**รหัสโครงการ 19p34c0414**

**แพลตฟอร์มสำหรับรวบรวมและแบ่งป​นความรู้ โดยวิธีการประมวลผลข้อความและดึงองค์ความรู้จากเอกสาร**

**การแข่งขันสุดยอดโปรแกรมป​ญญาประดิษฐ์ (Artificial Intelligence Application)**

**รายงานฉบับสมบูรณ์**

**เสนอต่อ**

**ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ**

**สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ**

**กระทรวงวิทยาศาสตร์และเทคโนโลยี**

**ได้รับทุนอุดหนุนโครงการวิจัย พัฒนาและวิศวกรรม**

**โครงการแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทย ครั้งที่ 19**

**ประจำปีงบประมาณ 2559**

**โดย**

**นายศุภณัฐ ทัตตินาพานิช**

**นายอินทัช แสงกระจ่าง**

**นายปุญญพัฒน์ เศรษฐ์สมบูรณ์**

**อาจารย์ที่ปรึกษา รศ.ดร.ธีรณี อจฉลากุล**

**ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรีวิทยาเขตบางมด**

**กิตติกรรมประกาศ**

โครงการแพลตฟอร์มสำหรับรวบรวมและแบ่งป​นความรู้ โดยวิธีการประมวลผลข้อความและดึงองค์ความรู้จากเอกสาร (Building a knowledge sharing platform with Text mining and knowledge extraction) สำเร็จลุล่วงได้ด้วยการสนับสนุน จากโครงการการแข่งขันพัฒนาโปรแกรมคอมพิวเตอร์แห่งประเทศไทย ครั้งที่ 19 โดยศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติสำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ กระทรวงวิทยาศาสตร์และเทคโนโลยี และสำนักงานส่งเสริมอุตสาหกรรมซอฟแวร์แห่งชาติ และขอขอบคุณมูลนิธิสดศรี-สฤษดิ์วงศ์ โดยการประสานของของพี่หญิง สมหญิง สายธนู ผู้ซึ่งให้ความช่วยเหลือทั้งทางด้านเอกสารที่นำมาใช้ คุณครูที่มาช่วยทั้งในด้านการจัดกลุ่มเอกสารและด้านอื่นๆ และทางสถาบันอาศรมศิลป์ โดยการประสานงานของคุณอภิษฎา ทองสอาด หรือพี่ปุ้ม ที่ช่วยให้คำแนะนำในด้านเอกสาร และให้เอกสารเพิ่มเติมจากทางมูลนิธิสดศรี-สฤษดิ์วงศ์

ขอขอบพระคุณ รองศาสตร์ตราจารย์ ดร.ธีรณี อจฉลากุล เป็นอาจารย์ที่ปรึกษาโครงการ ที่ให้คำปรึกษา คำแนะนำ และสนับสนุนอุปกรณ์ในการทำโครงการนี้

ขอขอบพระคุณ ผู้ช่วยศาสตร์ตราจารย์ สนั่น สระแก้ว เป็นหัวหน้าภาควิชา และ คณะทำงานภาควิชาวิศวกรรมคอมพิวเตอร์ทุกคน ที่อำนวยความสะดวก และสถานที่ในการทำโครงงานนี้

นายศุภณัฐ ทัตตินาพานิช

นายอินทัช แสงกระจ่าง

นายปุญญพัฒน์ เศรษฐ์สมบูรณ์

# บทคัดย่อ

การศึกษา ถือเป็นรากฐานที่สำคัญในการพัฒนาทรัพยากรมนุษย์ และช่วยในการพัฒนาประเทศชาติอย่างยั่งยืน ซึ่งในประเทศไทยนั้น รัฐบาลมีการให้ทุนสนับสนุนเพื่อการศึกษาค้นคว้า ทดลองหาวิธีการต่างๆ   
ที่จะช่วยเพิ่มประสิทธิภาพของระบบการศึกษาให้ดีขึ้น โดยผลที่ได้จากการวิจัยหรือทดลองนี้ จะถูกเขียนออกมาเป็นรูปเล่มรายงานและเก็บในรูปแบบของไฟล์ PDF หรือ DOCX ที่ประกอบไปด้วย รายงานสำหรับผู้บริหารที่แสดงถึงตัวเลขสถิติต่างๆ และรายงานสำหรับการเรียนการสอน ที่อธิบายรายละเอียดแนวทางการเรียนการสอน เป็นต้น อย่างไรก็ตาม เอกสารเหล่านี้ไม่ได้มีการเผยแพร่ในวงกว้าง และถูกเก็บในรูปแบบไฟล์เอกสารเหล่านั้น ที่ไม่สามารถสืบค้นได้ จึงไม่สามารถถูกนำมาใช้ประโยชน์ได้อย่างเต็มที่ การสร้าง knowledge sharing platform สำหรับคุณครู ผู้ปกครอง และผู้บริหาร ที่สามารถสืบค้นหาข้อมูลได้อย่างสะดวกจึงเป็นเรื่องจำเป็น เพื่อให้การศึกษากระจายตัวได้อย่างทั่วถึง ข้อเสนอโครงการฉบับนี้จึงเสนอแนวคิดในการใช้เทคโนโลยีเพื่อรวบรวม คัดกรอง จัดหมวดหมู่ รวมถึงสกัดเนื้อหาส่วนที่สำคัญจากเอกสาร เพื่อให้เอกสารการทดลองด้านนวัตกรรมการเรียนการสอนสามารถถูกนำไปใช้ประโยชน์ และสร้างสังคมของการแลกเปลี่ยนเรียนรู้ของคุณครูและผู้ปกครองได้

ทั้งนี้ ข้อมูลที่อยู่ในไฟล์ PDF หรือไฟล์ DOCX นั้นเป็นข้อมูลที่ไม่มีโครงสร้าง (Schemaless) ไม่สามารถนำมาประมวลผลเพื่อวิเคราะห์ด้วยซอฟท์แวร์ทั่วๆไปได้ จึงจำเป็นต้องมีการพัฒนาระบบจัดหมวดหมู่  
องค์ความรู้แบบอัตโนมัติขึ้น โดยอาศัยเทคโนโลยีทางด้าน Text Mining และเทคโนโลยีทางด้าน Machine Learning เข้ามาช่วยเพื่อให้สามารถใช้งานกับรายงานจำนวนมากจากทั่วประเทศได้อย่างมีประสิทธิผล

ในการพัฒนาระบบในระยะเริ่มต้นนั้น จะมีคุณครูอาสาสมัครจากทางมูลนิธิสดศรี-สฤษดิ์วงศ์เข้ามาช่วยอ่านและทำการระบุข้อความส่วนที่เป็นเนื้อหาใจความสำคัญของรายงานนั้นๆ และสร้าง Tag เพื่อบ่งบอกหัวเรื่องของเนื้อหา เพื่อใช้สำหรับการจัดหมวดหมู่ โดยคุณครูจะช่วยวิเคราะห์รายงานเพียงส่วนน้อยเท่านั้น หลังจากนั้นเนื้อหาและ Tag ที่คุณครูสร้างขึ้นจะถูกนำมาพัฒนา Machine Learning Model โดย Model จะถูกสอนให้เรียนรู้คำต่างๆที่เกี่ยวข้องกับ tag ที่อยู่ในรายงาน เมื่อมีรายงานเล่มใหม่เข้ามาในระบบ ระบบจะทำการวิเคราะห์เนื้อหาและสามารถแสดงส่วนที่เป็นใจความสำคัญ รวมถึงจัดประเภทหมวดหมู่ของรายงานได้โดยอัตโนมัติ ซึ่งช่วยลดการใช้เวลาและทรัพยากรบุคคล เมื่อผู้ใช้งานเข้ามาใช้ระบบนี้ จะสามารถสืบหาข้อมูลที่เกี่ยวข้องกับการพัฒนาการเรียนการสอน ด้วยการใส่ Keyword จากนั้น web application จะให้ผลลัพธ์ออกมาเป็นข้อความที่เกี่ยวข้องพร้อมทั้งแนบลิงค์สำหรับดาวน์โหลดเอกสาร ผู้ใช้งานสามารถอ่านสรุปใจความสำคัญที่ระบบแสดงก่อน และหากตรงกับความสนใจสามารถดาวน์โหลดรายงานทั้งเล่มไปเพื่อศึกษารายละเอียดต่อไป

# Abstract

The education is an important part of improving Thai citizens’ quality. It’s a main tool for creating creativity to children for a sustainable country improvement. In Thailand, Thai government gives funds for researching in improving an education in school and the research data are stored in digital document files such as PDF or DOCX files. The contents are about statistic, teaching techniques and so on. By the way, these files are not publicly published and cannot be searched due to their file type. So, if we created a knowledge sharing platform that can be used widely would be so helpful on making low quality schools performing better. This project provides a technology for collecting, analyzing, categorizing and extracting important parts of documents to make them searchable and be useful for educational society.

But, documents that stored in those formats are chunks of data that have no structure (schemaless). Therefore, these files cannot be analyzed by an ordinary software. So we decide to make an automatic document categorize system by using machine learning and text mining techniques to put each part of documents into groups. By using machines to categorize documents, we will be able to handle and categorize lots of documents from schools in Thailand.

In the first step of development, we need experts from Sodsri – Saritwong Foundation to read and tag cores of some documents into categories. This step will make a training data for system and the system will learn from this training data and create a machine learning model that learned about keywords for each category. Then the system will automatically tag each part of contents of documents that has been ingested into them. This machine learning system will reduce workload of humans a lot. Then we will create a web application that can search contents that are about a user’s searching keyword. A user can read an important part of each document and can download a document from a web for more details.

## คำสำคัญ

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

# บทนำ

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 ซึ่งรายงานเหล่านี้ล้วนเป็นตัวบ่งชี้ให้เห็นว่าการศึกษาของไทยยังอ่อนแอ และมีจุดบกพร่องเป็นอย่างมาก ควรที่จะต้องได้รับการพัฒนาอย่างเร่งด่วน

หนึ่งในปัจจัยสำคัญที่มีผลต่อการพัฒนาการศึกษา คือการสอนของคุณครู เพราะคุณครูเปรียบเสมือนผู้ที่ถ่ายทอดความรู้ต่างๆและพัฒนาเด็กให้เติบโตไปเป็นทรัพยากรมนุษย์ที่มีคุณภาพ ดังนั้น รัฐบาลไทยจึงมีการให้ทุนสนับสนุนกับครูในการศึกษาค้นคว้า ทดลองหาวิธี ในการพัฒนาการเรียนการสอนให้มีประสิทธิภาพมากยิ่งขึ้น และเข้าถึงเด็กนักเรียนได้มากขึ้น โดยเฉพาะการศึกษาขั้นพื้นฐานในระดับประถามและมัธยมศึกษา ซึ่งในปัจจุบัน ได้มีเอกสารที่ถูกเขียนออกมาเพื่อรายงานผลการทดลอง และวิธีในการพัฒนาการเรียนการสอนที่ดี (best practice) ซึ่งรายงานเหล่านี้มักจะประกอบไปด้วยจำนวนหน้าที่มาก และอยู่ในรูปแบบของไฟล์ PDF หรือ DOCX ทำให้ครูสามารถสืบค้นข้อมูลได้ยาก และต้องเสียเวลาในการอ่านหนังสือหลายร้อยหน้าจำนวนหลายเล่มเพราะ เอกสารไม่มีการรวบรวมและจัดเป็นหมวดหมู่ 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

# สารบัญ

กิตติกรรมประกาศ ก

บทคัดย่อ ข

Abstract ค

บทนำ ค

สารบัญ ง

สารบัญรูปภาพ ฉ

วัตถุประสงค์และเป้าหมาย 1

รายละเอียดการพัฒนา  **5**

Story Board 5

ทฤษฎีหลักการและเทคนิคหรือเทคโนโลยีที่ใช้ 7

เครื่องมือที่ใช้ในการพัฒนา

รายละเอียดโปรแกรมที่ได้พัฒนาเชิงเทคนิค (Software Specification)

Component Diagram

2.3 ภาษา, เครื่องมือ และซอฟต์แวร์ที่ใช้ในการพัฒนา 8

2.3.1

9 ขอบเขตและข้อจำกัดของโปรแกรม

กลุ่มผู้ใช้โปรแกรม

ผลของการทดสอบโปรแกรม

ปัญหาและอุปสรรค

แนวทางในการพัฒนาและประยุกต์ใช้งานอื่นๆ ในขั้นต่อไป

ข้อสรุปและข้อเสนอแนะ

เอกสารอ้างอิง

สถานที่ติดต่อของผู้พัฒนาและอาจารย์ที่ปรึกษา โทรศัพท์ มือถือ โทรสาร E-mail

ภาคผนวก

คู่มือการติดตั้งอย่างละเอียด

คู่มือการใช้งานอย่างละเอียด

# วัตถุประสงค์และเป้าหมาย

เป้าหมายของโครงการสร้างระบบจัดหมวดหมู่องค์ความรู้แบบอัตโนมัติด้วยเทคโนโลยี Text Mining และ Machine Learning คือการสร้าง knowledge sharing platform ที่จะช่วยรวบรวมข้อมูลรายงานตัวอย่างการเรียนการสอนที่ดี (best practice) มาทำการคัดแยก จัดหมวดหมู่ เพื่อให้คุณครูสามารถสืบค้นข้อมูลได้อย่างมีประสิทธิภาพและรวดเร็ว สามารถนำความรู้ไปพัฒนาและปรับใช้ตามเพื่อให้เกิดการเรียน  
การสอนที่ดีที่จะช่วยพัฒนาศักยภาพของเด็กนักเรียนได้ โดย platform มีขอบเขต ดังนี้

* สร้างเครื่องมือสำหรับรับไฟล์เอกสารและดึงข้อความภาษาไทยในแต่ละย่อหน้าออกมา โดยข้อความจะถูกประมวลผล Text และใส่เข้าใน Machine Learning Model เพื่อติด Tag ที่เกี่ยวข้องกับเนื้อหาต่อไป
* สร้าง Machine learning model ที่สามารถรับข้อมูลจากไฟล์ text ภาษาไทยจากขั้นตอนข้างต้น และทำการติด tag สำหรับแต่ละย่อหน้า โดยวิธีการ supervised classification ซึ่งจะแบ่งเป็น 2 ขั้นตอนคือ
  + เครื่องมือสำหรับการ train model โดยจะรับ text และ tag ของย่อหน้าจากผู้เชี่ยวชาญและนำมาปรับจนได้โมเดลทางคณิตศาสตร์ที่เหมาะสม
  + เครื่องมือสำหรับการทำนายหรือการติด tag อัตโนมัติด้วยโมเดลทางคณิตศาสตร์ที่พัฒนาขึ้นจะรับข้อความภาษาไทยเป็น input และทำการติด tag ให้ย่อหน้าต่างๆที่อธิบายประเด็นสำคัญของหนังสือ/รายงานแต่ละเล่ม ผลลัพธ์จะถูกเก็บลงในฐานข้อมูลเพื่อการสืบค้นต่อไป
* สร้าง Web application โดยมีคุณสมบัติหลักคือ
  + สามารถรับไฟล์รายงานในรูปแบบไฟล์ Text, PDF, DOCX เพิ่มเติมจากผู้ใช้ และนำไปประมวลผลโดยใช้ Machine Learning Model ซึ่งจะทำการติด tag โดยอัตโนมัติและเก็บลงฐานข้อมูล
  + สามารถสืบค้นข้อมูลตัวอย่างการเรียนการสอนที่ดี โดยผ่านการพิมพ์ข้อความหรือคีย์เวิร์ดลงในช่อง Search หรือผ่านการคลิกเลือกจากเมนู Advance Filter
  + สามารถแสดงผลการสืบค้นในหลายระดับ คือประเภทเอกสาร หมวดหมู่ของ tag และตัวอย่างข้อความที่สำคัญ นอกจากนี้ผู้ใช้ระบบจะสามารถดาวน์โหลดไฟล์ต้นฉบับออกจากระบบได้หากต้องการ

# รายละเอียดของการพัฒนา

## Story board

เมื่อต้องการที่จะค้นหาเอกสารที่เกี่ยวข้องกับเรื่องใดเรื่องหนึ่งขึ้นมาใช้งาน ผู้ใช้จะสามารถค้นหาข้อมูลได้อย่างรวดเร็วด้วย platform ที่ทางกลุ่มพัฒนาขึ้น โดย platform ที่พัฒนาขึ้นจะแบ่งออกได้เป็น 2 ส่วนหลักๆ

***- ส่วนของผู้พัฒนาและผู้ดูแลระบบ*** โดยผู้พัฒนาจะทำการเตรียมเอกสารที่เป็นไฟล์เอกสารตัวอย่าง ซึ่งเอกสารเหล่านี้จะมีผู้เชี่ยวชาญเฉพาะมาช่วยในการระบุคำสำคัญต่างๆเพื่อทำการเตรียม machine learning model โดยหลังจากที่ทำการสร้าง model เสร็จแล้ว เอกสารที่เหลือจะทำการ tag เอกสารได้โดยอัตโนมัติ โดยใช้ machine learning model ข้างต้น เช่น ถ้าต้องการให้ตัว model สามารถทำการจำแนกเนื้อหาที่เกี่ยวข้องกับเรื่อง “Problem based learning” ผู้พัฒนา/ผู้ดูแลจะต้องเตรียมเอกสารที่มีเนื้อหาที่เกี่ยวข้องกับเรื่อง Problem based learning และให้ผู้เชี่ยวชาญช่วยระบุว่า มีคำใดบ้างที่สามารถระบุได้ว่า ข้อความนี้มีความเกี่ยวข้องกับ " Problem based learning" และนำไปทำการเตรียม model โดยหลังจากสร้าง model เสร็จแล้ว ผู้พัฒนาสามารถนำเอกสารที่เกี่ยวข้องกับ "Problem based learning" มาทำการ tag เอกสารโดยอัตโนมัติได้

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

โดย platform นี้จะมีจุดเด่นที่เราสามารถนำ model นี้ ไปประยุกต์ใช้กับหัวข้ออี่นๆได้ 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, สรุปของบทความ หรือการสร้าง hashtag จาก model ของทางระบบที่สร้างไว้ ซึ่งโดยส่วนใหญ่ของโปรแกรมเหล่านี้ จะรองรับสำหรับภาษาในภาษาอังกฤษหรือภาษาที่รากศัพท์มาจากภาษาละติน เนื่องจากมี Library ในการจัดการทางภาษาศาสตร์จาก NLP Stanford

สำหรับเครื่องมือและเทคโนโลยีต่างๆที่ทางกลุ่มได้เลือกใช้ในขั้นตอนต่างๆ จะแบ่งเป็นการทำ Word Segmentation ซึ่งในปัจจุบันมีโปรแกรมสำหรับตัดคำภาษาไทยต่างๆได้แก่ LexTo เป็นโปรแกรมในการจัดคำที่จะใช้วิธี 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

## ทฤษฎีหลักการและเทคนิคหรือเทคโนโลยีที่ใช้

* **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 โดยในขั้นตอนนี้จะทำการนับจำนวนครั้งที่คำต่างๆปรากฎในบทความหนึ่งๆ และ  
  การทำ 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 ที่ใส่เข้าใป ซึ่งจะมีการเรียกใช้ vector ของคำที่ได้จากการทำ bag-of-word model   
  มาทำการหาความถี่ของคำเทียบกับเอกสารต่างๆ และทำการแปลงสร้าง model ความเกี่ยวข้องของ  
  คำต่างๆ เทียบกับเอกสารอื่นๆที่ได้ทำการเรียนรู้ เพื่อค้นหา Keyword ที่สำคัญสำหรับนำไปใช้งานต่อ ซึ่ง LDA นั้นจะมองเอกสารเป็นการรวมกันของ topics ต่างๆที่ซ่อนอยู่ โดยแต่ละ topic จะมีค่า   
  คำต่อความน่าจะเป็น 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

## 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 Apache Impala** [11]เป็นโปรแกรมจัดการฐานข้อมูลแบบ SQL ที่เป็น Open source ที่ถูกออกแบบมาให้ใช้งานร่วมกับ Hadoop ecosystem ซึ่งมีระบบการจัดการฐานข้อมูลที่ยืดหยุ่น ไม่ว่าจะใช้งานผ่านหน้า system shell หรือเรียกใช้งานผ่าน JDBC ซึ่งสาเหตุที่ทางกลุ่มเลือกใช้ Impala คือ เป็นโปรแกรมฐานข้อมูลที่รองรับการเก็บข้อความภาษาไทย สามารถทำงานได้อย่างรวดเร็ว เนื่องจากมีการทำงานในลักษณะ in-memory และมีความสามารถในการจัดการฐานข้อมูลเมื่อมีขนาดใหญ่มากๆ ได้
* **โปรแกรม unzip** เป็นโปรแกรมที่ที่ใช้การจัดการ compressed file ต่างๆ ใน linux operating system โดยโปรแกรมนี้สามารถนำมาใช้ในการแปลงไฟล์เอกสารที่เป็นรูปแบบ .docx ให้อยู่ในรูปแบบไฟล์ข้อความธรรมดา (text file) ได้ โดยสาเหตุที่เลือกใช่โปรแกรมนี้ในการแปลงไฟล์ docx ให้เป็นไฟล์เอกสารนั้น เนื่องจาก โปรแกรมนี้เป็นโปรแกรมที่มีอยู่ใน linux อยู่แล้ว และสามารถแปลงไฟล์เอกสารออกมาได้ในผลลัพธ์ที่น่าพอใจ
* **โปรแกรม PDFBox** [13] เป็นโปรแกรมสำหรับการแปลงไฟล์ในรูปแบบ PDF ให้เป็น text file ซึ่ง PDFBox เป็น Java API ที่ใช้สำหรับการดึงข้อมูลต่างๆออกมาจาก PDF Document เช่นตัวอักษรในภาษาต่างๆ เช่น ไทย อังกฤษ จีน และอื่นๆ โดยสำหรับโปรเจคนี้จะเน้นที่การดึงข้อความออกจาก PDF Document เพื่อสำหรับนำไป preprocess ต่อ ซึ่งฟังก์ชั่นที่ใช้ในการดึงข้อความออกมานั้นคือ ExtractText โดยเหตุผลที่เราเลือกใช้ PDFBox คือ การที่ตัวโปรแกรมสามารถแปลงไฟล์ PDF ที่เป็นภาษาไทยได้ค่อนข้างดีกว่าโปรแกรมอื่นๆ ที่ทดลองใช้
* **โปรแกรม LexTo** เป็นโปรแกรมที่ถูกพัฒนาด้วยภาษา Java โดยโปรแกรมนี้สามารถใช้ในการแบ่งคำต่างๆในภาษาไทยจากประโยคให้กลายเป็นคำซึ่งแบ่งด้วย delimiter ซึ่งคำต่างๆที่ใช้ในการแบ่งนั้น จะมี Dictionary ที่จะทำการเก็บคำทั้งหมดเอาไว้ แล้วโปรแกรมจะนำมาเปรียบเทียบเพื่อแบ่งคำตามที่ Dictionary ได้กำหนดไว้ ซึ่ง LexTo เป็นโปรแกรมตัดคำภาษาไทยแบบ Open Source ที่ทางกลุ่มสามารถนำมาใช้งานได้ และมีความแม่นยำในระดับที่พอรับได้ ทำให้ทางกลุ่มเลือกใช้โปรแกรม LexTo
* **ภาษา Python** เป็นภาษาโปรแกรมที่ทำงานในลักษณะ Scripting Language โดยจะเลือกใช้ Machine Learning Library ต่างๆเพื่อนำมาสร้าง Machine Learning Model
* **ภาษา Java** เป็นภาษาโปรแกรมในลักษณะ Object-Oriented Programming ที่ได้รับความนิยมสูงสุดในปัจจุบัน ถูกเลือกนำมาใช้ในการเขียนโปรแกรมสำหรับทำ Text preprocessing เนื่องจากมีความยืดหยุ่นในการทำงานสูง
* **ภาษา PHP** เป็นภาษาโปรแกรมในที่ทำงานในลักษณะ Server Scripting language ซึ่งใช้ในการทำ Web Application เพื่อใช้ในการติดต่อกับทาง Database ผ่าน JDBC และ Web Socket

## รายละเอียดโปรแกรมที่จะพัฒนา (Software Specification)

### **Input / Output Specification**

#### **Component Diagram**



ภาพที่ 1 Component Diagram

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

***Ingestion Web Application***

* Input – document file สำหรับการทำ prediction หรือ label ในเอกสารที่จัดเตรียมไว้แล้วสำหรับการ train model
* Output - Notification ว่ามีการรับ Input สำเร็จแล้ว

***Machine Learning: Train Model***

* Input – PDF, DOCX File ที่มีการระบุ tag ในแต่ละ paragraph โดยผู้เชี่ยวชาญเฉพาะทาง
* Output - Model ที่สามารถคาดเดา tag จาก paragraph และ Tag ที่ได้จากเอกสารเหล่านั้น เก็บลงใน Database

***Machine Learning: Prediction***

* Input – PDF, DOCX File ที่ไม่มีการระบุ tag มาล่วงหน้า
* Output - ข้อมูล Tag ที่ได้จากการ Prediction จาก PDF, DOCX ที่เป็น Input โดยใช้ model ที่สร้างขึ้น และเก็บข้อมูลไฟล์เอกสารและ 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, DOCX

#### **Flowchart**

flowchart_rotate

ภาพที่ 2 Flowchart

ส่วนประกอบในการทำงาน จะแบ่งขั้นตอนต่างๆออกเป็น 5 ส่วน ได้แก่

1. **Ingestion Web Application** จะมีเพียงทางโรงเรียนและคุณครูที่ได้รับการอนุญาตเท่านั้นจะสามารถ Log In เข้ามาในระบบและทำการใส่เอกสารและใส่ Tag เข้าไปยังเอกสารต่างๆได้ โดยจะแบ่งวิธีการรับข้อมูลเป็น 2 แบบคือ 1. ทำการรับ Label สำหรับใช้ใน Train Model และ 2. รับไฟล์เอกสารอย่างเดียวสำหรับ Predict Tag จากไฟล์เอกสารนั้น



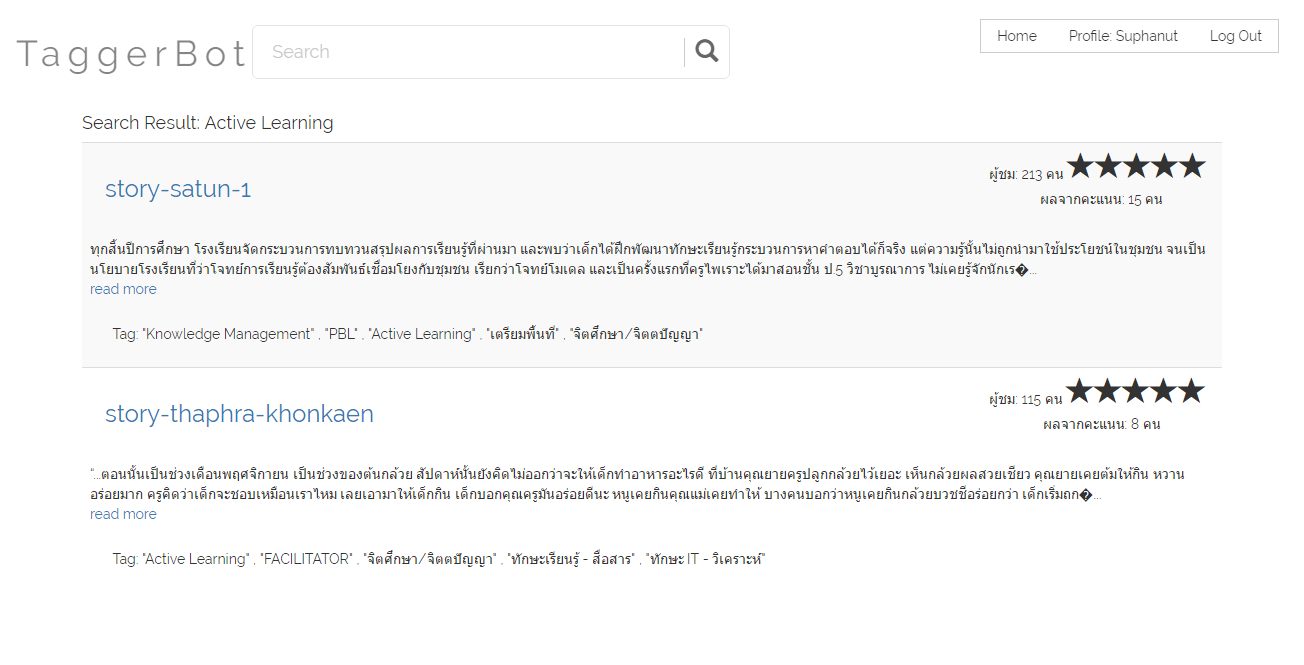
ภาพที่ 3 ตัวอย่าง Ingestion Web Application

1. **Preprocessing Data** โดยในขั้นตอนนี้จะทำการเปลี่ยนไฟล์เอกสารที่รับมาให้กลายเป็น 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 (Stopword Remover) ได้แก่ คำเชื่อม เช่น และ, หรือ, กับ เป็นต้น และคำขยายความต่างๆเช่น การ ความ เป็นต้น และสุดท้ายจะทำ bag-of-word เพื่อสร้าง vector ของคำและนำไปใช้ในขั้นตอน Topic Discovery
2. **Machine Learning: Classification** เป็นการสร้าง Machine สำหรับการจำแนกผลลัพธ์จากข้อมูลที่เข้ามา โดยจะแบ่งขั้นตอนการใช้งานเป็น 2 ขั้นตอนได้แก่

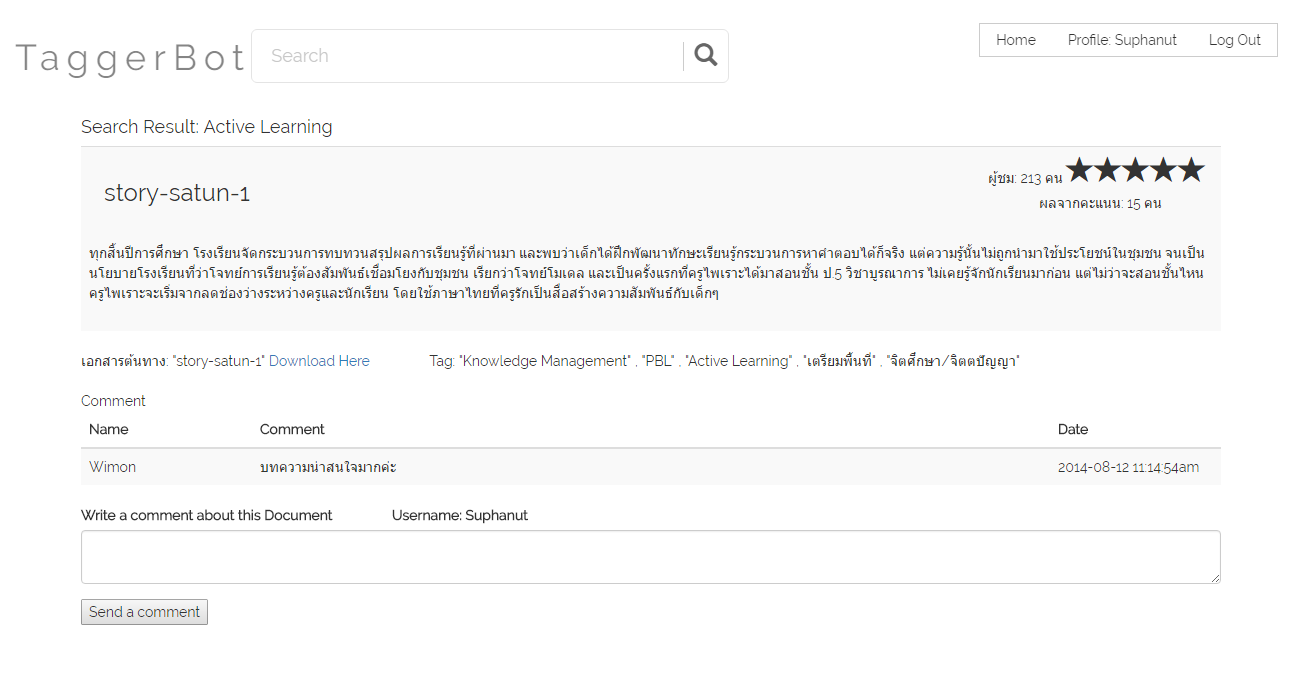
3.1 การทำ Training และ Testing Model โดยในขั้นตอนนี้จะนำคำต่างๆที่ได้จากขั้นตอนข้างต้น รวมกับ label ที่ผู้เชี่ยวชาญได้ระบุไว้มาสร้าง Model สำหรับการ Classification ออกมา

3.2 การ Prediction จากเอกสารต่างๆ เพื่อให้ได้ผลลัพธ์ออกมาเป็น tag เพื่อนำไปใช้ในการสืบค้นใน Database ต่อไป

1. **Query Web Application** จะเป็นการสร้าง Web Application เพื่อติดต่อกับ Database โดยตัว Web Application นั้น จะทำการ Query Tag ที่ต้องการสืบค้นจาก Database แล้วนำมาแสดงผลเอกสารที่ทำการค้นหาได้, ตัวอย่างเนื้อหาในเอกสาร, Tag, จำนวนผู้ชมและ rating ของเอกสารตามตัวอย่างข้างล่าง



เมื่อกดเลือกไปที่เอกสารที่ต้องการ ตัว Web Application จะแสดงเนื้อหาย่อหน้าทั้งหมดและ Tag ทั้งหมดในย่อหน้านี้ และ Comment ต่างๆที่ User ได้มาให้ข้อคิดเห็น และสามารถ Download เอกสารตัวเต็มได้จากหน้านี้



### **Functional Specification**

**Ingestion Web Application**

* Upload PDF, DOCX File มาเก็บไว้ในฐานข้อมูลและทำการส่งไปยังขั้นตอน Data Preprocess and NLP
* ผู้ที่ได้รับเลือกจะสามารถกรอก label ของ paragraph ต่างๆ สำหรับนำไปทำการ Training ได้

**Data Preprocess and NLP**

* ทำการแปลง PDF,DOCX ที่รับมา ให้เปลี่ยนเป็น xml fileโดยการใช้ PDFBox, unzip
* ทำการ preprocess text ที่ได้จาก pdf เพื่อลดแก้ไขคำผิดที่เกิดขึ้นจากการแปลง PDF เป็น text เช่น สระ อำ หลังจากแปลงจะได้ คำว่า “อ า” แทน “อำ”
* ทำการใส่ Tag ของคำเพิ่มเข้าไปใน paragraph สำหรับใช้เพื่อระบุในขั้นตอนต่อไป
* ทำการตัดคำจาก text file ด้วยโปรแกรม word segmentation สำหรับภาษาไทย เช่น LexTo
* ทำการกำจัดคำที่ไม่จำเป็นออกโดยการใช้ stopword remover
* สร้าง bag-of-word model สำหรับเก็บคำภาษาไทย และแปลงคำต่างๆจากในย่อหน้าให้กลายเป็น vector
* ใช้หลักการ TF-IDF ในการหาความถี่ของคำต่างๆ

**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

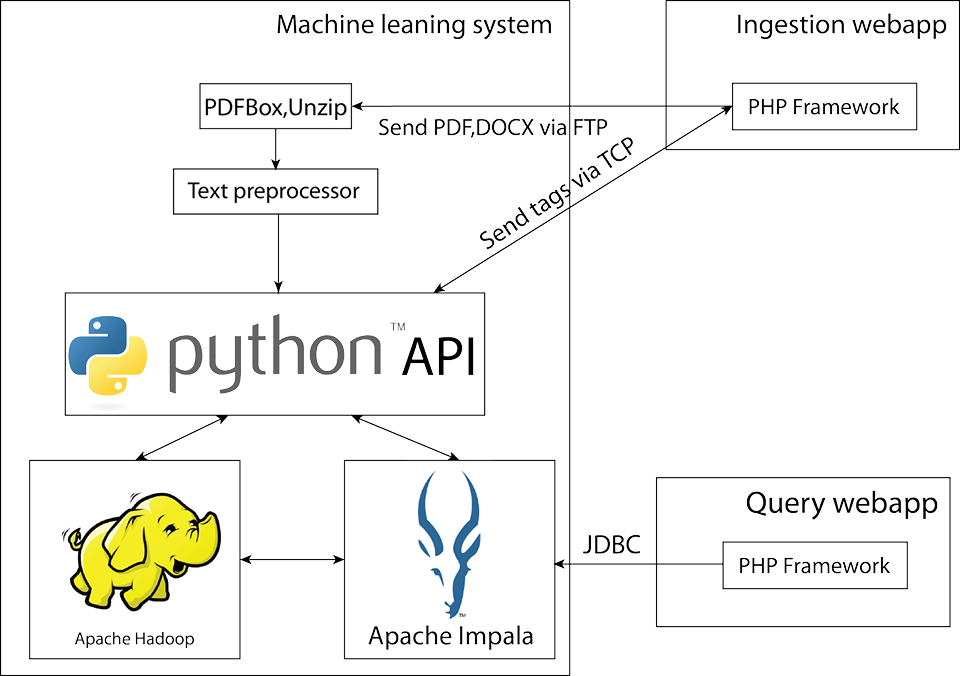
**Classification: Prediction**

* 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 ลิงค์สำหรับดาวน์โหลดไฟล์เอกสารที่สมบูรณ์

**Architecture Diagram**



ภาพที่ 5 Architecture Diagram

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

* โปรแกรม PDFBox ที่ใช้ในการแปลงไฟล์ PDF ให้เป็นไฟล์ข้อความ หรือโปรแกรม unzip ที่ใช้ในการแปลงไฟล์ DOCX ให้เป็นไฟล์ข้อความ
* ส่วนของโปรแกรมที่ใช้ในการจัดเตรียมข้อความเพื่อที่จะนำไปใช้ในการทำ Machine learning ได้แก่โปรแกรมสำหรับจัดเรียงข้อมูลที่ไม่เรียบร้อย (data cleaning), โปรแกรมแบ่ง paragraph และโปรแกรม LexTo
* โปรแกรมที่ใช้ในการทำ Machine Learning ซึ่งถูกพัฒนาขึ้นในภาษา Python
* โปรแกรม Apache Hadoop และ ที่ใช้ในการจัดเก็บข้อมูลตัวไฟล์เอกสารต้นฉบับ
* โปรแกรม Apache Impala เป็นโปรแกรมจัดการฐานข้อมูลที่ใช้ในระบบ hadoop ecosystem

ส่วนต่อมาเป็นส่วนของหน้าเว็บที่ใช้ในการรับ PDF ซึ่งจะพัฒนาขึ้นด้วยภาษา PHP โดยมีหน้าที่รับไฟล์ PDF,DOCX ที่อัพโหลดขึ้นมาจากผู้ใช้งาน หรือรับ label ที่ผู้เชี่ยวชาญใส่มาให้ในเอกสารตัวอย่างที่จัดเตรียมไว้เรียบร้อยแล้ว แล้วทำการส่งไฟล์เอกสารมาที่เครื่องที่ทำการทำ machine learning ด้วย FTP protocol และในส่วนของ label จะมีการส่งข้อความ tag ด้วย TCP protocol

ส่วนสุดท้ายเป็นส่วนของหน้าเว็บที่จะใช้ผู้ใช้ค้นหาเนื้อหาจาก tag ซึ่งพัฒนาขึ้นด้วยภาษา PHP เช่นเดียวกัน โดยหน้าเว็บจะรับ tag ที่ผู้ใช้ต้องการค้นหามาแล้วไปทำการ Query ใน Impala ออกมาแสดงผลให้ผู้ใช้งาน โดยติดต่อผ่าน JDBC

### **ส่วนที่ได้ทำการพัฒนาได้พัฒนาขึ้นเอง**

**Machine Learning**

* Pipeline ทั้งหมดในการทำ Machine Learning - ทีมงานพัฒนา
* การแปลง PDF/Doc เป็น Text - PDFBox และ Unzip
* การแปลง Text ภาษาไทยให้กลายเป็น Vector - ทีมงานพัฒนาขึ้นมาโดยดัดแปลงจาก CountVecterizer ให้สามารถเก็บ Vector คำภาษาไทยได้ จาก<http://scikitlearn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html>)
* การทำ Stopword Remover – ทีมงานพัฒนา
* Supervised Machine Learning – Scikit Library

**Web Application**

* หน้า UI ทั้งหมด - ทีมงานพัฒนา และเรียกใช้ Boostrap ในการสร้าง จาก <http://getbootstrap.com/>
* Function สำหรับการเรียก Imapala Database – ทีมงานพัฒนา
* Webserver – Ubuntu และ Apache Server

**Database**

* Database Server – CentOS, Cloudera Hadoop, Impala
* ข้อมูล Table สำหรับ User และเอกสารต่างๆ – ทีมงานพัฒนา

## 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 ดังนั้นถ้าหากเราไม่สามารถหาผู้เชี่ยวชาญที่จะมาระบุคำสำคัญให้ได้ เราก็จะไม่สามารถทำให้ระบบเรียนรู้หัวข้อใหม่ๆ ได้
* การระบุย่อหน้าจากไฟล์ PDF นั้นสามารถทำได้ยาก เนื่องจากการระบุย่อหน้าจาก PDF จำเป็นต้องใช้ค่าตำแหน่งของตัวอักษรต่างๆ เพื่อระบุว่าย่อหน้าควรจะอยู่ตำแหน่งไหน ซึ่ง PDF ที่ได้รับมานั้น มีรูปแบบการจัดหน้าและ font ที่แตกต่างกันรวมถึงรูปแบบคำภาษาไทยและภาษาอังกฤษในเอกสาร จะทำให้ตำแหน่งของคำเกิดการคลาดเคลื่อนซึ่งจะส่งผลให้ย่อหน้าที่ได้ออกมาอาจเกิดความผิดพลาดได้ ซึ่งในกรณีของไฟล์ DOCX นั้นจะไม่มีปัญหาดังกล่าว

# กลุ่มผู้ใช้โปรแกรม

1. ผู้ปกครอง สามารถสืบค้นเอกสารต่างๆ ที่เกี่ยวข้องกับการเรียนการสอน ตาม Tag ที่ค้นหา เพื่อใช้ในการค้นหาโรงเรียนที่มีจุดเด่นในการเรียนการสอนแบบต่างๆ
2. คุณครู สามารถสืบเอกสารต่างๆ เพื่อใช้ในการพัฒนาวิธีการสอนของตนเองได้
3. คุณครูที่มีเอกสารด้านการเรียนการสอนและต้องการแบ่งปันความรู้ สามารถอัพโหลดเอกสารต่างๆเข้ามาในระบบและทำการจัดเอกสารให้อย่างอัตโนมัติ

# ผลของการทดสอบโปรแกรม

ผลทดสอบ

# ปัญหาและอุปสรรค

เนื่องจากภาษาไทยเป็นภาษาที่มีการใช้ระบบสระและวรรณยุคต์ทำให้การแปลง PDF กลายเป็น text นั้นจะเกิดปัญหาเนื่องจากตัวอักษรดังกล่าว เนื่องจากไม่มีโปรแกรมสำหรับการดึง Text จาก PDF สำหรับภาษาไทยโดยเฉพาะ ทำให้จำเป็นต้องใช้โปรแกรมสำหรับภาษาอังกฤษโดยการใช้โปรแกรมสำหรับตัดคำทั่วไปในภาษาอังกฤษจะไม่มีระบบตัวอักษรเหล่านี้ทำให้เกิดความผิดพลาดได้

ในส่วนของการทำ Machine Learning นั้น เนื่องจากการทำการ tag เอกสารภาษาไทยนั้น ต้องใช้ทรัพยากรคนและเวลาเป็นอย่างมาก ทำให้เอกสารสำหรับการ training model นั้นไม่เยอะเพียงพอสำหรับการ train Machine Learning Model

เนื่องจาก Lexto นั้นจำเป็นต้องมีการอัพเดท Dictionary เพื่อรองรับคำใหม่ๆเพิ่มขึ้นในการตัดคำทำให้จำเป็นต้องทำการ update Lexto ตลอดเวลา

เวลาสำหรับการทำโปรเจคน้อยเกินไป ทำให้ชิ้นงานอาจจะยังไม่เสร็จสมบูรณ์ในบางส่วน ต้องใช้เวลาในการพัฒนาต่อในภายหลัง

# แนวทางการพัฒนาและประยุกต์ใช้ร่วมกับงานอื่นๆ ในขั้นตอนต่อไป

นำระบบ OCR ภาษาไทยที่รองรับภาษาไทยมาใช้ เพื่อให้ได้ข้อมูล Text จาก PDF ที่มีความถูกต้องมากกว่าเดิม

ติดตั้งระบบประมวลผลแบบ Parallel ด้วย Spark เพื่อเพิ่มประสิทธิภาพในการประมวลผล ในกรณีที่ระบบมีการใช้งานเป็น Community ขนาดใหญ่ โดยมีคนอัพโหลดเอกสารจำนวนมาก ทำให้จำเป็นต้องใช้กำลังในการประมวลผลเอกสารเหล่านั้น

ติดตั้งระบบ Feedback สำหรับรับความเห็นของผู้ใช้ที่มีความสนใจในเนื้อหา เพื่อที่จะใช้ในการสอน Model ให้มีความถูกต้องมากยิ่งขึ้น

# ข้อสรุปและข้อเสนอแนะ

ผลลัพธ์ที่ได้ยังไม่ดีพอเนื่องจากข้อมูลมีขนาดไม่มากพอ เราควรจะทำการหาข้อมูลเพิ่มเติมเพื่อใช้ในการสอน ML เพื่อเพิ่มความถูกต้องที่มากกว่าเดิม

# 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](https://www.scribd.com/document/113006633/2006-An-Empirical-Comparison-of-Supervised-Learning-Algorithms%20%5b2016), October 18]

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

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

# สถานที่ติดต่อของผู้พัฒนาและอาจารย์ที่ปรึกษา โทรศัพท์ มือถือ โทรสาร E-mail

ทีมพัฒนา

**นายศุภณัฐ ทัตตินาพานิช**

**สถานที​ติดต่อ เลขที่ ​126 ถ.ประชาอุทิศ แขวงบางมด เขตเขตทุ่งครุ กรุงเทพมหานคร 10140**

**โทรศัพท์ 024708000 มือถือ 0896202211 e-mail** [**zarkzaki@hotmail.com**](mailto:zarkzaki@hotmail.com)

**นายอินทัช แสงกระจ่าง**

**สถานที​ติดต่อ เลขที่ ​126 ถ.ประชาอุทิศ แขวงบางมด เขตเขตทุ่งครุ กรุงเทพมหานคร 10140**

**โทรศัพท์ 024708000 มือถือ 0850436161 e-mail** [artkrub7@gmail.com](mailto:artkrub7@gmail.com)

**นายปุญญพัฒน์ เศรษฐ์สมบูรณ์**

**สถานที​ติดต่อ เลขที่ ​126 ถ.ประชาอุทิศ แขวงบางมด เขตเขตทุ่งครุ กรุงเทพมหานคร 10140**

**โทรศัพท์ 024708000 มือถือ 0877170741 e-mail** [**perth.s28@gmail.com**](mailto:perth.s28@gmail.com)

อาจารย์ที่ปรึกษาโครงการ

**รศ.ดร.ธีรณี อจฉลากุล**

**สถานที​ติดต่อ เลขที่​126 ถ.ประชาอุทิศ แขวงบางมด เขตเขตทุ่งครุ กรุงเทพมหานคร 10140**

**โทรศัพท์ 024709380 มือถือ 0816437846 e-mail** [tiranee@cpe.kmutt.ac.th](mailto:tiranee@cpe.kmutt.ac.th)

# ภาคผนวก

ผลลัพธ์การทดลอง