **Hadoop Distributed File System (HDFS)** [9] เป็นระบบการจัดเก็บข้อมูลที่ออกแบบมาสำหรับการจัดการข้อมูลขนาดใหญ่ (Big data) โดย HDFS ถูกออกแบบมาสำหรับระบบที่มีคอมพิวเตอร์หลายๆ ตัวช่วยกันประมวลผล และ HDFS จะเหมาะกับการทำงานในลักษณะ “Write once, Read many” หรือข้อมูลที่เน้นการอ่านข้อมูลมากกว่าการเขียน,แก้ไข โดยลักษณะการทำงานของ HDFS ที่กล่าวไปข้างต้นนั้น มีความเหมาะสมกับรูปแบบการใช้งานของโครงการนี้เป็นอย่างมาก เนื่องจากข้อมูลที่เข้ามาในระบบนั้น จะถูกเขียนลงไปเพียงครั้งเดียว ไม่มีการแก้ไข และมีการอ่านข้อมูลขึ้นมาหลายๆ ครั้งในระหว่างการทำ Machine learning ซึ่งเข้ากันได้ดีกับรูปแบบการใช้งาน HDFS

- **โปรแกรม Spark ML** [10] เป็น library ที่มีอยู่ในโปรแกรม Apache Spark ซึ่ง Spark ML เป็น libraryที่ใช้ทำ Machine Learning โดยที่สามารถทำงานแบบขนาน (Parallel programming) ได้

  ซึ่ง Apache Spark เป็น engine สำหรับการทำการประมวลผลข้อมูลขนาดใหญ่ (Big data processing) ที่สามารถทำงานได้อย่างรวดเร็ว เนื่องจากใช้การประมวลผลในหน่วยความจำหลัก (In-memory processing) ทำให้การเข้าถึงข้อมูลทำให้รวดเร็วมากขึ้น ซึ่ง Spark ML นี้เป็น Machine learning library ที่ถูกใช้งานร่วมกับ big data platform อย่าง Hadoop กันอย่างแพร่หลาย และมีประสิทธิภาพในการทำงานสูง ทำให้ทางกลุ่มเลือกใช้โปรแกรมนี้
- **โปรแกรม Apache Hive** [11] เป็นโปรแกรมจัดการฐานข้อมูลแบบ SQL ที่เป็น Open source ที่ถูกออกแบบมาให้ใช้งานร่วมกับ Hadoop ecosystem ซึ่งมีระบบการจัดการฐานข้อมูลที่ยืดหยุ่น ไม่ว่าจะใช้งานผ่านหน้า system shell หรือเรียกใช้งานผ่าน JDBC ซึ่งสาเหตุที่ทางกลุ่มเลือกใช้ Hive คือ เป็นโปรแกรมฐานข้อมูลที่รองรับการเก็บข้อความภาษาไทย และมีความสามารถในการจัดการฐานข้อมูลเมื่อมีขนาดใหญ่มากๆ ได้
- **โปรแกรม Apache HBase** [12] เป็นโปรแกรมจัดการฐานข้อมูลแบบ NoSQL ที่เป็น Open source ที่ถูกใช้งานร่วมกัน Hadoop ecosystem โดยฐานข้อมูลแบบ NoSQL จะมีความยืดหยุ่นด้านโครงสร้างมากกว่าฐานข้อมูลแบบ SQL ดังนั้น ทางกลุ่มจึงนำ HBase มาใช้งานร่วมกันกับ Hive เพื่อเก็บข้อมูลที่เหมาะสมลงในฐานข้อมูลแต่ละโปรแกรม โดย HBase จะเก็บข้อมูลจำพวกเนื้อหาของแต่ละเอกสารที่ถูกแบ่งย่อหน้าแล้ว ซึ่งจำนวนย่อหน้าของแต่ละเอกสารจะมีไม่เท่ากัน ดังนั้นฐานข้อมูลแบบ NoSQL จึงเหมาะสมกับการเก็บข้อมูลลักษณะนี้ ส่วน Hive ที่เป็นฐานข้อมูลแบบ SQL จะจัดเก็บข้อมูลเรื่อง tag ของแต่ละย่อหน้าไว้ เพื่อให้สามารถทำการ Query ผ่านหน้าเว็บไซต์ได้อย่างรวดเร็ว
- **โปรแกรม PDFMiner** [13] เป็นโปรแกรมสำหรับการแปลงไฟล์ในรูปแบบ PDF ให้เป็น text file ซึ่ง PDFMiner เป็น Python API ที่ใช้สำหรับการดึงข้อมูลต่างๆออกมาจาก PDF Document เช่น ตัวอักษรในภาษาต่างๆ เช่น ไทย อังกฤษ จีน และอื่นๆ หรือสามารถดึงภาพออกจาก PDF ได้ โดยสำหรับโปรเจคนี้จะเน้นที่การดึงข้อความออกจาก PDF Document เพื่อสำหรับนำไป preprocess ต่อ ซึ่งฟังก์ชั่นที่ใช้ในการดึงข้อความออกมานั้นคือ PDF2TXT โดยคำสั่งต่างๆของฟังก์ชั่นนี้ สามารถเลือก page number, ชนิดของ output (text,tag,xml), ขนาดของ box ของคำใน pdf เป็นต้น โดยเหตุผลที่เราเลือกใช้ PDFMiner คือ การที่ตัวโปรแกรมสามารถ config ค่าต่างๆ ได้อย่างหลากหลาย ทำให้ผลการแปลงข้อมูลมีความแม่นยำมากยิ่งขึ้น เนื่องจากภาษาไทย

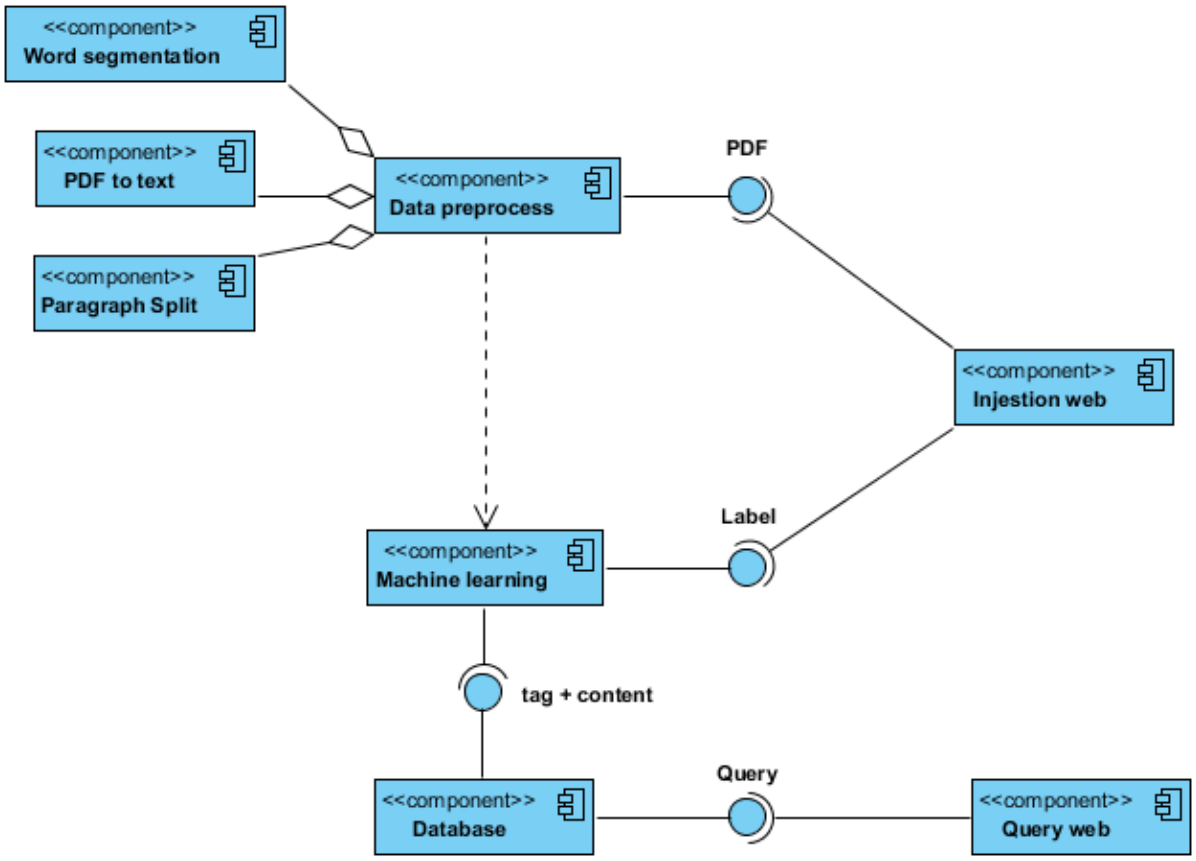
เป็นภาษาที่มีโครงสร้างซับซ้อนกว่าภาษาอังกฤษมาก ทำให้ต้องมีการดัดแปลงตัวโปรแกรมเพื่อให้สามารถทำงานได้อย่างถูกต้อง

- **โปรแกรม LexTo** เป็นโปรแกรมที่ถูกพัฒนาด้วยภาษา Java โดยโปรแกรมนี้สามารถใช้ในการแบ่งคำต่างๆในภาษาไทยจากประโยคให้กลายเป็นคำซึ่งแบ่งด้วย delimiter ซึ่งคำต่างๆที่ใช้ในการแบ่งนั้น จะมี Dictionary ที่จะทำการเก็บคำทั้งหมดเอาไว้ แล้วโปรแกรมจะนำมาเปรียบเทียบเพื่อแบ่งคำตามที่ Dictionary ได้กำหนดไว้ ซึ่ง LexTo เป็นโปรแกรมตัดคำภาษาไทยแบบ Open Source ที่ทางกลุ่มสามารถนำมาใช้งานได้ และมีความแม่นยำในระดับที่พอรับได้ ทำให้ทางกลุ่มเลือกใช้โปรแกรม LexTo

## 7.4 รายละเอียดโปรแกรมที่จะพัฒนา

### Input / Output Specification

### Component Diagram



จาก component diagram ข้างต้น จะเห็นได้ว่า โปรแกรมมีการแบ่งข้อมูลเป็น 3 ส่วนหลักๆ ได้แก่ ส่วนรับไฟล์ PDF เข้ามาในระบบ, ส่วนการทำ machine learning และส่วนของการ query ผลลัพธ์ออกมาแสดงผล โดยแต่ละส่วนจะมีการรับ-ส่งข้อมูลดังต่อไปนี้

***Ingestion Web Application***

- Input - PDF file อย่างเดียว หรือ PDF file ที่มีการระบุ tag ในแต่ละ paragraph แล้ว
- Output - Notification ว่ามีการรับ Input File สำเร็จแล้ว

***Machine Learning: Train Model***

- Input - PDF File ที่มีการระบุ tag ในแต่ละ paragraph โดยผู้เชี่ยวชาญเฉพาะทาง
- Output - Model ที่สามารถคาดเดา tag จาก paragraph และ Tag ที่ได้จาก pdf เก็บลงใน Database

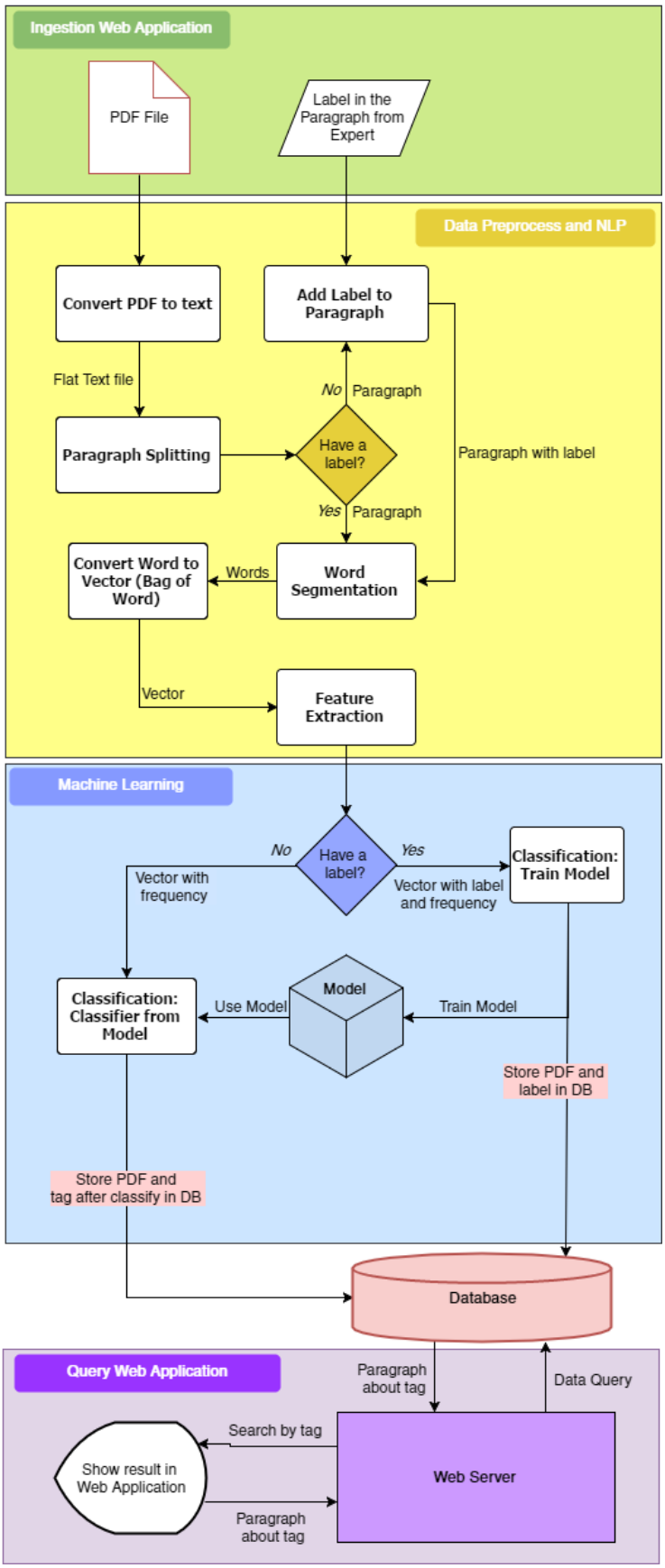
***Machine Learning: Prediction***

- Input - PDF File
- Output - ข้อมูล Tag ที่ได้จากการ Prediction จาก PDF ที่เป็น Input โดยใช้ model ที่สร้างขึ้น และเก็บข้อมูล PDF และ Tag ใหม่ที่ได้จากการ Prediction ลงใน Database

***Query Web Application***

- Input - tag ที่ต้องการจะสืบค้น
- Output - ย่อหน้าที่มีความเกี่ยวข้องกับ tag นั้นๆ และข้อมูลเกี่ยวกับย่อหน้านั้น ได้แก่ tag ของย่อหน้านั้นทั้งหมด, เอกสารที่เขียนข้อความนั้น และ link download เอกสารนั้นในรูปแบบไฟล์ PDF

Flowchart


Ingestion Web Application
PDF File
Label in the Paragraph from Expert
Data Preprocess and NLP
Convert PDF to text
Add Label to Paragraph
Flat Text file
No Paragraph
Paragraph Splitting
Have a label?
Paragraph with label
Yes Paragraph
Convert Word to Vector (Bag of Word)
Words
Word Segmentation
Vector
Feature Extraction
Machine Learning
No
Have a label?
Yes
Vector with frequency
Vector with label and frequency
Classification: Train Model
Classification: Classifier from Model
Use Model
Model
Train Model
Store PDF and label in DB
Store PDF and tag after classify in DB
Database
Query Web Application
Paragraph about tag
Data Query
Search by tag
Show result in Web Application
Web Server
Paragraph about tag

ส่วนประกอบในการทำงาน จะแบ่งขั้นตอนต่างๆออกเป็น 5 ส่วน ได้แก่

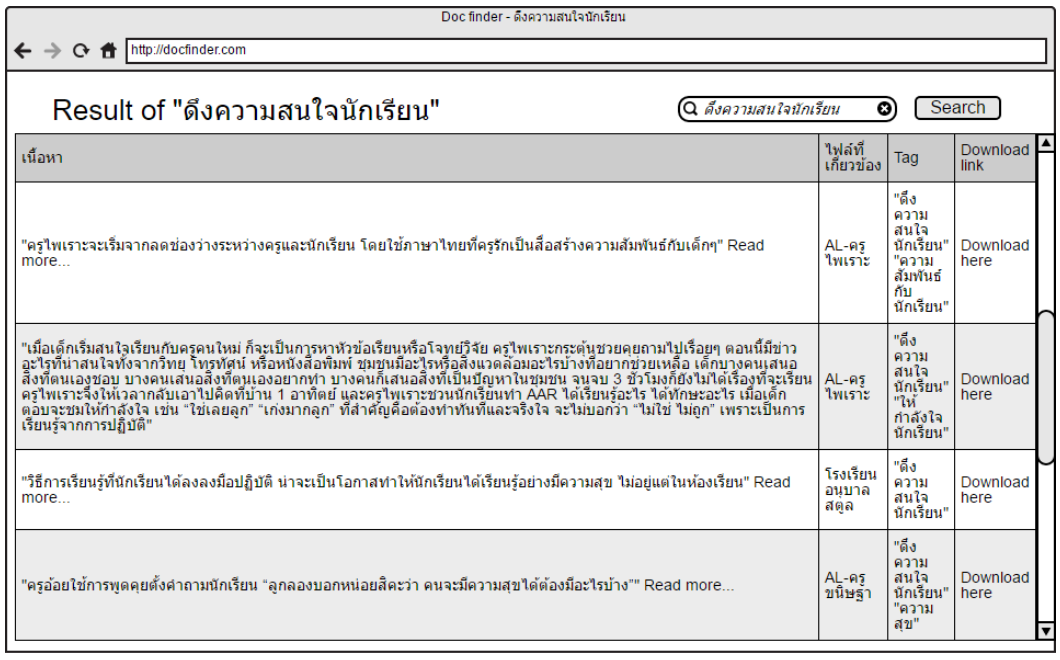
**1.Ingestion Web Application** โดยจะแบ่งวิธีการรับข้อมูลเป็น 2 แบบคือ 1.ทำการรับ PDF กับ Label สำหรับใช้ใน Train Model และ 2. รับ PDF อย่างเดียวสำหรับ Predict Tag จาก PDF นั้น

**2.Preprocessing Data** โดยในขั้นตอนนี้จะทำการเปลี่ยน PDF ที่รับมาให้กลายเป็น Flat Text และทำการรับ Label หากเป็นขั้นตอนการทำ Train Model ซึ่งหลังจากได้ Text มาแล้ว จะทำการเปลี่ยน Text เหล่านั้นมาสร้าง Vector ของคำ ซึ่งสำหรับภาษาไทยนั้น จำเป็นต้องมีการตัดแบ่งคำ (Word Segmentation) สำหรับประโยคออกเพื่อทำ NER (Name Entity Recognition) และทำการตัดคำต่างๆที่ไม่มีความหมายต่างๆทิ้งไป ได้แก่ คำเชื่อม เช่น และ, หรือ, กับ เป็นต้น และคำขยายความต่างๆเช่น การ ความ เป็นต้น และสุดท้ายจะทำ bag-of-word เพื่อสร้าง vector ของคำและนำไปใช้ในขั้นตอน Topic Discovery

**3. Topic Discovery** จะเป็นการดึงคำสำคัญหรือความเกี่ยวข้องต่างๆที่อยู่ใน paragraph ออกมา เช่นการทำ TF-IDF เพื่อหาความถี่ของคำ และการลดมิติของคำให้เหลือเพียงคำสำคัญต่างๆโดยการใช้ Machine Learning: Clustering คือ Latent Dirichlet Allocation

**4. Machine Learning: Classification** เป็นการสร้าง Machine สำหรับการจำแนกผลลัพธ์จากข้อมูลที่เข้ามา โดยจะแบ่งขั้นตอนการใช้งานเป็น 2 ขั้นตอนได้แก่ 1.การทำ Training และ Testing Model โดยในขั้นตอนนี้จะนำคำต่างๆที่ได้จากขั้นตอนข้างต้น รวมกับ label ที่ผู้เชี่ยวชาญได้ระบุไว้มาสร้าง Model สำหรับการ Classification ออกมา 2.การ Prediction จากเอกสารต่างๆ เพื่อให้ได้ผลลัพธ์ออกมาเป็น tag เพื่อนำไปใช้ในการสืบค้นใน Database ต่อไป โดยเทคนิค Classification คือ Neural Network

**5. Query Web Application** จะเป็นการสร้าง Web Application เพื่อติดต่อกับ Database โดยตัว Web Application นั้น จะทำการ Query Tag ที่ต้องการสืบค้นจาก Database แล้วนำมาแสดงผลตามตัวอย่างข้างล่าง



## Functional Specification

### Ingestion Web Application

- Upload PDF File มาเก็บไว้ในฐานข้อมูลและทำการส่งไปยังขั้นตอน Data Preprocess and NLP
- สามารถเลือกที่จะกรอก label ของ paragraph ต่างๆ สำหรับนำไปทำการ Training

### Data Preprocess and NLP

- ทำการแปลง PDF ที่รับมา ให้เปลี่ยนเป็น xml file โดยการใช้ PDFMiner
- ทำการ preprocess xml file ที่ได้ โดยจะทำการระบุย่อหน้าจาก xml schema, ทำการตัด xml schema ต่างๆ และดึงคำภาษาไทยออกมาและทำการแก้ภาษาไทยที่ผิดพลาดต่างๆจาก xml เช่น สระ อำ (อ า เป็น อำ)
- ทำการใส่ label ของคำเพิ่มเข้าไปใน paragraph สำหรับใช้เพื่อระบุในขั้นตอนต่อไป
- ทำการตัดคำจาก text file ด้วยโปรแกรม word segmentation สำหรับภาษาไทย เช่น LexTo
- ทำการกำจัดคำที่ไม่จำเป็นออกโดยการใช้ stop word remover
- สร้าง bag-of-word model สำหรับเก็บคำภาษาไทย และแปลงคำต่างๆจากในย่อหน้าให้กลายเป็น vector
- ใช้หลักการ TF-IDF ในการหาความถี่ของคำต่างๆ
- ใช้หลักการ LDA หรือ LSA กับ paragraph เพื่อช่วยลดมิติของคำ และนำคำที่ได้ไปรวมกับคำที่ label เพื่อกำหนดค่าที่ใช้สำหรับการทำ Classification

### Classification: Training model

- ทำการ Train Model ด้วย paragraph, label คำจาก expert และ keyword จากการทำ LDA หรือ LSA ด้วยวิธีการ Classification โดยทดสอบกับ Technique ต่างๆ ได้แก่ neural network, One vs Rest, Decision Tree เป็นต้น
- นำ label, paragraph และ keyword เก็บลงใน Database

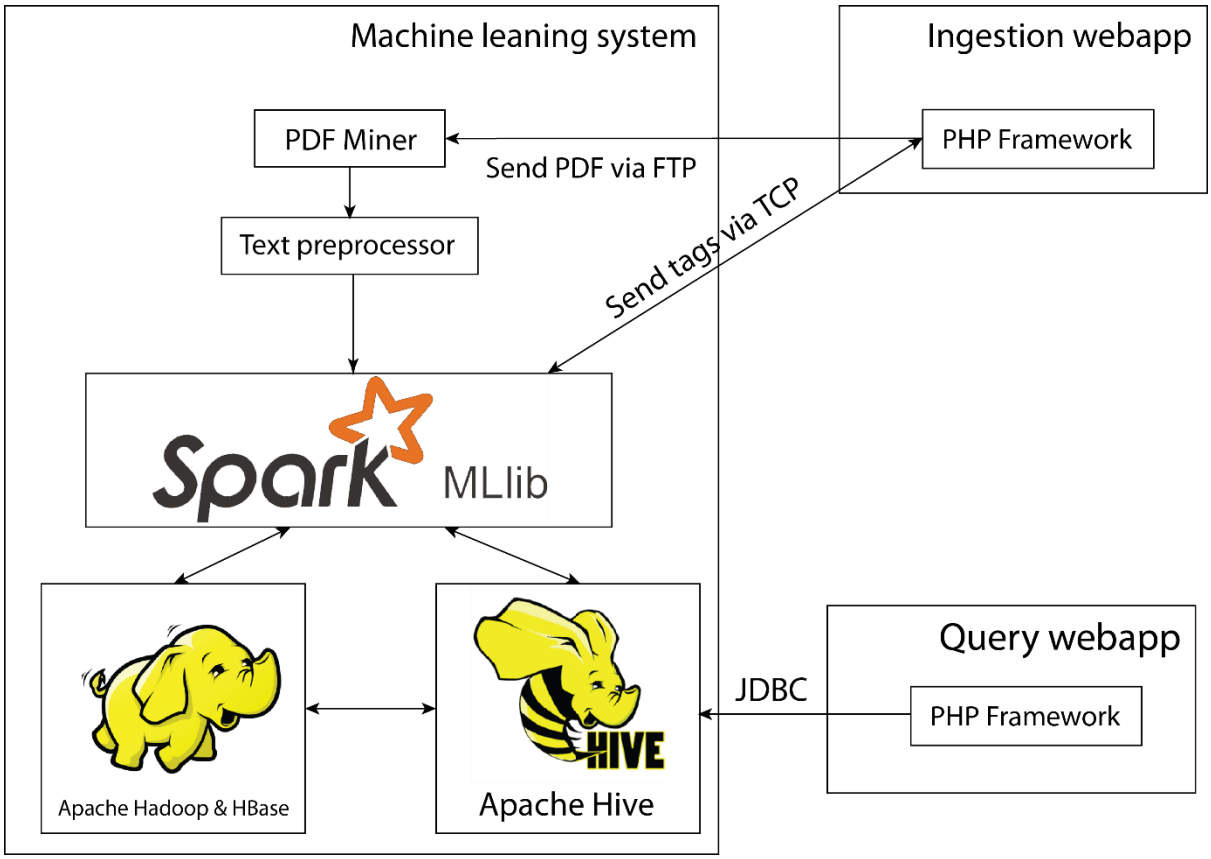
### Classification: Prediction

- นำ keyword ของคำที่ได้จากการทำ LDA หรือ LSA มาทำการ Classify ผ่าน Model ที่ได้ทำมาในขั้นตอน Training Model เพื่อหา Tag ของเอกสารเหล่านั้น
- นำ Tag ที่ได้จากการ Prediction, paragraph และ keyword ไปเก็บลงใน Database

**Query Web Application**

- รับคำที่ต้องการจะทำการค้นหาและทำการค้นหาคำในระบบ Database
- แสดงผลคำที่ต้องการค้นหา, paragraph ที่เกี่ยวข้อง, Tag ที่เกี่ยวข้องกับ paragraph นั้นและ PDF file ที่สมบูรณ์

**Architecture**




จากรูปข้างต้น จะเห็นได้ว่าโครงสร้างของตัวโปรแกรมจะถูกแบ่งออกเป็น 3 ส่วนหลักๆ ได้แก่ ส่วนของระบบในการทำ Machine learning, ส่วนของหน้าเว็บที่ใช้ในการรับ PDF และส่วนของหน้าเว็บที่ใช้ในการค้นหาข้อมูลจาก Tag โดยส่วนของระบบ Machine learning จะประกอบไปด้วย

- โปรแกรม PDF Miner ที่ใช้ในการแปลงไฟล์ PDF ให้เป็นไฟล์ข้อความ
- ส่วนของโปรแกรมที่ใช้ในการจัดเตรียมข้อความเพื่อที่จะนำไปใช้ในการทำ Machine learning ได้แก่โปรแกรมสำหรับจัดเรียงข้อมูลที่ไม่เรียบร้อย (data cleaning), โปรแกรมแบ่ง paragraph และโปรแกรม LexTo
- โปรแกรม Spark MLlib ซึ่งเป็นโปรแกรมที่ใช้ในการทำ machine learning สำหรับ hadoop ecosystem

- โปรแกรม Apache Hadoop และ HBase ที่ใช้ในการจัดเก็บข้อมูลเกี่ยวกับไฟล์ PDF และเนื้อหาภายในไฟล์นั้น
- โปรแกรม Apache Hive เป็นโปรแกรมจัดการฐานข้อมูลที่ใช้ในระบบ hadoop ecosystem

ส่วนต่อมาเป็นส่วนของหน้าเว็บที่ใช้ในการรับ PDF ซึ่งจะพัฒนาขึ้นด้วยภาษา PHP โดยมีหน้าที่รับไฟล์ PDF ที่อัพโหลดขึ้นมาจากผู้ใช้งาน และ tag ของเนื้อหาในไฟล์ PDF นั้น (ในกรณีที่เป็นไฟล์ PDF ที่ใช้ในการเรียนรู้ระบบ) แล้วทำการส่งมาที่เครื่องที่ทำการทำ machine learning ด้วย FTP protocol และส่งข้อความ tag ด้วย TCP protocol

ส่วนสุดท้ายเป็นส่วนของหน้าเว็บที่จะใช้ผู้ใช้ค้นหาเนื้อหาจาก tag ซึ่งพัฒนาขึ้นด้วยภาษา PHP เช่นเดียวกัน โดยหน้าเว็บจะรับ tag ที่ผู้ใช้ต้องการค้นหามาแล้วไปทำการ Query ใน Hive ออกมาแสดงผลให้ผู้ใช้งาน โดยติดต่อผ่าน OBDC

**7.5 ขอบเขตและข้อจำกัดของโปรแกรมที่พัฒนา**

- เนื่องจากภาษาไทยเป็นภาษาที่มีความซับซ้อนสูง ทั้งทางด้านตัวอักษร ที่มีสระบน-ล่าง และทางด้านรูปประโยคที่ไม่มีความแน่นอน ทำให้การเขียนโปรแกรมที่สามารถประมวลผลภาษาไทยได้อย่างสมบูรณ์แบบจึงเป็นเรื่องยาก ทำให้ความแม่นยำในการ tag และเลือกย่อหน้าที่มีความสำคัญกับเรื่องที่เลือก อาจจะต่ำกว่าการใช้งานกับภาษาอังกฤษ ที่มีรูปประโยคที่แน่นอนกว่า ทำให้สามารถใช้การดูรูปประโยคเข้ามาช่วยเสริมความหมายของคำได้ ซึ่งเป็นสิ่งที่ทำได้ยากมากในภาษาไทย
- ข้อมูลที่จะนำไปเข้าระบบ machine learning เพื่อให้ระบบทำการเรียนรู้ด้วยตนเองนั้น จะต้องใช้มนุษย์เป็นตัวช่วยในการกำหนดข้อมูลก่อนในเบื้องต้น (Supervised learning) เพราะฉะนั้น ถ้าเราต้องการให้ระบบเรียนรู้เนื้อหาเรื่องใหม่ๆ จะต้องมีการใช้ผู้เชี่ยวชาญที่เกี่ยวข้องกับเรื่องที่จะให้ระบบเรียนรู้มาช่วยทำการ label คำสำคัญก่อนที่จะนำข้อมูลเข้าไปในระบบ ดังนั้น
- ถ้าหากเราไม่สามารถหาผู้เชี่ยวชาญที่จะมาระบุคำสำคัญให้ได้ เราก็จะไม่สามารถทำให้ระบบเรียนรู้หัวข้อใหม่ๆ ได้
- การระบุย่อหน้าจาก PDF นั้นสามารถทำได้ยาก เนื่องจากการระบุย่อหน้าจาก PDF จำเป็นต้องใช้ค่าตำแหน่งของตัวอักษรต่างๆ เพื่อระบุว่าย่อหน้าควรจะอยู่ตำแหน่งไหน ซึ่ง PDF ที่ได้รับมานั้น มีรูปแบบการจัดหน้าและ font ที่แตกต่างกันรวมถึงรูปแบบคำภาษาไทยและภาษาอังกฤษใน

เอกสาร จะทำให้ตำแหน่งของคำเกิดการคลาดเคลื่อนซึ่งจะส่งผลให้ย่อหน้าที่ได้ออกมาอาจเกิดความผิดพลาดได้

**บรรณานุกรม**


[1] World Economic Forum. **Thailand Report** [Online]**.** Available: http://www3.weforum .org/docs/GCR2014-15/THA.pdf [2016, October 18]

[2] DeepDive: A Data Management System for Automatic Knowledge Base Construction. Ce Zhang.Ph.D. Dissertation, University of Wisconsin-Madison, 2015. Available: http://cs.stanford.edu/people/czhang/zhang.thesis.pdf [2016, October 18]

[3] AlchemyLanguageAPI. Available: https://alchemy-language-demo.mybluemix.net/. [2016, October 18]

[4] AYLIEN. Available: http://aylien.com/. [2016, October 18]

[5] Latent Dirichlet allocation. Blei, D. M., Ng, A. Y. and Jordan, M. I. In: Journal of Machine Learning Research 3, pp. 993-1022. 2003. Available: http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf [2016, October 18]

[6] Latent Semantic Analysis of Wikipedia with Spark. Available: http://www.slideshare.net/SandyRyza/lsa-47411625. [2016, October 18]

[7] Comparison between LSA-LDA-Lexical Chains. Costin Chiru, Traian Rebedea and Silvia Ciotec. 2014. Available: http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf [2016, October 18]

[8] An Empirical Comparison of Supervised Learning Algorithms. Rich Caruana, Alexandru Niculescu-Mizil. 2006. Avaliable: https://www.scribd.com/document/113006633/2006-An-Empirical-Comparison-of-Supervised-Learning-Algorithms [2016, October 18]

[9] Apache Hadoop. The Apache Software Foundation. 2014. Avaliable: http://hadoop.apache.org/ [2016, October 18]

[10] MLlib | Apache Spark, The Apache Software Foundation. Avaliable: http://spark.apache.org/mllib/ [2016, October 18]

[11] Apache Hive, The Apache Software Foundation. 2014. Avaliable: https://hive.apache.org/ [2016, October 18]

[12] Apache HBase, The Apache Software Foundation. 2016. Avaliable: http://hbase.apache.org/ [2016, October 18]

[13] PDFMiner, Yusuke Shinyama. 2013. Avaliable: http://www.unixuser.org/~euske/python/pdfminer/ [2016, October 18]