

2.1 สำคัญ

เนื่องจากในปัจจุบัน เอกสารความรู้ต่างๆ มากมาย ยังไม่ถูกนำมาจัดเก็บและจัดหมวดหมู่ให้เป็นระบบ ทำให้องค์ความรู้เหล่านั้นไม่สามารถนำมาใช้งานได้อย่างมีประสิทธิภาพ โดยในปัจจุบันนั้น การที่จะนำความรู้ที่มีอยู่เหล่านั้นมาจัดหมวดหมู่เพื่อใช้ในการสืบค้น จะต้องใช้มนุษย์เป็นผู้จัดการ ซึ่งจำเป็นต้องใช้แรงงานคนในการมาจำแนกหมวดหมู่เหล่านั้น โดยการพัฒนาการจัดการองค์ความรู้หรือ Knowledge management นี้ จะช่วยให้การนำความรู้ที่มีอยู่มาใช้ได้อย่างมีประสิทธิภาพมากขึ้น และช่วยส่งเสริมการพัฒนาประเทศชาติให้มุ่งไปสู่ความเป็นสังคมอุดมปัญญาได้ โดยที่ระบบ Knowledge management ที่มีอยู่ในปัจจุบันนั้น ยังไม่มีระบบที่สามารถนำเอาเอกสารไปทำการวิเคราะห์และ tag หมวดหมู่ของเนื้อหาได้โดยอัตโนมัติ โดยเฉพาะอย่างยิ่งในส่วนของภาษาไทย ซึ่งยังไม่มีนักพัฒนารายใดพัฒนาเทคโนโลยีในลักษณะนี้ออกมา ดังนั้น เราจึงคิดที่จะทำระบบที่สามารถนำเอาเอกสารหรือบทความต่างๆ ที่ถูกจัดเก็บไว้ในรูปของ PDF นำไปวิเคราะห์ข้อความและทำการสรุปว่า ข้อความนี้มีเนื้อหาที่เกี่ยวข้องกับเรื่องใดบ้าง และนำไปจัดเก็บลงไปยังฐานข้อมูล เพื่อให้สามารถทำการสืบค้นได้ โดยสิ่งที่ท้าทายสำหรับการทำโครงการขึ้นนี้ก็คือ การที่ภาษาไทยไม่มีการแบ่งคำที่ชัดเจนเหมือนภาษาอังกฤษที่มีการใช้ space คั่น ทำให้การวิเคราะห์รูปประโยคมีความยาก, การแบ่ง paragraph ต่างๆ ในรูปแบบ PDF file และการทำ machine learning ให้ได้ความแม่นยำในระดับที่สามารถนำไปใช้ได้จริงนั้น จะต้องใช้การเลือกใช้ algorithm และการปรับแต่งที่เหมาะสมกับข้อมูลที่นำมาใช้ จึงทำให้โครงการนี้มีความท้าทายในการดำเนินการ และเป้าหมายในการทำโครงการนี้ จะเป็นการพัฒนา model ที่สามารถนำข้อความจากเอกสารมา tag และทำการจัดหมวดหมู่ได้ และพัฒนา web application ที่สามารถสืบค้นข้อมูลที่ได้จาก model ข้างต้น โดย web application ที่จะสร้างขึ้นนั้น จะเป็น web application สำหรับสืบหาข้อมูลที่เกี่ยวข้องกับ การพัฒนาการเรียนการสอน ซึ่งจะสามารถทำการสืบค้นได้ด้วยการใส่ tag ที่ต้องการ และ web application จะให้ผลลัพธ์ออกมาเป็น ย่อหน้าที่เกี่ยวข้องกับ tag ที่เราทำการค้นหาพร้อมทั้งแนบลิงสำหรับ download เอกสาร

2.2 คำสำคัญ

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

3. หลักการและเหตุผล

ในยุคปัจจุบัน ที่มีการทำเอกสารในเรื่องต่างๆ ออกมาเป็นจำนวนมาก การทำ Knowledge management หรือการนำเอกสารข้อมูลเหล่านั้นมาจัดการให้เป็นระบบ นับเป็นเรื่องที่สำคัญมาก โดยเฉพาะในองค์กรหลายๆแห่ง การมีระบบ knowledge management จะช่วยให้องค์กรนั้นๆ สามารถบริหารจัดการงานองค์ความรู้ที่มีอยู่ได้อย่างมีประสิทธิภาพสูงสุด แต่ในปัจจุบัน เอกสารความรู้ต่างๆ ที่ถูกนำมาเผยแพร่อยู่นั้น มักจะอยู่ในรูปแบบของเอกสารในหน้ากระดาษ หรือเอกสารที่เป็นไฟล์ PDF ซึ่งยังไม่สามารถนำมาใช้งานได้อย่างมีประสิทธิภาพเท่าที่ควร เพราะว่า เอกสารเหล่านั้นมักจะมีข้อความอยู่มากมาย ที่เกี่ยวข้องกับเนื้อหาที่แตกต่างกัน แต่ว่าเมื่อผู้ที่ต้องการใช้งานความรู้เหล่านั้น ต้องการทำการหาเนื้อหาที่เฉพาะเจาะจงกับที่เขาสนใจในเอกสารนั้นๆ เขาก็ต้องทำการค้นหาด้วยตนเองโดยวิธีต่างๆ ไม่ว่าจะเป็นการไล่อ่านเนื้อหาทั้งหมดด้วยตนเอง ซึ่งใช้เวลามากในการอ่านและหาใจความสำคัญที่เขาต้องการ หรือใช้การค้นหา keyword ที่เขาต้องการด้วยวิธีต่างๆ เช่นการเปิดหาสารบัญ ซึ่งเอกสารบางฉบับก็ไม่มีสารบัญให้ หรือใช้การ search หา keyword ที่ต้องการ ซึ่งอาจจะเกิดการข้ามเนื้อหาในส่วนที่เกี่ยวข้องกับเรื่องที่ผู้ที่ค้นหาต้องการ แต่ไม่มี keyword ที่เขาใช้ค้นหาไปได้ ซึ่งสิ่งที่ได้กล่าวไปข้างต้นนั้น นับว่าเป็นปัญหาใหญ่ในการค้นคว้าหาข้อมูลเพื่อทำการศึกษาเป็นอย่างมาก เนื่องจากการที่ไม่มีระบบ knowledge management สำหรับเอกสารทุกๆ ไปนั้น ทำให้แหล่งความรู้ที่สามารถนำมาสืบค้นได้นั้นลดลงเป็นอย่างมาก และทำให้ความรู้จำนวนมากถูกทิ้งร้างไว้ไม่ได้ถูกนำมาใช้ให้เกิดประโยชน์ ดังนั้น ทางกลุ่มของเราจึงสนใจที่จะพัฒนา machine learning model ที่สามารถคัดแยกเนื้อหาในส่วนต่างๆ ในไฟล์เอกสาร และทำการ tag ข้อความเหล่านั้นได้โดยอัตโนมัติว่า เนื้อหาในส่วนนั้นๆ มีความเกี่ยวข้องกับเรื่องอะไรบ้าง และทำการจัดเก็บข้อมูลเหล่านั้นลงไปยังระบบฐานข้อมูลเพื่อให้สามารถทำการสืบค้นได้ง่ายและรวดเร็ว และทำให้การจัดการแหล่งความรู้ หรือ Knowledge management นั้น สามารถใช้งานกับเอกสารที่เป็นไฟล์ PDF ได้ ซึ่งส่งผลให้ความรู้ถูกนำไปใช้งานต่อ และเกิดการพัฒนาประเทศชาติในองค์กรรวมมากยิ่งขึ้น

4. วัตถุประสงค์

1. ศึกษาการทำ Image Processing แบ่งส่วนของ Paragraph ต่างๆ ออกจาก PDF
2. ศึกษาการทำ Text Processing ของภาษาไทย ได้แก่การตัดคำและการระบุประเภทของคำ
3. สร้าง Machine Learning Model สำหรับเรียนรู้เอกสารภาษาไทยและ tag ที่กำหนดเพื่อใช้สำหรับการประมวลผลออกมาเป็น Paragraph ที่สำคัญและ Tag ที่เกี่ยวข้องจากเอกสารใดๆในขอบเขตที่เกี่ยวข้อง
4. สร้าง Web Application สำหรับค้นหา Tag ที่สนใจ และแสดงผลลัพธ์ออกมาเป็น Paragraph และ Tag ที่เกี่ยวข้องพร้อมเอกสารฉบับสมบูรณ์ในรูปแบบ PDF

5. ปัญหาหรือประโยชน์ที่เป็นเหตุผลให้ควรพัฒนาโปรแกรม

ในปัจจุบันนั้น มีโปรแกรมที่ถูกพัฒนาโดยมหาวิทยาลัยสแตนฟอร์ด ที่ชื่อว่า Stanford Deepdive ที่ทำการนำข้อความ, ตารางหรือรูป (unstructured information) มาทำการวิเคราะห์เนื้อหาเหล่านั้นได้โดยการใช้ Machine Learning เพื่อให้ได้ผลลัพธ์ออกมาสร้างเป็นฐานข้อมูล SQL tables (structured information) เช่น GeoDeepDive ที่สามารถค้นหาข้อมูลทางธรณีวิทยาจากบทความทางวิชาการได้ แต่ว่าโปรแกรม Stanford Deepdive ที่กล่าวมานั้น ถูกพัฒนาขึ้นสำหรับการวิเคราะห์เอกสารในภาษาอังกฤษเป็นหลัก ซึ่งภาษาไทยที่มีรูปแบบของประโยค การจัดเรียงคำ การวางตำแหน่งคำ,ตัวอักษร และอื่นๆ ที่แตกต่างจากภาษาอังกฤษเป็นอย่างมาก ทำให้การทำระบบ tag เอกสารอัตโนมัติสำหรับภาษานั้น ไม่สามารถใช้ Stanford Deepdive ได้ และการ tag หมดหมัให้กับข้อความจำนวนมากโดยใช้มนุษย์ในการจัดการนั้น จะเป็นการเสียเวลาไปเป็นจำนวนมาก ทำให้ทางกลุ่มของเราสนใจที่จะพัฒนาโปรแกรมในลักษณะคล้ายกันกับ Stanford Deepdive ที่สามารถนำมาใช้กับภาษาไทยได้ เพื่อให้เอกสารต่างๆ ที่เป็นภาษานั้น ถูกนำมาใช้ประโยชน์ และนำมาศึกษาต่อได้อย่างมีประสิทธิภาพ เช่น การสร้างฐานข้อมูลที่สามารถสืบค้นได้จากเอกสารที่เกี่ยวข้องกับการพัฒนาการเรียนการสอนของโรงเรียนต่างๆ เพื่อให้ครูสามารถค้นหาเอกสารที่มีความเกี่ยวข้องหรือแนวทางที่ตนสนใจเพื่อจะนำไปใช้ในการพัฒนาการเรียนการสอนของตนเองต่อไป

6. เป้าหมายและขอบเขตของโครงการ

1. สร้างโปรแกรมสำหรับรับ PDF ภาษาไทย เพื่อทำการระบุและแบ่ง paragraph ต่างๆจาก PDF ที่รับมาและจะทำการดึง text ภาษาไทยใน PDF ออกมาจาก paragraph เหล่านั้นออกมาเพื่อทำการ Text processing ต่อ เพื่อให้ได้ผลลัพธ์สุดท้ายเป็น word ที่สำคัญต่างๆ สำหรับนำไปใช้ในการทำ Machine Learning

2. ทำการศึกษาและพัฒนา machine learning model ที่สามารถรับข้อมูล text file ภาษาไทยที่ได้จากขั้นตอนที่ 1 โดยวิธีการ supervised classification ซึ่งจะแบ่งเป็น 2 ขั้นตอนคือ

2.1 การทำ training model จะรับ text และ tag ของ paragraph

2.2 การทำ prediction จาก model โดยจะรับ text เป็น paragraph และแสดง tag เป็นผลลัพธ์และทำการเก็บลงใน Database

3. ทำการพัฒนา web application เพื่อที่จะใช้ในการสืบค้นข้อมูลที่ทำการ tag มาแล้วจากขั้นตอน prediction จากใน Database

รายละเอียดของการพัฒนา

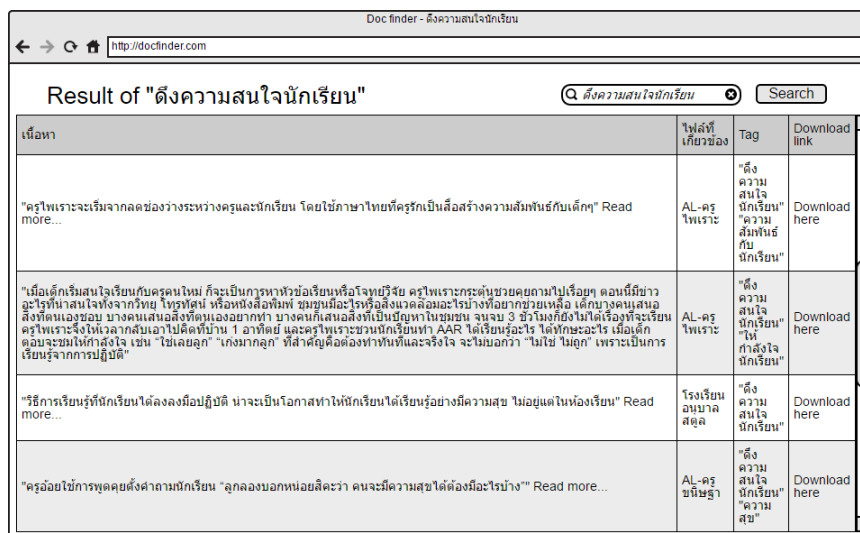
7.1 Story board

เมื่อต้องการที่จะค้นหาเอกสารที่เกี่ยวข้องกับเรื่องใดเรื่องหนึ่งขึ้นมาใช้งาน ผู้ใช้จะสามารถค้นหาข้อมูลได้อย่างรวดเร็วด้วย platform ที่ทางกลุ่มพัฒนาขึ้น โดย platform ที่พัฒนาขึ้นจะแบ่งออกได้เป็น 2 ส่วนหลักๆ

- ส่วนของผู้พัฒนาและผู้ดูแลระบบ โดยผู้พัฒนาจะทำการเตรียมเอกสารที่เป็นไฟล์ PDF เพื่อทำการเตรียม machine learning model ที่สามารถทำการ tag เอกสารได้โดยอัตโนมัติ โดยมีผู้เชี่ยวชาญเฉพาะมาช่วยในการระบุคำสำคัญของเนื้อหาในเรื่องต่างๆ ที่จะนำมาใช้ในการเตรียม model เช่น ถ้าต้องการให้ตัว model สามารถทำการจำแนกเนื้อหาที่เกี่ยวข้องกับเรื่อง “การดึงความสนใจนักเรียน” ผู้พัฒนา/ผู้ดูแลจะต้องเตรียมเอกสารที่มีเนื้อหาที่เกี่ยวข้องกับเรื่องการดึงความสนใจของนักเรียนไว้ และให้ผู้เชี่ยวชาญช่วยระบุว่า มีคำใดบ้างที่สามารถระบุได้ว่า ข้อความนี้มีความเกี่ยวข้องกับการดึงความสนใจของนักเรียน และนำไปทำการเตรียม model

- ส่วนของผู้ใช้งาน ผู้ใช้สามารถเข้ามาใช้งานผ่าน web application ที่ทางกลุ่มพัฒนาขึ้นมา แล้วทำการค้นหาเนื้อหาที่เกี่ยวข้องกับสิ่งที่ผู้ใช้งานต้องการ แล้วเนื้อหาส่วนนั้นก็จะปรากฏขึ้นมา และมีไฟล์เอกสารนั้นให้ผู้ใช้งานสามารถ download ไปอ่านได้ ยกตัวอย่างเช่น ครูสมศรีต้องการที่จะหาข้อมูลเรื่อง “การดึงความสนใจนักเรียน” เพื่อนำไปเตรียมการเรียนการสอนสำหรับชั้นเรียน สิ่งที่คุณครูต้องทำก็คือ ค้นหาด้วยคำว่า “ดึงความสนใจนักเรียน” ในหน้าเว็บ แล้วเว็บก็จะทำการแสดงผลย่อหน้าที่เกี่ยวข้องกับการดึงความสนใจนักเรียน และ tag ที่เกี่ยวข้องกับย่อหน้านั้นๆ โดยแต่ละย่อหน้าก็จะมี tag ที่เกี่ยวข้องเป็นของตัวเอง และมีลิ้งค์สำหรับดาวน์โหลดเอกสารที่มีข้อความนั้นอยู่ให้คลิกเพื่อดาวน์โหลดได้

โดย platform นี้จะมีจุดเด่นที่เราสามารถนำ model นี้ ไปประยุกต์ใช้กับหัวข้ออื่นๆได้ โดยไม่จำเป็นต้องออกแบบโปรแกรมใหม่ทั้งหมด เพียงแค่เตรียมเอกสารที่เกี่ยวข้องและคำสำคัญของหัวข้อนั้นๆ และให้ระบบทำการเรียนรู้ด้วยตัวเอง



The screenshot shows a web browser window with the address bar displaying 'http://docfinder.com'. The page title is 'Doc finder - ดึงความสนใจนักเรียน'. Below the address bar, there is a search bar with the text 'ดึงความสนใจนักเรียน' and a 'Search' button. The main content area displays the 'Result of "ดึงความสนใจนักเรียน"' in a table format. The table has four columns: 'เนื้อหา' (Content), 'ไฟล์ที่เกี่ยวข้อง' (Related files), 'Tag' (Tags), and 'Download link'. There are four rows of results, each containing a snippet of text, a file name, tags, and a download link.

เนื้อหา	ไฟล์ที่เกี่ยวข้อง	Tag	Download link
"ครูโหวตจะเริ่มจากลดช่องว่างระหว่างครูและนักเรียน โดยใช้ภาษาไทยที่ครูรักเป็นสื่อสร้างความสัมพันธ์กับเด็กๆ" Read more...	AL-ครูโหวต	"ดึงความสนใจนักเรียน" "ความสัมพันธ์กับนักเรียน"	Download here
"เมื่อเด็กเริ่มสนใจเรียนกับครูคนใหม่ ก็จะเป็นการหาหัวข้อเรียนหรือโจทย์วิจัย ครูโหวตจะกระตุ้นช่วยตามไปเรื่อยๆ คอยเตือนว่าอะไรที่นำสนใจจากครู โหวตให้ หรือหนังสือพิมพ์ ขอมุมมองอะไรเรื่องสิ่งแวดล้อมอะไรบ้างที่อยากช่วยเหลือ เด็กบางคนเสนอสิ่งที่ตนเองชอบ บางคนเสนอสิ่งที่ตนเองอยากทำ บางคนก็เสนอสิ่งที่ตนสนใจในชุมชน จนจบ 3 ชั่วโมงก็ยังไม่ได้เรื่องที่เรียน ครูโหวตจะรีบให้เวลากลับมาเปิดบ้าน 1 อาทิตย์ และครูโหวตจะชวนนักเรียนทำ AAR ได้เรียนรู้อะไร ได้ทักษะอะไร เมื่อเด็กตอบจะชมให้กำลังใจ เช่น "โอเคลูก" "เก่งมากลูก" ที่สำคัญคือต้องทำทันทีและจริงใจ จะไม่บอกว่า "โอเค ไม่ลูก" เพราะเป็นการเรียนรู้จากการปฏิบัติ"	AL-ครูโหวต	"ดึงความสนใจนักเรียน" "ให้กำลังใจนักเรียน"	Download here
"วิธีการเรียนรู้ที่นักเรียนได้ลงมือปฏิบัติ น่าจะเป็นโอกาสทำให้นักเรียนได้เรียนรู้ด้วยความสุข ไม่อยู่แต่ในห้องเรียน" Read more...	โรงเรียนอานาบาลิสชุด	"ดึงความสนใจนักเรียน"	Download here
"ครูอย่าใช้การพูดสั่งคำถามนักเรียน "ลูกลองบอกหน่อยสิคะว่า คนที่มีความสุขได้ต้องมีอะไรบ้าง"" Read more...	AL-ครูชนิษฐา	"ดึงความสนใจนักเรียน" "ความสุข"	Download here

7.2 เทคนิคหรือเทคโนโลยีที่ใช้

- Natural Language Processing เป็นอัลกอริทึมที่ใช้เพื่อวิเคราะห์ภาษาธรรมชาติของมนุษย์ ด้วยการรับค่าที่เป็นประโยค จากนั้นทำการวิเคราะห์ประมวลผลภาษาธรรมชาติเพื่อให้ได้ผลลัพธ์ในการใช้งานต่อ เช่นการทำ Word Segmentation จะเป็นการแบ่งคำออกจากกันให้เป็นคำที่มีความหมายเพื่อนำไปใช้งานต่อโดยวิธีการเหล่านี้จำเป็นต้องทำงานร่วมกับฐานความรู้อื่นด้วย ดังตัวอย่างการทำ Word Segmentation จำเป็นต้อง Dictionary Database เพื่อช่วยในการแบ่งคำ และการวิเคราะห์ภาษาธรรมชาติยังรวมถึงการทำการประมวลผลความหมายของคำต่าง เช่นการตรวจสอบ Sentiment ของคำต่างๆ อีกด้วย

- Term frequency – Inverse document frequency (TF-IDF) เป็นวิธีทางสถิติที่จะทำการตรวจสอบคำต่างๆในบทความเพื่อนำไปเปรียบเทียบกับบทความทั้งหมด เพื่อหาอัตราส่วนว่าคำๆนี้มีความสำคัญต่อบทความโดยรวมแค่ไหน โดย TF-IDF จะแบ่งขั้นตอนเป็น 2 ส่วนคือ Term frequency โดยในขั้นตอนนี้นั้นจะทำการนับจำนวนครั้งที่คำต่างๆปรากฏในบทความหนึ่งๆ และการทำ Inverse document frequency โดยในขั้นตอนนี้จะเป็นการนำคำต่างๆในบทความมาเปรียบเทียบกับบทความทั้งหมดและคำนวณหาค่าน้ำหนักความสำคัญนั้นๆ จากบทความทั้งหมด โดยการทำ TF-IDF นั้นสามารถใช้ประโยชน์ในการหาคำสำคัญในบทความต่างๆซึ่งสามารถนำไปประยุกต์ใช้ได้อย่างหลากหลายเช่นการทำ Search engine หรือการทำ Text Summarization
- Machine Learning: Classification เป็นวิธีในการจำแนกผลลัพธ์จากข้อมูล input โดยจะแบ่งขั้นตอนการใช้งานเป็น 2 ขั้นตอนคือการ Training Model และ Testing Model โดย การ Training Model นั้น จะทำการรับ Input จำนวนมากเพื่อสร้าง Model จาก Input เหล่านั้น และการทำ Testing Model จะเป็นการนำ Input ที่เราต้องการใช้ มาผ่าน Training Model ในขั้นตอนข้างต้นเพื่อให้ระบบประมวลผลและจำแนกผลลัพธ์ออกมา โดยการทำ Classification มีวิธีการที่หลากหลาย ซึ่งในที่นี้ เราจะเลือกใช้ Multiclass Classification เช่น One-vs-Rest, neural network และ Decision Tree

7.3 เครื่องมือที่ใช้ในการพัฒนา

- Hadoop Distributed File System (HDFS) เป็นระบบการจัดเก็บข้อมูลที่ออกแบบมาสำหรับการจัดการข้อมูลขนาดใหญ่ (Big data) โดย HDFS ถูกออกแบบมาสำหรับระบบที่มีคอมพิวเตอร์หลายๆ ตัวช่วยกันประมวลผล และ HDFS จะเหมาะกับการทำงานในลักษณะ “Write once, Read many” หรือข้อมูลที่เน้นการอ่านข้อมูลมากกว่าการเขียน,แก้ไข โดยการทำงานของ HDFS จะแบ่งคอมพิวเตอร์ที่อยู่ในระบบเป็น 2 ส่วนด้วยกัน ได้แก่ Name node และ Data node โดยเมื่อต้องการที่จะจัดเก็บไฟล์ลงในระบบ HDFS ไฟล์ที่เข้ามาจะถูกแบ่งออกเป็นหลายๆ ส่วนและแบ่งไปเก็บไว้ยัง data node แต่ละเครื่อง ส่วน name node จะมีหน้าที่เก็บข้อมูลว่า ไฟล์นั้นๆ ถูกแบ่งออกเป็นกี่ส่วน แต่ละส่วนถูกเก็บไว้ที่คอมพิวเตอร์เครื่องใดบ้าง และเมื่อต้องการนำไฟล์ออกมาใช้งาน name node จะทำการสั่งงานให้ data node ทุกเครื่องที่มีส่วนของไฟล์นั้นๆ ทำการอ่านส่วนของไฟล์ที่เก็บไว้ส่งไปยัง name node และตัว name node จะทำการรวมส่วนของไฟล์ทั้งหมดที่ได้รับมาให้เป็นไฟล์เดียวกัน เพื่อให้สามารถใช้งานไฟล์นั้นได้ต่อไป

- โปรแกรมสำหรับการแปลงไฟล์ในรูปแบบ PDF ให้เป็น text file โดยทางกลุ่มเลือกใช้ PDFMiner ซึ่ง PDFMiner เป็น Python API ที่ใช้สำหรับการดึงข้อมูลต่างๆออกมาจาก PDF Document เช่น ตัวอักษรในภาษาต่างๆ เช่น ไทย อังกฤษ จีน และอื่นๆ หรือสามารถดึงภาพออกจาก PDF ได้ โดยสำหรับโปรเจกต์นี้จะเน้นที่การดึงข้อความออกจาก PDF Document เพื่อสำหรับนำไป preprocess ต่อ ซึ่งฟังก์ชันที่ใช้ในการดึงข้อความออกมานั้นคือ PDF2TXT โดยคำสั่งต่างๆของ ฟังก์ชันนี้ สามารถเลือก page number, ชนิดของ output (text,tag,xml), ขนาดของ box ของ คำใน pdf เป็นต้น
- โปรแกรม Spark ML เป็น library ที่มีอยู่ในโปรแกรม Apache Spark ซึ่ง Spark ML เป็น library ที่ใช้ทำ Machine Learning โดยที่สามารถทำงานแบบขนาน (Parallel programming) ได้ ซึ่ง Apache Spark เป็น engine สำหรับการทำการประมวลผลข้อมูลขนาดใหญ่ (Big data processing) ที่สามารถทำงานได้อย่างรวดเร็ว เนื่องจากการประมวลผลในหน่วยความจำหลัก (In-memory processing) ทำให้การเข้าถึงข้อมูลทำให้รวดเร็วมากขึ้น
- โปรแกรม LexTo เป็นโปรแกรมที่ถูกพัฒนาด้วยภาษา Java โดยโปรแกรมนี้สามารถใช้ในการแบ่ง คำต่างๆในภาษาไทยจากประโยคให้กลายเป็นคำซึ่งแบ่งด้วย delimiter ซึ่งคำต่างๆที่ใช้ในการแบ่ง นั้น จะมี Dictionary ที่จะทำการเก็บคำทั้งหมดเอาไว้ แล้วโปรแกรมจะนำมาเปรียบเทียบเพื่อแบ่ง คำตามที่ Dictionary ได้กำหนดไว้
- โปรแกรม Apache Impala เป็นโปรแกรมจัดการฐานข้อมูลแบบ Open source ที่ถูกออกแบบ มาให้ใช้งานร่วมกับ Hadoop ecosystem โดย Impala จะเหมาะกับการเก็บข้อมูลที่ต้องการ นำมาวิเคราะห์แบบรวดเร็ว เนื่องจากตัวโปรแกรมมี latency ต่ำและมี throughput ที่สูง และยัง มีความสามารถในการเพิ่มประสิทธิภาพของระบบได้ง่าย (Scalable)

7.4 รายละเอียดโปรแกรมที่จะพัฒนา

Input/Output Specification

ส่วนของการ train ระบบ

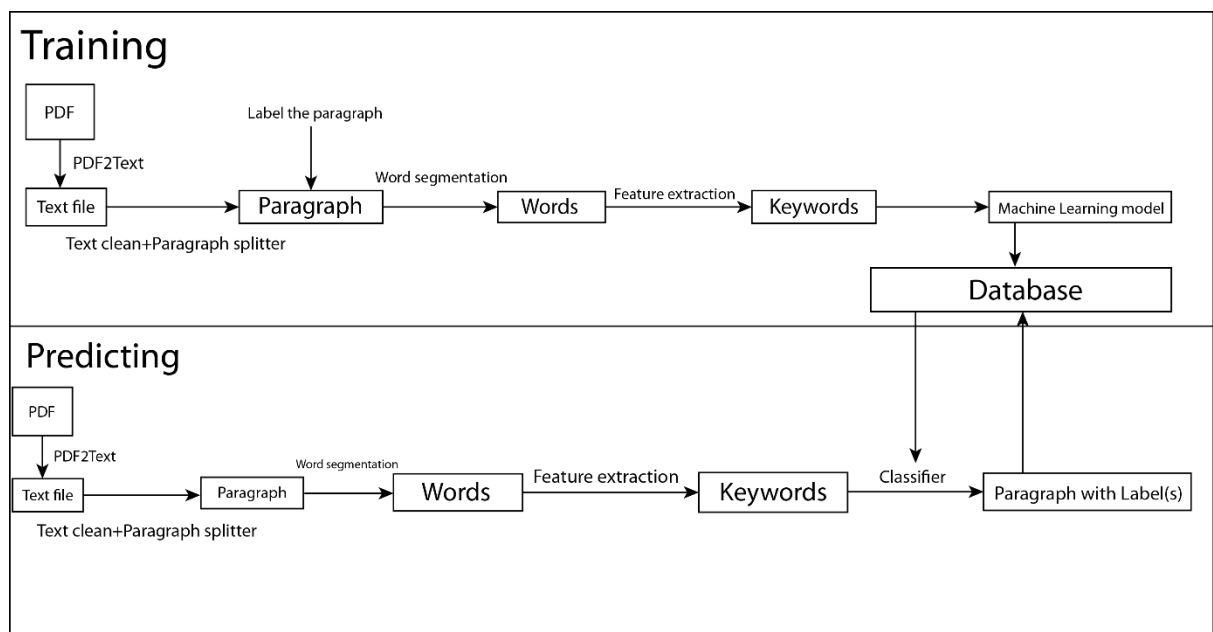
- Input เป็น paragraph ที่ถูก label เนื้อหาที่เกี่ยวข้องไว้แล้ว
- Output เป็น model ที่สามารถทำการคาดเดา label ของข้อความอื่นๆ

ส่วนของการคาดเดา label

- Input เป็น paragraph ที่ต้องการให้ระบบทำการ label เนื้อหา
- Output เป็น label ทั้งหมดที่เกี่ยวข้องกับข้อความนั้นๆ

ส่วนของ web application

- Input เป็น keyword ที่ต้องการนำไปค้นหา
- Output เป็นตัวอย่าง paragraph ที่เกี่ยวข้องกับ Keyword นั้นๆ และ Tag ต่างๆที่อยู่ใน Paragraph นั้นๆพร้อมลิงสำหรับการ download เอกสารฉบับสมบูรณ์



Functional Specification

Text Preprocessing

- ทำการแปลง PDF ให้เป็น xml file โดยการใช้ PDFMiner
- ทำการ preprocess xml file ที่ได้ โดยจะทำการระบุดูย่อหน้าจาก xml schema, ทำการตัด xml schema ต่างๆและดึงคำภาษาไทยออกมาและทำการแก้ภาษาไทยที่ผิดพลาดต่างๆจาก xml เช่น สระ อ่า (อ ่า เป็น อ่า)
- ทำการตัดคำจาก text file ด้วยโปรแกรม word segmentation สำหรับภาษาไทย เช่น LexTo
- ทำการกำจัดคำที่ไม่จำเป็นออกโดยการใช้ stop word remover

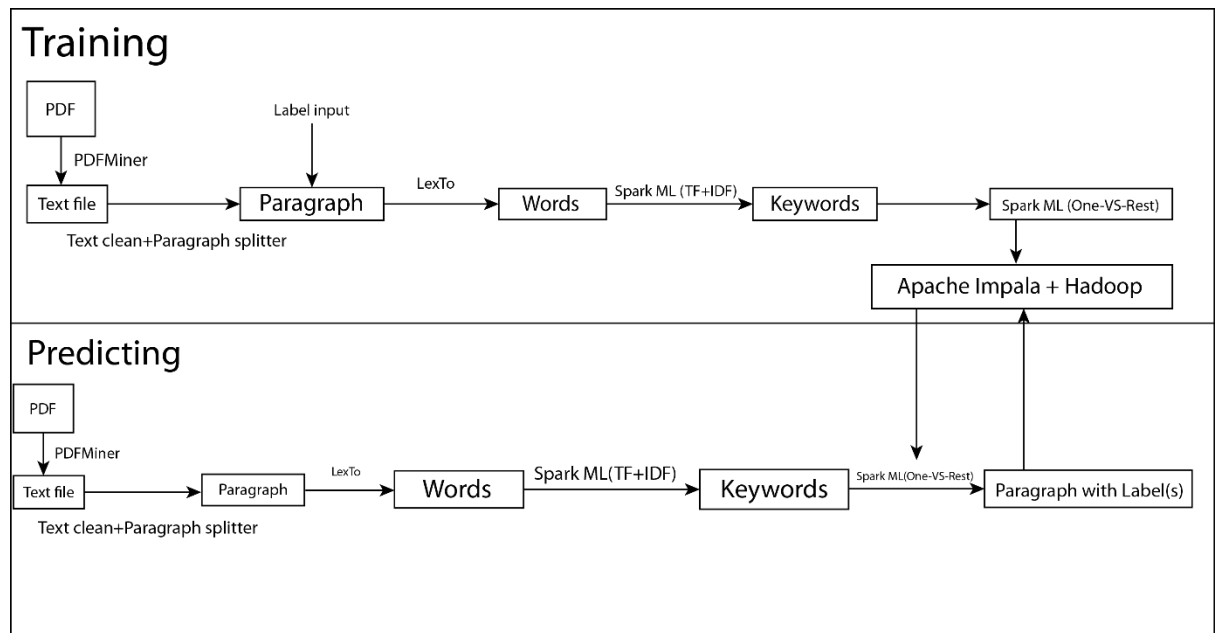
Classification: Training model

- ในแต่ละย่อหน้าจะมี tag ทั้งหมดที่เกี่ยวข้องกับย่อหน้านั้น ซึ่งทำการระบุด้วยมนุษย์
- สร้าง bag-of-words model สำหรับเก็บคำภาษาไทย และแปลงคำต่างๆจากในย่อหน้าให้กลายเป็น vector
- ทำการหา Feature ของคำแต่ละคำ โดยการหาความถี่การใช้คำในข้อความแต่ละอัน เทียบกับ tag ที่มีอยู่
- นำ Feature ไปให้ machine learning เรียนรู้

Classification: Prediction

- แปลงคำที่รับมาให้กลายเป็น vector สำหรับใช้ใน machine learning model จาก bag-of-words model ที่สร้างไว้
- ทำการหา Feature ของคำแต่ละคำ โดยการหาความถี่การใช้คำในข้อความแต่ละอัน เทียบกับ tag ที่มีอยู่
- นำ Feature ไปใส่ใน machine learning model ที่ได้ทำการ train ไว้ในข้างต้น เพื่อให้ได้ tag สำหรับ paragraph นั้น
- นำ tag ที่ได้และ paragraph ไปเก็บลงในฐานข้อมูล

Architecture



7.5 ขอบเขตและข้อจำกัดของโปรแกรมที่พัฒนา

- เนื่องจากภาษาไทยเป็นภาษาที่มีความซับซ้อนสูง ทั้งทางด้านตัวอักษร ที่มีสระบน-ล่าง และทางด้านรูปประโยคที่ไ้กับความแน่นอน ทำให้การเขียนโปรแกรมที่สามารถประมวลผลภาษาไทยได้อย่างสมบูรณ์แบบจึงเป็นเรื่องยาก ทำให้ความแม่นยำในการ tag และเลือกย่อหน้าที่มีความสำคัญกับเรื่องที่เลือก อาจจะต่ำกว่าการใช้งานกับภาษาอังกฤษ ที่มีรูปประโยคที่แน่นอนกว่า ทำให้สามารถใช้การดูรูปประโยคเข้ามาช่วยเสริมความหมายของคำได้ ซึ่งเป็นสิ่งที่ทำได้ยากมากในภาษาไทย
- ข้อมูลที่จะนำไปเข้าระบบ machine learning เพื่อให้ระบบทำการเรียนรู้ด้วยตนเองนั้น จะต้องใช้มนุษย์เป็นตัวช่วยในการกำหนดข้อมูลก่อนในเบื้องต้น เพราะฉะนั้น ถ้าเราต้องการให้ระบบเรียนรู้เนื้อหาเรื่องใหม่ๆ จะต้องมีการใช้ผู้เชี่ยวชาญที่เกี่ยวข้องกับเรื่องที่จะให้ระบบเรียนรู้มาช่วยทำการ label คำสำคัญก่อนที่จะนำข้อมูลเข้าไปในระบบ ดังนั้น ถ้าเกิดเราไม่สามารถหาผู้ที่จะสามารถระบุคำสำคัญให้ได้ เราก็จะไม่สามารถทำให้ระบบเรียนรู้หัวข้อใหม่ๆ ได้
- การระบุย่อหน้าจาก PDF นั้นสามารถทำได้ยากเนื่องจากการจะระบุย่อหน้าจาก PDF จำเป็นต้องใช้คำตำแหน่งของตัวอักษรต่างๆ เพื่อนำระบุว่าย่อหน้าควรจะอยู่ตำแหน่งไหน ซึ่ง PDF ที่ได้รับมานั้น มีรูปแบบการจัดหน้าและ font ที่แตกต่างกันรวมถึงรูปแบบคำภาษาไทยและภาษาอังกฤษใน

เอกสาร จะทำให้ตำแหน่งของคำเกิดการคลาดเคลื่อนซึ่งจะส่งผลให้ย่อหน้าที่ได้ออกมาอาจเกิดความผิดพลาดได้

บรรณานุกรม

- มาเริ่มเรียนรู้ Hadoop กันหน่อย, <http://www.somkiat.cc/start-with-hadoop/> (Accessed 2016-9-23)
- Apache Spark, <http://spark.apache.org> (Accessed 2016-9-23)
- Pdfminer, <http://euske.github.io/pdfminer/index.html>
- PDFMiner, <http://www.unixuser.org/~euske/python/pdfminer/>
- One-vs-Rest classifier, <https://spark.apache.org/docs/latest/ml-classification-regression.html#one-vs-rest-classifier-aka-one-vs-all>
- Lexto , <http://www.sansarn.com/lexto/>
- Latent Dirichlet allocation(LDA), <https://spark.apache.org/docs/1.6.0/ml-clustering.html#latent-dirichlet-allocation-lda>
- Impala, <http://impala.apache.org/>